Numéro d'ordre : 2009-ISAL-0077 Année 2009

Thèse

Contributions au tri automatique de documents et de courrier d'entreprises

Présentée devant
L'institut national des sciences appliquées de Lyon

Pour obtenir
Le grade de docteur

École doctorale : Informatique et Mathématiques de Lyon (InfoMaths)

Spécialité : Documents numériques, Images et Systèmes d'Information Communicants

Par
Djamel GACEB

COMPOSITION DU JURY

M. CHERIET	Professeur (École de Technologie Supérieure, Canada), Rapporteur
C. VIARD-GAUDIN	Professeur (École Polytechnique de l'Université de Nantes), Rapporteur
M. Jean-Marc OGIER	professeur (Université de La Rochelle), Examinateur
H. EMPTOZ	Professeur (INSA de Lyon), Directeur de thèse
V. EGLIN	Maître de conférences (INSA de Lyon), Directrice de thèse.
B. MAISONNEUVE	Directeur de la société CESA (Groupe VINCI) Lyon,

Laboratoire de recherche : Laboratoire d'InfoRmatique en Image et Systèmes d'information, UMR 5205 CNRS/INSA de Lyon.

A la mémoire de ma mère

Le dernier de tes fils, ta fierté, j'aime à croire ton préféré, te dédie son travail.

Comme le tendre regard que je sentais sur moi lorsque je quittais la maison pour rejoindre l'école, la force qui m'a portée jusqu'ici est ta présence que je sens encore à mes cotés.

Pour Abou Djamel qui t'est resté fidèle.

Pour celui dans les yeux duquel je lisais tes regards, ton petit frère Omar qui depuis peu est à tes cotés.

Pour ceux de ton sang qui ont comblé un peu du vide que tu as laissé.

Pour mon jeune disciple d'enfance qui dépassera bientôt son maître, l'ainé de tes petits fils.

Pour mes amis qui te connaissent sans jamais t'avoir vu.

Pour l'étoile qui m'a guidée ces dernières années.

Remerciements

Cette thèse s'est déroulée au sein du laboratoire LIRIS – INSA de Lyon et la société CESA de Lyon.

Merci à M. Hubert EMPTOZ, professeur à l'Institut National des Sciences Appliquées de Lyon et fondateur de l'équipe RFV et Document, de m'avoir accueilli au sein du laboratoire LIRIS dès mon Master et de m'avoir dirigé durant ma thèse. Grâce à son leadership, ces années au laboratoire ont été enrichissantes tant au niveau de la recherche qu'au niveau de l'enseignement.

Merci à Mme.Véronique EGLIN, Maître de conférences à l'Institut National des Sciences Appliquées de Lyon, de n'avoir jamais douté de mes capacités. Je tiens à la remercier pour son encadrement constructif depuis mon stage de Master, pour son soutien permanent, son écoute, ses conseils, son aide, sa patience, son respect, sa bonne humeur et ses encouragements. La liste est encore longue...

Merci à M. Bruno MAISONNEUVE, Directeur de la société CESA de Lyon groupe VINCI, de m'avoir accueilli au sein de sa société durant mes trois années de thèse, de m'avoir parfaitement expliqué les besoins de son entreprise et ses problèmes inhérents. Je le remercie de m'avoir accordé du temps et de l'énergie grâce auxquels j'ai su prendre en compte les contraintes et les exigences de son environnement de travail et apprendre à gérer les contraintes de la R&D en entreprise.

Je tiens à exprimer également mes plus vifs remerciements et toute ma gratitude à:

- M. Jean-Marc OGIER, professeur à l'université de La Rochelle, de m'avoir fait l'honneur de présider ce jury,
- M. Mohamed CHERIET, professeur à l'École de Technologie Supérieure de Canada, et M. Christian VIARD-GAUDIN, professeur de l'École Polytechnique de l'Université de Nantes, d'avoir accepté d'être rapporteurs de cette thèse et pour les judicieux conseils qu'ils m'ont apporté,

Merci à M. Frank LEBOURGEOIS, Maître de conférences à l'Institut National des Sciences Appliquées de Lyon, pour ses nombreux conseils scientifiques et sa grande culture humoristique.

Merci à tous les membres de l'équipe Imagine du laboratoire LIRIS avec qui j'ai pu échanger et à tous mes collègues de la société CESA pour leur esprit d'équipe et leur excellente présence durant cette thèse.

Un grand merci à ma mère et mon père à qui je dois cette éducation, cette soif de savoir et de science. Il n'existe pas de mot pour exprimer ma gratitude.

Merci à toute ma famille, petits et grands, pour leur soutien et leurs encouragements, je ne saurais m'arrêter si je parle de chacun de vous, vous avez tous été formidables.

Merci à tous mes amis avec qui j'ai partagé d'agréables souvenirs.

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **http://sakura.cpe.fr/ED206** M. Jean Marc LANCELIN Insa : R. GOURDON	M. Jean Marc LANCELIN Université Claude Bernard Lyon 1 Bât CPE 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél : 04.72.43 13 95 Fax : lancelin@hikari.cpe.fr
E.E.A.	**ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE** **http://www.insa-lyon.fr/eea** M. Alain NICOLAS Insa : C. PLOSSU ede2a@insa-lyon.fr Secrétariat : M. LABOUNE AM. 64.43 – Fax : 64.54	M. Alain NICOLAS Ecole Centrale de Lyon Bâtiment H9 36 avenue Guy de Collongue 69134 ECULLY Tél : 04.72.18 60 97 Fax : 04 78 43 37 17 eea@ec-lyon.fr Secrétariat : M.C. HAVGOUDOUKIAN
E2M2	**EVOLUTION, ECOSYSTEME, MICROBIOLOGIE, MODELISATION** **http://biomserv.univ-lyon1.fr/E2M2** M. Jean-Pierre FLANDROIS Insa : H. CHARLES	M. Jean-Pierre FLANDROIS CNRS UMR 5558 Université Claude Bernard Lyon 1 Bât G. Mendel 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cédex Tél : 04.26 23 59 50 Fax 04 26 23 59 49 06 07 53 89 13 e2m2@biomserv.univ-lyon1.fr
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTE** Sec : Safia Boudjema M. Didier REVEL Insa : M. LAGARDE	M. Didier REVEL Hôpital Cardiologique de Lyon Bâtiment Central 28 Avenue Doyen Lépine 69500 BRON Tél : 04.72.68 49 09 Fax :04 72 35 49 16 Didier.revel@creatis.uni-lyon1.fr
INFOMATHS	**INFORMATIQUE ET MATHEMATIQUES** http://infomaths.univ-lyon1.fr M. Alain MILLE Secrétariat : C. DAYEYAN	M. Alain MILLE Université Claude Bernard Lyon 1 LIRIS - INFOMATHS Bâtiment Nautibus 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél : 04.72. 44 82 94 Fax 04 72 43 13 10 infomaths@bat710.univ-lyon1.fr - alain.mille@liris.cnrs.fr
Matériaux	**MATERIAUX DE LYON** M. Jean Marc PELLETIER Secrétariat : C. BERNAVON 83.85	M. Jean Marc PELLETIER INSA de Lyon MATEIS Bâtiment Blaise Pascal 7 avenue Jean Capelle 69621 VILLEURBANNE Cédex Tél : 04.72.43 83 18 Fax 04 72 43 85 28 Jean-marc.Pelletier@insa-lyon.fr
MEGA	**MECANIQUE, ENERGETIQUE, GENIE CIVIL, ACOUSTIQUE** M. Jean Louis GUYADER Secrétariat : M. LABOUNE PM : 71.70 –Fax : 87.12	M. Jean Louis GUYADER INSA de Lyon Laboratoire de Vibrations et Acoustique Bâtiment Antoine de Saint Exupéry 25 bis avenue Jean Capelle 69621 VILLEURBANNE Cedex Tél :04.72.18.71.70 Fax : 04 72 43 72 37 mega@lva.insa-lyon.fr
ScSo	**ScSo*** M. OBADIA Lionel Insa : J.Y. TOUSSAINT	M. OBADIA Lionel Université Lyon 2 86 rue Pasteur 69365 LYON Cedex 07 Tél : 04.78.69.72.76 Fax : 04.37.28.04.48 Lionel.Obadia@univ-lyon2.fr

*ScSo : Histoire, Geographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Contributions au tri automatique de documents et de courrier d'entreprises

Résumé

Ce travail de thèse s'inscrit dans le cadre du développement de systèmes de vision industrielle pour le tri automatique de documents et de courriers d'entreprises. Ces systèmes sont par nature très exigeants en temps de traitement mais aussi en justesse et précision des résultats. Les systèmes actuels sont composés, pour la plupart, de modules séquentiels exigeant des algorithmes efficaces et rapides tout au long de la chaîne des traitements, depuis les étapes de bas niveau jusqu'aux étapes de niveau supérieur d'analyse fine et de reconnaissance des contenus. Les architectures existantes, dont nous avons balayé les spécificités dans les trois premiers chapitres de la thèse, présentent des faiblesses qui se traduisent par des erreurs de lecture et des rejets que l'on impute encore trop souvent aux OCR. Or, les étapes responsables de ces rejets et de ces erreurs de lecture sont les premières à intervenir dans le processus, à savoir celles de segmentation et de localisation de zones d'intérêts ; ces deux étapes qui s'impliquent mutuellement conditionnent les performances des systèmes et le rendement des chaînes de tri automatique.

Nous avons ainsi choisi porter notre contribution sur les aspects inhérents à la segmentation des images de courriers et la localisation de leurs régions d'intérêt (comme la zone d'adresse) en investissant une nouvelle approche pyramidale de modélisation par coloration hiérarchique de graphes ; à ce jour, la coloration de graphes n'a jamais été exploitée dans un tel contexte. Elle intervient dans notre contribution à toutes les étapes d'analyse de la structure des documents ainsi que dans la prise de décision pour la reconnaissance (reconnaissance de la nature du document à traiter et reconnaissance du bloc adresse). La partie de reconnaissance a été conçue autour d'un apprentissage traité à l'aide d'un modèle unique portant sur la b-coloration de graphe.

Notre architecture a été conçue pour réaliser essentiellement les étapes d'analyse de structures et de reconnaissance en garantissant une réelle coopération entres les différents modules d'analyse et de décision. Elle s'articule autour de trois grandes parties : une partie de segmentation bas niveau (binarisation et recherche de connexités), une partie d'extraction de la structure physique par coloration hiérarchique de graphe et une partie de localisation de blocs adresse et de classification de documents. Les algorithmes impliqués dans le système ont été conçus pour leur rapidité d'exécution (en adéquation avec les contraintes de temps réels), leur robustesse, et leur compatibilité. Les expérimentations réalisées dans ce contexte sont très encourageantes et offrent également de nouvelles perspectives à une plus grande diversité d'images de documents.

Mots-Clés: Extraction de la structure physique, catégorisation de documents, localisation de bloc adresse, coloration et b-coloration de graphes, tri de courriers en temps réel.

Contributions to the automatic sorting of company documents and mail

Abstract

This thesis deals with the development of industrial vision systems for automatic business documents and mail sorting. These systems need very high processing time, accuracy and precision of results. The current systems are most of time made of sequential modules needing fast and efficient algorithms throughout the processing line: from low to high level stages of analysis and content recognition.
The existing architectures that we have described in the three first chapters of the thesis have shown their weaknesses that are expressed by reading errors and OCR rejections. The modules that are responsible of these rejections and reading errors are mostly the first to occur in the processes of image segmentation and interest regions location. Indeed, theses two processes, involving each other, are fundamental for the system performances and the efficiency of the automatic sorting lines.
In this thesis, we have chosen to focus on different sides of mail images segmentation and of relevant zones (as address block) location. We have chosen to develop a model based on a new pyramidal approach using a hierarchical graph coloring. As for now, graph coloring has never been exploited in such context. It has been introduced in our contribution at every stage of document layout analysis for the recognition and decision tasks (kind of document or address block recognition). The recognition stage is made about a training process with a unique model of graph b-coloring.
Our architecture is basically designed to guarantee a good cooperation bewtween the different modules of decision and analysis for the layout analysis and the recognition stages. It is composed of three main sections: the low-level segmentation (binarisation and connected component labeling), the physical layout extraction by hierarchical graph coloring and the address block location and document sorting. The algorithms involved in the system have been designed for their execution speed (matching with real time constraints), their robustness, and their compatibility. The experimentations made in this context are very encouraging and lead to investigate a wider diversity of document images.

Key words: Physical layout extraction, document categorization, address block localization, graph coloring and b-coloring, mail sorting in real time.

Table des matières

Introduction générale

La quantité d'information disponible a eu une croissance exponentielle ces dernières années dans notre société. Le document sur papier que l'on croyait, un temps, menacé semble bel et bien perdurer à l'ère du tout numérique. L'exploitation du courrier papier est même en progression constante malgré l'augmentation des courriers électroniques. Parallèlement à cette progression, on voit s'afficher de façon très marquée de nouvelles exigences de qualité venant des entreprises qui exigent une fiabilité totale dans la distribution des courriers, d'excellents rapports qualité/prix et des solutions globales innovantes permettant de renforcer l'efficacité de leurs campagnes de marketing et d'optimiser la gestion de leurs documents et de leur courrier.

Le support papier remplit chaque jour des fonctions et répond à de nombreux usages pour lesquels peu d'utilisateurs préféreraient une version numérique. En conséquence, une très grande part de correspondances entre les personnes et les entreprises est toujours réalisée au moyen de documents papier. Les documents échangés sont aussi divers que les motifs de relations: bons de commande, formulaires, chèques, courriers, dossiers, contrats, questionnaires, colis, déclarations fiscales et sociales, feuilles de soin, etc.

Face à la persistance des documents physiques, à la charge qu'ils représentent, à la diversité des formats de documents reçus, à la multiplicité des destinataires et à la recherche constante de productivité, les organisations souhaitent se doter de solutions intelligentes et efficaces. Elles ont longtemps cherché des moyens techniques permettant de réduire les coûts et les délais de traitement.

C'est alors qu'il convient de définir et mettre en œuvre un processus de traitement informatique général allant de l'acquisition suivie par les meilleurs procédés Reconnaissance et Lecture Automatique de Documents (RAD/LAD) jusqu'à l'exploitation pour tirer pleinement parti des possibilités offertes par cette tâche. De ce fait, la conception des systèmes de vision à haute cadence suscite un intérêt sans cesse croissant. Cette tendance a été accompagnée et encouragée aussi par l'évolution rapide de la puissance des ordinateurs qui continue de respecter la loi de Moore.

Au cours de la dernière décennie, de nombreuses applications industrielles ont été mises en place avec des objectifs différents mais utilisant certains outils de solutions génériques. Les applications traitées ne présentent pas toutes les mêmes caractéristiques ni les mêmes difficultés. Il est possible de les répartir en un certain nombre de familles : la lecture optique de questionnaires, la lecture de documents normalisés, la lecture automa-

tique de formulaires, le traitement des chèques et les applications bancaires, le traitement de documents non structurés ou semi-structurés, la lecture automatique des adresses postales.

Les applications de reconnaissance d'adresses postales sont en service dans des centaines de sites de la plupart des pays développés. Des systèmes de lecture automatique de formulaires sont utilisés dans tous les secteurs de l'économie. Des logiciels permettent d'extraire les informations principales des factures à payer par une entreprise avant leur gestion en comptabilité, alors que d'autres sont capables de trier et de qualifier les contenus des courriers entrant dans une organisation pour les orienter vers un ou plusieurs destinataires internes auxquels seront déjà fournies les principales données du courrier à traiter. Le traitement automatique des documents physiques est ainsi devenu une véritable technologie industrielle.

Dans ce contexte, beaucoup d'efforts, de temps et de ressources ont été consacrés au développement des systèmes de vision spécialisés dans la lecture automatique de contenu de documents physiques. En particulier, le tri automatique de documents et de courriers d'entreprises est un domaine très actif où le nombre de produits commerciaux ne cesse de croître. Cette application engendre des études variées et elle requiert l'implication de plusieurs disciplines scientifiques pendant la conception d'un système de vision industrielle complet. Malgré la présence de résultats de qualité on peut encore apporter des améliorations.

Le panorama des solutions de LAD et de RAD destinées à ce type d'applications propose de distinguer deux types d'acteurs : les "historiques" de la reconnaissance d'écriture tels A2IA, Iris, Itesoft et SWT et d'autres, s'appuyant le plus souvent sur les technologies de ces derniers. Dans cette catégorie on peut placer la société CESA, un véritable expert du domaine de tri automatique.

C'est dans ce contexte, que s'inscrivent ces travaux de recherche. L'objectif de cette thèse se situe, donc, dans le cadre du développement d'un système complet de vision industriel pour le tri automatique de documents et de courriers d'entreprises. La société CESA partenaire de ce projet souhaite réduire à tout prix les taux de rejets et d'erreurs des systèmes de vision qu'elle conçoit et réalise en introduisant une meilleure coopération entre les différentes étapes du traitement et de l'analyse des images issues d'une acquisition en début de chaîne. La succession des étapes allant de la lecture optique au tri final des documents passe d'un point de vue méthodologique d'étapes fondamentales de reconnaissance de formes et d'objets à la prise de décision finale (comme le tri de courrier par exemple). L'ensemble des processus embarqués dans le système doit respecter la notion fondamentale de temps réel.

Notre objectif est centré sur l'étude d'un nouveau schéma d'organisation, non linéaire du processus de traitement de l'image, allant de la capture à la prise de décision. Il conviendra pour cela de concevoir une solution alliant rapidité de temps de traitement et échange d'information entre les différents étages du processus, afin de les faire contribuer intelligemment à la reconnaissance. Compte tenu de l'ampleur du problème, on ne développera pas d'outils particuliers pour la reconnaissance proprement dite des caractères ; pour cette tâche, nous ferons appel à un logiciel existant qui est paramétrable, celui qui a été conçu par la société A2IA. De plus, nous utiliserons les travaux sur la théorie des graphes liés à l'intégration de mécanismes de vision pré-attentive grâce aux apports de la multi-résolution afin d'élaborer une stratégie de recherche progressive d'information.

Notre travail s'articule, donc, en deux grandes phases :

1) la première consiste à développer les briques élémentaires constituées d'outils logiciels qui participent à la préparation des informations en vue de la reconnaissance ;

2) la seconde consiste à l'organisation optimale d'une réelle coopération entre les briques logicielles du système :

- extraction de la structure physique bas niveau des documents,
- localisation des zones de texte et marqueurs de zones graphiques,
- reconnaissance du type de documents / localisation de zones d'intérêt (bloc adresse),
- reconnaissance de la structure du document / classification des documents.

Nous allons structurer notre thèse qui s'intitule « Contributions au tri automatique de documents et de courrier d'entreprises » en cinq chapitres.

Nous allons présenter dans le premier chapitre les différentes briques logicielles nécessaires à la conception générale des systèmes de tri automatique de documents et de courriers. Nous évoquerons leurs spécificités et leurs limites.

Le deuxième chapitre est consacré à la présentation des principales approches existant dans la littérature pour chaque module de la chaîne de tri et à l'analyse des causes de leurs imprécisions ou lenteurs.

Nous présenterons tout d'abord les différents mécanismes de binarisation, et de détection des composantes connexes qui constituent généralement les premières étapes de segmentation de bas niveau. Nous aborderons ensuite les mécanismes d'analyse de structures des documents,

généralement présentés sous la forme d'approches ascendantes et descendantes. Nous ferons le constat qu'un grand nombre de méthodes de segmentation qui s'appliquent en temps réel sur des documents de type courriers et formulaires s'oriente vers la mise en place d'un mécanisme mixte (mi-ascendant / mi-descendant). Nous montrerons également comment la reconnaissance du type de document, la séparation texte / non texte des zones de document ou encore la séparation manuscrits / imprimés peut être initiée par une segmentation grossière du document et peut permettre d'aboutir à une segmentation plus fine des contenus. A ce stade, la reconnaissance et la segmentation constituent deux tâches qui s'influencent mutuellement et dont les interactions permettent des améliorations considérables à la fois de la segmentation et de la reconnaissance. Nous pourrions reformuler le paradoxe de Sayre [Sayre73] en écrivant que « *pour reconnaître une entité, il faut savoir la localiser, mais pour la localiser, il faut tout d'abord la reconnaître*». L'ensemble des contributions citées dans ce chapitre sera finalement argumenté afin de justifier nos choix et directions méthodologiques que les chapitres suivants détailleront et valideront.

La première partie du troisième chapitre est consacrée à la classification et la reconnaissance du type de document en argumentant leurs forces et leurs limites. Cette étape préalable permet de cibler les informations pertinentes pour le tri et de choisir un jeu de traitements plus adapté au contenu. La seconde partie de ce chapitre est destinée à la présentation de la problématique de la localisation automatique du bloc adresse qui constitue le cœur d'un système de tri. Elle consiste en une succession d'étapes allant de l'émergence des blocs informants à l'étiquetage et la décision.

Dans le quatrième chapitre, nous avons choisi porter notre contribution sur les aspects inhérents à la segmentation, la localisation des zones d'intérêt (localisation du bloc adresse) et à la reconnaissance automatique du type de documents en investissant une approche innovante de modélisation basée sur la théorie de graphe. Au début du chapitre nous introduisons les aspects théoriques de la coloration et de la b-coloration de graphes dans le cas général. Nous présentons ensuite les algorithmes que nous avons produits et qui s'adaptent aux besoins de notre application temps réel avec trois grands objectifs visés : l'extraction de la structure physique des images, la localisation du bloc adresse et la reconnaissance des familles de documents et de courriers.

Nous argumenterons donc la faisabilité d'une telle modélisation dans un contexte industriel où les contraintes de temps réels restent prévalantes. L'apport de la coloration et de la b-coloration de graphes dans les phases de segmentation et de reconnaissance de documents est ensuite vali-

dé et discuté. À ce jour, la coloration de graphes n'a jamais été exploitée dans un tel contexte.

Dans le cinquième chapitre « Proposition d'une nouvelle architecture pyramidale à base de coloration de graphes » nous allons décrire en détail le modèle retenu ainsi que les différents algorithmes développés pour résoudre les problèmes de segmentation et de reconnaissance des contenus. Dans ce chapitre, nous commenterons les résultats numériques des performances de notre système en les comparant à des approches plus usuellement employées dans un tel contexte. Nous conclurons ce chapitre sur le constat que la coloration de graphe employée dans la résolution des problèmes de tri de courriers d'entreprise est très efficace.

Dans la conclusion nous dresserons le bilan de nos contributions puis nous développerons quelques perspectives pour la suite de cette thèse.

Chapitre 1

Architecture générale des systèmes de tri

De l'acquisition d'images aux mécanismes de reconnaissance

1.1 Introduction

Face au nombre toujours croissant de documents physiques à traiter, à la diversité des formats, à la multiplicité des destinataires et à la recherche constante de productivité, les organisations souhaitent se doter de solutions intelligentes, économiques et rapides. Depuis plusieurs années, de gros enjeux économiques se jouent dans les entreprises chargées du traitement automatique des documents, lettres et courriers. Aussi elles s'intéressent à la mise en œuvre de processus innovant de traitement allant de l'acquisition à la Reconnaissance et Lecture Automatique de Documents (RAD/LAD). La conception des systèmes de vision à haute cadence suscite un intérêt sans cesse croissant, encore augmentée par l'évolution rapide de la puissance des ordinateurs.

Le tri automatique est une opération qui s'inscrit dans ce contexte. Sa spécificité est qu'il est fortement dépendant de la nature des objets à trier et des modes de dépôts, voir figure 1.1.

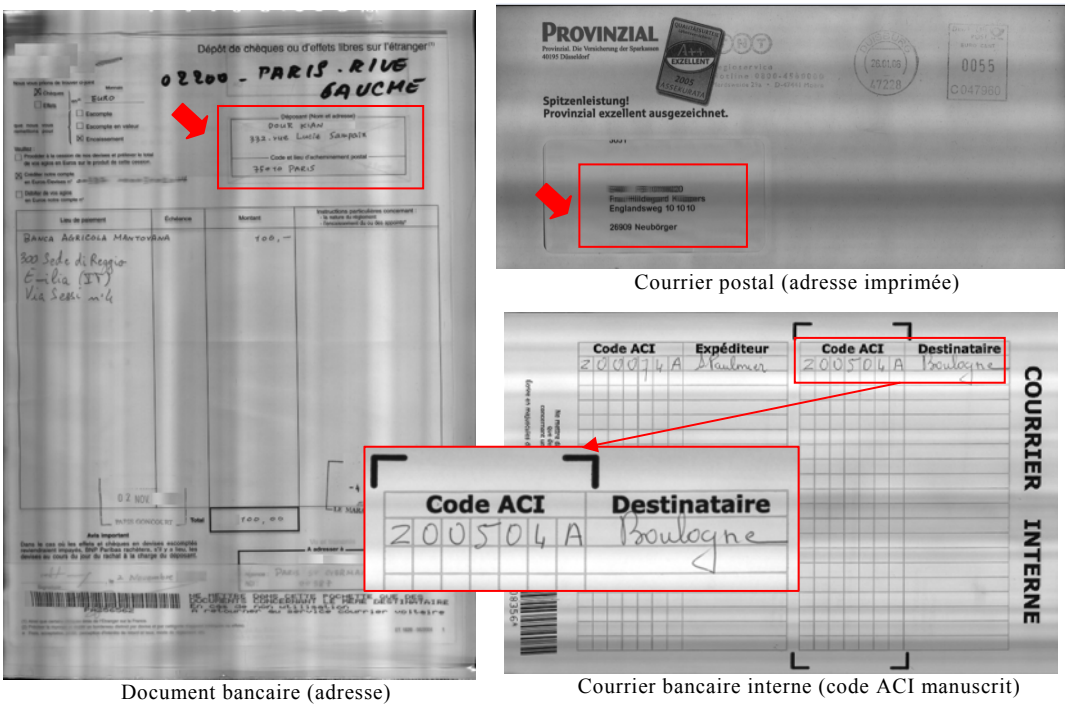
Figure 1.1 : Une très grande variété de documents et de courriers.

De l'enveloppe standard à l'adresse imprimée en passant par les revues sous plastique transparent, tout doit être, à terme, traité automatiquement. Certaines entreprises sont confrontées à la nécessité impérative d'automatiser leurs propres systèmes pour trier, en interne, leurs propres documents et courriers en plus de leur correspondance passant par La Poste. De plus cela permettre de les dégager de tâches fastidieuses et répétitives de saisie et de traitement qui entraînent leurs parts d'erreurs et de coûts.

La plupart des pays industrialisés utilisent des systèmes de tri automatique. Leurs performances sont difficiles à comparer dans la mesure où

les codes postaux et les objectifs du tri automatique présentent des difficultés variables : la taille des codes américains est de cinq ou neuf chiffres, les codes anglais comportent des chiffres et des lettres, les codes portugais seulement quatre chiffres…

En France, chaque jour, La Poste trie automatiquement des millions de lettres. Pour cela, elle dispose de plus de 100 machines de tri conçues et fabriquées par la société ALCATEL Postal Automation Systems. Ces équipements sont capables de saisir et d'interpréter les images des adresses de lettres défilant à 3 m/s à un débit maximal de 13 lettres à la seconde. Le résultat de la reconnaissance des adresses doit être produit en moins d'une seconde de manière à permettre la réalisation immédiate du tri physique des lettres. Les lettres soumises à la reconnaissance sont quelconques (manuscrites ou dactylographiées) et présentent la plupart des difficultés auxquelles doit s'attaquer un système de reconnaissance : localisation des blocs, séparation du manuscrit et du dactylographié, segmentation en lignes, mots et caractères, correction des erreurs éventuelles, voir figure 1.2.



Courrier postal (adresse imprimée)

Document bancaire (adresse)

Courrier bancaire interne (code ACI manuscrit)

Figure 1.2 : Exemples de courrier et de documents d'entreprises à trier où les informations de positions de la zone d'intérêt peuvent varier d'un modèle de courrier à l'autre.

Les informations reconnues permettent au centres de tri de courrier de réaliser différents niveaux de tri : tri d'acheminement vers le centre de tri d'arrivée puis le bureau distributeur fondé sur le code postal et le

nom de ville, tri de distribution correspondant aux différentes tournées de facteur fondé sur le nom de la voie et le numéro de porte. La Poste est ainsi aujourd'hui capable de trier automatiquement plus de 85 % des adresses dactylographiées (acheminement et distribution) et plus de 70 % des adresses manuscrites (acheminement seulement). Elle le fait, de plus, avec à peine quelques pourcentages d'erreur. Ces performances sont toutefois perfectibles et la Poste a, elle-même, développé au sein du Service de Recherche Technique de la Poste (SRTP) un logiciel capable de reconnaître une partie des rejets des machines de tri actuelles. Ce logiciel opère en temps différé, il ne rend pas son résultat dans un délai contraint par la nécessité d'un tri physique immédiat. Grâce à ce degré de liberté, des méthodes nouvelles ont pu être mises en œuvre permettant l'interprétation des images rejetées en première instance.

Dans le même temps, la banque BNP Paribas gère chaque jour une dizaine de tonnes de courrier et de documents envoyées aux 2300 agences du groupe et Immeubles Centraux. La fusion des banques BNP et Paribas en 1999 a engendré une nécessaire restructuration des services internes, notamment au niveau de l'entité chargée du traitement quotidien du courrier. Dès lors, BNP Paribas a décidé de réorganiser et de moderniser cette fonction, en recourant à un système d'automatisation du traitement du courrier. L'objectif de ce vaste projet consistait à intégrer ce système de tri automatisé dans le système manuel déjà existant. Après une vaste étude de marché notamment auprès de banques anglaises et allemandes, la Direction des Services Généraux de BNP Paribas a retenu le projet proposé par l'intégrateur français CESA (Conseil, Étude, Systèmes Automatisés) qui participe en même temps, à la modernisation des systèmes de tri de La Poste. Pour ce faire, BNP Paribas s'est doté d'une machine Vsort NPI (dont Prolistic est le revendeur exclusif en Europe) de 30 mètres de long et contenant 120 cases de tri, entièrement modulable et packagée pour répondre à ses besoins et intégrant la solution A2IA AddressReader. En termes de cadence (de 17 à 40 plis/seconde) et de type de documents plats à traiter (enveloppes, magazines, listings, planus…), la solution globale proposée par CESA, Prolistic et A2iA répondait en tout point, au cahier des charges de BNP Paribas, elle permettait l'automatisation de la machine de tri manuelle en place depuis 1986, dans le processus de traitement automatique du courrier et de documents. Le système OCR existant a été optimisé par l'intégration du moteur A2iA AddressReader.

Toute automatisation de système de tri repose sur des technologies de vision industrielle. L'innovation de ces technologies est un procédé capital qui conduit à un gain de temps et d'argent important pour les entre-

prises. Il peut sensiblement améliorer leur réactivité vis-à-vis de leurs clients et diviser par trois les coûts de traitements traditionnels.

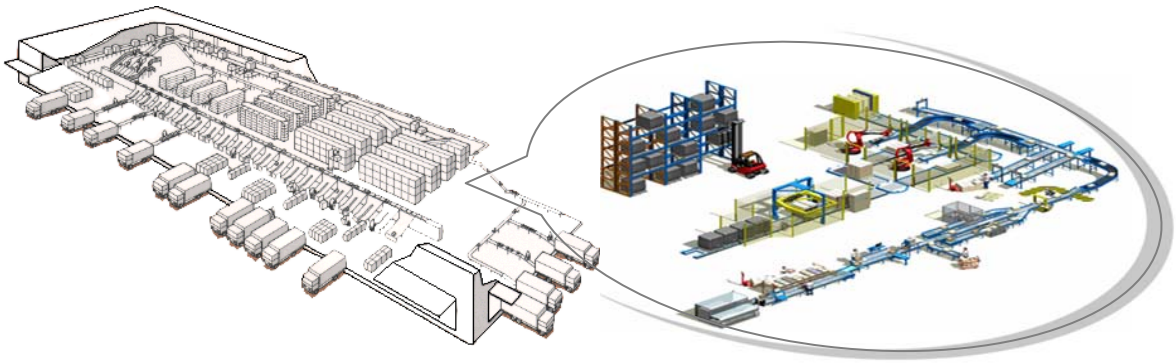


Figure 1.3 : Un exemple d'un centre de tri automatique de documents et de courrier.

Les atouts des solutions de LAD et de RAD sont multiples et couvrent aujourd'hui de nombreuses applications dans le domaine de tri automatique de documents et de courriers d'entreprise. Les tendances les plus visibles aujourd'hui se développent dans l'amélioration des systèmes industriels de vision où la performance a longtemps été associée à celle de l'OCR. Mais de notables évolutions technologiques ont permis d'affiner la précision et la pertinence de la recherche de l'information nécessaire à la reconnaissance permettant non seulement de prendre en compte des écritures différentes (imprimées ou manuscrites) mais aussi de procéder à une lecture intelligente du document qui conduit vers un tri plus efficace.

Dans ce chapitre, nous présentons les différentes briques logicielles nécessaires à la conception des systèmes de tri automatique de documents et de courriers. Nous évoquerons leurs spécificités et leurs limites.

1.2 Les architectures « standards » de tri de courriers

1.2.1 Les contraintes inhérentes au tri de courriers

Le domaine de tri automatique de documents et de courrier est assujetti à un certain nombre de contraintes. Au sein de la société, une évaluation préliminaire du système de tri actuel, effectuée en premier temps, nous a permis de dégager les limites et de fixer un certain nombre de contraintes qu'il conviendra de prendre en considération dans nos recherches :

- une très grande variété de documents de structures variables avec des contenus textuels manuscrits et/ou imprimés, sur des supports papiers de qualités, de couleurs et de textures souvent différentes. Les images à traiter sont réparties en catégories correspondantes aux familles de courriers des clients de l'entreprise : courrier postal, courrier interne manuscrit (CIM), courrier interne dactylographique (CID), formulaire (FRM), planus (PL), carte bleue (CB), listing A3(LA3), listing A4(LA4), NPAI, chèque circulant (CHC)… Ces images sont très différentes du point de vue de leur taille, de leur orientation, des couleurs du fond et du texte, de la position du texte dans l'image, de la taille des caractères et des types d'écritures (imprimés, imprimés matriciels, manuscrits…). Les documents sont traités par lots ou bien arrivent en vrac,
- un fonctionnement en temps réel : quelques fractions de secondes doivent suffire pour faire l'acquisition de l'image, la binarisation, la localisation des zones de textes,
- la capture des images par système de caméra linéaire (on devra développer les outils d'analyse d'images lié aux caractéristiques de cette prise d'image pour optimiser les temps de calcul),
- la maîtrise de la qualité des résultats (dans un marché très compétitif, le système doit être le plus performant possible pour éviter les coûteuses interventions manuelles),
- des résolutions spatiales des images élevées (200~300 dpi),
- le type de document doit être identifié automatiquement malgré les aléas de numérisation (rotations, décalages, plissement),
- l'existence d'une superposition de couches d'informations (tampons, notes manuscrites, …),
- les documents non reconnus sont immédiatement traités manuellement. L'échec de reconnaissance s'explique généralement par un dysfonctionnement des étages de prétraitements et en particulier des étages de segmentation et de localisation [GOR98][GOR99].

Ces contraintes fortes impliquent des niveaux d'actions ciblées que nous avons choisis de présenter en trois parties : mécanique, électronique et informatique.

1.2.2 Trois niveaux d'actions

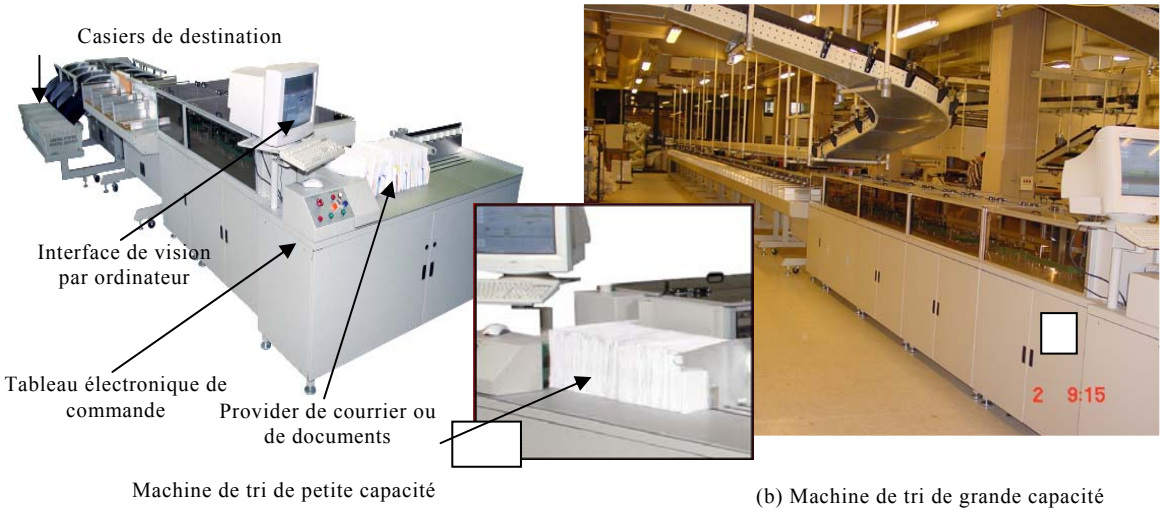
Les systèmes de tri de documents et de courriers se composent en général de trois parties : mécanique, électronique et informatique.

La partie mécanique : est constituée d'un ensemble d'actionneurs et pièces permettant de véhiculer dans une bande transporteuse (ou tapis roulant) tous les documents ou enveloppes physiques depuis leur acquisition jusqu'à leur distribution dans divers casiers de destinations.

La partie électronique : est constituée d'un ensemble de circuits de commandes et de relais qui gère les actionneurs de la partie mécanique (mise en marche, arrêt d'urgence de la machine de tri, etc.).

La partie informatique : la partie informatique est le cerveau de système qui contrôle les deux parties précédentes. Notre contribution s'inscrit dans cette partie :

Après le positionnement des documents (courriers) sur la bande porteuse (tapis roulant), un mécanisme d'entraînement les achemine tout au long d'un circuit jusqu'au casier de destination (emplacement ultime). Le système de lecture localise la zone d'intérêt puis lit les informations inscrites sur chaque objet (adresses, codes ou d'autres indications…) pour décider sa destination. Elles sont comparées avec un référentiel et transformées en un code-barres fluorescent qui est imprimé sur l'objet à trier. Les machines de tri (proprement dites) après lecture du code fluorescent orientent vers différentes cases les objets en fonction de l'information inscrite. Si ces machines n'arrivent pas à lire une adresse, le courrier est retraité manuellement, à l'aide d'un système de rectification par vidéo-codage. Un agent saisit l'adresse présente sur un écran de contrôle. La figure suivante illustre deux exemples de chaînes de tri et leurs différents composants.



Machine de tri de petite capacité

(b) Machine de tri de grande capacité

Figure 1.4 : Exemple des machines de tri de courrier d'entreprise.

1.2.3 Les principaux modules informatiques d'un système de tri

Les systèmes de tri de document et de courrier d'entreprises couvrent aujourd'hui toute la chaîne associée à la vision industrielle depuis la phase de l'acquisition jusqu'à la phase de prise de décision de tri en passant par les traitements et les analyses spécifiques de chaque solution. Nous montrons ci-dessous un exemple d'architecture informatique des systèmes de tri de documents et de courriers d'entreprise qui se compose de cinq modules (voir figure 1.5).

Cette architecture a été choisie car elle synthétise l'ensemble des mécanismes intervenant sur une chaîne de tri. Elle est également exemplaire car elle repose sur la grande linéarité des processus qui lui est inhérente.

PC Vision : Le PC Vision récupère l'image transmise par la caméra CCD linéaire sur un port Ethernet Giga Bit. Après traitement (binarisation, extraction de la structure physique), l'image optimisée est communiquée au PC OCR et au serveur de vidéo-codage depuis l'autre port Ethernet. La fonction d'acquisition d'image implémente le protocole de communication avec la caméra CDD. En parallèle de l'acquisition d'image, le module de compression comprime l'image au taux permettant la visualisation et le stockage. Les normes de compression les plus couramment utilisées sont le format TIFF et la compression CCITT T6 pour les images binaires, la compression JPEG pour les images en niveaux de gris alors que des principes de compression plus récents, tels que DjVu [BOT00], restent confidentiels. En parallèle de l'acquisition d'image, le système binarise puis il localise la zone d'intérêt (adresse). Cette opération est couplée à la segmentation en ligne du bloc adresse. Chaque ligne du bloc adresse est nettoyée, redressée. En fin la zone d'intérêt (adresse) ainsi que les positions des lignes de texte sont envoyées vers le PC OCR.

PC OCR : Le PC OCR reçoit l'image traitée par le PC Vision sur un port Ethernet. Chaque ligne du bloc adresse est transmise aux fonctions de reconnaissance de caractère spécialement paramétrées pour lecture du code postal et de la ville. Sur chaque réponse acceptable du traitement OCR (Découverte d'un code postal et d'une ville), une analyse sémantique du compte rendu est réalisée afin d'associer le code postal et la ville. A la fin de l'analyse de la zone d'adresse, un compte rendu est retourné au PC Vision pour qu'une décision sur la destination de l'objet soit prise.

Serveur Vidéo-codage : Le serveur Vidéo-codage reçoit l'image optimisée par le PC Vision sur un port Ethernet. En fonction de l'occupation de la station de vidéo-codage, le serveur Vidéo-codage transmet les images à vidéo-coder. Si aucune destination n'est trouvée l'image de l'objet est disponible pour la saisie assistée par l'image. Une fois que l'image est réalisée, un compte rendu est retourné au PC Vision pour qu'une décision sur la destination de l'objet soit prise.

Station Vidéo-codage : La station Vidéo-codage reçoit l'image optimisée par le serveur Vidéo-codage sur un port Ethernet. Chaque station de vidéo-codage permet l'encodage d'une destination ou d'une information suffisante pour qu'une décision soit prise sur la destination de l'objet rejeté. Depuis l'interface opérateur de la station de vidéo-codage, il est possible de réaliser les opérations de saisies assistées et de consultation d'image. Diverses fonctions d'affichage sont disponibles pour permettre à l'utilisateur de déchiffrer l'adresse de destination de l'objet (zoom, rotation, amélioration de contraste). Toutes ces fonctions sont accessibles depuis le clavier sous forme de combinaisons de touches programmables. Un espace de saisie permet l'encodage du code postal ou de la ville (Assistance d'un dictionnaire).

Contrôle commande trieur : Système en charge du pilotage du trieur.

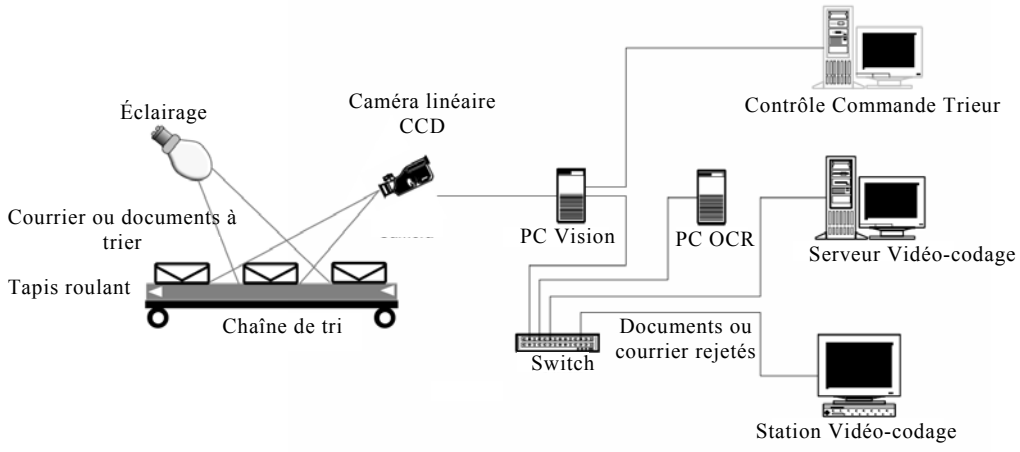


Figure 1.5 : Architecture informatique générale des systèmes de tri de documents et de courrier d'entreprises.

1.3 Systèmes de tri : de l'acquisition à la décision

1.3.1 Linéarité des processus d'analyse et de reconnaissance

Un système de vision industrielle pour le tri automatique de courrier et de documents fonctionne en trois phases. Parmi elles, on distingue :

1) l'acquisition,

2) le traitement, l'analyse, la localisation et la reconnaissance de contenu des images de documents (phase très importante sur laquelle va porter notre étude),

3) la reconnaissance optique, l'interprétation contextuelle et la décision de tri.

A l'issue de ces trois phases les documents à trier sont acheminés aux bons casiers de destinations.

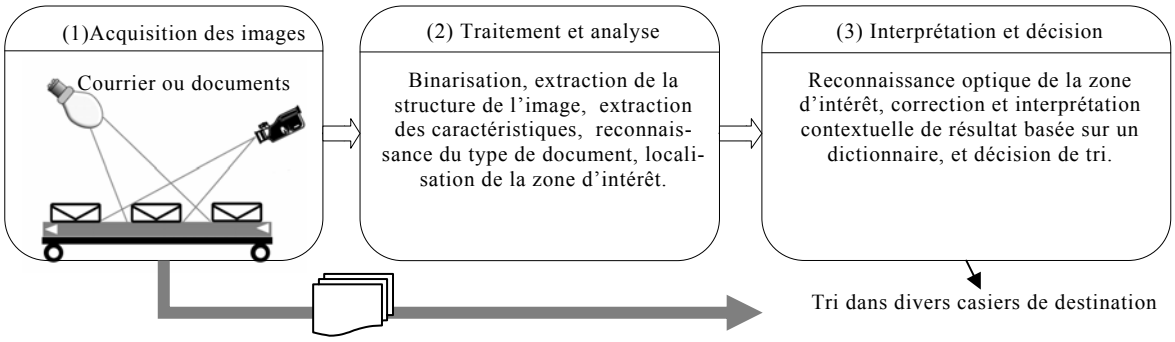


Figure 1.6 : Les trois phases d'un système de vision industrielle dans une chaîne de tri automatique de documents et de courrier.

La particularité de ces architectures repose sur leur très grande linéarité. Dans ce type d'architecture, les étapes d'extraction de la structure physique et la reconnaissance ne coopèrent pas (l'une précède l'autre). Les boucles de rétroaction, et la vérification de cohérence des résultats de reconnaissance ne sont en principe que très peu exploitées. La figure 1.7 ci-dessous synthétise la séquentialité des opérations engagées tout au long de la chaîne des traitements.

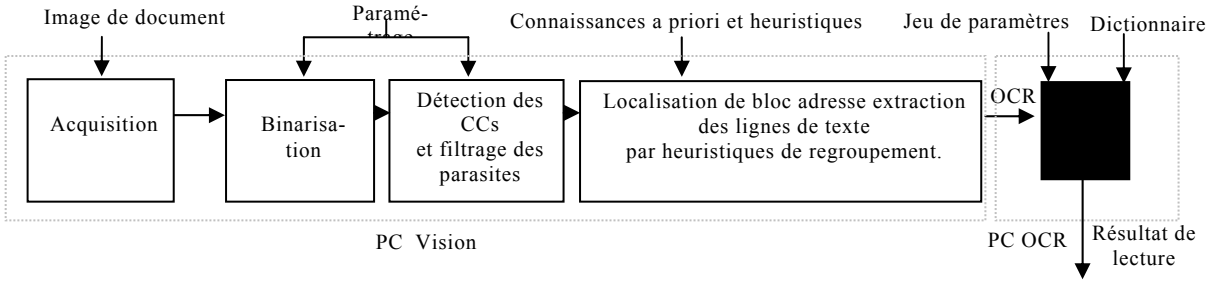


Figure 1.7 : Représentation de l'organisation linéaire des modules de traitement.

L'acquisition des images est liée au matériel de capture, aux conditions d'éclairage et à un ensemble de paramètres à calibrer. Ceux-ci nécessitent des connaissances métier très étendues et une véritable expérience de terrain pour garantir des prises de vues correctes minimisant l'impact du bruit (défaut d'éclairage, mauvais positionnement des images, défaut de cadrage…). Cette partie produit les supports images qui devront subir différents niveaux de traitements pour conduire la reconnaissance des zones d'intérêt et fournir la décision de tri. La contribution des techniques d'analyse et d'interprétation des images est donc de la première importance dans un système de tri automatisé.

Ces parties logicielles sont généralement conçues sous forme modulaire et de façon séquentielle. La figure ci-dessous illustre le principe général d'une architecture de tri automatique de courriers. Les étapes dites de « bas niveau » correspondant à une extraction d'information brute sont grisées sur la figure 1.8. Les étapes de reconnaissance (en encadré pointillé) regroupent les processus de lecture optique du texte de la zone d'intérêt et les corrections éventuelles des résultats de lecture par l'intermédiaire d'un dictionnaire. Elles aboutissent à la décision de tri permettant d'envoyer les documents vers les bons casiers de destination.

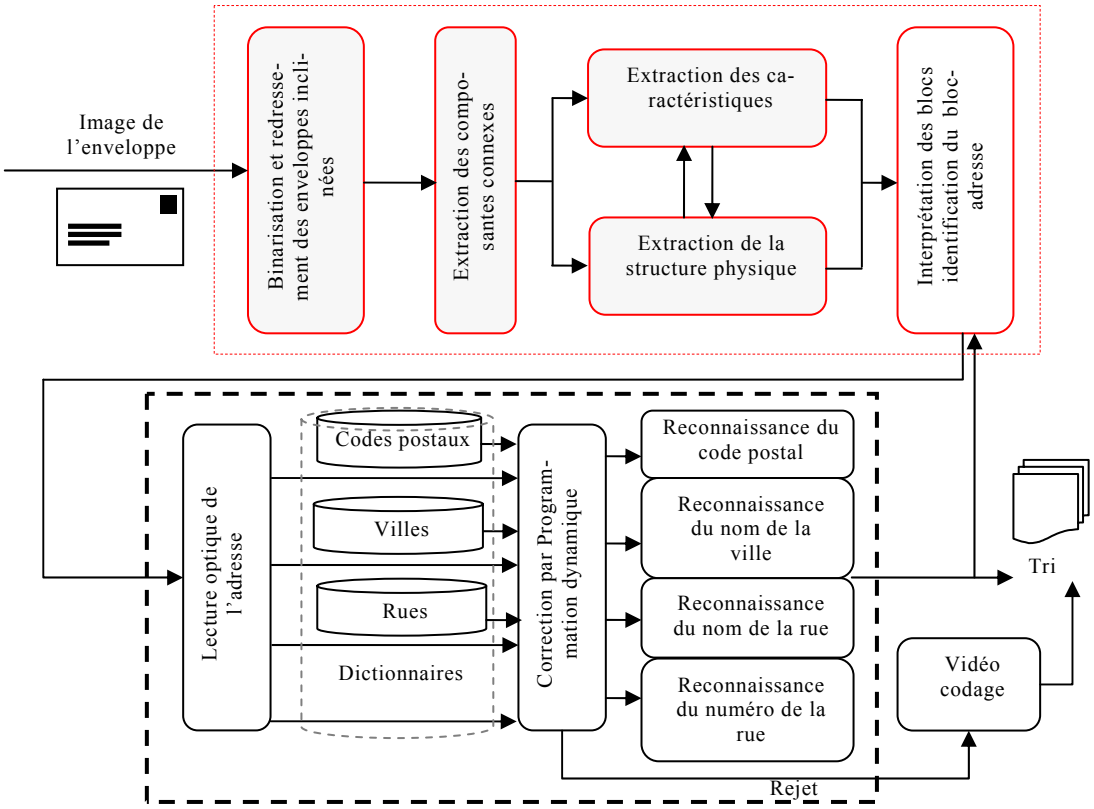


Figure 1.8. Architecture modulaire d'un système de tri de courriers.

1.3.2 L'acquisition des images de courriers

La phase d'acquisition de l'image de courrier ou des documents à trier est importante dans un système de tri automatique. Il s'agit d'une étape importante car, bien réalisée, l'acquisition permet de simplifier les étapes de traitement et d'analyse. Les choix de l'éclairage, de l'optique, de la caméra linaire CCD[1] à haute vitesse et de la carte d'acquisition sont cruciaux pour cette étape. L'étape d'acquisition est fortement dépendante des performances et de la rapidité des composants électroniques. A ce jour, les technologies inhérentes à cette étape sont bien connues. Nous ne rentrerons pas dans les détails techniques de ces éléments.

L'éclairage est important car il permet de fiabiliser et de simplifier le traitement et l'analyse (plus il est optimisé, plus l'analyse est facile). Ainsi, le choix du spectre permet d'optimiser le contraste dans les images en choisissant une couleur particulièrement révélatrice de l'objet. Mais

1. CCD : Charge Coupled Device.

d'autres paramètres interviennent : la puissance, la géométrie, etc. De plus, un éclairage mal choisi peut causer beaucoup de dégradations sur l'image (reflets, interférences, ombres, faible dynamique…). Les caractéristiques de la source lumineuse (longueur d'onde, direction, distribution spatiale) sont choisies en fonction de la surface de l'objet à trier (structure, transparence, couleur,…), ainsi que des caractéristiques de la caméra (qualité de l'optique, ouverture, sensibilité du capteur, gain, vitesse d'obturation,…). La définition d'un mode d'éclairage nécessite la définition du champ de vision ; le niveau de réflectivité de l'objet ; la géométrie de la surface en arrière plan. L'éclairage doit en premier lieu avoir les caractéristiques suivantes : luminance et chrominance stables au cours du temps et être homogène sur tout le champ de vision. On dispose d'une variété importante de sources lumineuses : éclairage incandescent, éclairage halogène (l'intensité lumineuse est quasiment constante, généralement n'est pas utilisé comme éclairage direct, mais plutôt comme source lumineuse pour des fibres optiques), lampes à arc, diodes électroluminescentes (elles sont fiables et de faible encombrement, sont souvent arrangées en matrices ou en anneaux ou associées à des fibres optiques), fibre optique (fournissent des éclairages très localisés et permettent d'ajuster précisément la distribution angulaire de l'intensité lumineuse aux besoins), éclairage stroboscopique (adapté aux scènes à mouvements rapides).

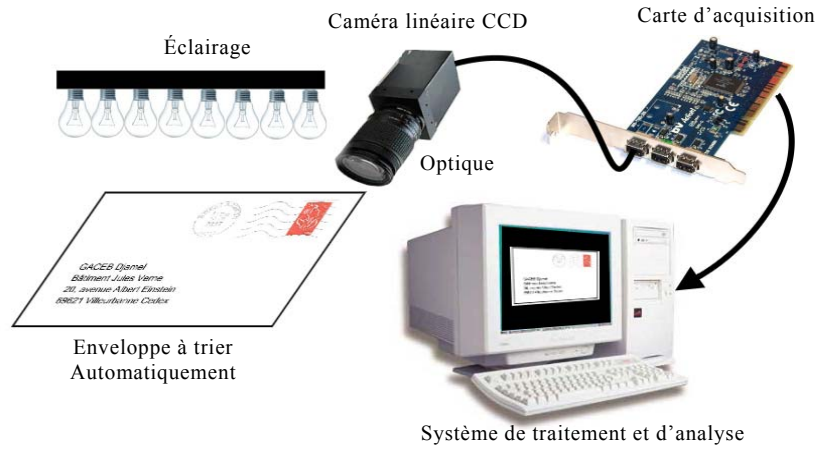


Figure 1.9 : Les composantes de l'acquisition.

L'optique est un objectif qui comporte un système de lentille et un ou plusieurs diaphragmes. Il contrôle la quantité de lumière qui va atteindre le capteur, ainsi que la profondeur de champ : une petite ouverture donne une grande profondeur de champ, mais aussi des effets de diffraction, une

grande ouverture donne des images floues pour les objets non plans (faible profondeur de champ). Les critères de choix d'un objectif sont : sa distance focale (liée au grossissement) ; son angle de champ ; son ouverture, qui caractérise la luminosité de l'objectif ; sa qualité (aberrations géométriques et chromatiques).

Le capteur : Une fois l'éclairage pris en charge, le capteur peut être choisi. Trois grands critères interviennent : sa nature (qui détermine sa sensibilité dans les différentes zones du spectre), son type (linéaire est plus adapté au tri que matriciel) et sa couleur (binaire ou en niveau de gris). Une caméra est dite linéaire lorsque son capteur CCD a une dimension de 1×n capteurs. Attention. Certaines caméras linéaires ont jusqu'à 96 lignes de pixels parallèles et somment les pixels dans le but d'obtenir une meilleur sensibilité et un meilleur rapport signal / bruit. Les caméras linéaires sont largement utilisées en vision industrielle et, plus particulièrement, en tri automatique de courriers et de documents. Elles permettent l'acquisition ligne par ligne d'une enveloppe ou d'un document (défilant séquentiellement devant la caméra sur une bande transporteuse de chaîne de tri). L'avantage de ces caméras linéaires par rapport aux caméras matricielles, c'est qu'elles permettent d'une part d'acquérir une image ayant une luminosité homogène et d'autre part de réaliser des images de grande taille (de grande résolution) pour un coût réduit. La résolution des capteurs est de 300 dpi (respectant le compromis vitesse-mémoire-précision), le plus souvent en niveau de gris. L'installation de la caméra doit respecter les critères suivants :

- elle sera correctement calibrée sinon des lignes de pixels sombres apparaissent,
- elle doit être fixée à une distance adéquate par rapport à l'objet à capter selon sa taille et selon l'optique utilisée,
- la caméra doit avoir une grande vitesse d'acquisition (le nombre de plis triés par minute est une unité de mesure de qualité),
- l'éclairage doit être relativement important car le temps d'exposition est court,
- la caméra doit être bien synchronisée en fonction de la vitesse de déplacement de la chaîne transporteuse qui véhicule les objets. Un système de déclenchement à grande précision des prises de vue pour chaque ligne doit être utilisé,

Pour améliorer le service de tri, il est indispensable d'améliorer la deuxième phase du système de vision : la phase de traitement et d'analyse. Cette phase constitue une étape logicielle toujours perfectible et sur laquelle reposent principalement les performances temps réel. Ces traitements s'amorcent généralement par des opérations d'amélioration de la qualité des images lorsque c'est nécessaire (redressement, suppression du

bruit…) afin de conforter la réussite des techniques de lecture optique (OCR). Ils s'accompagnent également de modules d'extraction de régions d'intérêt et de caractérisations des contenus pour pouvoir mieux les localiser et les identifier. Une connaissance complète des contenus est finalement nécessaire pour fournir la décision finale de tri : ce stade de reconnaissance correspond à la troisième phase du système (voir figure 1.6).

1.3.3 L'analyse des contenus avant lecture optique (OCR)

La seconde phase d'un système de vision industrielle est considérée comme le passage obligatoire qui relie l'acquisition à la décision de tri. Elle porte sur plusieurs niveaux de traitements et d'analyse qui, comme nous l'avons souligné, elle exerce une grande influence sur la vitesse et les performances du système pris dans son ensemble. Ces étapes seront détaillées dans les chapitres 2 et 3 à travers la présentation des différentes approches d'analyse existantes dans la littérature du domaine. La contribution principale de cette thèse s'inscrit précisément dans ce contexte. Nous avons donc choisi de présenter ici les définitions essentielles et qualitatives relatives à ces mécanismes permettant de faire la distinction entre les prétraitements, la caractérisation et l'analyse des contenus, et les phases décisionnelles et de reconnaissance. Les contenus seront présentés selon deux points de vue :

- leur nature qu'elle soit textuelle ou graphique, impliquant des caractéristiques physiques et des spécificités propres aux cas des images de courriers
- leur organisation sur la page qui nous conduit à considérer les diverses notions de structures inhérentes aux documents (physique, fonctionnelle, logique)

1.3.3.1 Spécificité des éléments de contenus des courriers d'entreprise

Les documents d'entreprise et plus spécifiquement les courriers ont des contenus très spécifiques où se mêlent les données textuelles en petit nombre (adresse de destination, parfois adresse de l'expéditeur, données publicitaires…) et les données graphiques (logos, tampons, vignettes et illustrations). Une particularité de ces documents est qu'ils contiennent tout à la fois des données imprimées et manuscrites. Dans la catégorie des documents (courriers) manuscrits, on distingue les manuscrits réguliers contraints (enveloppes précasées où l'écriture est guidée pour les adresses de courrier) des manuscrits irréguliers où la zone d'adresse peut apparaître un peu n'importe où. Indépendamment de la régularité de l'écriture, on tient compte de la difficulté supplémentaire apportée par le grand nombre de

scripteurs différents. Même si le vocabulaire est contraint et naturellement limité sur ce type de documents, c'est avant tout la variabilité des styles d'écritures et des mises en pages qui importent. Pour les courriers imprimés, c'est sans doute la richesse des styles typographiques et des polices de caractères employés qui constitue la principale caractéristique. Les documents mixtes, englobant à la fois de l'imprimé et du manuscrit, nécessitent des prétraitements pour différencier les différents types de texte et y adapter les modèles de segmentation.

Parallèlement à la nature hétérogène des contenus, les courriers d'entreprise possèdent des mises en page de complexité variable. Ces courriers sont, comme nous l'avons signalé, des documents composites incluant à la fois du texte et des images. La capture de l'organisation spatiale de ces informations sur la page est nécessaire à l'interprétation reconnaissance des structures. Et à ce stade il est nécessaire de disposer des informations de qualité optimale (en termes de reproduction et de quantité d'informations, donc de résolution) pour conduire une analyse pertinente. La résolution de l'image est un facteur important qui doit favoriser une séparation correcte des différentes composantes de l'image quand elle est déjà binaire. Dans les systèmes de vision actuelle, l'acquisition se réalise en niveaux de gris autorisant du même coup des résolutions plus faibles et donc des volumes de stockages plus petits. Même avec une résolution très élevée, une image déjà binarisée n'offre pas toujours suffisamment d'informations pour pouvoir séparer les objets différents qui se sont connectés. En revanche, l'information sur les nuances de gris permet d'améliorer la segmentation des images, même en basse et moyenne résolution. Enfin, la qualité des images en termes de bruit (perturbation du signal d'entrée dans la chaîne d'acquisition) et de qualité du support lui-même (tâches, déchirures, inclinaison du papier, etc.) est un facteur important à prendre en considération dans les performances d'une chaîne d'extraction des structures.

1.3.3.2 Les différents niveaux de structuration des contenus

Un document peut-être vu comme une disposition aléatoire d'objets graphiques et textuels de toutes sortes qui se manifestent sur 3 niveaux de structure selon Doermann dans [DOE98] qui propose une distinction fine entre les trois niveaux de structures suivants : la structure physique, la structure fonctionnelle intermédiaire et en fin la structure logique.

La structure physique : La structure physique d'un document se manifeste sur le plan matériel par la mise en page, correspond à l'organisation du document en regroupements de blocs géométriques, qu'ils soient textuels ou non. L'extraction de ces blocs constituant la structure

physique du document est appelée segmentation. Cette étape de traitement permet en particulier de séparer le texte des images. Elle permet ensuite de segmenter le texte selon des critères de proximité et d'alignement, en blocs de différents niveaux caractères, mots, lignes, etc.

La structure fonctionnelle : La structure fonctionnelle exprime la correspondance entre composants physiques et entités logiques. Le but est de représenter la manière dont le document transmet son contenu au lecteur. C'est à ce niveau que le lecteur peut identifier un bloc situé dans le tiers supérieur du document. C'est non seulement la localisation spatiale du bloc (niveau physique) qui est informative mais le type de contenu (niveau sémantique) qui permet une description fonctionnelle des éléments ; la position des éléments les uns par rapport aux autres attribue des priorités dans la capture de l'information. En particulier, certains documents administratifs possèdent des structures très spécifiques (localisation normalisée de l'adresse expéditeur, de l'adresse destinataire, de l'objet, de la date) qui rendent la structure fonctionnelle complètement dépendante de l'organisation physique des éléments.

La structure logique : La structure logique décrit l'organisation hiérarchique du texte contenu dans un document au moyen d'entités logiques telles que les titres, les notes, les citations, les montants, les numéros de téléphone, les codes à barres, les tableaux, les cellules ou les graphiques. Les entités logiques sont des concepts servant à structurer le message de l'auteur ; en retour, elles servent de repères au lecteur. Cette abstraction offre l'avantage de rendre la description du texte contenu dans un document indépendant de tout support physique.

Dans le cas précis des courriers d'entreprise, si on considère un bloc de texte se trouvant en haut droite d'une enveloppe, sa position, ses dimensions ou encore ses propriétés géométriques constituent ses caractéristiques au niveau physique. Celles-ci permettent de lui donner, indépendamment de format de l'enveloppe considérée, une description fonctionnelle et une visibilité particulière au sein de la page. Quant à l'interprétation logique, elle permettrait par exemple de préciser qu'il s'agit d'un bloc représentant les « Frais d'affranchissement du courrier » que l'on trouve généralement en ces endroits.

1.3.3.3 Quels mécanismes et traitements à engager ?

Dans cette partie nous allons distinguer les différents mécanismes nécessaires au traitement (au sens large) des images de courriers. Nous ferons la différence entre les notions de prétraitement, d'analyse et de reconnaissance des contenus.

Le prétraitement regroupe un ensemble de transformations, dont le but est d'améliorer la qualité de l'image de document et de la préparer à l'analyse. Les opérations de prétraitement sont relatives au redressement de l'image, à la suppression du bruit et de l'information redondante, à l'amélioration de contraste, à l'augmentation des caractéristiques de l'objet contenant les informations souhaitées et enfin à la sélection des couches de traitement utiles par binarisation.

L'analyse a pour but l'extraction de l'information caractéristique contenue dans une image de document. Elle regroupe les étapes préparatoires de segmentation (l'extraction des connexités, extraction de la structure physique de documents), d'extraction des caractéristiques. L'objectif de ces premières étapes est de décrire l'importante quantité d'informations contenues dans l'image de document en recherchant des indices visuels ou des primitives pertinentes permettant de la représenter sous une forme plus condensée et facilement exploitable. La performance des systèmes de vision artificielle est tributaire de la qualité de cette représentation. Les différents types de primitives peuvent être définis. On les regroupe généralement sous les dénominations de primitives structurelles, géométriques, statistiques, ou basées sur des transformations globales, comme les corrélations par exemple.

La reconnaissance de contenu liée à la nature des documents a pour but de localiser et identifier la zone d'intérêt nécessaire au tri, elle conduit le plus souvent à l'extraction de la structure logique et à la reconnaissance optique du texte. Cette étape regroupe toutes les opérations de classification de contenu (séparation texte non texte, imprimé /manuscrit), la localisation des zones d'intérêt porteuse de l'information nécessaire au tri (blocs adresses, numéros de série, codes…), la détection de lignes de caractères à reconnaître par l'OCR, la reconnaissance automatique de type de documents. Le niveau de complexité de la reconnaissance dépend de différents paramètres dont la nature des documents à trier et leur structure qui doivent être déterminées à partir d'une extraction de primitives adaptées aux contenus. Le choix des primitives est très important car ce sont elles qui conditionnent les décisions et interprétations. Une revue des caractéristiques généralement recherchées sur les images de documents pour satisfaire une reconnaissance de contenu sera présentée au chapitre 3.

Selon la nature des documents rencontrés, les approches de reconnaissance mises en jeu peuvent être très différentes. Il est donc très important de connaître, au préalable, la catégorie du document sur lequel reposera l'analyse. Les documents d'entreprise peuvent en effet revêtir des formes

extrêmement diverses (formulaires, colis, courrier postal, courrier d'entreprises entrant ou sortant, etc.) et peuvent être caractérisés de façon assez variable suivant la finalité recherchée. De plus, il est bien souvent indispensable de disposer de connaissances spécifiques, telles que des informations portant sur le contenu sémantique du document, ou encore des informations liées à la présentation du document qui apportent des renseignements supplémentaires pouvant servir à sa lecture. L'information de mise en page constitue en particulier une donnée importante pour une bonne localisation de la zone d'intérêt (le bloc adresse) et contient des renseignements précieux sur les données typographiques et la répartition spatiale des zones de texte. Ces connaissances pourront également servir à enrichir un modèle d'apprentissage pour une classification supervisée des documents par exemple. Du point de vue de l'analyse d'images, le nôtre, l'ensemble de ces connaissances peut servir de repère dans le choix des familles d'algorithmes de segmentation ou de reconnaissance à utiliser.

1.3.4 Les mécanismes de lecture optique

Les mécanismes de lecture optique des zones de texte sont sans doute les plus déterminants dans un système de tri. Les moteurs de reconnaissance sont généralement présentés comme des boîtes noires rendues inaccessibles pour des raisons évidentes de confidentialité industrielle. Ces moteurs atteignent la plupart du temps des taux de reconnaissance très élevés et même les erreurs de reconnaissances commises peuvent être supprimées par l'utilisation de corrections contextuelles basées sur les dictionnaires embarqués. La plus grande partie des rejets de courrier a une cause externe à ces moteurs, elle est liée à un dysfonctionnement d'un niveau de traitement amont ou d'un problème d'acquisition. Ainsi, on conçoit mieux que l'amélioration des performances et l'augmentation de la vitesse des systèmes de tri sont à étudier au niveau des étapes de traitement et d'analyse. Nous avons donc fait le choix de n'évoquer que les principes généraux de moteurs de reconnaissance optique (OCR, ICR…) sans en détailler les mécanismes sous-jacents, la plupart du temps confidentiels.

La reconnaissance optique de texte des zones d'intérêt est la technique qui permet de transformer un texte imprimé ou manuscrit (en mode analogique) en un texte numérique, composé de caractères ASCII, et non plus de pixels (texte analogique). Il s'agit donc de la phase complémentaire à celle de localisation de la zone d'intérêt (forcément obligatoire). Il est évident qu'un texte manuscrit aura un taux de reconnaissance très faible, même s'il est très bien écrit. Les systèmes de reconnaissance optique de texte ont acquis aussi de l'intelligence artificielle en ayant d'une part des

dictionnaires syntaxiques et grammaticaux qui contrôlent la cohérence de leur lecture, et d'autre part la possibilité d'exploitation des typographies qu'ils lisent fréquemment. Ces moteurs seront donc d'autant plus efficaces qu'ils liront un grand nombre de documents typographiés de façon identique, après une certaine période d'apprentissage.

Il faut garder à l'esprit qu'une reconnaissance à 99% représente environ une erreur de lecture de caractères sur 3 ou 4 zones d'adresse. L'article de [MOR92] contient une très bonne présentation historique de l'OCR. Il signale que le premier brevet sur ce concept est allemand et date de 1929. La première machine commercialisée (UNIVAC I) a été installée au bureau des statistiques américaines en 1951. L'OCR est utilisé dans les applications postales depuis 1970. Aujourd'hui, et malgré d'énormes progrès, on ne peut toujours pas clore la question de la lecture omni-texte. L'utilisateur d'un système OCR multi-police qui différencie certains styles (gras italique, souligné) et restitue des mises en page, ne sera pas forcément gêné par les limitations actuelles. Les performances vont évidemment dépendre aussi de la qualité du document. Comme les performances des meilleurs OCR sont très proches sur les documents de bonne qualité, c'est sur des documents dégradés que l'enjeu est le plus grand.

1.3.4.1 La reconnaissance des caractères imprimés par OCR[2]

La reconnaissance des caractères imprimés [MOR92], [NAG92], [SRI95] est une technique aboutie dès lors qu'elle s'applique à des cas simples correspondant à des fontes connues, si possible à espacement fixe et sans ligature, imprimées avec une qualité satisfaisante. Sous ces conditions favorables, des taux de reconnaissance individuelle de plus de 99,9 % peuvent être atteints. Dès que l'une ou l'autre de ces conditions n'est pas remplie, cette reconnaissance peut s'avérer difficile : il faut pouvoir séparer les caractères jointifs, reconnaître des caractères individuels déformés. Les techniques de segmentation / reconnaissance sont alors incontournables. Elles permettent, au moyen d'une boucle de rétroaction, de mener de pair la segmentation et la reconnaissance des caractères en vérifiant des hypothèses de segmentation par la reconnaissance des candidats caractères ainsi isolés [ELM96].

2. OCR : Optical Character Recognition.

1.3.4.2 La reconnaissance intelligente des caractères ICR[3]

La reconnaissance intelligente de caractères connue sous l'acronyme ICR s'adresse essentiellement aux écritures manuscrites, voire aux textes imprimés très dégradés. Cette technologie a un mécanisme d'apprentissage de nouveaux caractères qui permet au moteur d'ICR d'améliorer les performances lors des reconnaissances qui suivent. Autrement dit, si un caractère manuscrit qui représente un caractère *"V"* est identifié difficilement, il sera possible d'apprendre au moteur ICR qu'il s'agit d'un *"V"*. Lorsque de nouveau une matrice représentant potentiellement un *"V"* se présentera, le système utilisera sa base de caractères enrichie par apprentissage pour en déduire que la matrice correspond au caractère *"V"*. Ce cas est le plus fréquent dans les applications industrielles, notamment celles de traitement de formulaires. L'ICR est également associé à des règles permettant au moteur de prendre des décisions en cas de doute.

Les techniques de reconnaissance des caractères les plus souvent employées dans les applications industrielles se décomposent en deux étapes essentielles :

1) une étape d'extraction de caractéristiques invariantes au changement de styles et de typographie. Les techniques d'extraction de caractéristiques sont variées et généralement empiriques [TRI96]. Beaucoup constituent le savoir-faire caché et protégé des industriels.

2) une étape de classification automatique de signes à reconnaître dans les différentes classes possibles (chiffres, lettres). Les techniques relèvent souvent d'approches de type connexionniste ou neuronal comme la méthode de perceptron multicouche qui permet un apprentissage automatique par classification à partir d'exemples. Récemment, la technique des machines à vecteur support [VUU03] a bénéficié d'un essor rapide en offrant un apprentissage incrémental automatique (des nouveaux exemples peuvent être intégrés à l'apprentissage sans reconduire celui-ci en totalité). De nos jours, les applications industrielles font coopérer plusieurs moteurs de reconnaissance pour reconnaître une même forme. Cette approche de reconnaissance repose sur l'idée que plusieurs moteurs développés indépendamment présentent des caractéristiques complémentaires, et peuvent donc se compléter avec une combinaison efficace de leurs résultats. Différentes

3. ICR : Intelligent Character Recognition.

techniques de combinaison de classificateurs ont été employées à cet effet [RAH03] [BEL03].

Pour des raisons pratiques (temps de calcul, nombre d'exemples à fournir avec les classes de références), les moteurs de reconnaissance industriels ont une capacité limitée qui interdit toute progression de leurs performances au-delà d'une certaine quantité d'exemples.

1.3.4.3 La reconnaissance de l'écriture non contrainte

Ces méthodes sont appliquées pour reconnaître des mots manuscrits et des écritures cursives utilisant des modèles de Markov cachés (HMM), elles s'adressent notamment aux applications de vocabulaire restreint (reconnaissance des adresses postales). Les performances de ces méthodes dépendent naturellement de la taille du dictionnaire utilisé. Pour un dictionnaire de taille 1000 communes, plus de 90 % des mots manuscrits sont reconnus correctement (le reste est erroné) [GIL95].

1.3.4.4 La correction des erreurs de lecture optique des caractères

Durant la lecture optique des caractères on peut rencontrer principalement quatre types d'erreur :

1) Une confusion, en remplaçant un caractère par un autre, si les caractères ont des formes quasi similaires (par exemple : *« o, 0 », « c, (», « n, h », « s, 5 », « 1, I, l »*) ou les lettres sont accentuées.

2) Une suppression, en ignorant un caractère, considéré comme un bruit de l'image,

3) Un rejet, en refusant un caractère soit parce qu'il n'est pas connu par le système, soit parce que le système n'est pas sûr de sa reconnaissance ; dans ce cas, le système propose un caractère spécial, en général le ~ car celui-ci apparaît rarement dans les documents papier.

4) Un ajout, en dédoublant un caractère par deux autres dont la morphologie de leurs formes accolées peut être proche du caractère (par exemple : *« m, rn», « d,cl », « w,vv »).*

La correction des erreurs de lecture optique des caractères, appelées aussi (post-traitement ou correction contextuelle), cette opération est effectuée quand le processus de reconnaissance aboutit à la génération d'une liste de lettres ou de mots possibles, éventuellement classés par ordre décroissant de vraisemblance. Le but principal est d'améliorer le taux de reconnaissance en faisant des corrections orthographiques ou morphologiques à l'aide des probabilités d'occurrence de bi-gramme, de tri-gramme ou de n-grammes tiré de dictionnaires (liste des noms de villes, noms de rues, codes postaux...). De plus, chaque caractère corrigé pourra être retenu dans une

base de données. Ainsi, la base de données sera de plus en plus complète et on arrivera, à terme, à un OCR ayant des résultats de très bonne facture [HOC93] [GEN99] [BEN99] [KLA02] [STR03] [TAG04] [RIN05].

1.3.5 Les limites des systèmes actuels de vision

Les architectures logicielles existantes dans le domaine du tri de courrier sont essentiellement linéaires. Les limites atteintes par les systèmes de vision actuels sont dues à cette organisation du traitement de l'information (figure 1.7). Les temps de traitement, le taux de rejet et le taux d'erreur des systèmes industriels sont élevés à cause de l'indépendance des processus engagés dans la reconnaissance.

Cette séparation des processus est adaptée à la répartition des tâches sur plusieurs ordinateurs connectés, mais l'échec d'une seule étape du processus conduit irrémédiablement le système à rejeter ou bien à commettre une erreur d'interprétation. Certains travaux font déjà référence à des architectures plus avancées. [MIL96] propose un système multi-agents pour l'échange des données et la collaboration entre les différents modules d'acquisition et de reconnaissance. [SRI97] décrit une architecture coopérative des différents modules pour reconnaître les adresses et les codes postaux. [LU99] décrit une approche probabiliste pour combiner la localisation, la segmentation et la reconnaissance. Enfin, [ZHO02] décrit une segmentation en mots dirigée par l'étage de la reconnaissance.

1.4 Conclusion

L'efficacité du tri dépend pour une grande part de la facilité d'accès à l'information. Pour répondre à cet enjeu, les processus industriels mis en œuvre proposent des solutions dédiées à la capture, à l'identification automatique des documents entrants sur la chaîne de tri. Courriers, factures, commandes, chèques, contrats, formulaires, candidatures, publicités, sont autant de documents qui une fois scannés sont identifiés de manière automatique pour définir leurs destinataires.

Il devient ainsi primordial, au sein d'une telle hétérogénéité de contenus, de déterminer le meilleur scénario d'enchaînement des technologies (voting process) qui doivent être complémentaires et adaptables aux caractéristiques du flux documentaire entrant. Cette stratégie actuellement mise en œuvre par certains industriels soucieux de performances et de rapidité, doit permettre de faire coopérer tout un ensemble de technologies, afin de générer le meilleur diagnostic d'identification.

A ce jour, nous pouvons faire le constat que la majorité des technologies d'identification est encore régie par un fonctionnement très linéaire et séquentiel n'exploitant que très rarement une coopération interprocessus pourtant très prometteuse et trop peu souvent les ressources d'un fonctionnement en apprentissage automatique. Apprendre seul les nouveaux documents sur la seule base d'échantillons d'apprentissage mis à la disposition du système semble pourtant une voie ouverte très intéressante. Et c'est celle que nous avons choisi d'explorer. Les performances du système sont donc directement liées à la combinaison et la complémentarité d'approches technologiques et de ressources logicielles diverses, du niveau le plus bas (extraction de caractéristiques de présentation et de contenu) aux niveaux les plus élevés (reconnaissance de forme, recherche de régions d'intérêt localisées, analyse et interprétation du contenu…). La prise de décision issue d'une plus grande diversité de technologies (d'un bout à l'autre de la chaîne : de la binarisation à la reconnaissance) organisées en une véritable coopération de modules devraient permettre de multiplier les chances de succès de l'identification et donc d'augmenter sensiblement les taux d'automatisation du tri du courrier. Nous avons choisi de nous inscrire dans une telle voie. Voici donc, dans les chapitres suivants, la démonstration que nous proposons.

Chapitre 2

Les méthodes essentielles d'accès au contenu

Binarisation et segmentation

2.1 Introduction

Les systèmes de tri automatique de documents et de courriers d'entreprises exigent des algorithmes efficaces et rapides à toutes les étapes de traitements bas niveau ; c'est la condition nécessaire pour pouvoir aborder ensuite l'analyse fine et la reconnaissance des contenus.

Comme nous l'avons vu au premier chapitre, ces systèmes se composent de différents modules qui s'exécutent séquentiellement. Des algorithmes non optimisés conduisent alors à une accumulation de temps perdu non acceptable par ce type de systèmes. De même, le manque de précision d'un module du système pouvait entraîner, en cascade, des erreurs de décision ou des taux de rejet importants.

Nous allons donc rappeler les principales approches existant dans la littérature pour chaque module d'une chaîne de tri et argumenter les raisons de leurs imprécisions ou lenteurs.

Nous présenterons tout d'abord les différents mécanismes de binarisation,qui constitue généralement la première étape de segmentation de ce type de systèmes. Très peu d'approches « sans binarisation » ont réellement pu voir le jour ces dernières années en raison de l'augmentation non négligeable des temps de traitement nécessaires à l'analyse des images en niveaux de gris. Les mécanismes de binarisation sont généralement développés pour répondre aux besoins d'applications spécifiques : on trouve donc de nombreuses adaptations d'approches conventionnelles (Sauvola [SAU97], Niblack [NIB86]…) qui, comme nous le verrons, demeurent malheureusement inadaptées aux contraintes d'exécution temps réel. Avec des temps de calculs prohibitifs, une tendance accrue à la sur-segmentation des défauts et de la texture du support, les mécanismes usuels de binarisation présentent des faiblesses qu'il est indispensable de surmonter. Nous expliquerons les causes de ces insuffisances à travers la présentation de différentes approches globales de binarisation rapides et de différents mécanismes locaux plus adaptées aux changements locaux de contraste mais nécessitant plus de calculs et impliquant donc des temps plus importants.

La binarisation est souvent suivie par une étape de détection des composantes connexes (notées CCs), étape préalable indispensable à une recherche de formes et de régions connexes. Elle est souvent présente lorsqu'il faut regrouper des pixels connexes pour repérer rapidement les caractères, les lignes, les blocs de texte ou toutes autres formes graphiques. L'accomplissement de cette tâche en temps réduit a fait l'objet de nombreuses études qu'on présentera dans la suite de ce chapitre.

Nous aborderons ensuite les mécanismes d'analyse de structures des documents qui sont généralement présentés sous la forme d'approches

ascendantes et descendantes. Nous montrerons en particulier qu'un grand nombre de méthodes de segmentation qui s'appliquent en temps réel sur des documents de type courriers et formulaires s'oriente vers la mise en place d'un mécanisme mixte (mi-ascendant / mi-descendant). Les méthodes descendantes bien que rapides sont moins précises sur des documents de structure complexe ou variable, tandis que les méthodes ascendantes plus précises sont très consommatrices en temps de calcul. La coopération rendue nécessaire entre ces deux mécanismes a permis d'obtenir de bien meilleurs compromis temps / précision. A cela s'ajoute ces dernières années l'introduction de la multi-résolution (hiérarchie de décomposition) et des changements d'espace de représentation.

Dans cette présentation de l'existant en matière de segmentation des images de documents, nous sommes partis de l'hypothèse forte que la reconnaissance du type de document, la séparation texte / non texte des zones de document ou encore la séparation manuscrits / imprimés peut être initiée par une segmentation grossière du document et peut ensuite permettre d'aboutir à une segmentation plus fine des contenus. Comme nous l'avons déjà dit dans l'introduction générale, la reconnaissance et la segmentation constituent deux tâches qui s'influencent mutuellement et dont les interactions permettent des améliorations considérables à la fois de la segmentation et de la reconnaissance. Nous avions déjà reformulé le paradoxe de Sayre [Sayre73] comme suit « *pour reconnaître une entité, il faut savoir la localiser, mais pour la localiser, il faut tout d'abord la reconnaître*».

Dans la suite du chapitre, nous poursuivrons l'analyse des premières étapes de bas niveau par la présentation des mécanismes de séparation entre texte imprimé et texte manuscrit permettant de choisir l'OCR et d'activer les traitements appropriés. La littérature dans ce domaine regorge d'outils permettant de discriminer les deux types de texte. Avant même de procéder à cette distinction, il est nécessaire de procéder en une séparation entre entités textuelles et graphiques. Dans le cas des courriers ou des documents d'entreprise, la structure physique à extraire est souvent composée de deux couches distinctes : une couche textuelle qui comporte l'essentiel de l'information et une couche non textuelle qui peut contenir des graphiques, des tableaux, du bruit, et d'autres informations additionnelles. Cette tâche représente une tâche centrale au sein des chaînes de tri automatique de courriers et de documents d'entreprises car elle doit permettre un accès rapide à l'information nécessaire à la reconnaissance (adresse de destination, codes, zones textuelles d'intérêt sur les formulaires et les documents bancaires).

L'ensemble des contributions citées dans ce chapitre sera finalement argumenté afin de justifier nos choix et directions méthodologiques que les chapitres suivants détailleront et valideront.

2.2 Binarisation des images des documents

L'acquisition des images des documents et des courriers livrés s'effectue la plupart du temps en niveaux de gris. L'analyse par OCR qui constitue une étape clé dans le processus de tri de courrier nécessite une réduction de la quantité d'informations passant par une étape préliminaire incontournable de binarisation qui, à elle seule, a un impact très fort sur les performances de toutes les étapes ultérieures de traitement automatique du document. Binariser une image, c'est convertir une image non adaptée en image binaire adaptée aux traitements ultérieurs (extraction des composantes connexes, extraction de la structure, OCR). Cette étape est un passage irréversible d'une image en niveaux de gris en une image bimodale : elle facilite ainsi la classification entre le fond généralement blanc (arrière plan image du papier) et les objets noirs (traits, graphiques, caractères…).

Le problème serait simple si le niveau de gris associé au fond était uniforme, si le niveau de gris associé aux objets l'était également, et enfin si ces niveaux de gris étaient suffisamment différents pour que, par comparaison avec un seuil supposé connu, on puisse étiqueter blanc tous les pixels de niveau de gris supérieur ou égal à ce seuil et noir tous les pixels de niveau de gris inférieur à ce même seuil. Dans la pratique, cette situation idéale ne se rencontre que très rarement. Les niveaux de gris associés au fond et aux objets présents sur l'image sont supposés être suffisamment différents pour qu'une bonne discrimination puisse être faite. Cependant, cette dichotomie n'est évidemment pas parfaite en raison de défauts d'éclairage ou de bruits introduits par le capteur lui-même. Par conséquent, un mauvais choix d'un seuil de binarisation peut détruire une grande part d'information utile contenue dans l'image en dégradant notamment la qualité des caractères à reconnaître par l'OCR (figure 2.1), ces caractères peuvent ainsi être fragmentés ou fusionnés.

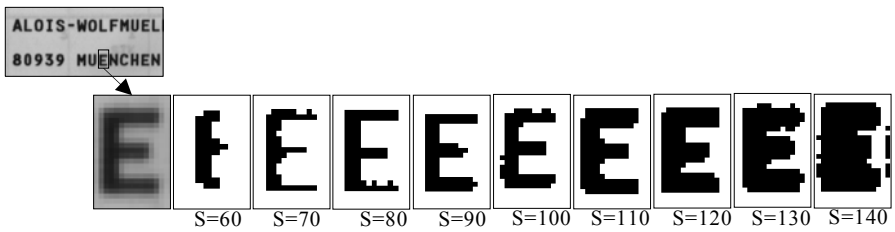


Figure 2. 1 : Effet de seuillage sur la qualité des caractères.

Figure 2. 2 : Résultat de la binarisation de différentes images par le même seuil S=120.

Ainsi, une binarisation appliquée directement sur les images de documents dégradés introduit de nombreux artéfacts qui entraînent des erreurs dans les modules suivants d'analyse. Par conséquent, il est nécessaire d'appliquer des prétraitements de rehaussement du contraste, d'égalisation de l'histogramme et de réduction du bruit par filtrage afin d'améliorer la qualité de cette binarisation. Dans ce contexte on peut citer par exemple les travaux de Ramponi, de Shan et de Fontanot [RAM93], [FON93], [SHA98] qui ont proposé des approches de filtrage quadratique pour améliorer la qualité des images de courrier pour la phase de binarisation (figure 2.3).

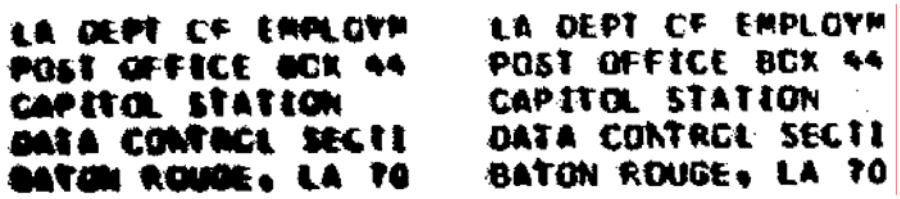


Figure 2. 3 : Exemple d'image binaire (à gauche) sans prétraitement et à droite avec prétraitement (filtrage quadratique).

Cette figure illustre le fait qu'une bonne binarisation doit être capable de conserver à la fois tous les caractères et les objets sans récupérer trop de bruit. Pour résoudre ce problème, nous avons pu relever un très grand nombre de travaux concernant la binarisation des documents. On distingue essentiellement trois catégories de méthodes selon la nature de seuillage utilisé: les méthodes globales, les méthodes locales et les méthodes hybrides qui exploitent les deux approches précédentes. Si on désigne par P un pixel, $I(P)$ son niveau de gris et par $B(P)$ le résultat d'un opérateur local agissant sur un voisinage $V(P)$ du pixel P, le seuillage peut alors être associé à un opérateur $T_1 (P, V, B)$. On parle alors de seuillage global si T ne dépend que de $I(P)$, et de seuillage local si T dépend à la fois de $I(P)$ et de $V(P)$. Le seuillage hybride se définit alors par l'action conjuguée d'une analyse locale dans le voisinage d'un point et d'une analyse globale dans une région plus étendue autour de ce point (voir de l'image toute entière).

2.2.1 Les méthodes de seuillage global

Les méthodes globales déterminent un seul seuil pour toute l'image en partant du point de vue que les objets doivent avoir une distribution des niveaux de gris relativement distincte de la partie fond. Dans ce cas, la recherche de seuil s'effectue par l'analyse de l'histogramme des niveaux de gris et par la détermination d'un minimum local (voir figure 2.4). Les pixels ayant un niveau de gris inférieur au seuil sont mis en noir et les autres en blanc.

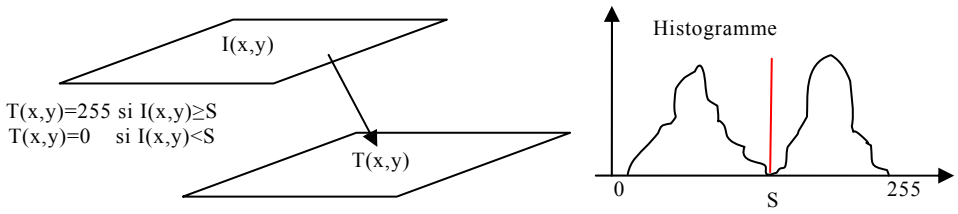


Figure 2. 4 : Principe de la binarisation par seuillage globale.

2.2.1.1 Approches basées sur la séparation de distribution

La localisation du seuil de binarisation se fait par séparation des distributions des niveaux de gris de l'image toute entière. Celle-ci peut être réalisée par une séparation de deux gaussiennes qui modélisent l'histogramme ou à l'aide d'un critère de minimisation de la somme des inerties, associées aux deux classes (objets, fond), [COC95]. La célèbre méthode de Fisher [FIS58] consiste à modéliser l'histogramme bimodal des niveaux de gris d'une image par une somme pondérée de deux distributions gaussiennes et à localiser un seuil vu comme le séparateur des distributions. Pour cela, on utilise un critère de minimisation de la somme des inerties associées aux deux classes de niveau de gris (C_1 et C_2). Cela revient à trouver leurs bornes afin de minimiser l'inertie I donné par l'expression suivante :

$$\sum_{k\in C1} h(k)\times\left[k-G(C_1)\right]^2+\sum_{k\in C2} h(k)\times\left[k-G(C_2)\right]^2 \quad (2.\ 1)$$

G : centre de gravité de la classe, k : niveau de gris, $h\ (k)$: densité du niveau de gris k dans l'histogramme. Un développement de cette expression permet de formuler le problème en terme de maximisation d'une quantité J :

$$J(P)=\frac{\left(\sum_{k\in C1} k\times h(k)\right)^2}{\sum_{k\in C1} h(k)}+\frac{\left(\sum_{k\in C2} k\times h(k)\right)^2}{\sum_{k\in C2} h(k)} \quad (2.\ 2)$$

Le niveau de gris solution de cette maximisation correspond au seuil recherché permettant de distinguer les deux classes. D'autres méthodes consistent à trouver un seuil en séparant l'histogramme en deux classes itérativement avec la connaissance a priori des valeurs associées à

chaque classe comme c'est le cas pour la méthode ISODATA, [COC95]. Une autre méthode très utilisée est celle des nuées dynamiques (K-means) qui consiste à affecter à chaque classe un pixel qui constituera son centre de gravité initial. Chaque pixel de l'image est ensuite affecté à la classe dont le centre de gravité est le plus proche. Les centres de gravité sont à nouveau calculés et le processus continue itérativement jusqu'à ce qu'il ait convergence. Le k-means peut aussi être soit global et appliqué directement à toute l'image soit local et appliqué à chaque fenêtre de l'image dont on choisit la taille [TRE04]. La sérialisation du k-means consiste à initialiser les centres de gravité de chaque fenêtre avec les centres de gravité finaux de la fenêtre précédente. La sérialisation donne des résultats très intéressants mais est très consommatrice en termes de temps de calcul. Elle nécessite de plus une initialisation des centres par l'utilisateur [LEY04].

2.2.1.2 Approches basées sur l'analyse discriminante

Otsu [OTS78] formule le problème de la binarisation comme une analyse discriminante, pour laquelle il utilise une fonction critère particulière comme mesure de séparation statistique. Des statistiques sont calculées pour les deux classes de valeurs d'intensité séparées par un seuil *i*. On calcule les statistiques pour chaque niveau d'intensité *k*, c'est-à-dire pour tous les seuils possibles. Dans le cadre de la binarisation par la méthode d'Otsu, la séparation s'effectue à partir de la moyenne et de la variance. On calcule donc :

$$\mu(i) = \sum_{k=1}^{i} k \times h_{Normalisé}(k) \quad et \quad \omega(i) = \sum_{k=1}^{i} h_{Normalisé}(k) \qquad (2.3)$$

Enfin, on calcule pour chaque valeur de k la valeur :

$$S^2(i) = \omega(i) \times \left[1 - \omega(i)\right] \times \left[\mu(255) \times \omega(i) - \mu(i)\right]^2 \,|\, i = 1...255 \quad (2.4)$$

Le niveau qui maximise la fonction critère est retenu comme seuil de binarisation. Ainsi la valeur du seuil est obtenue pour *i* tel que *S2(i) = max (S2)* pour toute valeur de *i* variant de *1* à *255*. Une variante de cette approche a été proposée par Tsai dans [TSA07] : elle consiste initialement à découper l'image récursivement en quadtree, et à appliquer ensuite un seuillage de type d'Otsu dans chaque bloc. Cette méthode s'adapte bien à la forme des tracés et permet de résoudre les problèmes liés à une distribution non uniforme de l'intensité lumineuse. L'inconvénient de cette approche est son coût calculatoire et le risque de générer des blocs entièrement noirs.

2.2.1.3 Approches basées sur la notion d'entropie

Ces méthodes sont basées sur l'optimisation d'une fonction critère, l'entropie dans ce cas [KAP85], [ESQ02] et [ZHU02].

2.2.1.4 Approches basées sur la transformation d'histogramme

Dans ces approches, le seuil n'est pas sélectionné directement, mais après transformation de l'histogramme des niveaux de gris de l'image. Cette transformation a pour but d'élever les pics et d'abaisser les vallées, en associant à chaque pixel un poids dépendant de ses propriétés locales, ce qui permet de bien discriminer les modes d'histogramme [YAN06].

2.2.1.5 Approches basées sur la matrice de cooccurrences

La matrice de cooccurrences $M(d,\theta)$ est une matrice dont les entrées sont les fréquences relatives aux deux pixels voisins Ng_i et Ng_j, séparés par une distance d avec une orientation θ. KOHLER [KOH81] donne une mesure du contraste en utilisant les matrices de cooccurrences, dont les éléments correspondent à des couples de valeurs de niveaux de gris. Le seuil optimal est déterminé pour un contraste maximal. Zhu dans [ZHU 02] utilise la notion d'entropie sur la matrice de cooccurrences.

2.2.1.6 Approches basées sur les réseaux de neurones

Babaguchi et al. [BAB 90] ont proposé une méthode de binarisation basée sur un modèle connexionniste (*CMB*). Cette technique s'articule sur deux phases (apprentissage et binarisation). Dans la phase d'apprentissage, le réseau utilise l'algorithme de rétro propagation sur toute la base d'images, chacune étant représentée par son histogramme propre et un seuil désiré. Dans la phase de binarisation, le réseau reçoit à l'entrée un histogramme d'une image inconnue et retourne en sortie le seuil optimal de binarisation, voir figure 2.5.

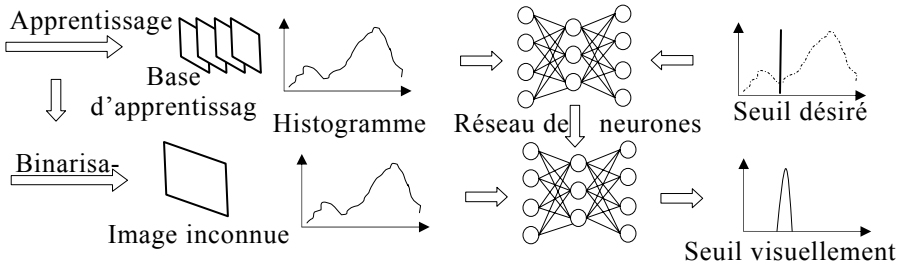


Figure 2. 5 : Principe général de la méthode de binarisation par réseaux de Neurones.

2.2.1.7 Discussion des méthodes de seuillage global

Les méthodes de seuillage global ont l'avantage d'être extrêmement rapides, mais leur gros inconvénient vient du fait qu'on ne tient pas compte des relations spatiales entre pixels d'un objet. Par conséquent, rien ne permet d'assurer que les pixels sélectionnés soient bien adjacents et que

les formes soient effectivement bien isolées de l'arrière plan. En ce sens, des pixels du fonds peuvent assez facilement être intégrés dans les objets et inversement des pixels des objets peuvent être classés en points de fond. Ce phénomène se retrouve particulièrement au voisinage du contour et pour les objets très bruités. Un autre inconvénient fréquemment relevé de la méthode de seuillage global vient d'une illumination non nécessairement constante sur l'image : la variation d'éclairage sur le document fait chuter la qualité de la binarisation en créant des zones entièrement noires une fois l'image binarisée (voir figure 2.6). On doit dans ce cas envisager des approches utilisant un seuillage local : le seuil en tout point de l'image est alors défini comme une fonction de l'illumination dans le voisinage de chaque point.

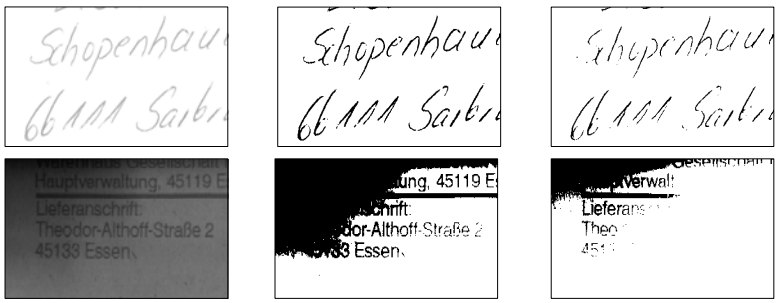


Figure 2. 6 : Seuillage global, à gauche par la méthode de Fisher, à droite par la méthode d'Otsu.

2.2.2 Les méthodes de seuillage local

Les méthodes de seuillage local (adaptatif) s'adaptent au contexte de chaque pixel par le calcul d'un seul seuil pour chaque pixel de l'image en fonction de l'information contenue dans son voisinage. Cela permet de compenser les variations de luminosité et les dégradations locales d'une image. Si la fenêtre couvre une zone de l'image faiblement contrastée, la sensibilité du seuil de détection est automatiquement augmentée. Cette adaptation aux changements locaux de contraste explique la popularité de ces méthodes sur les images de documents dégradés ou ceux qui utilisent des couleurs d'encre différentes. Habituellement, l'adaptation est obtenue en balayant l'image en zigzag par une fenêtre d'analyse centrée sur chaque pixel dans laquelle on réalise le calcul d'un seuil local. La complexité est de l'ordre de $N \times M \times P$ où $N \times M$ représente le nombre de pixels de l'image et P le nombre de pixels de la fenêtre d'analyse locale.

Il existe plusieurs méthodes de seuillage locale. Nous allons présenter dans ce qui va suivre celles qui sont les plus utilisées dans le domaine de traitement automatique de documents.

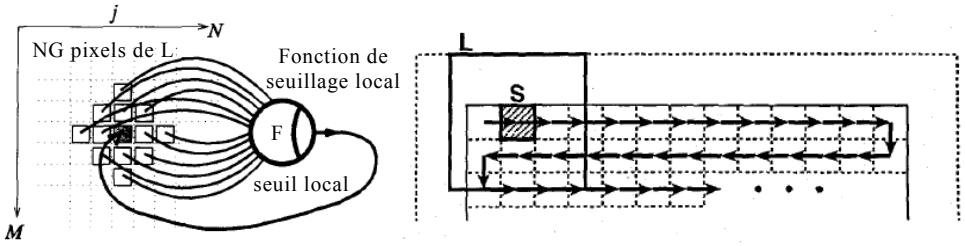


Figure 2. 7 : Principe des méthodes de binarisation locales et parcours de la fenêtre d'analyse L (s pixel à seuiller).

2.2.2.1 Méthodes de seuillage local basées sur le concept de Niblack

Le concept de Niblack [NIB86] repose sur le calcul d'une valeur de seuillage en faisant glisser une fenêtre sur l'image. Pour chaque pixel un seuil T est calculé à partir de quelques statistiques comme la moyenne locale m et l'écart type local s calculés sur les niveaux de gris des pixels voisins dans la fenêtre par la formule suivante :

$$T = m + k \cdot s \qquad (2.\ 5)$$

Avec k, constante négative.

Une étude comparative effectuée par Trier et Jain [TRI95b] a montré que la méthode de Niblack segmente bien les caractères de texte et donne de meilleures performances sur les images de documents par rapport aux autres méthodes locales et globales de binarisation. Cette efficacité a été confirmée aussi par [HEJ05] qui a comparé la méthode de Niblack avec d'autres méthodes plus récentes. Cependant, cet algorithme produit du bruit sur les images dont l'arrière plan est dégradé, ayant pour conséquence la nécessité d'un post-traitement très coûteux en temps. Sauvola [SAU97] a choisi d'améliorer la formule (2.5) en ajoutant une hypothèse sur les valeurs des niveaux gris des pixels de texte et de fond (tous les pixels de texte ont des niveaux de gris proches de 0 et tous les pixels de fond ont des niveaux de gris proches de 255), la formule de seuillage local devient donc:

$$T = m \cdot (1 - k \cdot (1 - \frac{s}{R})) \qquad (2.\ 6)$$

k est un paramètre fixé à $k = 0.5$ et R est la dynamique de l'écart type, fixé à $R = 128$. Les résultats obtenus avec la méthode de Sauvola et al montrent une réelle diminution du bruit du fait des hypothèses initiales sur les données. De point de vu qualité, le problème a été résolu dans le cas des images de documents d'entreprise et de courriers par Sauvola en ajoutant dans le calcul le fait qu'un pixel sombre appartienne plus probablement au texte qu'au fond. Aujourd'hui un grand nombre de méthodes de segmentation utilise les algorithmes de Niblack et de Sauvola comme approches de base qu'elles améliorent soit en adaptant la formule [WOL02][FENG04] soit en ajoutant des post-traitements, [VAL00] [HAM05].

Valverde et al. dans [VAL00] ont tenté de pallier les inconvénients de la méthode de Niblack par l'association de deux étapes de post-traitement pour avoir plus de fiabilité sur des documents techniques. Les étapes de post-traitement utilisent essentiellement une fermeture morphologique pour réduire le bruit et combler les trous et un gradient de Sobel pour améliorer la forme des caractères. Dans le cas des documents vidéo caractérisés par des propriétés différentes (faibles contrastes, écarts de niveaux de gris importants, etc.), les hypothèses choisies par Sauvola ne sont pas toujours justifiées, provoquant parfois des « trous » dans les caractères. La méthode proposée par Wolf [WOL02] résout ce problème pour les documents vidéo et améliore les performances au niveau du bruit par une normalisation des contrastes et des moyennes de niveaux de gris de la façon suivante:

$$T = (1-k)\cdot m + k\cdot M + k\cdot \frac{s}{R}\cdot (m-M) \qquad (2.\ 7)$$

Où M est le minimum des niveaux de gris de toute l'image et la dynamique d'écart type R est fixée au maximum de l'écart type s de toutes les fenêtres. Cet algorithme se concentre sur l'écart type maximum de l'image entière ce qui risque malgré tout de causer des dégradations sur des images vidéo exposées à des grands changements de luminance du fond. Face à cet inconvénient, Feng [FEN04] propose une version plus fiable par une nouvelle approche du calcul de m, de M et de s estimés dans une fenêtre locale primaire dont la taille est assez grande pour couvrir un ou deux caractères de texte, voir figure 2.8. Afin de compenser l'effet de variation de luminance, la dynamique de l'écart type R est calculée cette fois ci sur une fenêtre locale secondaire d'une taille plus grande au lieu de l'image toute entière. Le seuil T peut alors être donné par la formule suivante :

$$T = (1-\alpha_1)\cdot m + \alpha_2\cdot \frac{s}{R_s}\cdot (m-M) + \alpha_3\cdot M \quad \text{avec} \quad \alpha_2 = k_1\cdot \left(\frac{s}{R_s}\right)^{\gamma} \text{ et } \alpha_2 = k_2\cdot \left(\frac{s}{R_s}\right)^{\gamma} \quad (2.\ 8)$$

Où α_1, γ, k_1 et k_2 sont des constantes positives.

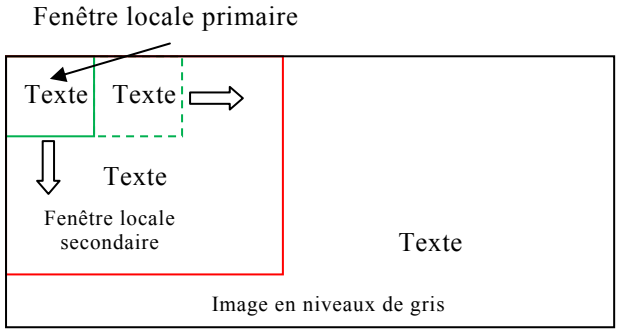


Figure 2. 8 : Définition des fenêtres locales utilisées par le concept de Feng.

2.2.2.2 Méthodes de seuillage basées sur les réseaux de neurones

Ces méthodes utilisent des réseaux de neurones pour donner au système de binarisation la capacité d'apprendre à optimiser les valeurs du seuil. Dans ce contexte, Chigusa et al. [CHI92] utilisent un réseau de Hopfield dans un algorithme parallèle. Le seuil de chaque neurone est choisi d'une manière adaptative en fonction des niveaux gris de pixels voisins au pixel central représenté par ce neurone. Les poids synaptiques entre les neurones distants sont à *0*, et entre les neurones voisins sont à *1*. Ce voisinage est mesuré par la distance euclidienne et l'état d'équilibre du réseau représente l'image binaire finale. Chi et Wong [CHI01] proposent de faire coopérer la phase de binarisation avec la phase de la segmentation en blocs. Au départ, ils binarisent l'image initiale par un seuillage global, puis ils segmentent en blocs l'image binaire. Ils appliquent ensuite un feed-back pour rebinariser chaque bloc par un réseau de neurones. Cette méthode ne peut pas être efficace sur les documents qui possèdent une luminance non uniforme car la binarisation globale risque de provoquer la fusion des blocs et de dégrader la qualité de la segmentation en blocs. Afin d'éviter cet écueil, Hamza et al. [HAM05] combinent une carte auto-organisatrice (SOM) avec les méthodes K-Means, Sauvola et Niblack. Cette combinaison donne de meilleurs résultats sur des images de documents dégradés mais la performance est toujours conditionnée par le choix de nombre de neurones et par la représentativité de la base d'apprentissage.

Figure 2. 9 : (a) Image en niveaux de gris dégradée, résultat de binarisation par la méthode **(b)** Sauvola, **(c)** SOM_Sauvola, **(d)** Niblack, **(e)** SOM_Niblack.

2.2.3 Méthodes de seuillage hybrides

Ces méthodes utilisent une analyse globale et locale en même temps. Trier et Taxt [TRI95] ont proposé une méthode adaptée aux documents techniques. Pour cela, ils calculent le Laplacien à partir des dérivées partielles du résultat *A(x,y)* issu d'un filtrage médian de l'image initiale, puis ils créent une image étiquetée *L* telle que:

$$L(x,y)=\begin{cases}'0' & si \quad A(x,y)<T_A \\ '-' & si \quad A(x,y)\geq T_A \; et \; \nabla^2 Z(x,y)<0 \\ '+' & si \quad A(x,y)\geq T_A \; et \; \nabla^2 Z(x,y)\geq 0\end{cases} \qquad (2.9)$$

Ensuite, ils séparent le fond de la forme selon l'algorithme suivant:

```
Algorithme 2.1 : Seuillage_hybride_Trier_Taxt
Début :
 Pour chaque région r 4-connexe de L
 - Compter les nombres N- et N+ de pixels respecti-
    vement marqués '-' et '+' et 8-connexes à r.
 - Si N+ > N-, alors re-étiqueter tous les pixels
    avec le label '+'.
 - Seuiller ensuite L de telle sorte que les pixels
    marqués '+' appartiennent à la forme, et ceux
    marqués '-' ou '0' au fond.
Fin.
```

À la fin, ils appliquent des post-traitements présentés dans [YAN89] pour améliorer la qualité de l'image binaire. Cette méthode donne de bons résultats sur des documents de bonne qualité, mais elle a l'inconvénient d'être régie par plusieurs paramètres qui restent difficiles à régler en pratique.

Après avoir séparé l'arrière plan de premier plan, Chang et al [CHA95] proposent d'égaliser l'histogramme de premier plan de manière à faciliter la distinction entre les caractères et le bruit. Ils se servent par la suite d'un Laplacien pour reconstruire la forme des caractères binaires. Savakis dans [SAV98] a proposé deux algorithmes de seuillage des images adaptés à la numérisation de documents à haute vitesse. Le premier algorithme utilise un seuillage adaptatif qui fonctionne en commutation : il applique soit un seuillage local sur des pixels qui possèdent un gradient local répondant à une forte transition au niveau de tracé, soit un seuillage global sur des pixels à faible gradient qui appartiennent aux zones homogènes de fond. Le second algorithme est basé sur le suivi de tracé utilisant un regroupement basé sur une variante d'algorithme de type K-Means. Les deux approches peuvent être utilisées indépendamment ou combinées pour un meilleur résultat. Dans le même principe, Kamada et al [KAM99] proposent une technique séquentielle destinée aux images à faible résolution comme les images issues d'une caméra de téléphone portable. Cette binarisation s'effectue par deux tâches : la première sépare le premier plan de l'arrière plan par un seuillage global, alors que la seconde applique une extraction de voisinage de caractères et une interpolation linéaire de l'image puis elle utilise un seuillage local sur les pixels de premier plan, améliorant ainsi la qualité des caractères pour l'OCR, voir figure 2.10.

Figure 2. 10 : Étapes de binarisation proposées par Kamada, **(a)** image en niveaux de gris, **(b)** seuillage global, **(c)** extraction de voisinage de caractères, **(d)** seuillage local.

Sue Wu et Adnan Amin [WUS03] proposent une méthode hybride de seuillage des images d'enveloppes postales qui s'applique en deux étapes. La première étape applique un seuillage global sur l'image originale. La deuxième étape ajuste la valeur de seuil en fonction des caractéristiques spatiales des composantes connexes formées dans la première étape. La méthode donne de bons résultats sur l'ensemble des images d'enveloppes simples, bien contrastées et de bonne qualité. Badekas et Papamarkos proposent dans [BAD03] un système de binarisation intégrant le résultat de six techniques de binarisation indépendantes locales et globales. Chacune de ces techniques a un coefficient (poids) de contribution qui doit être attribué par l'utilisateur de telle façon que la somme de tous les coefficients soit égale à *100%*. Cette méthode est principalement destinée aux documents présentant de fortes dégradations ou une mauvaise luminance. Dans le même contexte, Thillou et Gosselin ont proposé dans [THI04] une coopération entre la phase de binarisation et la phase de segmentation des caractères s'améliorant mutuellement.

2.2.4 Bilan des approches de binarisation

Les conditions de vitesse d'exécution imposées par les systèmes de tri automatique de documents et de courriers exigent des algorithmes qui doivent être non seulement efficaces, mais également très rapides. Chacune des méthodes de binarisation présentées ci-dessus a été conçue ou adaptée pour une application bien définie. Aucune de ces méthodes n'est bien adaptée à notre application de temps réel. En effet, les méthodes les plus simples utilisant un seuillage global ont l'avantage d'être extrêmement rapides mais la variation d'éclairage, la présence de divers graphiques imprimés sur les enveloppes ayant des couleurs d'encres différentes font très rapidement chuter la qualité de la binarisation. Les méthodes locales dépassent ces limites et sont plus adaptées aux changements locaux de contraste. Cependant elles nécessitent plus de calculs, sont donc plus lentes et de ce fait mal adaptées aux applications temps réel. Bien qu'elles assurent une bonne efficacité sur les documents qui concernent notre application, ces approches de binarisation locales possèdent principalement les inconvénients suivants :

- des temps de calculs prohibitifs en fonction la taille de la fenêtre d'analyse,
- une sur-segmentation des défauts et de la texture de l'arrière plan de l'image,
- un traitement difficile des documents dont les caractères sont de taille variable (la fenêtre d'analyse étant fixe durant tout le traitement).

Face à ces inconvénients et au regard de notre application industrielle temps réel, il est nécessaire de concevoir une méthode de binarisation respectant des temps de calculs très faibles tout en restituant des images binaires de très bonne qualité. La qualité des images binarisées reste cependant un point difficile à évaluer. Actuellement, quelques travaux ont été réalisés afin de résoudre le problème de l'évaluation du résultat de binarisation. Il est possible de les regrouper en trois familles d'approches : La première famille d'approches évalue la qualité de la binarisation en mesurant la performance de la reconnaissance du texte [TRI95] [ZHA96] [HE05]. Les résultats d'évaluation effectués par Trier [TRI95] et par He [HE05] montrent que les méthodes de Niblack et de Sauvola donnent de meilleurs résultats sur des images de documents. La seconde famille d'approches évalue cette qualité en mesurant sa similarité avec une vérité-terrain [SEZ01]. La troisième famille d'approches se base sur des critères d'évaluation non supervisés permettant d'estimer la qualité d'un résultat de segmentation et peut ainsi être utilisée pour évaluer le résultat d'une binarisation [PHI01].

Nous avons choisi une approche de l'évaluation de notre méthode de binarisation des documents basée sur l'évaluation directe des résultats de l'OCR sur site.

2.3 Extraction des composantes connexes

L'extraction des composantes connexes (CCs) est la base d'un grand nombre de chaînes algorithmiques en traitement automatique de documents, dès qu'il s'agit de segmenter une image binaire d'un document ou d'analyser ses éléments constitutifs. En effet, elle est souvent présente lorsqu'il faut regrouper des pixels connexes pour trouver les caractères, les lignes ou les blocs de texte. Cette procédure également appelée étiquetage des pixels représente une source de grande vitalité pour les applications de RAD (Reconnaissance automatique de documents). Formellement, une composante connexe est un ensemble de pixels connectés deux à deux où la connexité est une propriété de liaison entre deux pixels qui fait qu'on les considère comme faisant partie du même objet dans une image binaire. Sur une image binaire d'un document ou d'une enveloppe, les composantes connexes peuvent être, par exemple, des mots (dans le cas de texte manuscrit), des caractères alphanumériques, des morceaux de caractères, des ponctuations, des accents, des symboles, des éléments de parties graphiques et même des tâches dues au bruit ou à la mauvaise qualité du support physique. L'étiquetage des pixels transforme une image binaire (objets/fond) en une image symbolique de telle sorte qu'une étiquette identique soit attribuée pour tous les pixels d'une même composante. Chaque composante

connexe reçoit une étiquette disjointe des autres et peut ensuite être facilement isolée. On en déduit que deux composantes connexes distinctes sont toujours disjointes, voir figure 2.11.

Figure 2. 11 : Étiquetage des composantes connexes d'un bloc adresse.

La connexité d'ordre *4* se distingue de la connexité d'ordre 8 selon le critère de voisinage qui comprend les *4* ou les *8* voisins d'un pixel, voir figure 2.12.

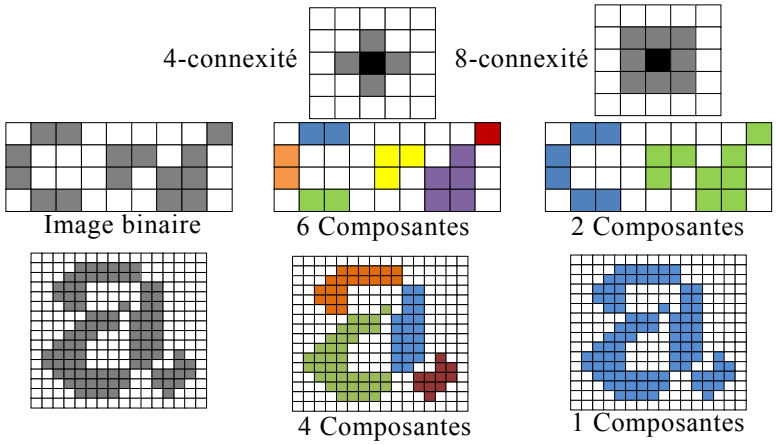


Figure 2. 12 : Effet de la connexité sur l'extraction des composantes et prédécesseurs d'un pixel.

Pour réduire la complexité de l'exploitation, chaque composante connexe peut être représentée par les coordonnés (x_d, y_d, x_f, y_f) du plus petit rectangle qui l'englobe. L'utilisation des rectangles circonscrits pour représenter des connexités peut poser des problèmes dans la mesure où deux connexités différentes peuvent avoir des rectangles circonscrits imbriqués ou se chevauchant, voir figure 2.13. L'étiquetage des composantes connexes représente la tâche la plus coûteuse en temps d'exécution et en mé-

moire dans la chaîne de traitement et peut être véritablement pénalisant pour l'accomplissement des tâches ultérieures.

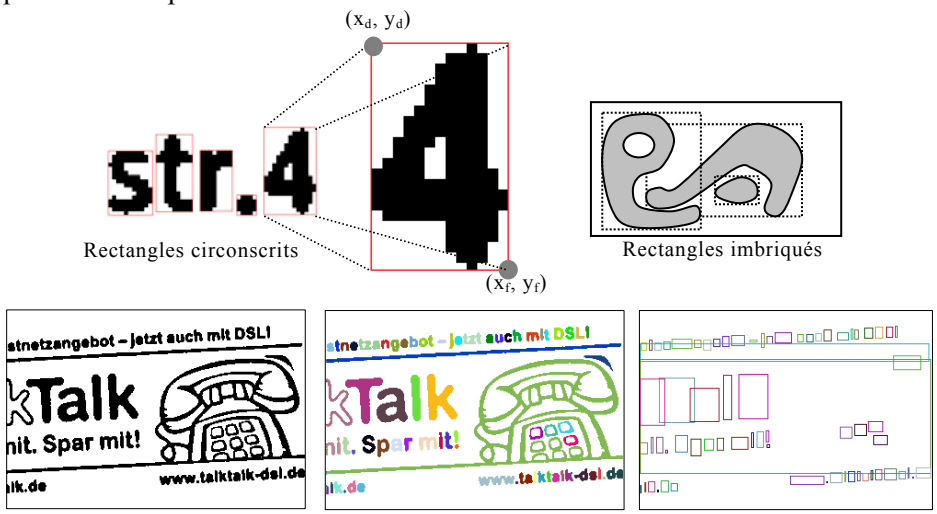


Figure 2. 13 : Ambiguïté dans la représentation des connexités par des rectangles circonscrits.

L'accomplissement de cette tâche en temps réduit a fait l'objet de nombreuses études qui ont abouti à plusieurs algorithmes séquentiels [ROS 66][HAR81][LUM83] ou parallèles [NAS80][KUM86][YAN88][RAN91]. Ces algorithmes utilisent l'un des deux principes suivants. Le premier principe consiste à suivre le contour d'un objet jusqu'à revenir au point de départ : à ce moment, une composante connexe est délimitée. Les contours intérieurs correspondant aux éventuels trous ne sont pas pris en compte [SUR99] [CHAN04]. Le second principe utilise la propagation d'un étiquetage des pixels lorsque l'on effectue un balayage des lignes et des colonnes de l'image [GLA01][LIF08]. Les divers algorithmes proposés dans la littérature peuvent être regroupés en deux grandes catégories: algorithmes récursifs et algorithmes à passes.

2.3.1 Algorithmes récursifs

Les algorithmes de cette catégorie consistent à étiqueter récursivement les composantes connexes de l'image binaire. L'algorithme floatfill proposé par Glassner [GLA01] est un très bon exemple de ces méthodes. Son principe repose sur le parcours de l'image pixel par pixel : il consiste à affecter pour chaque objet une nouvelle étiquette qui sera diffusée de manière récursive entre tous les pixels noirs qui lui appartiennent (figure 2.14). À chaque pixel de premier plan, si aucune étiquette ne lui a encore été affectée, on lui en attribue une nouvelle qu'on va aussitôt diffuser à tous ses pixels voisins, ainsi qu'à leurs voisins respectifs, ce mécanisme se poursuit alors récursivement. Chaque pixel non étiqueté est ensuite empilé dés qu'on lui attribue une nouvelle étiquette ou qu'il reçoit

celle de son voisin. Par la suite il est dépilé dès qu'il diffuse son étiquette avec ses pixels voisins non étiquetés. Quand la pile est vide, le balayage de l'image continue selon un parcours normal dans le but de trouver le prochain pixel noir qui n'a pas encore été étiqueté.

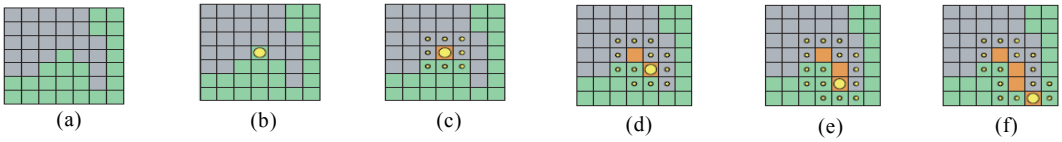


Figure 2. 14 : Propagation de l'étiquette utilisant un voisinage de 8 connexités.

Dans le cas des documents qui ne contiennent que du texte, les composantes connexes des caractères sont donc suffisamment petites pour que l'algorithme donne de très bons résultats. Si le nombre de pixels connexes est très grand, l'algorithme utilisera beaucoup de mémoire pour les appels récursifs et risquera à tout instant de causer un « overflow » (débordement de la mémoire tampon de la machine). En effet le nombre limité de récursions possibles dus à la gestion de la mémoire par nos ordinateurs ne permet pas d'employer cette méthode sur de très longues connexités (gravures, cadres, dessins au traits) ou des connexités possédant une grande surface (zones noires à l'emplacement de certaines photos). Dans ce cas il faut absolument éviter d'utiliser ce type d'algorithmes et s'intéresser à des mécanismes impliquant un nombre de récursions réduit ne détectant, par exemple, les connexités que sur les pixels des contours des objets. Plusieurs algorithmes ont été proposés à cet égard. On peut citer par exemple, l'algorithme de Sural et Das [SUR99] et l'algorithme de Chang et al [CHA04] qui appliquent un suivi récursif des contours des objets afin de détecter directement leurs rectangles circonscrits (figure 2.15). Ces méthodes peuvent très bien réduire le problème de débordement de mémoire sur des documents complexes, mais elles doivent résoudre le problème des rectangles imbriqués et nécessitent ainsi une étape préalable de détection de contours qui peut être coûteuse en temps de traitement. L'ensemble de ces constats (temps de traitement coûteux, débordement mémoire…) nous pousse à nous intéresser aux algorithmes itératifs (ou algorithmes à passes).

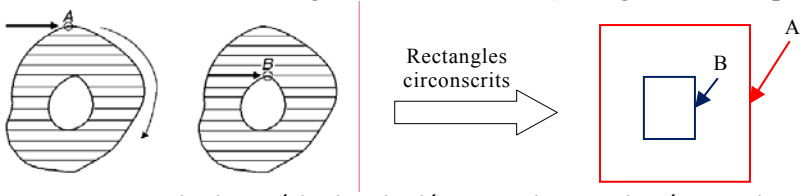


Figure 2. 15 : Exemple des méthodes de détection des CCs basées sur le suivi de contours.

2.3.2 Algorithmes à passes

Les algorithmes d'étiquetage de cette catégorie utilisent généralement deux passes (parcours avant et parcours arrière) ou une séquence de deux passages sur l'image binaire. La figure suivante représente les masques de voisinage définissant les pixels précédents au pixel courant P et qui ont déjà été analysés dans un premier parcours. Les indices des lignes et de colonnes sont croissants dans le cas d'un parcours avant et décroissants dans le cas d'un parcours arrière.

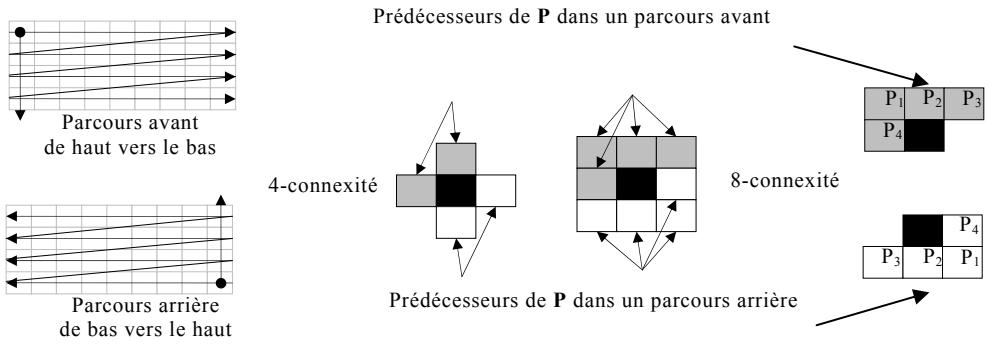


Figure 2. 16 : Les deux parcours de l'image et prédécesseurs d'un pixel.

Rosenfeld [ROS66] [ROS76] [ROS82] ont révolutionné le domaine en proposant un algorithme purement séquentiel d'étiquetage des composantes connexes ne comprenant que deux passes sur l'image. L'idée de cet algorithme est d'exploiter l'étiquetage effectué lors du premier parcours pour une affectation finale des étiquettes ne demandant ainsi qu'un seul parcours supplémentaire. Le principe consiste à balayer dans un parcours avant les pixels de l'image ligne par ligne et ne considérer que ceux des objets (pixels noirs).

Algorithme 2.2 : Rosenfeld

Début :

- ***Si*** *le pixel n'a pas de voisin connexe déjà étiqueté, lui créer une nouvelle étiquette et l'affecter à ce pixel.*
- ***Si*** *le pixel a exactement un seul pixel connexe étiqueté, lui affecter cette étiquette.*
- ***Si*** *le pixel a plus d'un pixel connexe (avec des étiquettes différentes) lui affecter la valeur minimale de ces étiquettes et mémoriser le fait que dans une table de correspondances Tc toutes ces étiquettes sont équivalentes.*
- ***Refaire*** *une passe sur l'image dans un parcours arrière en groupant pour tous les points d'un même objet les étiquettes équivalentes sur une seule étiquette.*

Fin.

Pour une forme longiligne, les événements continuité/fusion/séparation signalent les différentes composantes d'une même connexité et leurs relations sous la forme d'un graphe d'adjacence de lignes (LAG) qui est souvent utilisé comme descripteur de formes [Pav77] (Voir le chapitre 5, section 5.3.2). L'algorithme Rosenfeld et Pfaltz [ROS66] induit la génération de nombreuses étiquettes temporaires dans le cas d'un objet de forme complexe, ce qui implique la définition d'une table T_c de grande taille. Ceci peut augmenter le temps de traitement. Pour compenser cette faiblesse, Haralick [HAR81] a conçu une méthode itérative séquentielle qui ne nécessite que la structure image et un compteur d'étiquettes comme structure de données. Le contenu de la matrice image est modifié de manière itérative en alternant des parcours avant et des parcours arrière jusqu'à stabilisation. La première passe affecte des étiquettes temporaires aux pixels, elle est définie par:

$$g(x,y)=\begin{cases} F_B & \text{si} \quad b(x,y)=F_B \\ m, m=m+1 & \text{si} \quad \forall\{i,j\in Ms\}\ g(x-i,y-j)=F_B \\ g_{\min}(x,y) & \text{sinon} \end{cases} \qquad (2.\ 10)$$

Avec $g_{\min}(x,y)=\min\left[g(x-i,y-j)\,|\,i,j\in Ms\right]$

Où *b* est l'image d'entrée binarisée, $g(x, y) \in \{F_B, F_O\}$ est la valeur du pixel à la position *(x, y)* dans l'image binarisée, F_B la valeur d'un pixel de fond (blanc), F_O la valeur d'un pixel de texte (noir). *g(x, y)* contient l'étiquette du pixel à la position *(x, y)*, m est l'étiquette courante a affecter au prochain nouveau label, et *Ms* est le masque des voisins (celui de parcours avant dans ce cas). La première passe va donner des étiquettes temporaires à g de telle sorte que chaque pixel possède la plus petite étiquette de ses voisins qui ont déjà été analysés ou une nouvelle étiquette s'il ne possède aucun voisin. Une fois les étiquettes temporaires affectées à g, différentes passes sont effectuées sur g jusqu'à que plus rien ne soit modifié :

$$g(x,y)=\begin{cases} F_B & \text{si} \quad b(x,y)=F_B \\ \min\left[g(x-i,y-j)\,|\,i,j\in Ms\right] & \text{sinon} \end{cases} \qquad (2.\ 11)$$

Le nombre d'itérations dépend donc de la complexité des objets. Cette approche permet d'éviter la gestion d'une table de correspondances mais consomme en contrepartie plus de temps. Notez que le nombre d'itérations ainsi que la durée de l'étiquetage pour cet algorithme dépendent de la nature de l'image d'entrée. Lumia et al [LUM83] ont proposé une technique à deux passes qui donne un meilleur compromis temps/mémoire par rapport aux deux méthodes précédentes [ROS66] et [HAR81]. Cette technique utilise une table de correspondances pour chaque ligne et consomme de ce fait moins de mémoire que l'algorithme de Rosenfeld. Par ailleurs elle ne né-

cessite pas l'exploitation de plusieurs itérations comme l'algorithme Haralik. Une combinaison de ces deux algorithmes séquentiels a été implémentée sur machine parallèle par Cheng et a1 [CHE94]. L'algorithme à deux passes proposé par Stefano et Bulgarelli [STE99] analyse cette fois-ci les équivalences durant le premier balayage en utilisant une table de correspondances à une seule dimension. Cette approche conduit à la fusion des classes dès qu'une nouvelle équivalence est trouvée. Elle donne de meilleures performances sur des images binaires tramées par rapport à un algorithme similaire déjà présenté par Samet[SAM86].

L'affectation des pixels à une table d'équivalences a été également utilisée par Suzuki et al [SUZ03] et améliorée par une réduction significative du nombre de passes à effectuer pour arriver à l'étiquetage définitif de l'image. Cet algorithme a été finalement repris par Wu et al [WU05] et amélioré par l'introduction d'une forêt d'arbres représentée par la table d'équivalence renommée pour l'occasion. Chaque nœud de cet arbre possède pour parent un nœud qui lui est connexe dans l'image. Les différents nœuds de l'arbre sont réunis au fur et à mesure de l'avancement de l'algorithme de façon à ce que tous ceux qui font partie de la même composante connexe soient dans le même arbre. La seconde phase affecte les labels définitifs dans la table d'équivalence sans passage sur l'image. La dernière passe met à jour la valeur des pixels de l'image des composantes connexes.

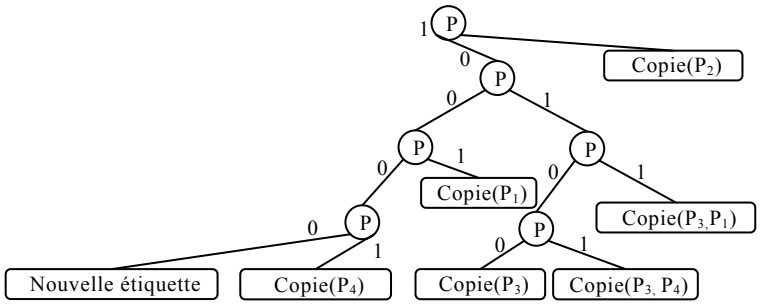


Figure 2. 17 : Arbre de décision utilisé par l'algorithme de Suzuki et Wu.

On peut remarquer dans l'arbre de décision ci-dessus que les 4 voisins analysés dans une passe ne sont pas tous nécessaires à la déduction de l'étiquette courante. Le fait que le pixel voisin P_2 dans le masque d'analyse soit scanné avant le pixel courant P, l'étiquette de P doit être celle de P_2. Par contre, si le pixel P n'a aucun voisin, alors on créé une nouvelle étiquette dans la première passe ou on ne fait rien dans les passes suivantes. Si le pixel n'a qu'un seul voisin on prend sa valeur. Si le pixel possède deux voisins, par exemple P_3 et P_1, alors on lui donne la valeur minimale, en utilisant la table d'équivalences et en modifiant la table pour les deux voisins. La comparaison de l'algorithme de Wu avec celui de Su-

zuki montre qu'il est plus rapide sur des images complexes qui contiennent des grandes composantes connexes et qu'il est légèrement plus lent sur des documents qui ne contiennent que du texte simple.

En plus des algorithmes séquentiels mentionnés ci-dessus, d'autres algorithmes parallèles ont été proposés également. Nassimi et Sahni [NAS80] ont proposé un algorithme *O(N)* utilisant un réseau de NxN mailles pour des images d'entrée de taille *N×N*. Kumar et Eshraghian [KUM86] on proposé un algorithme *O(log4N2)* utilisant un réseau en arbre de *NxN* mailles. L'algorithme parallèle proposé par Yang[Yan 88] tente d'étiqueter en deux passes les séquences noires à la place des pixels . Cette technique est implémentée dans une architecture hardware *VLSI* spécialement dédiée à l'analyse d'images en temps réel dans des applications de vision robotique. La complexité de la méthode a été réduite par Ranganathan dans [RAN91-95] par une architecture *VLSI* systoliques simple qui peut être implémentée dans une puce ou dans un circuit intégré. Bien que, l'algorithme ait une complexité de temps égale à *O(N2)*, le matériel utilisé se composait de 128 processeurs et avait le pouvoir de traiter une image de *128 x 128* en *85* milliseconde. Pour une image de *MxN* pixels, l'algorithme nécessite *3N+MN* cycles pour effectuer la première passe et *2N+ MN* pour la seconde. Par conséquent, toute la durée de l'étiquetage consomme *5N+2MN* cycles. Une autre architecture légèrement améliorée a été aussi proposée dans [RAS 97] par Rasquinha et Ranganathan.

L'algorithme à deux passes proposé plus récemment par Lifeng et al [LIF07][LIF08] s'applique directement sur des séquences noires. Les données de séquences noires obtenues durant la première passe sont enregistrées dans une file d'attente afin d'être utilisées pour détecter les connexités. Durant cette passe, toutes les étiquettes temporaires affectées à une composante connexe sont arrangées dans un ensemble d'étiquettes temporaires, et la plus petite étiquette est utilisée comme représentant. Ainsi, chaque séquence noire de la ligne courante reçoit une nouvelle étiquette, si seulement si elle n'a aucune connexité avec les séquences noires de la ligne précédente. Sinon elle prend l'étiquette de la séquence connexe la plus gauche et l'ensemble des étiquettes temporaires correspondant à ses séquences connexes est fusionné dans un même ensemble qui aura ainsi la plus petite étiquette comme représentante.

Durant la deuxième passe, chaque étiquette temporaire doit être remplacée par celle de son représentant.

À l'inverse des algorithmes à passes classiques qui cherchent les équivalences entre toutes les étiquettes temporaires, cet algorithme consomme moins de temps et de mémoire et ne cherche les équivalences qu'entre les ensembles d'étiquettes temporaires (figure 2.18).

Pour notre application de tri automatique de courrier et de document, l'algorithme de Lifeng présente le meilleur compromis temps/ mémoire et peut être facilement adapté et amélioré en l'intégrant à la structure LAG adoptée dans notre architecture pyramidale.

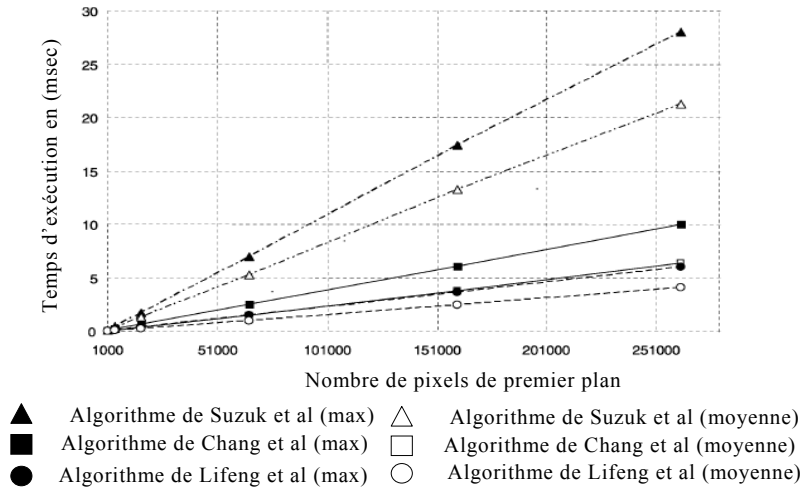


Figure 2. 18 : Performance en temps d'exécution de l'algorithme de Lifeng par rapport à l'algorithme de Chang et à l'algorithme de Suzuk considérés comme méthodes performantes.

2.4 Extraction de la structure physique des documents : Une clé pour la compréhension des documents

Lors du passage des documents ou des enveloppes physiques dans une chaîne de tri, les images sont acquises par une caméra CCD linéaire à haute vitesse. Les images résultantes se composent d'une variété d'entités physiques ou de régions comme les blocs, les lignes de texte, les mots, les figures, les tableaux et le fond. Ces éléments et leurs relations peuvent être automatiquement décrits par le processus de reconnaissance de documents ou plutôt par l'analyse d'images de documents. Cette description peut être physique, i.e. révélant le format de mise en page, ou logique, i.e. décrivant l'enchaînement des sous-structures, ou encore sémantique, i.e. portant sur le sens affecté à certaines parties [EGL99]. Toutes ces descriptions exigent une étape d'extraction de la structure physique qui détermine une partition géométrique de l'image du document numérisé de sorte à isoler tous ses éléments constitutifs. Ces éléments sont souvent espacés et forment des blocs géométriques homogènes, à base de rectangles dans la grande majori-

té des cas. Dans une application de tri postal, la phase de l'extraction de la structure physique a un grand impact sur la performance de système de tri en entier. Par exemple, durant la séparation de l'image de l'enveloppe en blocs, la moindre erreur de segmentation peut facilement conduire à une lecture erronée de l'adresse de destination.

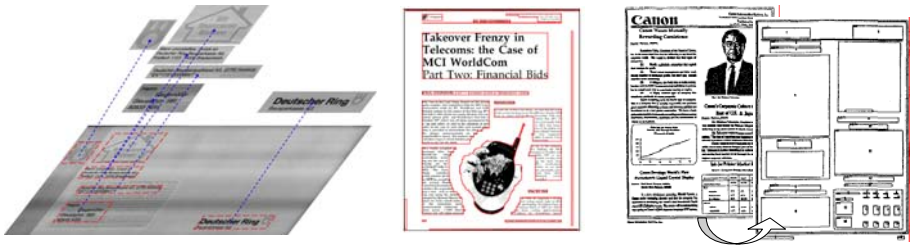


Figure 2. 19 : Exemples d'extraction de la structure physique de différents documents.

L'étape de segmentation est bel et bien une étape clé dans le processus de lecture de l'adresse. Plus généralement, il s'agit d'une étape préliminaire indispensable à tout processus de reconnaissance des contenus. Dans les applications industrielles de vision en temps réel, on se trouve souvent devant l'obligation de choisir ou de concevoir une méthode de segmentation qui respecte le mieux possible le compromis temps / performance. Cependant, la moindre erreur de segmentation d'un document se paye par l'impossibilité de retrouver les informations clés portées par ce dernier. Cela nous oblige donc à prendre en compte plusieurs facteurs d'erreurs qui ont une grande influence sur le résultat de la segmentation comme l'état, la qualité, la couleur, la texture de papier, la complexité, la résolution spatiale, le contenu, l'espacement entre les éléments et les artefacts (bruit, inclinaison) des images. Ceci nous soumet aussi à revenir à la définition de la segmentation qui est assez proche du sens littéral du mot analyse avant d'aller à la présentation des différentes techniques de segmentation. On parle donc, de sur-segmentation lorsque l'élément constitutif est lui-même fragmenté, et de sous-segmentation lorsque plusieurs éléments constitutifs n'ont pas pu être isolés. Ces deux types d'erreurs décrites dans [AGN00] et qu'il faut pouvoir réduire le mieux possible, sont présentes en pratique sous plusieurs formes (Figure 2.20):

- La fusion horizontale (ou fusion verticale de blocs ou de lignes de texte),
- Le découpage horizontal ou vertical de blocs ou de lignes de texte,
- La fusion ou la confusion de texte avec le graphique ou le bruit,
- Le non détection de blocs ou de lignes textuels.

Segmentation correcte | Fusion horizontale de zones textuelles

Segmentation correcte | Fusion verticale de zones textuelles

Découpage horizontal de zones de texte | Découpage vertical de zones de texte

Zones non détectée et Parasites confondus au texte | Fusion Graphique/ bruit avec le texte

Figure 2. 20 : Exemple de différentes erreurs d'extraction de la structure physique de l'image de la zone d'adresse.

Dans cette section, nous allons présenter les mécanismes usuels de segmentation tenant compte dans la plupart des cas des principales sources d'erreurs précédemment citées. Nous aborderons dans une dernière partie, les approches innovantes du domaine ainsi que les facteurs importants influençant les performances des systèmes.

2.4.1 Les mécanismes usuels de segmentation

Pour extraire la structure physique de l'image d'un document, on procède habituellement soit par découpage descendant récursif à partir des espaces, soit par fusion ascendante récursive des objets élémentaires entre eux, soit encore par la combinaison des deux lorsque l'une ou l'autre n'est pas satisfaisante. Ces techniques sont en général fondées sur une analyse spatiale, structurelle ou spectrale. Dans la littérature, nous pouvons trouver un grand nombre de travaux concernant l'extraction de la structure des documents. Les méthodes peuvent être classées en trois grandes familles : les méthodes descendantes guidées par un ensemble de règles ou par un modèle, les méthodes ascendantes guidées par les données et les méthodes mixtes qui reposent à la fois sur des méthodes descendantes de recherche

d'information et sur des méthodes ascendantes de segmentation guidée par le contenu de l'image [EGL99].

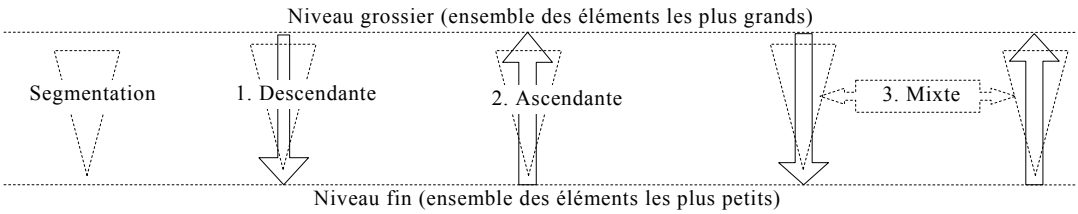


Figure 2. 21: Les trois mécanismes de segmentation.

2.4.1.1 Les mécanismes de segmentation descendants (top-down)

Ils caractérisent les techniques de segmentation procédant par découpage récursif des images traitées. La récursivité se poursuit jusqu'à ce que la composante physique la plus élémentaire soit obtenue (par exemple : décomposition de l'image d'une enveloppe en blocs, les blocs textuels en lignes et les lignes en mots). Dans ces techniques, la découpe d'une image est fondée, en général, sur l'analyse des caractéristiques globales. Elle a l'avantage d'être moins coûteuse en temps de calcul que les méthodes ascendantes.

2.4.1.1.1 Segmentation par projection de profils et découpage récursive de document

La méthode descendante la plus simple est le découpage récursif X-Y(ou RXY-Cut). Cette méthode a été largement utilisée pour obtenir une représentation hiérarchique de la structure physique des documents sous forme d'arbre où la racine correspond au document entier et les nœuds correspondent aux différentes régions d'intérêt du document, [DUY02]. Alors que Nagy [NAG84], Akiyama [AKI86] et Krishnamoorthy [KRI93] projettent les pixels noirs horizontalement et verticalement pour découper récursivement le document le long des espaces blancs, Ha [HAJ95] préfère projeter les profils des composantes connexes. Dans les deux cas, la position de chaque point de découpage est déterminée à partir des pics des projections qui indiquent les lignes et les colonnes du document. En choisissant judicieusement les seuils, il est possible de décomposer des documents assez complexes à condition qu'ils soient bien redressés et que les blocs soient rectangulaires. La décomposition peut être poussée jusqu'aux caractères ou être stoppée à un niveau supérieur, selon les besoins. Dans tous les cas, il est nécessaire de faire appel à des connaissances a priori sur le document pour bien définir le critère d'homogénéité utilisé.

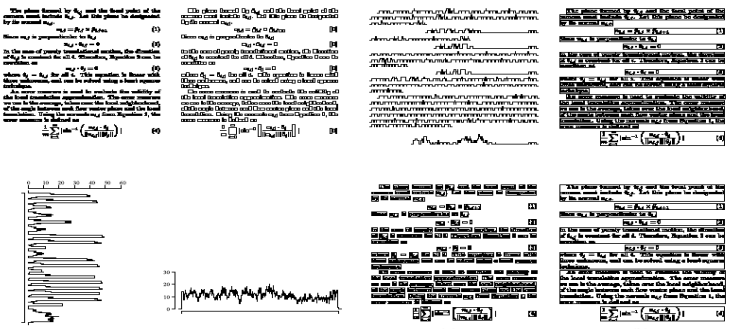


Figure 2. 22 : Découpage récursif d'un document.

L'algorithme de découpage X-Y peut être comparé à d'autres techniques à base de représentations plus évoluées. Akindele et Belaid [AKI93] lui ont ajouté une méthode de suivi de segments (suivi de contours) pour aboutir à une segmentation en blocs polygonaux. Wolf et al [WOL97] l'ont amélioré pour l'appliquer sur des images d'enveloppes en niveaux de gris dans une application de tri postal. Enfin, Cesarini [CES99] et Appiani [APP01] l'ont modifié (MXY-Cut) pour l'adapter à des documents contenant des tableaux (formulaires, factures…) sur lesquels la méthode classique perd son efficacité en présence de segments linéaires des tableaux de grandes tailles. Dans le même contexte, on peut citer les travaux de Han dans [HAN07] qui vise une application industrielle d'accès au contenu des documents off-line en téléphonie mobile. Dans les travaux de Han, l'algorithme de segmentation a été repris de l'approche MXY-Cut de Cesarini et combinée à un réseau de neurones de type perceptron multicouches (MLP). Cela a permis de rejeter tous les éléments de bruit qui risquaient de causer des erreurs de segmentation inacceptables dans une telle application.

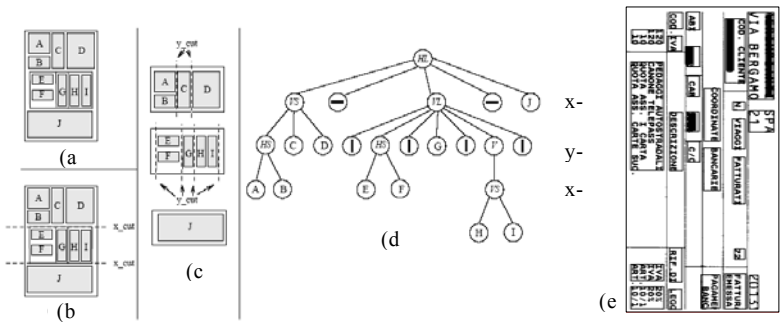


Figure 2. 23 : Exemple d'arbre-MXY, **(a)** une page, **(b) et (c)** le premier et le deuxième niveau de segmentation, **(d)** arbre MXY, **(e)** Résultat de segmentation d'un formulaire binaire par la méthode MXY.

Dans tous les cas, la difficulté inhérente à ce type de techniques est fortement liée à la différenciation entre les minima locaux et globaux dans l'histogramme des profils. Certaines séparations correspondent no-

tamment à des minima locaux non significatifs dans le cas des images mal redressées. Il est donc difficile d'utiliser ces méthodes seules sur les documents qui concernent l'application de tri où la mise en page des documents est variable et les zones de texte sont fréquemment mal redressées.

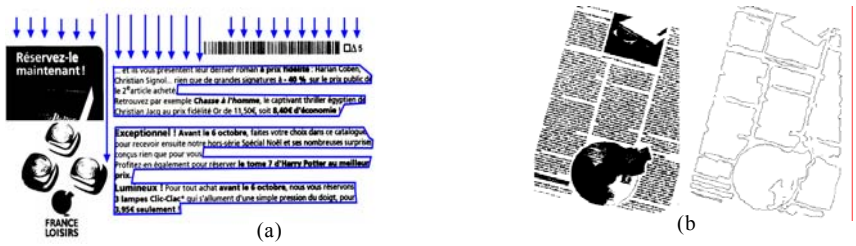
2.4.1.1.2 Segmentation par analyse des espaces blancs (analyse de l'arrière plan)

Lorsqu'il est impossible de détecter correctement les zones de texte par la projection des profils des pixels du premier plan, il peut être possible de prendre le problème à l'envers: analyser le fond pour extraire les grands espaces blancs permettant d'extraire les zones de texte formant le complément, [BAI90], [PAV91], [BAI92], [ANT94]. Cette méthode ne fonctionne que si les blocs sont rectangulaires et bien espacés. Selon le même principe d'analyse de l'arrière plan de la page, il a été proposé de suivre et de squelettiser les espaces blancs, rendant ainsi la méthode insensible aux rotations [KIS96]. Ces méthodes sont très sensibles au contenu des documents et au temps de calcul nécessaire à l'amincissement morphologique. Dans le cas particulier de l'analyse des images de courrier, une localisation du bloc adresse par extraction des espaces blancs a été proposée par Kise dans [KIS96] puis améliorée par Yip et Chi [YIP01] en réduisant globalement les temps de traitement. Malgré tout cela, l'extraction et la synchronisation des plages blanches entre elles (nécessaire pour générer la continuité des segments blancs), et la fixation des seuils et des critères d'arrêt dans ce type d'approches posent toujours des problèmes de généralisation.

2.4.1.1.3 Segmentation par suivi de contours ou par pavage

Les méthodes de suivi de contour comme la méthode de Kruatrachue [KRU01] procèdent généralement par balayage vertical périodique de haut en bas pour détecter les zones informatives, puis elles tracent les contours de ces zones. L'inconvénient majeur de ces méthodes est la répétition du balayage point par point de l'image conduisant à une lenteur des traitements.

Les méthodes de segmentation par pavage comme la méthode d'Antonacopoulos et al [AN94] utilisent un lissage vertical pour noircir les zones informatives et des rectangles de différentes tailles pour couvrir le fond de l'image. L'extraction s'applique uniquement sur les bords des rectangles coïncidant avec les bords des zones noircies. Ces méthodes sont peu utilisées dans des applications industrielles en raison des temps de traitement élevés et de leur inefficacité sur des polices de grande taille.



(a) (b)

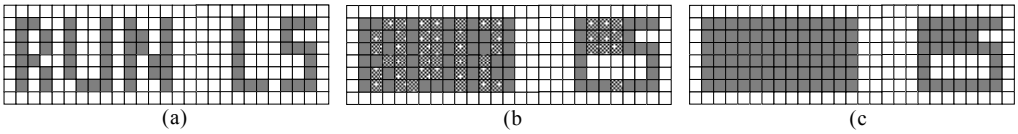
Figure 2. 24 : Exemples de segmentation, **(a)** par suivi de contour, **(b)** par pavage.

2.4.1.2 Mécanismes de segmentation ascendants (bottom-up)

Ils caractérisent les techniques de segmentation procédant par fusion hiérarchique des entités physiques. La fusion se poursuit jusqu'à ce que la structure complète qui représente la racine de la hiérarchie de l'image traitée soit obtenue. Dans ce genre de techniques, la fusion est fondée, en général, sur une analyse des caractéristiques locales par rapport à chaque niveau (pixels noirs, séquences noires, composantes connexes, mots, lignes, ou blocs), [LEB92]. Si de telles techniques sont plus efficaces que celles s'inscrivant dans une stratégie descendante, elles sont toutefois moins fiables que ces dernières pour deux raisons essentielles : la propagation des erreurs locales de fusion le long des lignes de segmentation et la fusion de deux composantes connexes qui ne font pas partie du même bloc et qui conduisent à la fusion incorrecte de leurs blocs.

2.4.1.2.1 Segmentation par lissage RLSA

La méthode de Wong et al [WON82] utilise un algorithme appelé RLSA (Run Length Smoothing Algorithm) qui consiste à noircir, dans une image, les petites plages blanches de longueur inférieure à un seuil S fixé pour obtenir des blocs noirs continus. Ce lissage est appliqué horizontalement et verticalement sur l'image, produisant deux images. Un ET logique est appliqué sur ces deux images produisant une image lissée ou image des composantes connexes (figure 2.25, 2.26). A partir de cet algorithme, plusieurs variantes de méthodes ont été proposées pour s'adapter à la nature de nouveaux documents à segmenter [WAH82][YAM96][PAP96][PAR03] [SHI04][SUN05].



(a) (b (c

Figure 2. 25 : Étapes de la méthode RLSA, **(a)** image binaire d'une ligne de texte, **(b)** lissage horizontal avec un seuil = 3, **(c)** résultat de lissage horizontal.

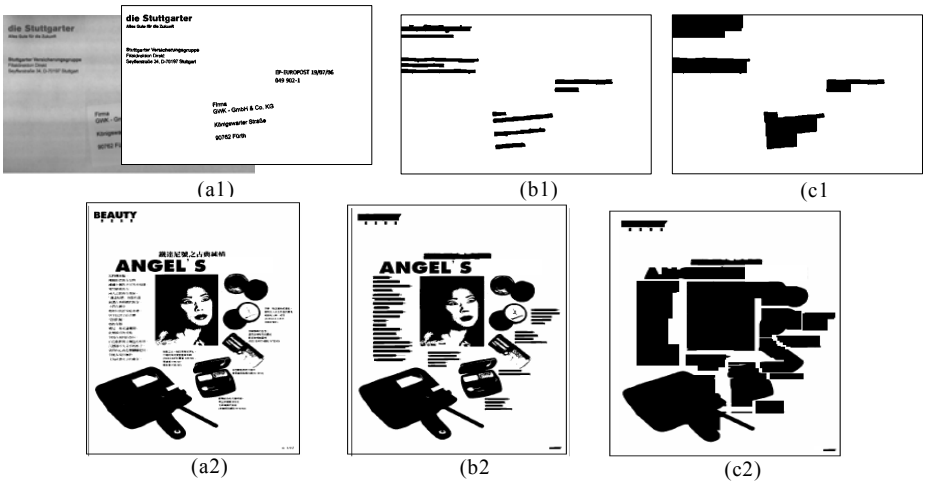


Figure 2. 26 : Application de RLSA sur une image d'enveloppe (**haut**) et sur une image de magazine (**bas**), (**a1,2**)image binaire, (**b1,2**) lissage horizontal , (**c1,2**) lissage vertical.

Le choix du seuil de lissage utilisé par les méthodes RLSA est à la fois crucial et difficile. Le moindre changement de la taille de la police de texte ou de l'espacement entre les objets du document, que ce soit des caractères, des mots, des lignes ou des blocs, affecte beaucoup le résultat de la segmentation. Un seuil légèrement élevé peut causer facilement une sur-segmentation de ces objets alors qu'un seuil légèrement faible risque de leur causer une sous-segmentation. Pour rendre la tâche de segmentation plus robuste, Yamashita [YAM96] a proposé d'utiliser un seuil adaptatif qui s'ajuste en fonction de la taille et de l'espacement entre les objets. Dans le même esprit, Papamarkos et al [PAP96] ont proposé une méthode non-supervisée pour ajuster automatiquement le seuil de lissage par RLSA en fonction de la distribution des séquences (longueurs des plages) noires et blanches. Dans le même principe de RLSA, d'autres méthodes de segmentation utilisent des opérations morphologiques ou des filtres de lissage directement sur les images en niveaux de gris sans avoir recours à une binarisation. Prasanna et al [PAR03] ont utilisé un lissage par filtrage de type passe bas pour former les mots puis les regrouper en blocs dans une application de lecture automatique des adresses indiennes. Shi et Govindaraju [SHI04] ont proposé une méthode plus avancée pour segmenter des documents et des courriers postaux complexes basée sur l'application de longueurs de plages directionnelles floues (fuzzy directional runlength). L'avantage principal de la méthode est sa capacité à détecter la direction dominante et à transformer le contenu en zones texte ou non texte. L'estimation de la direction dominante repose sur l'optimisation de la distance des moindres carrées et sur le regroupement des composantes. La performance de cette segmentation a été testée sur des images de courriers

postaux et sur des images de documents manuscrits anciens comme ceux de Newton, de Galilée et de Washington. Quel que soit le type de textes imprimés, manuscrits ou mixtes le taux de bonne segmentation est de *93%* calculer sur la base USPS de *1864* images.

(a) (b) (c)

Figure 2. 27 : Segmentation d'une enveloppe et séparation de contenu texte / non texte par la méthode de Shi et Govindaraju.

Sur des documents ou sur des enveloppes complexes, ce genre de méthodes ne doit pas être utilisé sans réduire l'échelle de l'image ou changer l'espace de représentation à cause de la grande quantité d'information présente dans l'espace des niveaux de gris.

2.4.1.2.2 Segmentation par fusion ou par regroupement des composantes connexes

Le principe de ces méthodes ascendantes consiste à regrouper ou à fusionner séquentiellement les CCs en éléments de plus en plus importants créant les mots à partir des caractères, les lignes de textes à partir des mots, les blocs à partir des lignes et ainsi de suite. Les propriétés utilisées pour la fusion sont connues a priori ou déduites directement à partir des calculs statistiques sur les espacements de l'image.

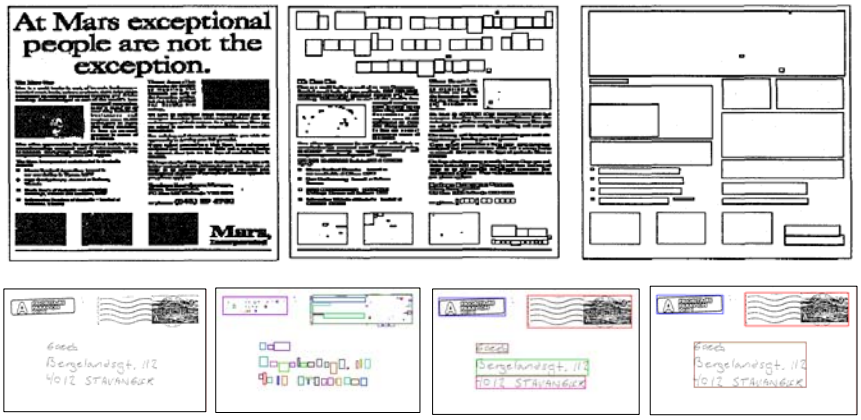


Figure 2. 28 : Exemple de regroupement hiérarchique des composantes connexes appliqué, **en haut**, sur une image d'article, **en bas**, sur une image de courrier manuscrit.

L'approche de regroupement proposée par Drivas et Amin [DRI95] repose sur l'analyse de voisinage entre les CCs et sur le regroupement des CCs ayant une même dimension. La méthode est destinée à la segmentation

des documents extraits de rapports techniques, de revues et de cartes de visites. D'autres techniques de regroupement ascendant ont été déjà proposées visant des applications de vision industrielle [PAL92] [LII93] [JEO04]. Palumbo et al [PAL92] ont intégrée cette segmentation dans un système de localisation de blocs adresse en temps réel développé dans le centre CEDAR[4] qui a trié au États-Unis plus de *200* milliards courriers en 1992. Jeong et al [JEO04] l'ont également intégrée dans un système de tri de courrier Coréen.

Contrairement aux méthodes descendantes, le regroupement ascendant des CCs n'est pas limité aux blocs de forme rectangulaire ni à l'inclinaison des documents ou à la complexité de l'alignement. Cependant, son inconvénient majeur revient à sa grande sensibilité aux discontinuités des objets, à l'interligne, à l'espacement variable entre les caractères et à la faible résolution de la numérisation. Ces limitations peuvent être réduites par des architectures plus avancées comme la méthode de Simon et al [SIM97] qui s'adapte à une très large gamme de documents contre une complexité de calcul considérablement réduite par rapport aux méthodes ascendantes classiques. Cette méthode utilise une distance particulière entre les composantes, en se basant sur l'algorithme de Kruskal qui recherche l'arbre recouvrant de poids minimum pour détecter la hiérarchie de la structure physique. La méthode a l'avantage d'être indépendante au changement d'espacement du texte et d'alignement des blocs, mais elle peut souffrir des problèmes habituels des stratégies ascendantes, telles que la segmentation incorrecte due à une erreur de regroupements précoces et le coûteux temps de calcul des connexités. Cependant, l'interaction avec d'autres outils de traitement qui calculent les connexités peut être une des moyens utilisés pour éviter de calculer les connexités à nouveau. D'autres méthodes utilisent un regroupement ascendant de pixels au lieu de CCs. La méthode de Etemad et al [ETE97] utilisée dans ce contexte, repose sur des règles de décision floues avec un réseau de neurones pour regrouper les pixels d'un document en plusieurs blocs homogènes.

Les méthodes ascendantes nécessitent en général des connaissances a priori très fortes sur les caractéristiques typographiques des textes. Pour éviter ces limitations ainsi que celles qui sont liées aux méthodes descendantes, des méthodes mixtes ont été proposées par de nombreux auteurs.

4. CEDAR : Center of Excellence for Document Analysis and Recognition.

2.4.1.3 Mécanismes mixtes de segmentation (mi-ascendants, mi-descendants)

Les méthodes de segmentation ascendantes et descendantes apportent des connaissances différentes qu'il ne faut pas négliger lorsque l'on désire augmenter la robustesse. Contrairement aux méthodes classiques qui comportent plusieurs sources d'erreurs, les méthodes mixtes rassemblent les deux mécanismes en même temps. Leur point fort repose sur le fait qu'elles se servent des avantages de l'une pour combler les inconvénients de l'autre. Les méthodes mixtes peuvent êtres décomposées en trois classes selon l'ordre de combinaison : fusion puis décomposition, décomposition puis fusion, interaction.

2.4.1.3.1 Méthodes basées sur la stratégie fusion puis décomposition (merge & split)

Ces méthodes reposent sur un regroupement des éléments les plus petits pour former des blocs de taille supérieure. Par la suite, elles perfectionnent le résultat par une décomposition en sous blocs homogènes. Se basant sur ce principe, la méthode de segmentation proposée par Lee et Kim [LEE94] utilise des caractéristiques qui servent ensuite à la phase de localisation du bloc adresse présent sur les images de documents de type courrier Coréen manuscrit. Les CCs sont regroupées hiérarchiquement en mots, en lignes et en blocs. Puis une étape de décision basée sur la théorie de Bayes est appliquée sur chaque bloc pour valider sa décomposition ou sa fusion avec un autre bloc jusqu'à l'obtention du bloc adresse. Xue et al [XUE99] ont introduit quelques optimisations de cette méthode pour segmenter des enveloppes manuscrites chinoises. Une fois les blocs formés à partir des CCs, la projection des profils est utilisée avec la transformée de Hough pour décomposer chaque bloc en lignes de texte. Jiang et al [JIA07] ont adopté le même principe dans un système plus évolué de reconnaissance des adresses manuscrites chinoises.

2.4.1.3.2 Méthodes basées sur la stratégie décomposition puis fusion (split & merge)

Ces méthodes sont basées sur le principe d'une décomposition de l'image suivie d'un regroupement. Kruatrachue et Suthaphan [KRU01] ont proposé une décomposition de l'image de documents en plusieurs blocs polygonaux. Cette technique extrait les différents blocs par un algorithme descendant de suivi de contour, elle applique ensuite un algorithme ascendant sur les rectangles englobant des blocs pour identifier les blocs de texte, les images et les tableaux.

La méthode de Cinque et al [CIN03] utilise un principe similaire à la méthode XY-cut proposée par Nagy [NAG84] mais au lieu d'appliquer un découpage purement descendant de l'image, elle utilise les lignes hori-

zontales et verticales pour former des petites régions qui sont ensuite fusionnées à l'aide d'un arbre quaternaire, voir figure 2.29.

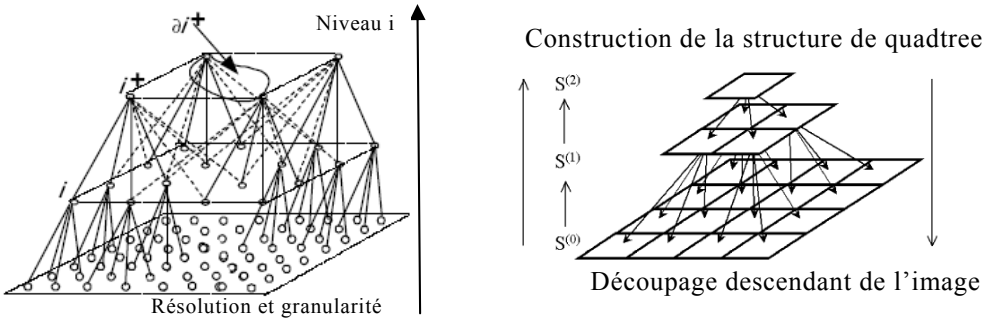


Figure 2. 29 : Principe de la méthode décomposition fusion proposée par Cinque [CIN03].

2.4.1.3.3 Méthodes basées sur une analyse interactive

Ces méthodes utilisent les mécanismes ascendants et descendants simultanément. La méthode proposée par Ramel et Leriche [RAM04] pour segmenter les documents imprimés anciens est un très bon exemple de ces méthodes. Les auteurs ont présenté un algorithme basé sur la superposition de deux représentations de l'image : une carte des formes qui se focalise sur les CCs présentes dans l'image et une carte du fond qui fournit de l'information sur les espaces blancs séparant les blocs constituant la page. En fonction de ses dimensions, l'une des étiquettes suivantes est affectée à chaque CC: Bruit (CC de petite taille), Graphique (CC de grande taille), Texte (autres CC de taille moyenne). La liste des CCs étiquetées Texte est utilisée pour reconstruire les paragraphes. Ce type d'interaction devient très intéressant si on veut effectuer plusieurs traitements (segmentation, séparation entre le texte, le graphique et le bruit) simultanément afin de gagner en temps.

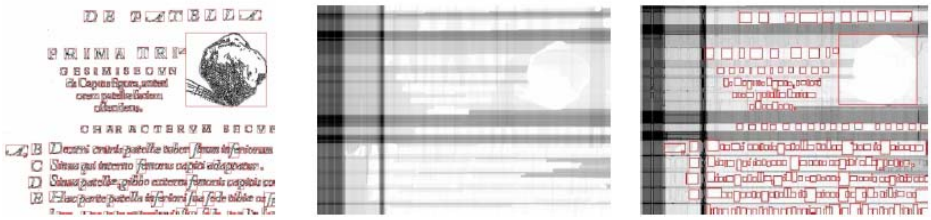


Figure 2. 30 : (a) Carte des formes, **(b)** carte du fond, **(c)** superposition des deux cartes [RAM04].

Han et al [HAN05] utilisent une interaction entre la segmentation et la reconnaissance de caractères dans une application de tri postal. Cette

méthode utilise dans une première phase une projection verticale des profils pour faire émerger les blocs sur des enveloppes manuscrites chinoises. Cette segmentation descendante seule ne suffit pas pour séparer correctement les entités du bloc adresse. Les caractères manuscrits chinois y sont souvent écrits selon différents styles, employant différentes tailles de police, et des espaces intra et inter-caractères souvent très variables. A cela, il faut également ajouter le recouvrement entre caractères qui constitue un facteur majeur d'erreurs de segmentation. Pour y remédier, les auteurs proposent une étape de regroupement ascendant des CCs qui succède à l'étape initiale de segmentation. Ainsi, la séparation des caractères qui se touchent peut être résolue par une interaction avec la phase de reconnaissance de caractères. Cette méthode a été testée sur 589 enveloppes manuscrites, plus de *79,46%* ont été correctement triées, ce qui correspond à un très bon score.

Comme il a été évoqué dans les approches présentées ci-dessus, les méthodes mixtes sont en général plus efficaces et plus utilisées dans le domaine de tri de courrier que les méthodes purement ascendantes et descendantes. Les méthodes mixtes permettent généralement de se dégager des fortes connaissances nécessaires à l'analyse descendante des documents, en évitant dans la plupart des cas de manipuler sur l'ensemble de l'image la totalité des données responsable de la lenteur de certains algorithmes ascendants. De plus, les combinaisons ascendantes et descendantes offrent la possibilité de traiter des documents non contraints (documents contenant des éléments graphiques, des images, des portions de texte d'orientations variées ...) et ont pour ambition commune de parvenir à extraire du texte mêlé à d'autres éléments informatifs, [EGL99][RAM05].

2.4.2 Les innovations par changement d'espace de représentation, sous-échantillonnage ou/et analyse de la texture

Cette famille d'approches innovantes de segmentation ne s'intéresse pas uniquement au processus de segmentation de l'image par l'exploitation d'un des trois mécanismes cités précédemment, mais elle procède par des changements d'espace de représentation (du spatial au fréquentiel par exemple) ou par des changements de résolutions de l'image permettant de considérer les éléments de contenus de manière plus ou moins rapprochée.

Si l'image à segmenter est de grande taille on peut facilement réduire sa taille par des opérations de sous-échantillonnage, évitant ainsi d'analyser la totalité des éléments présents dans les images. Par exemple, si la segmentation porte sur les traits grossiers de l'image, il n'est pas utile de conserver une haute définition pour l'image, au risque de surcharger les

calculs. Dans cette optique, l'analyse multi-résolution offre un cadre idéal de décomposition de l'image en détails et en approximations permettant une reconstruction parfaite des contenus. Un grand intérêt a été accordé à ces techniques multi-résolutions qui par leurs décompositions hiérarchiques garantissent à la fois une localisation globale des régions d'intérêt et un accès ensuite facilité des constituants les plus fins.

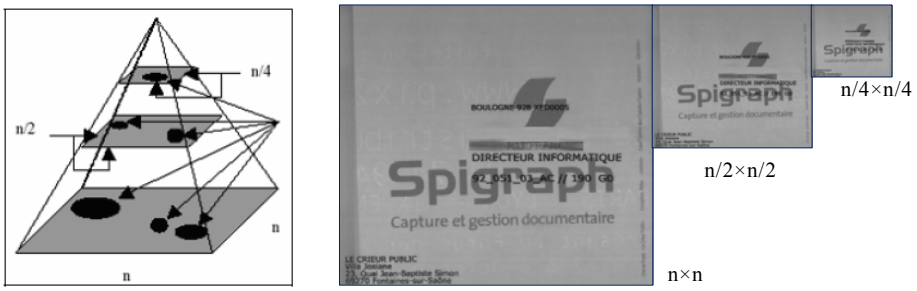


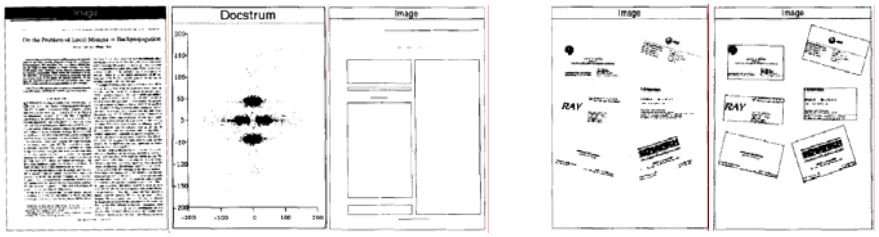
Figure 2. 31: Représentation multi-résolution/multiéchelle.

Déforges dans [DIF94] et [DIF95] ont introduit une méthode ascendante générique basée sur une représentation multi-résolution / multiéchelle de l'image de document utilisée pour localiser le bloc-adresse. Une structure presque similaire a été utilisée par Wang [WAN95] pour distinguer les blocs de textes des blocs de graphiques, et les représenter dans un modèle structurel. Les méthodes consistant à réduire la résolution de l'image ou à analyser les différentes textures contenues dans l'image sont difficilement utilisables sur des images qui contiennent des objets hétérogènes très proches. Les espaces entre éléments de contenus sont indétectables lorsqu'on diminue la résolution : les zones de texte ne présentent alors aucune différence de texture significative.

Les méthodes de segmentation par changement d'espace de représentation (ou par transformation), emploient des transformées mathématiques permettant de changer l'espace de représentation de l'image vers un autre espace où les zones de séparation entre les objets proches deviennent facilement détectables. Ces techniques sont fondées sur une transformation globale des images en vue de déterminer soit les critères de découpe, soit les critères de fusion, soit les deux. L'objectif est d'augmenter la disparité entre les objets pour faciliter leur séparation et d'intégrer à la segmentation, des propriétés d'invariance à certaines déformations que peut subir la forme, telles que la rotation, le changement d'échelle ou la distorsion.

Plusieurs méthodes existent. Parmi elles, on peut citer la transformée de Fourrier [HAS85], [ITT93], la transformée de Hough [LIK94], les décompositions par ondelettes, [LIJ98], les bancs de filtres de Gabor [JAI92b]. D'autres travaux sont fondés sur des transformations plus évo-

luées. O'Gorman [GOR93] a proposé une méthode spectrale, le Docstrum qui consiste en un regroupement hiérarchique des CCs fondé sur une analyse du graphe des k plus proches entités voisines. L'auteur définit par un spectre du document le graphe associé à ce dernier. Les nœuds du graphe sont initialement constitués de CCs. Le Docstrum, mis en place pour la segmentation des documents en blocs textuels, présente l'avantage d'être indépendant de toute inclinaison globale ou locale.



(a) (b)

Figure 2. 32 : Segmentation en blocs par l'analyse de Docstrum **(a)** d'une image de journal, **(b)** d'une image qui contient six cartes de visites de différents formats et orientations.

D'autres techniques innovantes se sont un peu éloignées du concept classique en faisant coopérer la phase de segmentation avec celle de localisation des zones d'intérêt. Govindaraju et Tulyakov [GOV03] proposent dans ce contexte de combiner la phase de localisation du bloc adresse manuscrit avec la phase de segmentation. La méthode utilise un regroupement des CCs par Kmeans dans l'espace des caractéristiques où chaque composante est représentée par un vecteur d'attributs extraits directement à partir de leurs contours. Ce changement d'espace rend la méthode invariante à la rotation et au changement d'échelle. Cela facilite le regroupement des composantes en ensembles homogènes. Basée sur des règles heuristiques, cette approche permet ensuite de déceler le bloc qui correspond le mieux au bloc adresse.

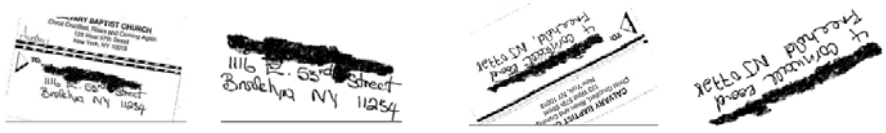


Figure 2. 33 : Exemple de segmentation de blocs d'adresse manuscrite et exemple de rejet des blocs non pertinents par la méthode de Govindaraju et Tulyakov.

Les auteurs ont testé cette méthode dans un système de tri connu sous le nom HWAI développé au CEDAR [PAL92]. Les résultats montrent que cette méthode de segmentation de bloc adresse donne de meilleurs résultats que la méthode existante dans le système HWAI. Eiterer et al

[EIT04] ont suivi le même principe en exploitant la représentation des images en niveaux de gris des enveloppes manuscrites par leurs dimensions fractales et en y réalisant une classification par Kmeans des pixels comme éléments de fond, bruit ou objets sémantiques (avec les labels : timbre, cachet postal ou bloc adresse). Ce principe reposant sur une classification simple a permis de réaliser conjointement plusieurs tâches : une extraction de premier plan, la suppression de bruit, la séparation des entités physique et la localisation du bloc adresse. Plus récemment encore, Sun et al [SUN08] ont proposé dans un concept innovant, une combinaison entre plusieurs techniques d'analyse de données pour segmenter les documents à partir de l'analyse de leurs textures et de connaissances contextuelles. Cette méthode exploite les aspects multi-échelles autour d'une segmentation Bayésienne et porte sur la transformée en ondelettes et un modèle d'arbre de Markov caché pour isoler la texture de texte de celle de graphique et de bruit. Ce type de combinaison de techniques est très coûteux en temps de calcul car il ne vise pas des applications de temps réel et se focalise en priorité sur la réduction des erreurs de segmentation sans être limité par le facteur de temps.

2.4.3 Optimisations des temps de calcul : Vers de nouveaux mécanismes de coopérations

Jusqu'ici, nous avons présenté plusieurs méthodes d'extraction de la structure physique correspondant à des besoins spécifiques, notamment en analyse de courrier, de formulaires ou de documents administratifs. En terme d'évaluation de l'ensemble de ces approches, nous pouvons citer quelques travaux phares du domaine [KAN95][ZHA96][YAN98][THU99] [PEN01]. Nous pouvons ainsi constater qu'un grand nombre de méthodes de segmentation qui s'appliquent en temps réel (sur des documents de type courriers et formulaires) s'oriente vers l'emploi d'un mécanisme mixte (mi-ascendant / mi-descendant). Les méthodes descendantes sont rapides mais moins précises sur des documents de structure complexe ou variable, alors que les méthodes ascendantes sont plus précises mais très consommatrices en temps de calcul. L'alliance des deux mécanismes a montré que les meilleurs compromis temps / précision pouvaient être obtenus. On constate aussi que l'intérêt actuel du domaine de la vision industrielle se porte davantage vers de nouvelles approches innovantes, où l'on cherche à s'affranchir des modèles classiques en ajoutant notamment à la coopération ascendant/descendant les propriétés intéressantes de la multi-résolution, de la hiérarchie de décomposition et des changements d'espace de représentation.

Dans un marché compétitif, une segmentation sans erreur n'existe pas en pratique, mais elle représente une référence de convergence pour ceux qui veulent conquérir le marché de tri automatique de documents et de courriers. Il faut noter qu'à ce jour, il n'existe pas encore de critères d'extraction de la structure physique suffisamment génériques à toutes les classes ou à toutes les qualités de documents. Toutefois, la combinaison de la tâche de segmentation avec d'autres tâches peut permettre de choisir les critères les plus adéquats. Nous sommes partis de l'hypothèse forte que la reconnaissance du type de document, la séparation texte / non texte des zones de document, la dichotomie manuscrits / imprimés, l'estimation de la complexité, de la qualité et de la résolution du document peuvent être obtenues par une segmentation grossière du document et peuvent en même temps permettre de sélectionner automatiquement l'ensemble des critères pour aboutir à une segmentation cette fois plus fine des contenus. Cette combinaison de tâches doit permettre au système de tri d'avoir certaine autonomie et une bonne adaptation au contenu pour obtenir une meilleure séparation des éléments constitutifs en temps réduit. Nous allons porter toute notre attention dans la suite des chapitres a exposé une méthodologie générique répondant à ces hypothèses fondamentales.

Pour finir, l'analyse des approches de segmentation que nous avons produit nous montre qu'une architecture modulaire ne permet pas à elle seule d'assurer toute la souplesse d'échanges inter-modulaires nécessaire pour faire interagir la segmentation avec les autres tâches. Sans cette interaction toute tentative d'amélioration de la segmentation devient très délicate, elle est d'ailleurs toujours aussi bien exprimée à travers le paradoxe de Sayre [Sayre73] « *pour reconnaître une entité, il faut savoir la localiser, mais pour la localiser, il faut tout d'abord la reconnaître* ».

Partant de l'ensemble de ces constatations, notre proposition s'inspire de la théorie des graphes. Elle consiste à utiliser une stratégie de segmentation pyramidale mixte plus adaptée aux courriers postaux. Les étapes de haut niveau reposent sur un outil qui permet d'assurer une bonne séparation entre les entités physiques permettant à la fois une combinaison souple et efficace entre la phase de segmentation et les autres phases de traitement et d'analyse comme la localisation des zones d'intérêt et la reconnaissance due type de document.

2.5 Les méthodes de discrimination texte/non texte

Pour reconnaître le texte dans un document quel qu'il soit, il faut pouvoir l'isoler ou l'extraire du document (figure 2.34). Dans le cas des courriers ou des documents d'entreprise, la structure physique à extraire est souvent composée de deux couches distinctes selon des critères purement structurels ou géométriques:

- une couche textuelle qui comporte l'essentiel de l'information contenue dans le document, généralement, constituée de caractères alpha-numériques.
- une couche non textuelle qui peut contenir des graphiques, des tableaux, du bruit, et d'autres informations additionnelles.

Après l'étape de segmentation des documents en plusieurs blocs constitutifs, les méthodes de séparation servent, en général, à distinguer les blocs textuels des autres blocs non textuels. Cette tâche représente donc l'une des tâches les plus importantes dans la chaîne de tri automatique de courriers et de documents d'entreprises (factures, formulaires, planus…) puisque elle permet un accès rapide à l'information et une reconnaissance (adresse de destination, codes, zones textuelles d'intérêt sur les formulaires et les documents bancaires) plus rapide également. Les méthodes qui abordent cette problématique se divisent généralement en deux grandes familles d'approches : méthodes basées sur l'analyse de la texture et méthodes basées sur l'analyse des composantes connexes.

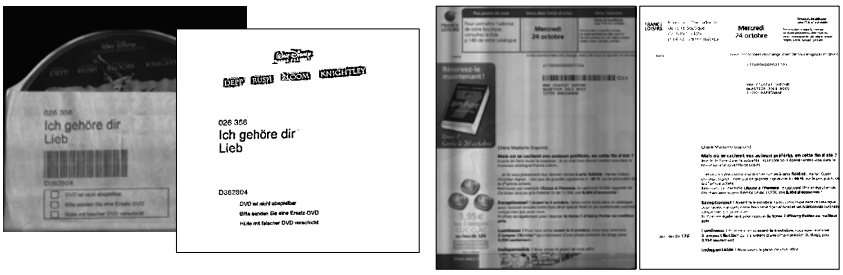


Figure 2. 34 : Exemple de séparation texte/non texte sur des pochettes de DVD.

2.5.1 Méthodes basées sur l'analyse de la texture

Dans ces méthodes, le problème de séparation texte/ non texte peut être traité comme un cas particulier de segmentation de la texture où les caractères de texte sont analysés comme des entités texturées. Dans ce cas, l'image d'entrée est habituellement considérée comme un composé de deux (texte et non-texte) ou de trois (texte, graphiques et fond) classes de texture. La séparation entre ces classes utilise généralement une fenêtre de classification ou un bloc d'une certaine taille dans l'espoir que toute ou partie des pixels contenus appartiennent à la même classe de texture.

Dans ce contexte, Wang et Srihari [WAN89] ont présenté une technique de séparation entre le texte, les graphiques et les photographies.

La méthode est fondée sur le calcul de transitions de zones noires et blanches correspondant à certaines régularités dans le dessin du caractère. Elle consiste à calculer les occurrences de paires NB (noir / blanc) et de triplets NBN (noir / blanc / noir) correspondant aux transitions noires et blanches horizontales dans les régions de l'image du document. Une paire NB est un ensemble de pixels noirs suivis par un ensemble de pixels blancs dans la direction horizontale (figure 2.35). La longueur de la paire est égale au nombre total de pixels contenus dans la paire. La proportion de pixels blancs dans une paire est codée par un entier compris entre 1 et 9, appelé catégorie.

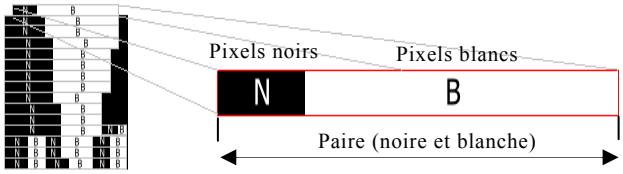


Figure 2. 35 : Séquence NB des pixels (noirs/blancs).

Les résultats d'extraction des paires *NB* sont rangés dans une matrice M_1, où $m_1(i,j)$ exprime le nombre de fois où l'on rencontre dans l'image des paires *NB* de longueur j et de catégorie *i*. De manière similaire, on détermine une matrice M_2, dans laquelle on range les triplets *NBN*. La longueur du triplet est donnée par la longueur de la zone blanche. La longueur des pixels noirs est donnée par une catégorie de *1* à *3*. $m_2(i,j)$ donne le nombre de triplets de longueur j et de catégorie *i*. Ensuite, on déduit à partir de ces matrices trois quantités F_1, F_2, et F_3 exprimant des mesures de texture :

$$F_1 = \frac{\sum_{i=1}^{n_1}\sum_{j=1}^{L_1}\left(\frac{m_1(i,j)}{j^2}\right)}{\sum_{i=1}^{n_1}\sum_{j=1}^{L_1} m_1(i,j)} \qquad (2.\ 12)$$

Où n_1 est le nombre de catégories et L_1, la plus grande longueur d'une paire. On constate que F_1 a une valeur plus grande pour les petites lettres que pour les grandes lettres car les espacements entre les tracés dans les petites lettres sont plus petits que dans les cas de grandes lettres. Par conséquent, F_1 a la valeur la plus importante pour les photographies.

$$F_2 = \frac{\sum_{i=1}^{n_1}\sum_{j=1}^{L_1}\left(j^2 \cdot m_1(i,j)\right)}{\sum_{i=1}^{n_1}\sum_{j=1}^{L_1} m_1(i,j)} \qquad (2.\ 13)$$

F_2, décrit les grandes transitions et a donc une valeur importante pour les grandes lettres.

$$F_3 = \frac{\sum_{j=s_1}^{L_2} j^2 \left(\sum_{i=1}^{n_2} p(i,j) \right)}{\sum_{j=s_1}^{L_2} \sum_{i=1}^{n_2} p(i,j)} \quad \text{avec} \quad p(i,j) = \begin{cases} m_2 & si \quad m_2(i,j) > s_2 \\ 0 & si \quad m_2(i,j) \le s_2 \end{cases} \qquad (2.\ 14)$$

où L_2 est la plus grande longueur d'un triplet, n_2 le nombre de catégories et s_1, et s_2 sont les valeurs de seuils choisis pour mettre l'accent sur les traits longs. Une classification par discrimination linéaire dans l'espace des paramètres F_1, F_2 et F_3 permet, dans une image donnée, de séparer le texte, les graphiques et les photographies.

Un autre algorithme basé sur l'analyse des séquences noires a été proposé par Sivaramakrishnan et al [SIV95]. Afin d'être classée dans l'une des neuf classes différentes, chaque région est caractérisée par la moyenne et la variance des longueurs des séquences noires et blanches, la moyenne et la variance spatiale, le pourcentage des pixels noirs dans la région, et le rapport de largeur de la région avec la largeur de ses colonnes. Les caractéristiques relatives aux séquences sont calculées selon les quatre directions canoniques (horizontale, verticale, diagonale gauche et diagonale droite). Un arbre de décision est utilisé pour classer chaque région sur la base de son vecteur de *67* caractéristiques. Cette technique utilise la projection de profils pour faire émerger les lignes de texte et donne de meilleurs résultats lorsque les documents sont bien redressés.

La technique proposée par Jain et Bhattacharjee [JAI92a] a été dédiée à un système de lecture automatique de l'adresse de destination de courrier postal. L'extraction des zones textuelles utilise 8 filtres de Gabor selon les fréquences centrales et les orientations suivantes :

$$\begin{aligned} f_0 &= 32\sqrt{2} \rightarrow \quad \theta = 0° \quad \theta = 45° \quad \theta = 90° \quad \theta = 135° \\ f_0 &= 64\sqrt{2} \rightarrow \quad \theta = 0° \quad \theta = 45° \quad \theta = 90° \quad \theta = 135° \end{aligned} \qquad (2.\ 15)$$

Chaque filtre permet de caractériser la texture dans une orientation différente. L'ensemble de caractéristiques est utilisé par la suite pour réaliser un clustering non supervisé afin de classer chaque pixel comme texte ou non - texte. Cette méthode est invariante à la rotation et robuste au changement de polices et de tailles des caractères de texte, mais elle est moins robuste sur des enveloppes et des documents complexes.

Figure 2. 36 : Localisation de texte manuscrit sur des enveloppes par la méthode de Jain et Bhattacharjee.

WU et al [WUV99] ont proposé un système d'extraction automatique de texte sur des images de différentes sources (chèques, vidéo, journaux, petites annonces, certificats d'actions et photographies). Dans ce cadre, le texte est traité comme une texture distincte et sa détection repose sur une segmentation multi résolution de la texture et sur des critères de cohérence spatiale. La première phase du système consiste en un filtrage linéaire suivi d'un filtrage non-linéaire pour segmenter la page en régions de différentes textures. Pour cela neuf opérations de convolutions de type dérivées de gaussienne du second ordre $\sigma = \{1, \sqrt{2}, 2\}$ sont utilisées. Puis l'image subit une transformation non-linéaire de type *tanh(αt)* avec $\alpha = 0.25$. En utilisant les sorties délivrées par cette transformation, certaines énergies sont mesurées localement pour chaque pixel sous forme d'un vecteur de caractéristiques. Une classification de type K-means (avec $K=3$) est appliquée sur l'ensemble des vecteurs de caractéristiques pour pouvoir séparer les différentes formes de texture présentes sur la page.

La seconde phase du système consiste à appliquer un ensemble d'heuristiques conçues pour générer les différentes couches de symboles d'une même classe de texte. Pour détecter le texte dont la taille de police change de manière significative, les couches issues d'une hiérarchie de trois niveaux sont combinées dans une troisième phase. Les boîtes englobant le texte contiennent des intensités similaires. Un algorithme basé sur l'analyse de l'histogramme est donc suffisant pour calculer le seuil de binarisation des zones textuelles. Une étape de raffinement est utilisée par la suite pour supprimer quelques fragments non textuels qui peuvent être présents dans les zones textuelles.

Cette méthode perd son efficacité dans les situations où la texture des images ressemble à celle du texte (présence de feuilles d'arbres ou d'herbes. Semblables aux petites formes des caractères de texte). Huiping et al dans [HUI00] traitent cette situation difficile, souvent rencontrée dans les séquences vidéo de scènes naturelles, par un réseau de neurones à trois couches. Cette approche permet de décider localement si le contenu de la fenêtre glissante (typiquement d'une taille 16×16) est textuel ou non. Pour faciliter la détection de diverses tailles de caractères de textes, ce concept utilise une représentation pyramidale de l'image initiale. Cette technique montre plus d'efficacité sur ce type de document par rapport aux méthodes non-supervisées proposées dans [JAI92b] et [WUV99].

D'autres méthodes font coopérer la phase de l'extraction de zones textuelles avec la phase de l'estimation de l'inclinaison des lignes de texte. Dans ce contexte, Zhu et Yin [ZHU02] utilisent la transformée de Fourier des projections de profils pour définir un vecteur de caractéristiques pour chaque bloc. Une machine à vecteurs de supports SVM est ensuite utilisée

pour classer chaque bloc comme textuel ou non. Cette séparation permet d'avoir une bonne estimation de l'angle d'inclinaison sur les lignes de texte en évitant la prise en compte d'éléments graphiques ou bruités.

Plus récemment, Journet et al dans [JOU05] proposent dans leur article une méthode de caractérisation d'images de documents imprimés datant de la Renaissance. Cette approche se base sur une extraction des différentes orientations présentes sur la totalité de la surface de la page et qui sont caractéristiques de la présence de différentes entités textuelles, ou graphiques (incluant les enluminures, les ornements et bandeaux, les lettrines, ainsi que diverses illustrations). Cette caractérisation s'appuie sur le calcul et l'exploitation de la fonction d'autocorrélation qui a la particularité, lorsqu'elle est estimée sur une zone de texte ou de dessin, de générer une signature unique facilement identifiable. Ce choix permet de séparer le texte des dessins minimisant la quantité d'a priori relatif aux images traitées. Cette méthode est non seulement robuste aux bruits fréquemment rencontrés dans ce genre d'images (détériorations de l'encre, défauts de numérisation…) mais elle se veut aussi complètement indépendante de la typographie employée, de la taille des caractères, de la présence de parties graphiques.

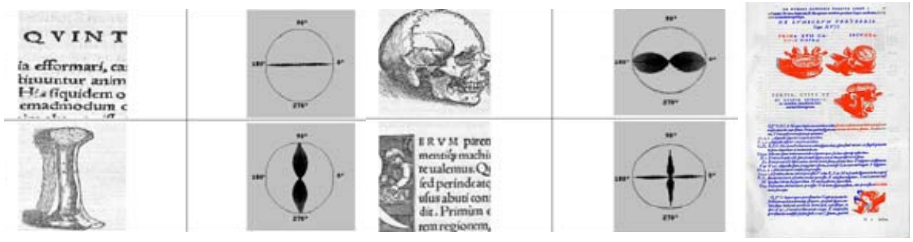


Figure 2. 37 : Exemples de roses des directions et détermination de zones graphiques [JOU05].

2.5.2 Méthodes basées sur l'analyse des composantes connexes

Ces techniques sont essentiellement basées sur l'analyse des composantes connexes, des espaces qui les séparent et de leur taille, leur régularité et même leur texture.

La méthode de Wong, Casey et Wahl [WON82] en est un exemple très connu. Elle repose sur les approches agrégatives classiques de type RLSA qui consiste à noircir, dans une image, les petites plages blanches de longueur inférieure à un seuil *S* fixé pour obtenir des blocs noirs continus. Ce lissage est appliqué horizontalement et verticalement sur l'image, produisant deux images. Un "ET" logique est ensuite appliqué sur ces deux images produisant une image lissée ou image des composantes connexes (figure 2.38).

Figure 2. 38 : Lissage par la méthode RLSA avec un seul s=15.

Leur méthode de séparation de zones utilise l'image des composantes connexes et se base sur le calcul de quatre paramètres:

- la hauteur H ;

-la densité D (ou taux de pixels noirs dans le rectangle circonscrit de la connexité)

- la longueur moyenne L des traits noirs de la composante dans l'image d'origine.

- l'excentricité E= largeur/hauteur.

Les quantités L et H permettent de localiser les lignes du texte, car elles expriment une certaine régularité au sein de formes de petites tailles. Les quantités E et H permettent de distinguer entre les lignes verticales, les graphiques et les photographies. Cette méthode est plus rapide que les méthodes basées sur l'analyse de la texture car elle ne nécessite pas trop de transformations. Cependant elle est sensible à l'inclinaison des documents et nécessite une bonne orientation et un parfait alignement des lignes de texte.

Pour augmenter la robustesse de cette approche Fletcher et Kasturi [FLE88] ont appliqué la transformé de Hough sur des composantes connexes colinéaires de tailles similaires. Ils les ont ensuite regroupées en chaînes de texte. Certaines améliorations ont été apportées par Tombre et al dans [TOM02] pour l'adapter aux documents graphiques de structure riche. Un post-traitement a également été proposé pour récupérer les composantes textuelles attenantes aux graphiques.

Dans le cadre d'une application de tri postal, Jain et Yu [JAI96] ont proposé une méthode de localisation automatique de bloc adresse en temps-réel sur des courriers complexes. La méthode repose sur une classification directe des lignes de texte généralisées (GTLs) issues d'une segmentation ascendante par regroupement hiérarchique des composantes connexes en caractères, en lignes et en blocs homogènes. La technique analyse les

tailles et l'alignement horizontal des composantes de chaque GTL pour les classer comme textuelle ou non.

Dans cette famille d'approches, nous pouvons également citer les travaux de Belaïd et Akindele dans [BEL93] qui proposent une classification des blocs selon les catégories suivantes : petit texte, texte moyen, grand texte, graphiques et photographiques. Leur méthode repose sur l'analyse des composantes connexes, les espaces inter composantes, leur taille et leur régularité.

2.5.3 Conclusion

Parmi l'ensemble des approches proposées pour la discrimination texte/non-texte, celle qui nous semble la plus générique est la méthode de Zheng et al [ZHE03]. En effet, elle porte sur la combinaison d'un module de séparation texte/ non texte et un module de séparation imprimé/ manuscrit. Cette coopération permet de gagner du temps et de la robustesse en n'appliquant qu'une seule classification. Nous l'avons détaillée dans la section « Séparation imprimé/ manuscrit »

2.6 Le cas particulier de la séparation imprimé/manuscrit

Pour des raisons industrielles et économiques, les outils de lecture automatisée des textes imprimés (OCR) cherchent à être de plus en plus proches de résultats temps réel dans des environnements toujours plus complexes et contraints. On a pu faire le constat au début de ce chapitre que la coopération entre plusieurs méthodes de prétraitement, d'analyse et de reconnaissance de documents était essentielle pour améliorer la qualité de la reconnaissance et s'adapter à des domaines d'application plus difficiles. On a notamment pu constater que la capacité de choisir automatiquement l'algorithme approprié selon le type de données permettait au système de reconnaissance d'avoir une grande autonomie et un meilleur rendement dans le traitement de documents hétérogènes. Dans le monde réel, l'écriture de l'adresse sur le courrier et le remplissage des formulaires ou des chèques peuvent être effectués soit à la main, soit par impression avec différentes polices, soit encore par les deux modes combinés. Une personne pourra, par exemple, écrire à la main l'adresse de son destinataire sur son propre courrier, et une organisation envoyer des lettres à ses clients ou à ses employés en imprimant automatiquement l'adresse de destination sur l'enveloppe. Enfin, des situations duales où fragments manuscrits et imprimés sont combinés pourront se constater lors du remplissage des chèques bancaires ou

des formulaires, ceux-ci pouvant tout à la fois contenir quelques zones déjà pré-remplies par impression et d'autres remplies à la main (figure 2.39).

Le tri automatique de ces papiers nécessite de localiser les zones textuelles d'intérêt puis de les lire par OCR quelque soit la nature des textes pour finalement les trier en fonction de caractéristiques communes. La présence de texte imprimé et manuscrit dans la même image de document ou dans la même rame d'une chaîne de tri, pose donc des difficultés supplémentaires aux mécanismes d'automatisation de la reconnaissance optique des caractères.

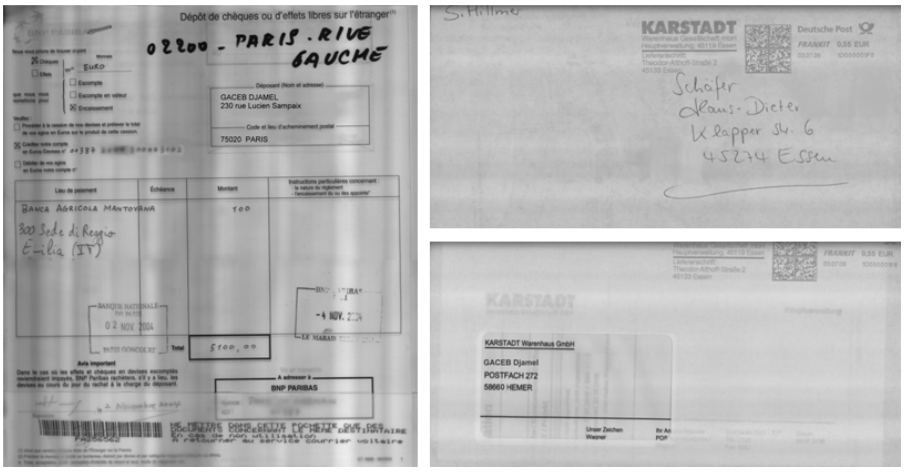


Figure 2. 39 : Exemples de documents traités dans une chaîne de tri.

Quand les deux types de texte se réunissent sur la même chaîne de tri ou sur le même document (formulaires de demande, questionnaires, courriers, chèques, corrections et instructions dans les documents imprimés), il est nécessaire de traiter différemment les deux types de texte et d'apporter des solutions logicielles efficaces permettant :

- La récupération des informations pertinentes (par exemple, identification de l'écriture manuscrite dans un formulaire)
- La suppression des informations inutiles (par exemple, suppression des notes manuscrites dans les documents officiels)
- La reconnaissance de chaque type de texte et l'automatisation du paramétrage selon le type d'OCR employé.

A ce jour, il existe sur le marché différents types d'OCR capables, soit de reconnaître le texte imprimé avec des polices différentes, soit de reconnaître le texte manuscrit. Cependant, il n'existe pas encore d'OCR capable de reconnaître les deux types de textes en même temps. Par conséquent, les systèmes de reconnaissance de documents mixtes utilisent nécessairement deux types d'approches complémentaires: un OCR pour re-

connaître le texte imprimé et un système de reconnaissance (transcription assisté) pour reconnaître le texte manuscrit. Les deux systèmes sont généralement considérés comme des « boîtes noires » rendant toute intervention sur leur fonctionnement propre impossible. En revanche, il est possible de les faire travailler en commutation et de les paramétrer en fonction du type de texte qu'on souhaite lire. La distinction préalable entre le texte imprimé et le texte manuscrit est donc une étape indispensable au système de reconnaissance permettant ainsi de choisir le type d'OCR qu'il faut activer et les traitements appropriés qu'il faut appliquer. Dans la pratique, c'est bien souvent un opérateur qui réalise ces activations par ajustement manuel de l'ensemble des paramètres initiaux en fonction du type de texte à reconnaître.

La littérature dans ce domaine regorge d'outils permettant de discriminer les deux types de texte. Nous allons présenter les mécanismes sous jacents permettant d'y parvenir.

Pour commencer, portons notre attention sur les aspects morphologiques des formes des traits manuscrits et imprimés. De ce point de vue, on peut remarquer plusieurs différences notables entre les deux types de texte. Dans le texte imprimé, les lignes de base sont habituellement droites, les caractères majuscules sont de taille uniforme ainsi que les minuscules et les espacements entre caractères ou composantes connexes. Ces caractéristiques sont largement différentes de celles du texte manuscrit (figure 2.40).

(a) (b)

Figure 2. 40 : Différences entre **(a)** le texte manuscrit et **(b)** le texte imprimé.

L'extraction de ces différences peut facilement assurer la distinction entre les deux types de texte. Généralement, les systèmes de séparation automatique entre le texte imprimé et le texte manuscrit peuvent être décomposés en trois modules (figure 2.41). Le prétraitement et la segmentation en blocs sont utilisés en premier pour réduire les temps de calcul en normalisant et décomposant l'image de document en plusieurs blocs homogènes. Le troisième module du système consiste en une étape de classification, portant sur une extraction des caractéristiques adaptées (par la prise en compte de connaissances a priori et d'une quasi systématique réduction de dimensionnalité). Elle permet précisément au système de décider si le type de texte est imprimé ou manuscrit.

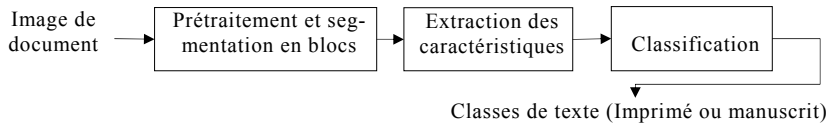


Figure 2. 41: Exemple d'un système de distinction entre le texte imprimé et le texte manuscrit.

Plusieurs techniques ont été développées pour classer les deux types de texte. Les travaux précédents s'intéressent à classer le texte au niveau des blocs, des lignes, des mots ou plus finement au niveau de caractères sur des documents Latins, non-Latins ou bilingues.

Palumbo et al [PAL92] ont intégré un module de discrimination entre le texte imprimé et le texte manuscrit, qui porte le nom HWMP, dans un système de tri automatique de courrier postal en temps réel. Ce module permet de déterminer le type de texte du bloc-adresse issu du module de localisation et de segmentation en blocs. Cette classification est basée sur la fréquence des hauteurs des composantes connexes dans le bloc adresse. Elle suppose qu'un bloc avec des hauteurs largement différentes est manuscrit et un bloc avec des hauteurs de composantes uniformes est imprimé.

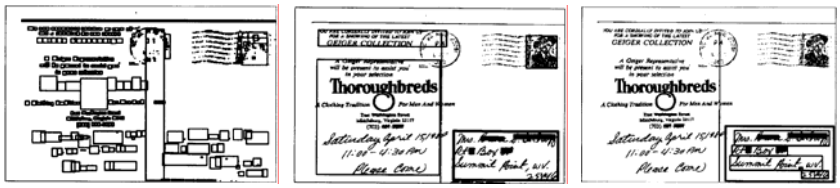
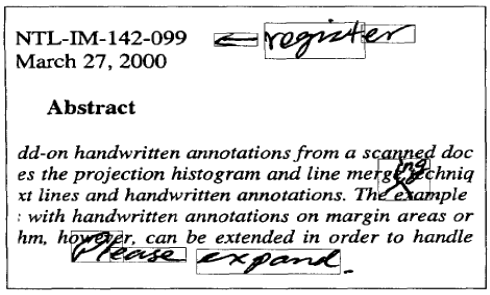


Figure 2. 42 : Exemple de regroupement des composantes connexes en blocs pour localiser le bloc adresse sur une enveloppe.

Franke et Oberlander [FRA93] ont présenté une méthode permettant de contrôler chaque champ d'information, et de savoir s'il est imprimé ou manuscrit. Ils ont utilisé quatre ensembles de caractéristiques géométriques extraites à partir des connexités représentées par des rectangles circonscrits. Un classifieur statistique a été ainsi mis en application pour chaque ensemble de caractéristiques. La méthode proposée par Imade et al [IMA93] segmente les documents japonais selon les classes suivantes : Kanji et Kana imprimés, Kanji et Kana manuscrits, photographies et images imprimées. Des caractéristiques extraites à partir de l'histogramme de gradient et de luminance de l'image du document ont été utilisées comme descripteur alimentant l'entrée d'un classifieur basé sur un modèle de réseau de neurones multicouches de type feed-forward. Ce type de classifieur a également été utilisé sur des caractères Romains par Kuhnke et al. [KUH95] avec trois couches actives composées respectivement de 55, 6 et 2 nœuds sigmoïdaux utilisant des caractéristiques directionnelles et symétriques. Après avoir extrait les contours et les rectangles englobant les ca-

ractères, des mesures de symétrie des pixels et de droiture des segments horizontaux et verticaux sont calculées pour décrire leur structure et leur forme géométrique. Fan et al [FAN98] ont décrit une méthode de classification des blocs en texte manuscrit ou imprimé sur des scripts Anglais, Japonais et Chinois. Chaque bloc est représenté par un ensemble de caractéristiques spatiales et structurelles comme les mesures de variances estimées sur la structure des caractères. Pal et Chaudhari [PAL99] ont utilisé un classifieur d'arbres avec des caractéristiques structurelles et statistiques sur les nœuds de l'arbre pour séparer les lignes de texte de script Bangla et Devanagiri Indien. Les caractéristiques statistiques sont représentées par une variante du calcul de la variance de la structure utilisée dans [FAN98].

Guo et Ma [GUO01] proposent de distinguer des notes écrites à la main sur le texte imprimé ou dans son voisinage. La méthode utilise la projection des profils verticaux pour caractériser les mots segmentés et un modèle de chaînes de Markov caché (HMM) comme classifieur (figure 2.43). Ces méthodes nécessitent une étape de séparation texte/non texte et ne s'appliquent pas facilement à d'autres scripts (qui n'ont pas été appris dans la phase d'apprentissage) ou sur des documents bruités.


NTL-IM-142-099
March 27, 2000

Abstract

dd-on handwritten annotations from a scanned doc
es the projection histogram and line merg
xt lines and handwritten annotations. The example
: with handwritten annotations on margin areas or
hm, however, can be extended in order to handle

Figure 2. 43 : Localisation de notes manuscrites sur un texte imprimé par la méthode de Guo et Ma.

Afin de permettre de traiter également les documents bruités, Zheng et al [ZHE03] ont proposé de traiter le bruit comme une classe distincte en utilisant un mélange de caractéristiques à partir des histogrammes des séquences noires, de la structure des composantes connexes, des histogrammes de croisements, des orientations des segments obtenues par l'application de bancs de filtres de Gabor et de l'analyse de texture issue d'une description par matrices de cooccurrences. Le classifieur de Fisher a été utilisé pour faire la distinction entre le texte imprimé, le texte manuscrit et le bruit. Un champ de Markov aléatoire a également été utilisé dans ce contexte pour corriger les erreurs de classification et modéliser la structure géométrique de chaque classe. L'extraction d'un grand nombre de caractéristiques rend cette méthode très coûteuse en temps de calcul.

Avec un nombre restreint de caractéristiques, Kavallieratou et Stamatatos [KAV04] proposent d'utiliser l'analyse discriminante sur des lignes pour distinguer les deux classes de texte. Le principe consiste à combiner linéairement les variables d'entrée de telle sorte que les classes soient statistiquement les plus distinctes possible. L'idée de l'approche consiste à tirer partie des propriétés structurelles qui contribuent à la discrimination humaine entre le texte imprimé et le texte manuscrit. Il est facilement remarquable que dans une ligne, la hauteur des caractères imprimés est plus ou moins stable et celle des caractères manuscrits est très fluctuante. Les mêmes remarques peuvent être faites pour la hauteur du corps médian des caractères et pour la hauteur des hampes ascendantes et des jambages descendants. Les rapports hauteur des hampes ascendantes/hauteur de corps et hauteur des jambages descendantes/hauteur de corps seraient donc stables sur le texte imprimé et variable sur le texte manuscrit. L'extraction de ces caractéristiques est basée sur le profil des points supérieurs et inférieurs de chaque ligne de texte (c'est-à-dire, la position du premier et de dernier pixel noir sur chaque colonne).

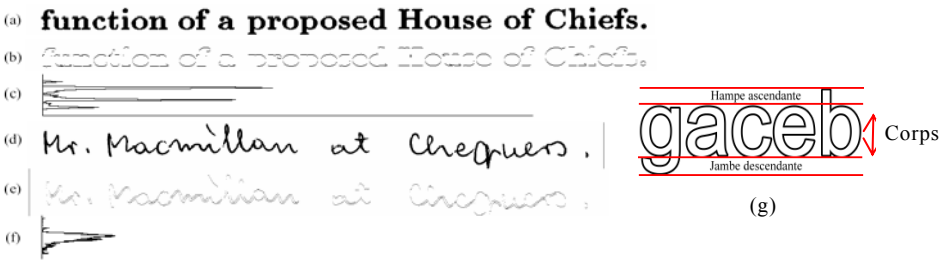


Figure 2. 44 : Exemple de profils des points supérieurs et inferieurs, **(a)** une ligne de texte imprimé, **(b)** son profil des points supérieurs et inferieurs, **(c)** l'histogramme horizontal des profils, **(d)** une ligne de texte manuscrit, **(e)** son profil des points supérieurs et inférieurs, **(f)** l'histogramme horizontal des profils, **(g)** hampes ascendantes, jambages descendants et corps d'une ligne de texte.

Le nombre limité de caractéristiques utilisées par cette méthode augmente la vitesse de traitement mais la projection des profils rend la méthode sensible à l'inclinaison des lignes de texte qu'on rencontre souvent sur les images d'enveloppes en provenance des chaînes de tri automatique. Dans ce contexte, une méthode plus robuste à l'inclinaison des lignes a été proposée par Seung et al [SEU04] pour classer le texte de chaque adresse comme manuscrit ou imprimé dès sa localisation sur le courrier Coréen. Cette méthode consiste à extraire des caractéristiques géométriques à partir de rectangles minima englobant les composantes connexes de l'adresse et à les fournir à un réseau de perceptron multicouche pour la classification de texte. Pour mesurer la régularité des textes, les auteurs proposent trois catégories de caractéristiques : l'histogramme des largeurs, l'histogramme des variances des largeurs et l'histogramme des positions des composantes

connexes. Certaines caractéristiques ont déjà prouvé leur efficacité dans la méthode Franke et Oberlander [FRA93] et l'algorithme de sélection et de regroupement des composantes connexes a été amélioré pour pallier aux différentes dégradations des images et pour réduire les erreurs de segmentation en blocs.

Récemment, Farooq et al [FAR06] ont proposé une méthode de séparation adaptée au script arabe. Cette méthode s'applique directement sur chaque mot pour savoir s'il est imprimé ou manuscrit. Un filtre de Gabor est appliqué sur l'image de mots pour extraire un jeu de caractéristiques selon les différentes orientations des graphèmes (voir figure 2.45). La phase d'apprentissage utilise une classification par maximisation de l'espérance (EM) basée sur un réseau de neurones probabiliste. L'algorithme EM utilisé permet aussi de réduire les effets de sur-apprentissage de la base d'entrainement sur la performance de séparation.

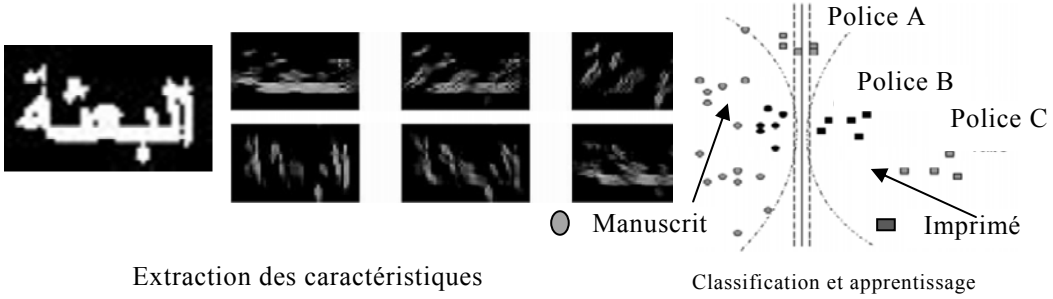


Figure 2. 45 : (**à gauche**) La sortie de filtre directionnel de Gabor selon six directions, (**à droite**), classification et apprentissage sur des images de mots.

D'autres méthodes qui s'appliquent sur les caractères utilisant une analyse plus fine ont été proposées plus récemment dans [MAJ06] et [LAK07]. Majumdar et Chaudhuri [MAJ06] ont proposé un nouveau concept de lecture automatique de formulaires en appliquant une coopération entre la phase de séparation et la phase de reconnaissance des chiffres écrits en script Bangla indien. La classification utilise un réseau de neurones de type perceptron multicouche pour séparer et reconnaître les chiffres manuscrits en même temps que les chiffres imprimés selon 23 polices différentes. Ce principe de combinaison a été adopté également par Lakshmi et al [LAK07] pour reconnaitre les chiffres manuscrits et imprimés des chèques et des enveloppes. Les caractéristiques sont basées sur l'histogramme des orientations des contours et la classification est basée sur l'ACP.

La moyenne des taux de séparation entre le texte imprimé et le texte manuscrit obtenue par les méthodes présentées ci-dessus atteint un taux de *96%*. Ce chiffre reste loin d'être idéal dans des chaînes de tri qui

doivent pouvoir traiter jusqu'à plusieurs dizaines de millions de courriers chaque jour. La méthode de Seung et al [SEU04] atteint un meilleur taux de séparation à *98,9%* suivi par la méthode de Pal et Chaudhari [PAL99] à *98%* (figure 2.46). Grâce à leur cohérence avec la phase de prétraitement et de segmentation, ces deux dernières méthodes ont été conçues de telle sorte à pouvoir s'adapter aux exigences du domaine du tri automatique de courriers. A l'heure actuelle toute tentative d'amélioration de la séparation imprimé / manuscrit dans ce domaine doit donc préférentiellement s'orienter selon ce principe. La figure suivante montre les taux de bonne séparation pour les principales approches présentées dans ce chapitre.

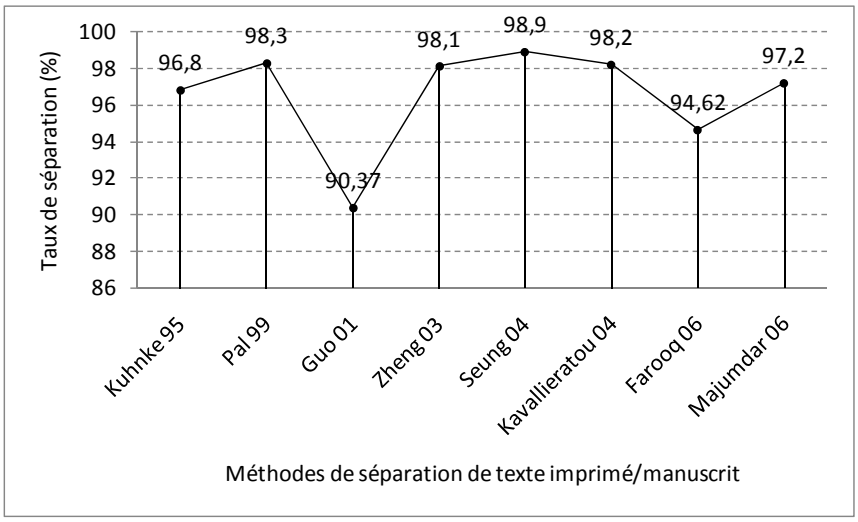


Figure 2. 46 : Taux de séparation de texte imprimé / manuscrit.

Chapitre 3

Les méthodes fondamentales de localisation et de reconnaissance

3.1 Introduction

Le tri automatique de documents est un terrain d'expérimentations de nouvelles technologies de premier choix car à lui seul il contient toutes les étapes d'analyse allant du niveau le plus bas (prétraitement et segmentation) au niveau le plus élevé (reconnaissance et décision). Les projets techniques observés ces dernières années ont permis des gains de temps et d'argent très importants pour les organisations bénéficiaires de tels progrès. Les tendances les plus visibles actuellement se portent sur l'augmentation de la précision et de la pertinence des approches de reconnaissance embarquées permettant notamment de traiter des images de contenus hétérogènes (présence d'écritures différentes imprimées ou manuscrites) mais aussi de procéder à une reconnaissance intelligente du document en s'intéressant notamment à sa structure et sa mise en page.

A ce titre, la reconnaissance automatique des documents (RAD) fait partie intégrante d'un processus complet d'analyse des documents en permettant de classifier les documents selon leur typologie. La connaissance délivrée par cette étape préalable aux traitements permet de cibler les informations pertinentes au tri et choisir un jeu de traitements plus adapté au contenu.

Malgré l'intérêt réel de tel processus de RAD au sein du système de tri, on constate à ce jour des résultats encore imparfaits car des difficultés non résolues doivent encore faire l'objet de recherches. Nous verrons notamment dans ce chapitre les composants essentiels des système de RAD, leurs performances réelles au regard des performances attendues liées notamment à des contraintes qu'il est à ce jour encore difficile d'intégrer totalement : contraintes de temps réel, limitation de l'intervention manuelle en cas de rejet, prise en compte des aléas de la numérisation, superposition fréquente de couches d'informations textuelles et graphiques… Ces contraintes impliquent d'avoir à la fois une description simple des contenus et une description discriminante de la structure afin de garantir un classement rapide de tous les documents susceptibles d'apparaitre dans la chaîne.

Dans ce contexte, nous présentons en première partie de ce chapitre les outils généraux de classification et leur application à la classification de documents en argumentant leurs forces et leurs limites.

La localisation du bloc adresse constitue un point central dans l'élaboration d'un système de tri. Elle consiste en une succession d'étapes allant de l'émergence des blocs informants à l'étiquetage et la décision. Nous aborderons dans ce chapitre les différentes catégories de méthodes. Nous verrons notamment que les méthodes mises en place pour l'analyse d'images binaires qui exploite des stratégies de segmentation mixte (mi-ascendantes et mi- descendantes) sont plus performantes que les approches

exclusivement ascendantes ou descendantes. De même, nous monterons que les méthodes basées sur des ensembles de règles déterministes qui sont encore à ce jour les plus courantes nécessitent un nombre très élevé de critères et des connaissances en grand nombre qui conduisent à des délais de prise de décision souvent très importants.

Dans ce contexte, nous avons porté notre attention sur les solutions basées sur des mécanismes d'apprentissage qui s'avèrent beaucoup plus souples et qui permettent de contrôler des situations complexes (mises en page complexes ou structures variables) et nouvelles de façon optimale. Dans ce chapitre, nous passerons notamment en revue différentes approches de LBA et argumenterons les mécanismes relevant des réseaux de neurones qui s'avèrent très efficaces car possédant de très bonnes propriétés de prédiction. Même si leur validation est souvent difficile à argumenter, ces approches n'en demeurent pas moins de très bons modèles d'analyse ayant la capacité de s'accommoder de valeurs très bruitées ou des données partielles.

3.2 Première partie : La reconnaissance automatique du type de document (RAD)

Les documents échangés entre les entreprises sont nombreux et leur traitement automatique devient une nécessité. En effet, il contribue à créer une réelle valeur ajoutée à l'entreprise en valorisant son patrimoine documentaire et en le rendant plus accessible. La mise en place de nouveaux services liés à l'automatisation de ces traitements contribue à améliorer son propre processus organisationnel.

Dans ce contexte, le tri automatique de documents permet aujourd'hui un gain de temps et d'argent considérable pour les organisations. Les tendances les plus visibles actuellement se développent dans l'amélioration des systèmes de vision, la fiabilité des calculateurs et les performances des OCR au cœur même des systèmes de traitement automatique des documents. Ces dernières années, nous avons pu observer de notables évolutions technologiques qui ont permis d'affiner la précision et la pertinence de la reconnaissance. Non seulement, elles ont permis de prendre en compte des écritures différentes (imprimées ou manuscrites) mais aussi de procéder à une reconnaissance intelligente du document en s'intéressant notamment à sa structure et sa mise en page.

De manière très générale, on peut affirmer que tout système de reconnaissance de documents nécessite l'introduction de connaissances liées au type de document à reconnaître. Dans la plupart de ces systèmes, la con-

naissance est totalement dissimulée dans le code et est, de ce fait, difficile à adapter à de nouveaux types de documents. On y retrouve les documents structurés dont la mise en forme est relativement stable avec des éléments de mise en page récurrents (logo, entête, zones de texte, etc.). On peut citer à titre d'exemples : les factures, les bons de livraison ou encore d'autres documents comptables émanant de fournisseurs et partenaires de l'entreprise.

Dans ce contexte, nous pouvons affirmer que la reconnaissance automatique des documents (RAD) fait partie intégrante d'un processus complet d'analyse des documents en permettant de classifier les documents selon leur typologie (figure 3.1). La connaissance délivrée par cette étape préalable aux traitements permet de cibler les informations pertinentes au tri et choisir un jeu de traitements plus adapté au contenu.

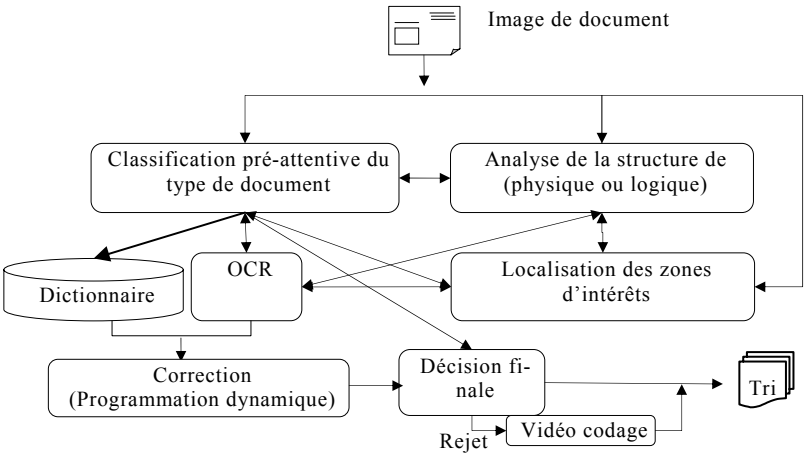


Figure 3.1 : Architecture modulaire d'un système de tri.

Malgré l'intérêt théorique d'une telle modélisation, les résultats obtenus à ce jour sont loin d'être parfaits. L'introduction de la RAD dans une application de tri reste un problème complexe qui bute encore sur des difficultés non résolues. Elle fait l'objet à ce jour de nombreuses recherches qui doivent s'adapter aux contraintes essentielles des systèmes de vision industriels :

- un fonctionnement en temps réel (quelques fractions de secondes doivent suffire à la reconnaissance),
- la maîtrise de la qualité des résultats (le système doit être le plus performant possible pour éviter le coûteux traitement manuel).
- le type de document doit être identifié automatiquement malgré les aléas de l'étape de numérisation (rotations, décalages, plissement),

- une résolution spatiale des images élevée (300 dpi),
- la superposition quasi systématique de couches d'informations (tampons, notes manuscrites, …).

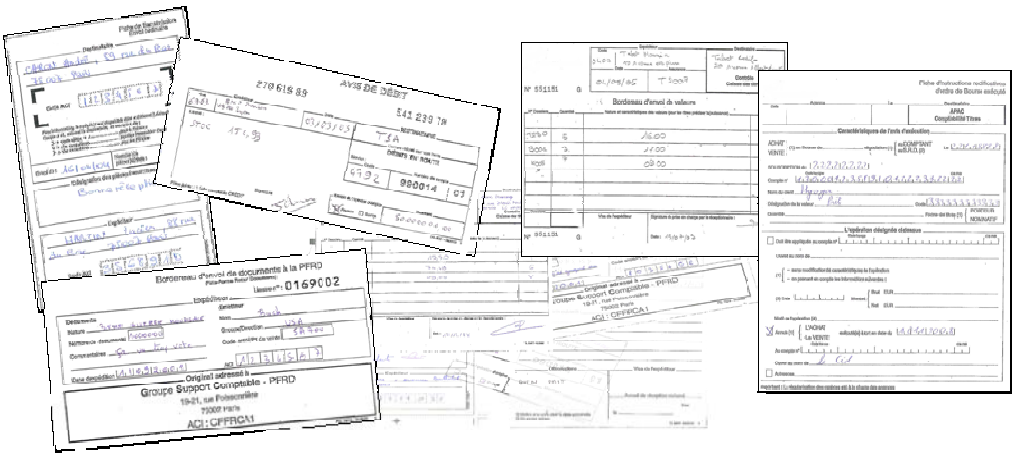


Figure 3. 2 : Une très grande variété de documents dans l'entreprise.

Dans cette section, nous allons donc présenter les composantes essentielles d'un système de classification de documents. Nous aborderons les différentes notions théoriques et un ensemble de définitions liées à la classification de documents puis une section plus descriptive d'architectures de systèmes qui détaillera leurs spécificités.

3.2.1 Composantes et définitions essentielles des systèmes de classification de documents

La « littérature » scientifique montre qu'il existe une grande diversité de classifieurs de documents. Selon le type d'applications à résoudre, chaque classifieur a sa propre façon d'exploiter les caractéristiques des documents, de choisir les algorithmes d'apprentissage et d'utiliser la base d'apprentissage pour construire les modèles des classes de documents [GAC08]. Chen et Blostein [CHE07] ont montré que la conception d'un système de classification de documents dépend essentiellement de trois composantes : l'énoncé du problème, l'architecture de classifieur, et l'évaluation de la performance (voir la figure 3.3).

3.2.1.1 Positionnement du problème et formalisme

Un énoncé du problème doit définir deux aspects pour un classifieur de document : l'espace des documents (espace des représentants) et l'ensemble des classes. Le premier définit l'ensemble des échantillons de documents d'entrée qui permet de construire deux bases représentatives : une base d'apprentissage et une base de test. Le second, définit les sorties possibles produites par le classificateur et sert à étiqueter les échantillons

de documents. La plupart des systèmes de classification étudiés utilisent des définitions de classes "manuelles" ou intuitives basées sur des similitudes de contenu, de forme, ou de style.

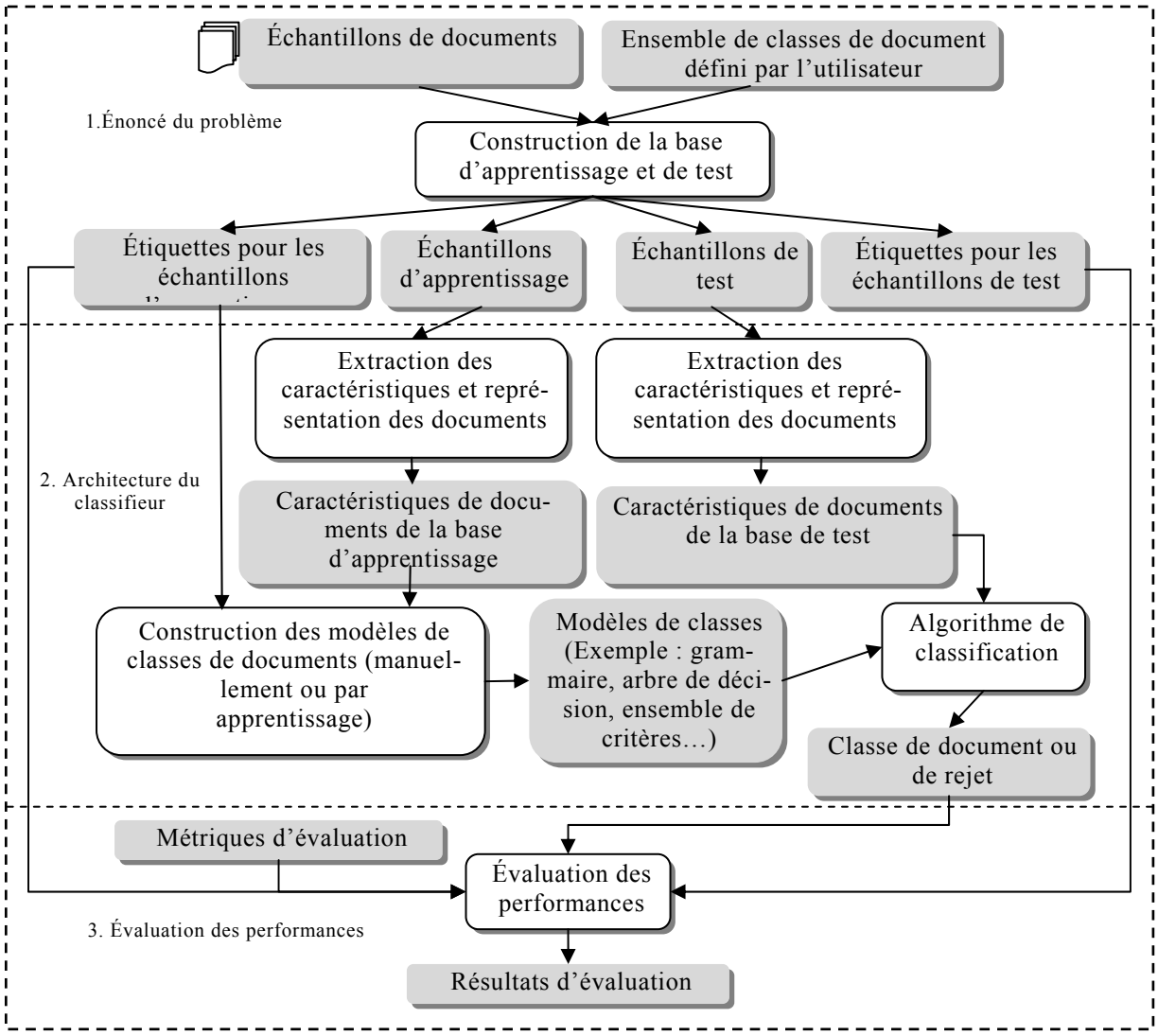


Figure 3. 3 : La structure conceptuelle des trois composantes d'un classifieur de documents (l'énoncé du problème, l'architecture de classifieur, et l'évaluation de la performance. Les boîtes rectangulaires représentent les processus. Les régions ombrées représentent les données).

3.2.1.1.1 Espace des documents (ED)

L'espace des documents (noté E_D) est un ensemble hétérogène de documents à traiter par le classifieur. Cet espace peut-être caractérisé par un domaine d'application (documents bancaires, formulaires administratifs, pièces d'identité etc.) ou par les caractéristiques propres des images à traiter (tableaux, nombre de blocs et de lignes, taille variable, etc.). La forma-

tion des échantillons d'apprentissage et de test est réalisée à partir de cette espace qui peut inclure d'autres documents qui n'appartiennent à aucune des classes et doivent être rejetées. La base d'apprentissage représente donc l'ensemble des classes définies. Plusieurs classifieurs de documents avec option de rejet sont présentés dans [LAM94] [TAY95] [HER98] [ESP00] [CES01] [OGA03]. Deux types de rejet doivent être distingués:

Le rejet de distance : Si la forme à reconnaître ne correspond pas du tout à une forme que le système a appris à reconnaître, la réponse du classifieur ne peut donc pas être pertinente et il faut donc la rejeter. Ce type de rejet permet donc de délimiter les connaissances du classifieur pour rejeter les documents qu'il n'a pas appris à reconnaître.

Le rejet de confusion : Si la forme à reconnaître est associée par le système à deux classes distinctes de façon équivalente, aucune décision sûre ne peut être prise et les risques de se tromper sont donc de *50%* : il faut donc rejeter la forme. Ce type de rejet permet donc de rejeter les documents que le classifieur ne sait pas classer sans risquer de se tromper : il permet d'augmenter de manière sensible la fiabilité du classifieur.

3.2.1.1.2 Notions de classes de documents

L'ensemble des classes de document définit la répartition de l'espace des documents où les noms des classes représentent les résultats de classifieur. Soit E_D un espace qui contient un ensemble de documents répartis en N_{cl} classes $\{CD_{i=1...Ncl}\}$ avec $Ec = \cup CD_i \mid i = 1...N_{cl}$ est un espace occupé par la réunion de toutes les classes. Quatre répartitions de E_D sont donc possibles :

a) $E_D = E_C$ *et* $\cap CD_i = \varnothing \mid i = 1...N_{cl}$,

b) $E_D > E_C$ *et* $\cap CD_i = \varnothing \mid i = 1...N_{cl}$, l'espace de rejet de distance $E_{\text{Rejet}}^{\text{Distance}} = E_D - E_C$ doit être défini,

c) $E_D = E_C$ *et* $\cap CD_i \neq \varnothing \mid i = 1...N_{cl}$, l'espace de rejet de confusion $E_{\text{Rejet}}^{\text{Confusion}} = \cap CD_i \mid i = 1...N_{cl}$ doit être défini,

d) $E_D > E_C$ *et* $\cap CD_i \neq \varnothing \mid i = 1...N_{cl}$, les deux espaces $E_{\text{Rejet}}^{\text{Distance}}$ et $E_{\text{Rejet}}^{\text{Confusion}}$ doivent être définis.

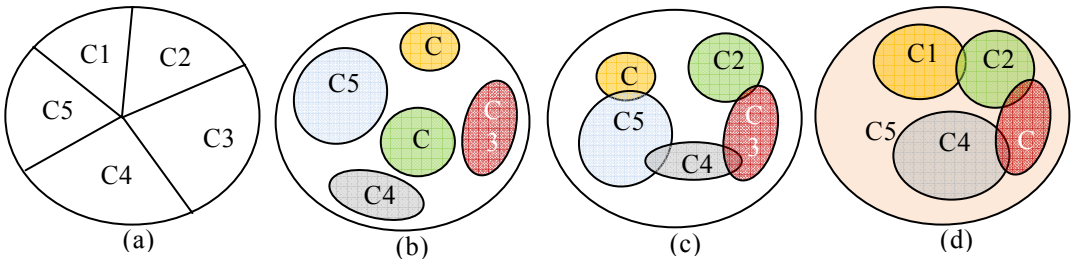


Figure 3. 4 : Les trois répartitions possibles de E_D sur E_C.

Une classe de document (également appelée type de document ou catégorie de document) est définie comme un ensemble homogène de documents de caractéristiques communes.

La notion même de classes de documents peut être vu selon différent point de vue : d'un point de vue purement visuel, il est possible de définir des classes similaires à partir des formes en présence, des structures ou des modèles de mise en page engagés dans le formatage (bas niveau) du document ; à l'opposé et selon un point de vue plus sémantique, il est également possible de définir des classes de documents induites par des métriques (ou similarités) sémantiques basées sur le contenu textuel et le sens des textes.

3.2.1.2 Les composantes essentielles des architectures des classifieurs

L'architecture d'un classifieur comprend principalement trois composantes: 1) l'extraction des caractéristiques permettant la description des contenus et leur reconnaissance 2) la représentation des documents et les modèles de classes 3) les algorithmes proprement dit de classification, de décision et les mécanismes d'apprentissage.

3.2.1.2.1 Les caractéristiques du document : une approche portant sur une analyse des contenus

Tous les systèmes de classification de documents s'appuient, de façon explicite ou implicite, sur certaines caractéristiques. Celles-ci sont essentielles : elles représentent les mesures utilisées comme information de base pour faciliter les décisions de classification. Ces caractéristiques peuvent être globales, décrivant ainsi l'image de document dans son intégralité ou locales décrivant les blocs ou les sous-sections identifiés pendant la phase de la segmentation de document. On peut calculer ces caractéristiques directement à partir des images (on parle souvent d'étape d'extraction d'indices visuels de bas niveau), à partir de l'analyse de la structure physique (disposition et présence de blocs homogènes sur la page, [ESP00]) ou logique du document (interprétation de la présence de blocs d'information, [TAN98]) ou même à partir des résultats d'OCR, en relation directe avec le contenu textuel cette fois [MOH07]. L'ensemble de ces approches est présenté en détail dans la section 3.3.2 de ce chapitre.

Le choix des caractéristiques employées reste toujours délicat et dépend de la nature des documents considérés. Ce choix se base souvent sur un compromis entre la précision et la généralité des caractéristiques. La reconnaissance des éléments de contenu précédant l'étape de classification est en relation directe avec le choix des caractéristiques exploitables. Il faut noter à ce stade, que la classification de documents peut, selon la stratégie

déployée, être effectuée après une segmentation en blocs, avant l'analyse de la mise en page, ou après, parfois même en même temps que l'étiquetage logique ou encore à partir des résultats de reconnaissance du texte.

3.2.1.2.2 La représentation du document

Elle décrit la façon dont les propriétés de chaque classe de documents sont vues par le système. La représentation peut être vectorielle (à base d'un vecteur de caractéristique géométrique), structurelle (à base de relations spatiales entres les composantes constitutifs d'un document sous forme d'arbre ou de graphe) ou basée sur un ensemble de règles. Cette étape est une des étapes clés d'un système efficace de reconnaissance du type de documents. En effet, l'utilisation d'un classifieur très performant ne peut en aucun cas compenser une représentation mal adaptée ou peu discriminante.

3.2.1.2.3 Les modèles de classes et les algorithmes de classification :

Supervisée ou non, il s'agit de la méthode de décision qui peut être inspirée de différentes techniques parmi lesquels on peut citer les bases de connaissances, les arbres de décision, les chaînes de Markov, les méthodes statistiques, les isomorphismes de graphes, les réseaux de neurones, les SVM, etc. Les méthodes sont nombreuses, nous en présenterons les caractéristiques essentielles dans la section 3.3.

3.2.1.2.4 Les mécanismes d'apprentissage

Ils sont utilisés pour ajuster automatiquement le paramétrage du classifieur. Le choix du mécanisme d'apprentissage dépend fortement du choix de l'algorithme de classification et des caractéristiques utilisées.

3.2.1.3 L'évaluation des performances

L'évaluation des performances est une étape très importante dans la conception d'un classifieur de documents. Elle est utilisée pour mesurer la précision et la justesse d'une méthode de classification à partir de la comparaison de plusieurs classifieurs. La diversité des systèmes rend la comparaison de performances généralement difficile, d'autant plus que les approches rencontrées sont rarement appliqués à des documents de même type et présentant les mêmes contraintes.

De façon très générale, les critères d'évaluation d'un classifieur peuvent être regroupés en trois catégories:

– les critères analytiques qui permettent de caractériser un algorithme de classification en termes de principes, de besoins, de complexité,

de convergence, de stabilité, etc... sans référence à une implémentation concrète de l'algorithme, ou à des données de test,

– les critères empiriques de justesse qui calculent une métrique dite de justesse sur un résultat de classification, le plus souvent d'un point de vue statistique,

– les critères empiriques de divergence qui calculent des mesures de dissimilarité entre le résultat de classification et le résultat de classification désiré.

Dans la mesure où nous sommes intéressons à la qualité d'une classification, ce sont les deux derniers types de critères qui nous intéressent. Ceux-ci peuvent encore être divisés en trois sous-groupes.

– Les critères d'évaluation non supervisés ne nécessitant aucune connaissance sur les classifications à évaluer. Leur principe consiste à estimer la qualité d'une classification à partir de statistiques calculées sur chaque classe formée.

- Les critères basés sur l'exploitation de connaissances a priori. Ces connaissances peuvent être aussi bien une classification de référence appelée vérité terrain ou des données sur les documents à reconnaître,

- Les critères génériques qui peuvent être adaptés en tous contextes supervisés ou non selon les besoins d'évaluation de l'utilisateur.

Après avoir mis en avant les concepts fondamentaux de la classification de documents, nous présentons ici quelques techniques de base de classification qui nous ont semblés essentiels pour positionner et introduire notre contribution dans le chapitre suivant. Nous citerons notamment ici les différentes stratégies de classification de documents existantes en présentant les outils de base et leurs limites en relation avec les applications de tri automatique de documents, objets de nos recherches.

3.2.2 Les méthodes essentielles de classification des documents

Un document peut-être vu comme une organisation d'objets (de symboles textuels et graphiques de toutes sortes) ayant une disposition aléatoire ou structurée. La classification des documents consiste à regrouper divers documents en sous-ensembles homogènes à partir de la description leur structure physique ou de leur contenu textuel selon des critères de similarité entre documents d'une même classe qu'il faut s'attacher à définir avec le plus grand soin. Symétriquement, les observations faites sur des documents issus de classes différentes doivent traduire la plus grande dissimilarité possible. Il existe de nombreuses méthodes de classification automatique de documents. Une première analyse des approches de classification conduit bien souvent à faire la distinction entre méthodes supervisées et non supervisées. Le choix relève généralement de l'application concernée

et repose sur la connaissance a priori du nombre de classes. La stratégie de décision de chacune de ces approches peut être inspirée de divers concepts : les bases de connaissances, les nuées dynamiques (K-Means), les chaînes de Markov, les arbres de décision, les isomorphismes de graphes, les machines à support de vecteurs (SVM), les réseaux de neurones, mais également les méthodes statistiques comme les analyses factorielles, etc. [CAR04]. Ces méthodes se basent en général sur plusieurs niveaux de représentation des informations de contenus des images de documents:

- description de l'image seulement et/ ou,
- description de la structure physique et/ou,
- description de la structure logique et/ou,
- description du contenu textuel.

Dans ce qui suit, nous présentons une brève présentation des principes des modèles principaux utilisées dans la classification de documents. Nous présentons ensuite une revue assez complète des méthodes de reconnaissance du type de documents associant cette fois les descripteurs aux méthodes de classifications. Cette dernière partie est essentielle car elle met en relation les liens, souvent difficiles à montrer, qui existent entre les descripteurs bas niveau et les mécanismes et outils techniques de classification.

3.2.2.1 Les outils généraux de classification

Nous présentons ici les principaux mécanismes algorithmiques utilisés en classification de documents et abondamment cités dans la littérature.

3.2.2.1.1 Les K-means

Il s'agit d'une des techniques de classification non supervisée les plus utilisées [EGL03-04]. Étant donné un entier *K*, l'algorithme dit des K-means partitionne les données en *K* classes ne se chevauchant pas. Ce résultat est obtenu en positionnant *K* "prototypes ou centres" dans les régions de l'espace les plus peuplées. Chaque observation est alors affectée au prototype le plus proche (règle dite "de la Distance Minimale"). Chaque classe contient donc les observations qui sont plus proches d'un certain prototype que de tout autre prototype (figure 3.5). Les prototypes sont positionnés par une procédure itérative qui les amène progressivement dans leur position finale stable en recalculant à chaque itération le nouveau centre des classes amenées à évoluer et à grossir. La grande popularité de K-means vient de sa simplicité conceptuelle, de sa rapidité et de ses faibles exigences en taille mémoire. Mais elle souffre également de certains défauts dont le plus important est lié au problème du choix initial du nombre K de classes qui

peut conduire, en cas de mauvaise estimation à une typologie et une décomposition en classes sans rapport avec la réalité. Pour une valeur donnée de K, les classes obtenues dépendent beaucoup de la configuration initiale des prototypes, ce qui rend l'interprétation des classes parfois difficile.

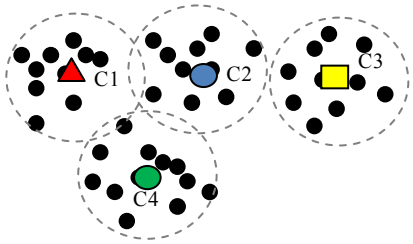


Figure 3. 5 : Exemple de K-means (K=4).

3.2.2.1.2 Le classifieur par K-PPV

Connus en anglais sous le nom K-nearest neighbor (K-NN), la méthode des plus proches voisins est une méthode supervisée de classification géométrique non bayésienne très utilisée en reconnaissance de formes et en classification de documents [YAN99][BAL03][HER98], en raison de sa simplicité et de sa robustesse aux données bruitées. Cette méthode n'exige pas de connaître la loi de distribution des variables et diffère des traditionnelles méthodes d'apprentissage car aucun modèle n'est induit à partir des exemples. Les données restent telles quelles : elles sont simplement stockées en mémoire. Ce classifieur est une extrapolation du classificateur euclidien. Au lieu d'utiliser le vecteur de caractéristiques moyen comme unique prototype d'une classe, la méthode du plus proche voisin fait intervenir tous les exemplaires des vecteurs caractéristiques disponibles. Le principe est le suivant : étant donnée une base d'apprentissage de documents étiquetés correctement et un entier k, le classifieur k-ppv détermine la classe d'un nouveau document en lui attribuant la classe des k documents lui ressemblant le plus dans la base d'apprentissage. L'erreur produite par cette méthode peut être au maximum deux fois plus grande que celle introduite par le classifieur Bayésien. Notez aussi que, si le temps d'apprentissage est inexistant puisque les données sont stockées telles quelles, la classification d'un nouveau cas est très coûteuse puisqu'elle nécessite la comparaison de ce cas à tous les exemples déjà classés.

Le choix de *K* est donc essentiel dans cette approche, sa valeur a une grande influence sur le résultat : plus elle est grande, plus l'erreur de classification est petite. Voici un exemple (figure 3.6) où on cherche à classer le nouveau document *P*. Si on choisit *K = 1*, *P* sera classé *A*. Si *K = 3*, le même *P* sera classé *B*. Pour remédier à cet inconvénient, une autre méthode appelée Category-Based Search a été conçue par Iwayama et Tokunaga [IWA95]. Elle consiste à représenter tous les documents rangés dans

une catégorie par un cas unique selon le contenu textuel (par exemple la moyenne des documents associés à une catégorie). Pour classer un nouveau document, on cherche le représentant le plus proche du document à classer, et non les k plus proches. Il suffit ensuite de modifier le représentant de cette catégorie si on veut prendre en compte ce nouveau document comme un nouvel exemple de la classe. On gagne ainsi en rapidité puisqu'on ne compare plus tous les documents deux à deux avec le nouveau document à classer, mais uniquement le nouveau document avec le représentant de chaque catégorie.

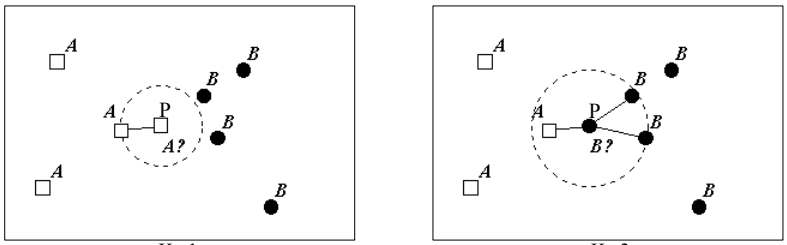


Figure 3. 6 : Influence de paramètre *K* sur le résultat de classification.

3.2.2.1.3 Les classifieurs bayésiens

Dans cette catégorie d'approches, Souafi [SOU02], par exemple, a travaillé sur la reconnaissance de la structure logique pour la classification des documents à structures riches et récurrentes selon une approche bayésienne.

Dans une approche par modélisation bayésienne, on cherche un modèle de prédiction $p(x|c_i)$, qui donne pour chaque classe de documents ci et chaque observation x, la distribution des données qui lui sont associées. Dans une approche par discrimination, on cherche à approximer la distribution $p(c_i|x)$ qui représente la probabilité à posteriori des données étant donné la classe c_i. En pratique, cette information n'est pas toujours fournie. La règle de décision bayésienne est une théorie clé en classification qui permet d'estimer la probabilité à posteriori à partir de la probabilité conditionnelle et d'émettre un vote d'appartenance du document traitée. Elle s'écrit :

$$p(c_i|x) = \frac{p(x|c_i) \times p(c_i)}{p(c_i)} \quad (3.\ 1)$$

Pour *K* classes, la décision bayésienne cherche la classe ci qui maximise la probabilité a posteriori. Elle s'écrit :

$$c(x) = \arg\max_i p(c_i|x) \quad (3.\ 2)$$

Le terme $p(c_i)$ représente la probabilité a priori de la classe c_i. Il est particulièrement utile si les classes ne sont pas équilibrées dans

l'échantillon de données considéré. La vraisemblance de l'observation *p(x),* est une quantité constante qui peut être omise du processus de décision.

3.2.2.1.4 Les arbres de décision

Les arbres de décision sont très populaires en Data Mining. Ils sont également largement utilisés dans la classification de documents [LEW94], [DEN96], [WAT95], [HER98], [CES01]. La construction d'un arbre de décision à partir de données est déjà ancienne : c'est une technique bien éprouvée par les premiers statisticiens. On considère généralement que cette approche a connu son apogée en 1984 avec la méthode CART (Classification and Regression Tree) de Breiman et al.[BRE84] Mais son origine est attribuée à Morgan et Sonquist, [MOR63], qui en 1963 ont été les premiers à avoir utilisé les arbres de régression dans un processus de prédiction et d'explication. Le principe de construction d'un arbre de décision repose sur la division récursive des exemples de l'ensemble d'apprentissage à partir d'heuristiques jusqu'à obtenir des sous-ensembles d'exemples appartenant tous à une même classe. Ces arbres peuvent traiter d'importants volumes de données, et leurs décisions peuvent être transcrites sous forme de "règles logiques" très prisées mais aux pouvoirs explicatifs souvent limités. Ces avantages sont contrebalancés par un certain manque de précision dans les prédictions comparées à celles de techniques plus sophistiquées comme les Réseaux de Neurones.

Figure 3. 7 : Exemple des arbres de décision [CES01].

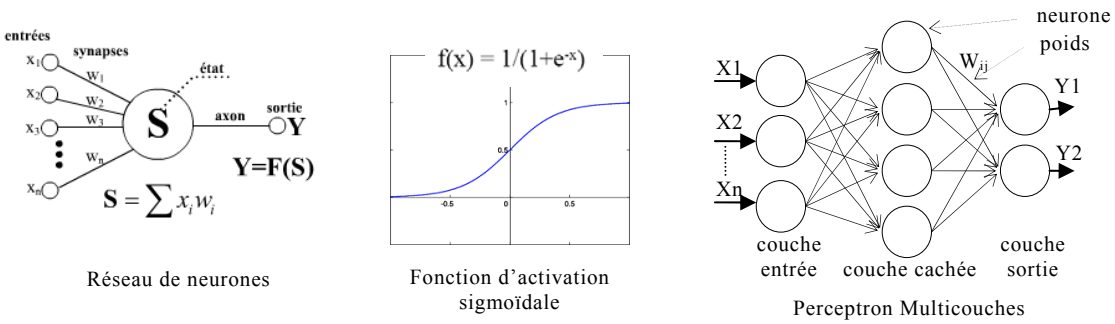
3.2.2.1.5 Les réseaux de neurones

Les réseaux de neurones sont souvent exploités comme alternatives efficaces aux approches portant sur des arbres de décision. Leur performance est souvent bien meilleure. Leur grand avantage réside dans leur capacité d'apprentissage automatique, ce qui permet de résoudre des problèmes sans nécessiter l'écriture de règles complexes, tout en étant tolérant aux erreurs.

Un réseau de neurones est un outil d'analyse statistique permettant de construire un modèle de comportement à partir de données qui sont des exemples de ce comportement. Essentiellement utilisés en classification, les réseaux de neurones peuvent être représentés par une boite noire à l'entrée

de laquelle on présente un vecteur de n dimensions, représentant les données du problème, et à la sortie de laquelle on récupère un vecteur de dimension m qui représente la solution déterminée par le système. Construit à partir d'exemples de chaque classe dont il a fait l'apprentissage, un réseau de neurones est normalement capable de déterminer à quelle classe appartient un nouvel élément qui lui est soumis. Grâce à leur grande capacité d'apprentissage automatique à partir de données, les réseaux de neurones permettent de remplacer efficacement des modèles mathématiques même extrêmement complexes. Malheureusement, le manque de lisibilité des modèles générés est un frein à leur utilisation. En cas d'erreurs, il est impossible d'en déterminer la cause.

Il n'est pas possible d'énumérer la totalité des variations existantes des réseaux de neurones disponibles à ce jour. Les chercheurs n'ont de cesse que d'inventer de nouveaux types de réseaux toujours mieux adaptés à des problèmes à chaque fois spécifiques. Cependant, à titre d'exemple nous présentons dans la suite les réseaux les plus utilisés à ce jour, voir figure 3.8.



Réseau de neurones — Fonction d'activation sigmoïdale — Perceptron Multicouches

Figure 3. 8 : La structure de base d'un réseau de neurones.

.

Les Perceptrons Multicouches (PMC ou MLP en anglais) : Le PMC est sans doute le plus simple et le plus connu des réseaux de neurones. La structure est relativement simple : une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées. Chaque neurone n'est relié qu'aux neurones des couches précédentes, mais à tous les neurones de la couche précédente. Les réseaux de neurones ont connu un essor important grâce à l'algorithme de rétro-propagation du gradient de l'erreur attribué à Werbos [WER74], Rumelhart [RUM86] et LeCun [LEC01]. Ce classifieur a trouvé application dans beaucoup de domaines tels que la reconnaissance du type de documents [HER98][CES01][MOH07] , la reconnaissance de caractères, la reconnaissance de visages, etc. Il est capable d'inférer n'importe quelle fonction de décision non linéaire moyennant une seule couche de neurones cachées et des fonctions d'activation sigmoïdales.

L'apprentissage du PMC est réalisé en minimisant une fonction de coût quadratique de l'erreur. Elle s'écrit :

$$E = \frac{1}{2l}\sum_{i=1}^{l}\sum_{k=1}^{r}\left[f_k(x_i) - t_i)\right]^2 \qquad (3.3)$$

r et l étant respectivement le nombre de classes et la taille des données considérées. Lorsque le bruit des données est de nature gaussienne, il est démontré que les sorties du PMC estiment les probabilités à posteriori des classes [BIS95]. Pendant la phase d'apprentissage, ce classifieur est assez rapide à entraîner. En test, il présente un temps de réponse qui dépend de sa complexité. Une architecture réduite est beaucoup plus rapide en test. Cependant, il n'existe pas de méthode permettant de choisir une taille adéquate du réseau de neurones. Cette limitation importante altère le pouvoir de généralisation du classifieur qui peut subir l'effet du sur-apprentissage. Ce phénomène prend ampleur lorsque le rapport du nombre de connexions au nombre de données est important. Aussi la dimensionnalité des données peut-elle dégrader de façon significative la performance du classifieur.

Les Réseaux de Kohonen : Les réseaux de Kohonen désignent trois familles de réseaux de neurones :

VQ : Vector Quantization (apprentissage non supervisé) :

Introduite par Grossberg en 1976 dans [GRO76], la quantification vectorielle est une méthode généralement qualifiée d'estimateur de densité non supervisé. Elle permet de retrouver des groupes sur un ensemble de données, de façon relativement similaire à un algorithme de type k-means que l'on préfèrera d'ailleurs généralement à un VQ si la simplicité d'implémentation n'est pas un élément majeur de la résolution du problème.

SOM : Self Organizing Map (apprentissage non supervisé) :

Les cartes auto-organisatrices de Kohonen (terme anglophone : SOM) sont issues des travaux de Fausett [FAU94] et Kohonen [KOH95]. Ces réseaux sont très utilisés pour l'analyse de données. Ils permettent de cartographier en deux dimensions et de distinguer des groupes dans des ensembles de données. Les SOM sont encore largement utilisés mais les scientifiques leur préfèrent maintenant les LVQ.

LVQ: Learning Vector Quantization (apprentissage supervisé) :

Les réseaux utilisant la méthode LVQ ont été proposés par Kohonen (1988), [KOH88]. Des trois types de réseaux présentés ici, la LVQ est la seule méthode qui soit réellement adaptée à la classification de données par "recherche du plus proche voisin".

Les Réseaux de Hopfield : Ces réseaux sont des réseaux récursifs, un peu plus complexes que les perceptrons multicouches. Chaque cellule est connectée à toutes les autres et les changements de valeurs de cellules s'enchainent en cascade jusqu'à un état stable. Ces réseaux sont bien adaptés à la reconnaissance de formes.

3.2.2.1.6 Les Machines à support vectoriel (SVM)

Depuis leur fondation en 1979 par Vapnik [VAP79], aujourd'hui, le SVM (de l'anglais «Support Vector Machine») n'a cessé de susciter l'intérêt de nombreuses communautés de chercheurs de différents domaines scientifiques. Cette méthode de classification est basée sur la recherche d'un hyperplan qui permet de séparer au mieux des ensembles de données. Le SVM est un modèle discriminant qui tente de minimiser les erreurs d'apprentissage tout en maximisant la marge séparant les données des classes. La maximisation de la marge est une méthode de régularisation qui réduit la complexité du classifieur. Ce processus produit un ensemble réduit de prototypes faisant partie de l'ensemble d'apprentissage qu'on appelle communément vecteurs de support. Le SVM compte principalement deux cas. Dans le cas où les données sont linéairement séparables, il s'agit de définir l'hyperplan qui sépare les points selon leur classe, en prenant les plus grandes marges possibles. La position du séparateur est obtenue à l'aide d'une optimisation quadratique. Dans le cas d'une séparabilité non linéaire des données, la méthode consiste à projeter les données dans un espace de grande dimension par une transformation basée sur une fonction noyau (Kernel) linéaire, polynomial ou gaussien comme le montre la figure 3.9. Dans cet espace transformé, les classes sont séparées par des classifieurs linéaires qui maximisent la marge. La complexité d'un classifieur SVM va donc dépendre non pas de la dimension de l'espace des données, mais du nombre de vecteurs supports nécessaires pour réaliser la séparation, donc de la taille de l'ensemble d'apprentissage. Par ailleurs, le SVM est habituellement applicable à des tâches de classification à deux classes, mais il existe des extensions pour la classification multi classe. En particulier des extensions à la classification multi-classes ont été proposées pour la catégorisation des documents, [JON05] : elles requièrent la combinaison d'un ensemble de SVMs, chacun se spécialisant en une partie du problème. Parmi les schémas de combinaison les plus utilisés, on citera l'approche un-contre-tous et l'approche un-contre-un. La première consiste à entraîner chacun des classifieurs pour séparer une classe du restant des classes. La deuxième consiste à entraîner les SVMs afin d'obtenir toutes les frontières de décision séparant les classes une à une. De nombreux travaux ont démontré la supériorité du SVM sur les méthodes discriminantes classiques telles que le PMC, le discriminant de Fisher, le réseau RBF, etc. Des versions modifiées du SVM ont permis d'obtenir d'excellentes performances sur plusieurs bases de données standards [LEC01]. La robustesse des SVM vis à vis de la dimensionnalité des données et leur pouvoir accru de généralisation, en font un outil de classification très avantageux. Quand les données sont non linéairement séparables le problème de séparation devient

plus compliqué et le nombre de paramètres à régler augmente : en particulier,la sélection de la bonne famille de fonctions noyau devient un point crucial souvent déterminée empiriquement.

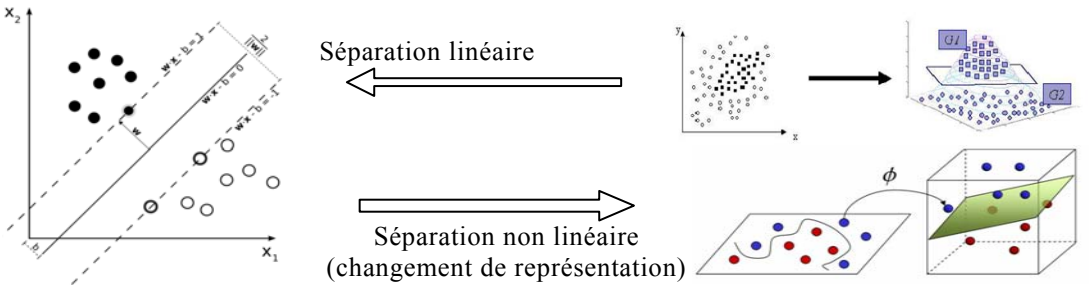


Figure 3. 9 : Principe des SVM (deux types de séparation).

3.2.2.1.7 Les Champs de markov et Modèles de Markov cachés

Une chaine de Markov est une collection d'états correspondant chacun à un résidu, où le passage d'un état à l'autre est associé à une probabilité. Les probabilités de passage d'un état à l'autre sont appelées probabilités de transition. On peut calculer la probabilité qu'une séquence appartienne à un modèle donné: il suffit d'observer les transitions qui apparaissent dans cette séquence, puis de se reporter au modèle pour obtenir les probabilités correspondantes. La probabilité finale est le produit des probabilités de transition. Le système a besoin d'une phase d'entrainement préalable pour déterminer les probabilités de transition pour toute nouvelle séquence de famille à reconnaître. Cette étape peut ensuite servir à toute autre séquence pour déterminer si elle appartient bien à la famille.

Les modèles de Markov cachés (HMM) introduit en 1989 par Rabiner dans [RAB89] sont une autre évolution possible des chaînes de Markov. Ces modèles se basent cette fois sur deux processus stochastiques dépendants l'un de l'autre. L'état du système n'est plus directement observable, il est caché par un processus d'observation. Les HMM sont des outils très utilisés dans de nombreux domaines de l'analyse des images, [LI95]. Citons à titre d'exemple : la reconnaissance de la parole, la reconnaissance de l'écriture [ANI92], la classification de documents [DIL03], l'analyse de mise en page de documents manuscrits peu structurés [NIC06].

3.2.3 Des primitives bas niveau à la décision : quelques approches essentielles

3.2.3.1 Les mécanismes de classification portant sur des primitives bas niveau sans segmentation

Les méthodes que nous allons évoquer ici font la synthèse entre d'une part les primitives jugées pertinentes pour caractériser efficacement les contenus et les outils de classification que nous venons de présenter. Tantôt basées sur des primitives bas niveau de l'image, (géométriques, structurelles ou encore texturelles), tantôt sur une analyse plus poussée de la structuration physique ou logique des contenus, ces méthodes cherchent toutes à mettre en adéquation une caractérisation des images et un mécanisme de classification le plus adapté.

Certaines approches comme celle développée dans [SHI01] utilisent des caractéristiques extraites directement de l'image sans avoir besoin de la segmenter en différents blocs. Ces caractéristiques peuvent être liées à des informations de densité de contenu de l'image (par le biais de calcul de moments par exemple), à des statistiques calculées sur l'ensemble des composantes connexes, à des informations structurelles de mise en page telles que le décalage entre les lignes et les colonnes et d'autres mesures relatives à la taille de la police et autres effets typographiques associés. Shin et al [SHI01] ont travaillé sur ce type d'approches en calculant des caractéristiques de l'image à partir de quatre types de fenêtres : fenêtres rectangulaires, fenêtres de bandes horizontales, fenêtres de bande verticales et fenêtre de la page. Une mesure de similarité basée sur les correspondances entre les différentes fenêtres est ensuite utilisée pour comparer les images de documents. Deux types de classifieurs ont été utilisés pour la classification de documents : Un arbre de décision et une carte auto-organisatrice.

3.2.3.2 Les mécanismes de classification portant sur l'analyse de la structure physique

D'autres approches, en revanche s'intéressent à la description complète de la structure physique du document. La plupart des méthodes basées sur ce principe utilisent une représentation hiérarchique des éléments physiques sous forme de zones (blocs ou lignes de texte, graphiques, grilles, cases à cocher, tableaux…). Cette représentation permet de mettre facilement en relation les différents éléments constitutifs d'un document.

Watanabe et al [WAT95] proposent une méthode de classification de formulaires basée sur un arbre de décision en utilisant la description de la grille. Toujours sur les formulaires, Héroux et al. [HER 98] représentent chaque document par un arbre, où les nœuds sont formés à partir de blocs issus de l'analyse de la structure physique. Ils appliquent ensuite un appariement hiérarchique entre les arbres pour regrouper les documents en classes homogènes. Ce classifieur structurel a été comparé à deux autres

classfieurs classiques, le premier est de type K-PPV et le second est un perceptron multicouche (MLP). Ces deux classifieurs utilisent un vecteur de caractéristiques extraites directement de l'image de formulaire avant sa segmentation en bloc. Les résultats montrent que la classification structurelle offre plus de robustesse, dans sa capacité à former des classes homogènes et à créer des classes de rejets. Esposito et al [ESP 00] manipulent les attributs et les relations entre blocs dans un langage du premier ordre. Ce langage est utilisé avec certaines règles par la phase d'apprentissage. Cesarini propose dans [CES 01] un algorithme de construction de l'arbre X-Y basé sur une stratégie de segmentation descendante. Le document est découpé récursivement selon les directions horizontale ou verticale d'après des zones homogènes de séparations. Le résultat est alors un arbre où chaque nœud représente une zone de l'image. Les techniques d'apprentissage par réseaux de neurones de type PMC (perceptron multicouches) consistent à minimiser un critère d'erreur en adaptant l'ensemble de poids du réseau représentant les modèles. Les arbres X-Y produits par les algorithmes descendants de segmentation sont également très utilisés, mais ils induisent des risques d'être insuffisamment discriminants à cause de la rotation des images. Baldi [BAL 03] et Diligenti [DIL 03] ont proposé une extension de l'arbre X-Y en arbre XYM. Baldi projette les distances d'édition entre arbres dans un espace de K-PPV, alors que Diligenti l'utilise pour construire un modèle d'arbre de Markov caché (HTMM). D'autres travaux de classification portant sur la théorie des graphes ont été proposés par Bagdanov et al. dans [BAG 03]. La technique est principalement basée sur la construction de FOGGs (First Order Gaussian Graphs) où des probabilités sur les nœuds et sur les sommets sont utilisées lors de l'apprentissage pour créer les modèles de reconnaissance.

3.2.3.3 Les mécanismes de classification portant sur la description de la structure logique

Cette description repose essentiellement sur l'analyse des étiquettes logiques des blocs physiques extraits lors de la segmentation physique du document. Les étiquettes logiques servent à exprimer la sémantique de chaque bloc du document (titre, logo, date, nom, code ACI, adresse, montant, signature, etc).

Dengel et Dubiel ont présenté dans [DEN96] un système de classification de lettres d'entreprises basé sur ce type de description. Ce système est basé sur la construction d'une hiérarchie d'objets portant sur un étiquetage logique manuelle des blocs et un classement des lettres en catégories spécifiques. Pour cela, le système définit intialement les relations spatiales entre les différents blocs en utilisant l'ensemble des étiquettes initiales (su-

jet, expéditeur, destinataire, etc.). Puis il construit un arbre de décision à partir de l'ensemble des documents de la base d'apprentissage. La classification d'un nouveau document s'effectue alors par le parcours de l'arbre de décision en fonction des éléments extraits sur l'image à classifier. Cette approche est simplement limitée par les problèmes de segmentation qui peuvent survenir pendant l'extraction des blocs. En utilisant les résultats de l'étiquetage fonctionnel, Eglin et Bres [EGL03,04] ont présenté une méthodologie complète pour la caractérisation et la catégorisation des documents. Cette méthode utilise des mesures statistiques basées sur des primitives de textures s'inspirant des mécanismes de la perception visuelle humaine. Le processus de séparation des blocs en sous classes fonctionnelles est basé sur une classification non supervisée par la méthode des k-means. On peut citer à titre d'exemples d'autres méthodes de classification qui utilisent la description des deux structure physique et logique à base de n-grams [BRU97], d'appariement de modèle [KOC99], de l'algorithme de Winnow [NAT01] ou d'isomorphisme logique de graphe [LIA02a], etc.

3.3.3.4 Représentation basée sur la sortie OCR (contenu textuel)

La description du contenu textuel qui provient de la sortie de l'OCR utilise généralement les fréquences d'apparition des caractères, des n-grams ou même de certains mots clés. Les méthodes de classification de documents concernées par cette description sont fondées sur une analyse syntaxique et un calcul sémantique. On peut aussi leur adjoindre des méthodes d'apprentissage, incluant des modèles de régression, l'approche des k-PPV [Yan99], des approches Bayésienne naïves, des arbres de décision [LEW 94], des méthodes à base de connaissances ou de réseaux de neurones [MOH07]. D'autres méthodes présentées dans la littérature combinent la description de contenu textuel avec celle de la structure physique [SAK03] ou logique [LIA02b], mais elles ne sont pas adaptées à dés applications de temps réel car elles sont très coûteuses en temps du calcul.

3.2.4 Bilans des approches de classification : vers des outils plus adaptés au contenu

Dans cette partie, nous avons tenté de présenter de manière simple et complète les différentes étapes de la conception d'un classifieur, les outils généraux de classification et leur application à la classification de documents. Nous avons décelé des limites de chacune de ces méthodes et projeté les effets sur leurs exigences par rapport à notre application de tri de documents. Nous avons classé également ces méthodes selon quatre principes de description (image, structure physique, structure logique et contenu textuel) qui correspondent aux approches habituelles de description des

contenus des images de documents. Nous pouvons remarquer que les systèmes de RAD utilisateurs de la structure logique [DEN96] [EGL03,04] ou du contenu textuel [MOH07] sont lents et très difficiles à mettre en œuvre dans une application de tri qui doit fonctionner en temps réel. Par ailleurs, la quantité d'informations apportée par une simple description de l'image de document sans analyser sa structure physique ne peut pas être discriminante sur des documents présentant une grande variabilité de mise en forme [HER98][SHI01]. Ces contraintes impliquent d'avoir à la fois une description simple des contenus et discriminante de la structure afin de permettre un classement rapide de tous les documents susceptibles d'apparaitre dans une chaine de tri. Afin de s'adapter au mieux aux exigences de rapidité et d'efficacité imposées par notre application, nous nous sommes intéressés aux approches basées sur la description de la structure physique des pages.

Afin de répondre au mieux aux besoins du système industriel de notre entreprise partenaire, il a fallu choisir un outil efficace garantissant des résultats cohérents par rapport aux exigences des applications temps réel. Nous avons donc choisi de produire une caractérisation rapide et complète des contenus qui ne nécessite pas ni calculs prohibitifs ni redondants. L'architecture que nous avons choisie de mettre en place doit également être capable de produire une partitionnement efficace des objets présents sur les pages à traiter en minimisant les situations d'erreurs et les rejets. C'est la raison pour laquelle, nous avons choisi de mettre en place une architecture complète basée sur la coloration des graphes que nous allons présenter dans le chapitre suivant.

Nous nous sommes intéressés essentiellement à la puissance de cet outil et à ses avantages : regroupements efficaces et automatiques des données en ensembles homogènes, approche non paramétrique, simple, propriété de bonne sélection des représentants des classes, bon contrôle des différentes phases de classification, et surtout obtention de résultats correspondant à l'information réellement présente dans la base d'apprentissage. Ces caractéristiques permettent à notre système de reconnaissance d'avoir une adaptation non supervisée à la nature des documents à trier. Cette adaptation peut être réalisée soit en mode « par lots », soit en mode incrémental en gardant toujours à l'esprit, la possibilité d'une interaction toujours possible et simple avec l'utilisateur minimisant les connaissances à avoir pour faire fonctionner le système et interagir avec lui.

3.3 Deuxième partie : La localisation du bloc-adresse (LBA)

Nous abordons dans cette partie un cas particulier de documents, il s'agit des documents postaux ou des courriers d'entreprises de structure variable. Ces documents contiennent tous une région d'intérêt commune imprimée ou manuscrite, le bloc adresse qui doit être localisé en vue de l'automatisation du processus de tri. Cette application concerne aussi bien des lettres que des colis ou des plis (lettres de grand format, magazines, revues, journaux, documents publicitaires). De très grands progrès ont été réalisés depuis ces trente dernières années dans ce domaine très pointu. Les banques, les entreprises et les compagnies postales des pays développés utilisent désormais différents types de machines de tri pour traiter une très grande quantité de courriers avec des technologies de type OCR toujours plus spécialisées et complètes : plus de *200* milliards de plis par an aux Etats-Unis, plus de *26* en France. C'est la raison pour laquelle ces établissements ont joué un rôle très important dans l'innovation des techniques de traitement des documents en finançant la recherche et le développement de machines de lecture automatique d'adresses. Le principe de ces techniques consiste à trier autant de plis que possible au niveau du point de distribution, minimisant ainsi le temps passé par les facteurs dans le bureau de poste.

Depuis les années 70, la poste américaine utilise des machines à OCR pour trier automatiquement toute sorte de courriers. En 1984, 252 machines ont été réparties dans 118 centres traitant jusqu'à 24000 plis à l'heure. Cette cadence est passée à 45 000 plis à l'heure en 1990. En 1997, les lecteurs de la poste américaine ne lisaient que 2 % des adresses manuscrites. Ces performances atteignaient 53 % en 1999. En 1998, la société Lockheed Martin recevait un contrat de 168 millions d'euros pour porter rapidement ce taux à 80 % [SRI97]. Aujourd'hui, les performances de reconnaissance manuscrite approchent rapidement celles du l'imprimé et les systèmes de tri traitent automatiquement plus d'une dizaine de tonnes de courriers chaque jour avec des cadences allant jusqu'à 68 000 plis à l'heure.

L'automatisation du tri des objets postaux en fonction de l'adresse de leurs destinataires nécessite d'abord un code puis un traitement. Elle a donc donné naissance, dans un premier temps, à deux types de machines : les machines d'encodage qui permettent de marquer le pli d'un code fluorescent et les machines de tri qui après lecture du code fluorescent orientent vers différentes casiers les plis en fonction de résultat de lecture de l'adresse de destination. Cette lecture nécessite que le bloc adresse soit ra-

pidement et précisément identifiable par le module de reconnaissance, et que les lignes d'adresse soient correctement organisées sur l'image de courrier (figure 3.10). Les images pour lesquelles les adresses ne sont pas reconnues sont rejetées puis transmises à des positions de saisie manuelle dites de vidéo-codage.

Les performances des systèmes de tri automatique de courrier ne cessent de croître, avec la mise sur le marché de machines de plus en plus puissantes et le développement de nouvelles architectures matérielles, logicielles dans ce domaine). Chaque jour, ces systèmes traitent automatiquement plus d'une dizaine de tonnes de courrier avec des cadences allant jusqu'à 17 plis par seconde, ce qui nécessite que le bloc adresse soit rapidement et précisément identifiable par le module de reconnaissance, et que les lignes d'adresse soient correctement organisées sur l'image de courrier (figure 3.10).

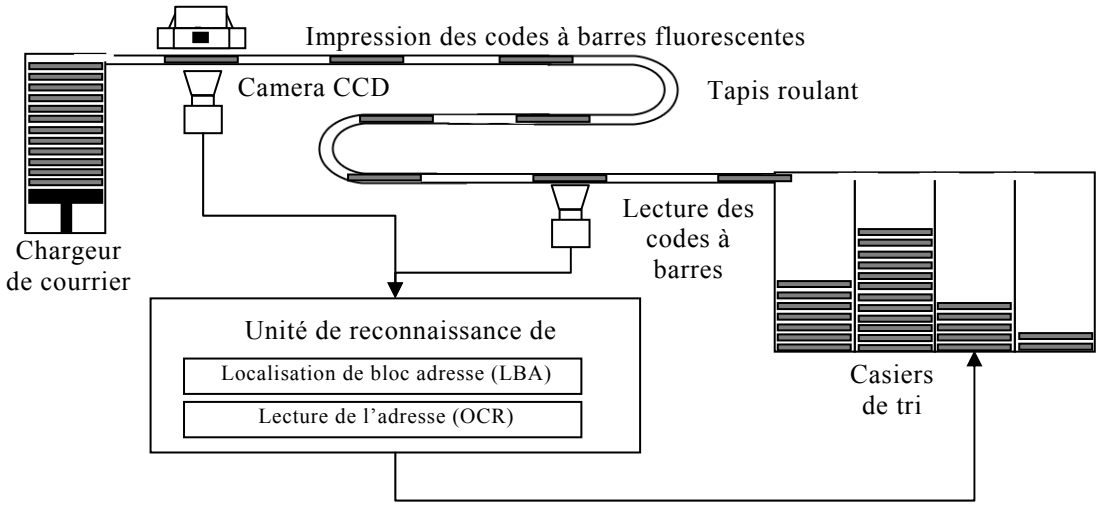


Figure 3. 10 : Structure générale de la chaîne de tri de courrier postal.

3.3.1 Contraintes et spécificités des images de courrier

Contrairement aux idées reçues, la localisation du bloc-adresse (notée LBA dans la suite) est une opération non triviale du fait du temps de traitement limité, du nombre important de blocs informatifs parasites présents dans le voisinage direct de la zone d'adresse et de la très grande variabilité des caractéristiques de cette zone (voir la figure 3.11) :

- l'adresse peut avoir une taille variable.

- l'adresse peut être manuscrite avec des styles variables ou imprimée avec des polices différentes et des mises en forme différentes (taille, espacement…).

- l'adresse peut avoir quatre orientations différentes (*0°, 90°, 180°* et *270°*) et des inclinaisons variables entre*]-45°, 45° [*,

- l'encre d'impression de l'adresse peut être de différentes couleurs,

- les technologies d'impression sont différentes : machines à écrire, imprimante matricielle, laser, à jet d'encre…etc.

- la zone d'adresse peut être mal contrastée et l'image de l'enveloppe peut avoir une luminance non uniforme et contenir des dégradations,

- l'adresse peut être écrite directement sur l'enveloppe, ou sur une étiquette adhésive de fond clair sur ou sous un film plastique, ou encore elle peut être visible par une fenêtre par transparence sur l'enveloppe.

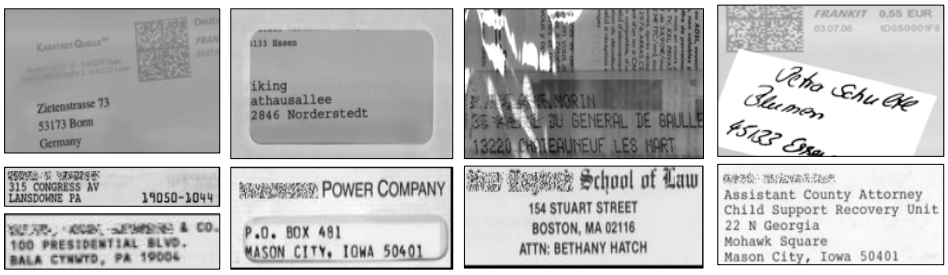


Figure 3. 11 : Illustration de la très grande variabilité des caractéristiques de la zone d'adresse.

Par ailleurs, notons que pour une grande partie des courriers traités dans cette étude, l'adresse de destination n'est pas systématiquement écrite au coin inférieur droit : certaines mises en page ne respectent pas cet arrangement strict. La présence de timbres, de marques de la poste, de logos imprimés, de diverses annonces et autres informations parasites sur le courrier rend la tâche de localisation très difficile (voir la figure 3.12).

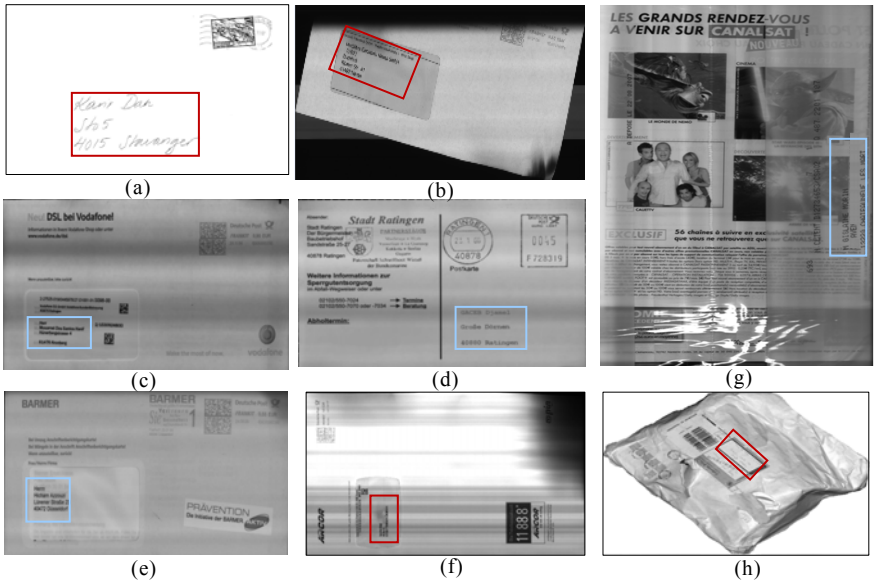
3.3.2 Complexité de la structure des courriers

Les courriers à trier sont de complexité variable. Parfois, ils peuvent avoir des structures simples présentant un rapport signal/bruit (SNR) élevé : lettres imprimées ou manuscrites, ou une structure complexe ayant un SNR faible : courriers d'entreprise, magazines, journaux, colis (voir la figure 3.12). Ce rapport peut être donné par l'équation suivante [ORI95] :

$$SNR_{Enveloppe} = \frac{Card\left[I(x,y) \in Objet \subset \text{Adresse} \middle| \begin{matrix} x_{adr}^0 \le x \le x_{adr}^1 \\ y_{adr}^0 \le y \le y_{adr}^1 \end{matrix} \right]}{Card\left[I(x,y) \in objet \subset (\text{Image} - \text{Adresse}) \middle| \begin{matrix} 0 \le x \le W_{\text{img}} \\ 0 \le y \le H_{\text{img}} \end{matrix} \right]} \quad (3.4)$$

Jain et al [JAI96] définissent automatiquement la complexité à partir du nombre des CCs dans l'image de courrier $M=|\{CC_i\}|$. Si $M<T_{cn}$, l'image du courrier est classée comme simple sinon elle est classée comme complexe, T_{cn} est un seuil fixé empiriquement dans les expériences. Dans

les deux cas, la complexité est définie en relation avec le nombre d'éléments sur la page et non leur agencement. Voyons désormais quelles sont les caractéristiques essentielles à extraire des images pour caractériser la complexité des structures des pages.



(a) (b) (c) (d) (g) (e) (f) (h)

Figure 3. 12 : Positions et orientations variables de la zone d'adresse et présence d'informations « parasites » autour du bloc adresse. Cas de courrier de structure simple : **(a)** lettre manuscrite, **(b)** lettre imprimée. Cas de courriers de structure complexe : **(c, d, e, f)** courrier d'entreprises, **(g)** magazine, **(f)** colis.

Les courriers dits de structure simple sont le plus souvent composés de quelques blocs assez bien isolés. Ces derniers présentent presque toujours un bloc adresse, un bloc affranchissement et occasionnellement quelques logos et vignettes. Les deux premiers blocs principaux respectent une disposition relativement structurée et l'adresse peut être indifféremment imprimée ou manuscrite alors qu'elle présente une grande diversité dans les caractéristiques du texte.

Les courriers dits complexes se distinguent quant à eux par une structure physique très variable. Le texte ne correspond pas uniquement à l'adresse, mais il représente d'autres éléments publicitaires et des paragraphes de tailles de caractères et d'orientations variables. Les éléments textuels ne sont pas forcément majoritaires et on retrouve aussi d'autres éléments non textuels comme les graphiques, les photos, les cadres, et les tableaux de positions et de couleurs différentes. Ces éléments que l'on peu caractériser comme parasites dans le processus de LBA sont parfois en inverse vidéo ou même superposés à l'adresse. Les cas les plus difficiles se retrouvent sur les images de pochettes plastiques transparentes. Le seul élément qui peut réduire la complexité de cette situation, est la connais-

sance a priori du type d'écriture de la zone d'adresse presque toujours imprimée. Les tests montrent que les méthodes de LBA donnent de meilleurs résultats sur des enveloppes simples que sur des enveloppes complexes [ORI95][JAI96].

3.3.3 Les chaînes de localisation du bloc adresse (LBA) : revue de l'existant

3.3.3.1 Des systèmes basés sur des architectures modulaires

La chaîne de localisation du bloc adresse dans les images de courriers se décompose de façon modulaire : après l'acquisition de l'image de l'enveloppe par une camera CCD en 300 dpi de résolution, trois modules principaux peuvent ainsi être retenus (voir la figure 3.13) :

- la segmentation de l'image d'enveloppe,
- l'analyse de la structure de l'enveloppe,
- l'interprétation des blocs.

Après une phase de binarisation de l'image de l'enveloppe, le premier module permet la détection des composantes connexes (CCs). Le second module effectue l'analyse hiérarchique de l'agencement de ces CCs sur l'enveloppe pour recomposer les blocs et établir leur description. Les caractéristiques usuelles extraites à partir des blocs sont les mesures utilisées comme information de base pour les décisions de classification. Nous les avons synthétisées dans le tableau 3.1.

Type	Caractéristiques
Blocs	Position, hauteur, rapport W/H | Moyenne (W/H) de lignes | Nombre : lignes, caractères Densité : NG, pixels noirs | Alignement gauche des lignes | Variance et moyenne : hauteur et largeur des lignes | Texture : coefficient de fourrier, Gabor | Relations spatiales Espace moyenne : inter-lignes, inter-mots.
Lignes	Position, taille, rapport W/H | Nombre : caractères | Densité : NG, pixels noirs Variance et moyenne : hauteur et largeur des CCs| Espace moyenne inter-CCs Tailles moyennes des CCs.
CCs	La position, taille, rapport W/H, Densité : NG, pixels noirs, Moyenne de NG, épaisseur de traits.

Tableau 3.1 : Bilan des caractéristiques utilisées à différents niveaux d'échelle par les méthodes de LBA présentées dans la littérature.

Une phase décisionnelle inspecte l'ensemble des données obtenues pour identifier le bloc adresse. Pratiquement, un dysfonctionnement sur l'un de ces modules réduit les performances des autres, et par conséquent peut conduire à une imprécision ou même une erreur de localisation du bloc adresse. Cela entraîne ainsi une lecture optique erronée de son contenu.

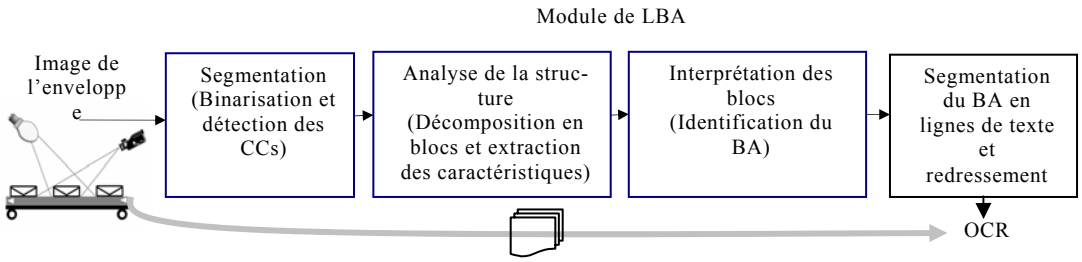


Figure 3. 13 : Modules principaux de LBA.

Dans tous les travaux relatifs à la localisation du bloc adresse, il est supposé que le bloc adresse représente la zone d'intérêt contenant l'information nécessaire pour reconnaître la destination. Par conséquent, toute adresse mal localisée (et donc mal reconnue) conduit à un rejet immédiat du courrier.

Dans cette partie, nous présentons les travaux essentiels qui ont été consacrés ces dernières années à l'amélioration de la LBA. Le but n'est pas d'en dresser une liste exhaustive mais de dresser un panorama des approches les plus fréquemment utilisées. Nous présentons ici une classification de ces techniques selon les stratégies mises en jeu.

3.3.3.2 Les stratégies basées sur une analyse des contenus et des structures

La binarisation des images de courriers n'est pas systématique : certaines approches de LBA analysent directement les images en couleurs ou en niveaux de gris. Ces méthodes sont largement basées sur l'analyse de la texture de l'image en utilisant, par exemple, les filtres de Gabor [JAI94], l'inhomogénéité à base de gradients et de coefficients de Fourrier [MWO97], les lignes de partage des eaux [YON03] ou les dimensions fractales [EIT04]. Ces méthodes consomment beaucoup de temps et s'appliquent uniquement à des enveloppes de petite taille ou représentées en basse résolution, exigeant, de ce fait, que les blocs soient suffisamment éloignés entre eux. Pour gagner en rapidité de traitement les autres approches que nous référençons nécessitent une étape supplémentaire de binarisation séparant le premier plan du fond, [DOW90], [LEE94], [JAI96],[PRA03], [JEO04], [ROY05], etc. Ces méthodes sont les méthodes les plus répandues dans ce type d'application.

Les méthodes de LBA procèdent généralement par une décomposition des contenus en blocs. On distingue deux familles d'approches : les méthodes qui sélectionnent le BA parmi plusieurs blocs candidats tel que cela est présenté dans [YEH87] et [WAN88] et celles qui extraient directement le BA à partir de l'image de l'enveloppe, [XUE99][LEE94].

Les méthodes de la première classe utilisent diverses techniques de segmentation pour faire émerger les différents blocs existants sur l'image de l'enveloppe. Elles extraient ensuite des descripteurs pour chaque bloc, afin d'identifier celui qui contient explicitement l'adresse de destination.

Les méthodes de la deuxième classe se limitent à localiser directement le BA sans décomposer l'image de l'enveloppe en plusieurs blocs candidats. Ces méthodes sont souvent appliquées à des enveloppes où la séparation entre les différents blocs est presque impossible. Par exemple, dans les enveloppes de langue orientale (Chinoise, Coréenne), les adresses manuscrites de l'expéditeur et du destinataire se chevauchent souvent, c'est pourquoi il n'existe pas de méthode pour les séparer. Au final, ces méthodes restent moins performantes sur des enveloppes de blocs séparés : nous avons donc porté toute notre attention dans le carde de notre application industrielle aux méthodes de la première classe.

Quelle que soit l'approche d'analyse du contenu choisie, les méthodes de LBA se divisent en trois classes selon la stratégie d'analyse de structure mise en jeu (voir le chapitre 2, section 2.4.1). Comme les méthodes mixtes offrent un meilleur compromis temps/ précision, un grand nombre des méthodes de LBA s'oriente vers ce type de stratégies.

3.3.3.3 Les stratégies basées sur l'apprentissage ou les règles de décision

Les stratégies de prise de décision se décomposent principalement en deux classes de méthodes : les méthodes à base de règles déterministes et d'heuristiques, et les méthodes à base d'apprentissage.

3.3.3.3.1 Méthodes à base de règles déterministes et d'heuristiques

Ces méthodes appliquent un ensemble de règles et d'heuristiques à toutes les phases de traitement telles que l'extraction des caractéristiques, le filtrage des éléments parasites, la segmentation en blocs homogènes et l'interprétation des blocs. La stratégie consiste à générer une liste de blocs candidats (groupes de lignes spatialement proches et de caractéristiques géométriques et structurelles semblables) et à classer ces blocs en fonction de règles afin de ne retenir que le bloc adresse le plus vraisemblable. Dans le cas d'un courrier simple, le classement est assez simple puisque le nombre de candidats est généralement réduit et que deux ou trois caractéristiques comme la position, la taille et le nombre de lignes dans le bloc sont suffisamment discriminantes. Dans le cas d'un courrier complexe, on a besoin d'un nombre de règles et de caractéristiques plus élevé [ORI95][JAI96][YUB97]. La base de connaissances regroupe et structure de façon déclarative les connaissances utilisées, ou générées. On peut en

distinguer trois types : les connaissances observables, descriptives, et stratégiques.

Les connaissances observables incluent les données à traiter (images d'enveloppes, blocs) et leurs informations contextuelles (qualité, type, etc.). Les connaissances descriptives décrivent les blocs susceptibles de figurer sur les enveloppes (la position, la densité…). Plusieurs formalismes de représentation peuvent donc être utilisés : les approches statistiques/structurelles [LEE94], les règles [YUB97], etc. Ces connaissances descriptives peuvent être acquises par des techniques d'apprentissage [LEE94], ou introduites par des spécialistes de domaine [YUB97]. Les connaissances stratégiques décrivent un scénario de LBA [JAI96]. Dans ce contexte, la plus ancienne étude de la littérature a été effectuée par Yeh et al dans [Yeh87]. La phase de la segmentation en blocs est basée sur l'opérateur Laplacien et un processus de regroupement des CCs. Les CCs de caractéristiques communes sont regroupées et représentées dans un arbre de voisinage. Le découpage de cet arbre produit plusieurs blocs, où chacun doit appartenir à une des catégories suivantes : adresse de destination, adresse de retour, timbre, et cachet de la poste. L'utilisation du Laplacien augmente la sensibilité de la méthode aux dégradations du fond et au bruit.

D'autres travaux ont tenté d'aboutir à une localisation « intelligente » du bloc adresse en utilisant les systèmes experts comme outil d'aide à la décision [WAN88] [ANT88][APP89]. Ce type de systèmes se compose, en général, de trois parties : une base de faits (les blocs candidats), une base de règles et un moteur d'inférence. Wang et Srihari dans [WAN88] ont utilisé ce type de stratégies pour superviser le choix et la localisation du bloc adresse. Le système utilise un tableau noir (blackboard) pour stocker et exploiter les attributs géométriques des blocs. Une base de données statistiques et un moteur d'inférence à base de règles sont utilisés pour attribuer des scores aux blocs candidats. Trois mille blocs de courrier ont été utilisés pour développer les règles. Dans le même esprit, Appiani et al [APP89] ont proposé un schéma hiérarchique multiprocesseur d'un système expert de LBA dans le but de garantir une grande modularité mais avec un temps de traitement plus réduit. Les approches fondées sur des systèmes experts se composent souvent de règles de type « si – alors ». On imagine bien les limites quand on souhaite traiter de gros volumes de données manipulant des vecteurs de caractéristiques de grandes tailles. Mais avec de telles approches se posent alors rapidement les problèmes de gestion de connaissances, rendues très coûteuses. Initialement ces systèmes étaient utilisés essentiellement dans le domaine de l'intelligence artificielle. Pour cette raison, leur adaptation au domaine de LBA a été rapidement abandonnée.

Dans les années 90, les études se sont à nouveau orientées vers une exploitation classique de règles. Downton et Leedham [DOW90] ont réduit la phase de LBA à la seule détermination de seuils sur des caractéristiques physiques de régions. Viard-Gaudin et Barba [VIA91] ont utilisé une structure de données pyramidale représentant l'image avec une approche multi-résolution. Une segmentation ascendante est utilisée pour construire la structure de données, et une analyse descendante est menée conjointement pour construire un arbre d'inclusion des CCs et interpréter les blocs segmentés aux différents niveaux de résolution. Palumbo et al [PAL92] ont proposé une architecture modulaire pour un système de LBA multiprocesseurs appliquant des règles sur les caractéristiques géométriques des blocs et sur les relations spatiales entre blocs.

La méthode proposée par Lee et Kim [LEE94] repose sur la construction d'une base de connaissances permettant de déterminer avec une plus grande précision la localisation du BA. Le théorème de Bayes est utilisé dans l'inférence statistique pour actualiser la description des blocs. Selon le même principe, une base de connaissances descriptives a été utilisée dans les travaux de Yu et al [YUB97] pour localiser le BA sur des structures plus complexes (magazines et journaux). Un regroupement ascendant de CCs est utilisé avec la stratégie BAG (block adjacency graph) pour former les blocs.

D'autres travaux à base de règles simples ont été développés dans [ORI95] [JAI96]. Il reposent sur l'adaptation des règles à la complexité de la structure des documents à analyser. Cela permet de dérouler l'algorithme le plus efficace dans la chaîne de traitements en fonction du document en présence.

Oriot et al [ORI95] ont présenté une localisation du bloc adresse de destination sur des grands objets postaux. Les objets à trier ont été décomposés en deux catégories principales (objets plats peu et très chargés). Une méthode a été adaptée à chaque catégorie : les auteurs mettent en œuvre une segmentation originale fondée sur deux représentations complémentaires des zones de texte. Ces représentations sont obtenues à partir de caractéristiques de texture. Un intérêt particulier a été porté sur la phase d'optimisation des traitements et d'évaluation des performances de la LBA. Jain et al [JAI96] localise le BA sur des images d'enveloppes en couleurs de basse résolution. Une approche ascendante de segmentation et quelques heuristiques sont ainsi utilisées pour produire un modèle pyramidal, localiser, et estimer l'inclinaison du bloc adresse.

D'autres difficultés sont liées à la langue. Dans le cas des enveloppes manuscrites chinoises, par exemple, les blocs sont difficilement séparables. Les règles s'appliquent donc directement sur les mots et les lignes de texte, sans avoir besoin de séparer les blocs pour chercher par exemple le code postal, [XUE99].

Plus récemment, une nouvelle méthode a été proposée par Jeong dans [JEO04] visant des enveloppes de structure stable (par exemple celles des entreprises qui imposent des consignes strictes de mise en page). La méthode consiste à décomposer une enveloppe en *9* zones de même taille (1 et 2 : Adresse de retour, *3 et 2* : timbre, *7* blocs divers, *5, 6, 8, 9* : adresse de destination).

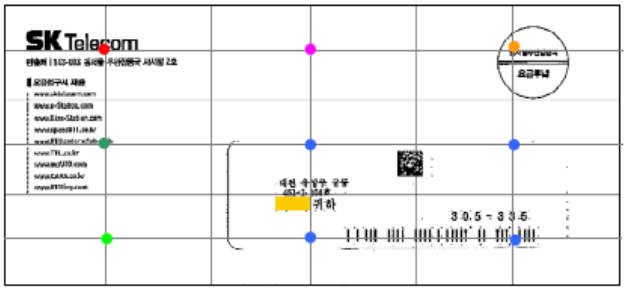


Figure 3. 14 : Initialisation des centres de 9 zones.

Un ensemble de règles est alors appliqué lors du regroupement des CCs en blocs. En utilisant quelques heuristiques et de la position de chaque bloc par rapport aux 9 zones un label est affecté. Cette méthode fortement dépendante de la position des blocs ne peut pas être appliquée sur des enveloppes de structures variables.

Les systèmes classiques à base de connaissances et en particulier à base d'heuristiques déterministes ou de règles conduisent à la LBA en deux étapes. Une étape d'extraction de caractéristiques géométriques de bas niveau est réalisée en amont afin de permettre l'émergence de blocs adresse candidats et une étape de tri basaée sur l'utilisation de règles strictes. Typiquement, ces systèmes utilisent des centaines de règles et d'heuristiques et leur gestion nécessite des mécanismes très complexes. Si à ce jour ces systèmes sont encore les plus utilisés, d'autres travaux ont été récemment menés visant des concepts très différents et plus évolués. Les recherches se tournent actuellement vers des méthodes basées sur l'apprentissage, souvent avec succès.

3.3.3.3.2 Méthodes basées sur l'apprentissage

La plupart des méthodes basées sur l'apprentissage utilisent des réseaux de neurones formels (RN). Dans notre application, avant de de-

mander à un RN de choisir un bloc adresse parmi plusieurs candidats, il faut adapter automatiquement sa connaissance au problème posé en utilisant une étape d'apprentissage. Cette étape vise l'un des deux objectifs suivants : apprentissage pour séparer les blocs non textuels des blocs textuels ou apprentissage pour distinguer le bloc adresse directement à partir des autres blocs candidats.

Apprentissage pour séparer les blocs non textuels des blocs textuels :

Une fois la séparation texte / non texte effectuée, le choix du bloc adresse à partir d'autres blocs textuels candidats devient relativement aisé. En général, quelques règles simples sont suffisantes à la prise de décision. Ce type de méthodes est très efficace sur des enveloppes de structure simple et variable, où le bloc adresse représente souvent le seul contenu textuel. Dans ce cadre, Jain et Bhattacharjee dans [JAI92a] ont utilisé les filtres de Gabor pour distinguer les différentes textures présentes sur l'image de l'enveloppe où le texte est défini comme une zone de texture particulière. Un classifieur à base de réseau de neurones d'une seule couche est utilisé, pour séparer les régions de texte des régions graphiques. Les éléments textuels sont regroupés en blocs candidats, et la sélection du bloc adresse est établie par une simple heuristique. Bien que les auteurs aient montré des exemples réussis, ils ont limité leurs résultats sur des courriers simples. En outre, la méthode basée sur le filtre de Gabor est très coûteuse en temps. Aucune évaluation de l'exactitude de la méthode et aucun taux de LBA n'ont été présentés dans leurs travaux. Sur d'autres types de documents comme les feuilles d'impôts de structures complexes et stables, Srihari et al [SRI96] ont utilisé le discriminateur linéaire de Fisher pour séparer le texte du graphique. La localisation du bloc adresse dans le contenu textuel repose donc sur les techniques de mise en correspondance avec un modèle prédéfini. Ce système lit 2,36 feuilles d'impôts par seconde.

Apprentissage pour distinguer le bloc adresse directement à partir des autres blocs candidats :

Sur des enveloppes de structure complexe et variable, il est plus intéressant de faire un apprentissage directement sur les blocs pour accélérer la reconnaissance de chacun et trouver celui qui contient l'adresse de destination. Au lieu d'utiliser des caractéristiques de bas niveau, Wolf et John [RWO94] ont engagé un apprentissage par RN afin de produire un résumé des caractéristiques de haut niveau du bloc adresse. Cette approche a permis d'éviter la fastidieuse tâche de mise au point d'un grand nombre de règles ou de modèles explicites du bloc adresse. L'apprentissage a été ef-

fectué sur une base de 800 images et les tests sur une base de 500 images d'enveloppes imprimées. Le taux de bonne localisation est de 98.2%. Ce taux semble être le meilleur taux estimé par rapport aux autres méthodes. Eiterer et al [EIT04] ont suivi le même principe en exploitant la représentation des images en niveaux de gris des enveloppes manuscrites dans les dimensions fractales et en y réalisant une classification des pixels de type K-means. Les ensembles des pixels sont reconnus comme appartenant aux classes de fond, de bruit ou d'objets logiques (avec les labels : timbre, cachet postal ou bloc adresse). Ce principe repose sur une classification simple. Il a permis de réaliser conjointement plusieurs tâches : une extraction de premier plan, la suppression de bruit, la séparation des entités physiques et la localisation du bloc adresse.

Plus récemment, Roy et al 2005 [ROY05] ont proposé un système à base de RN pour lire des adresses postales écrites en deux langues : anglaise et indienne (Bangla). La méthode RLSA et un ensemble de caractéristiques de différentes composantes d'image sont utilisés pour extraire la zone de timbre, le cachet et les graphiques. Des relations spatiales ont été utilisées par la suite pour localiser les zones candidates. La recherche de code-pin en Bangla et en anglais utilise deux réseaux de neurones. La ligne de partage des eaux est utilisée avec une HMM et un réseau de neurone pour décomposer le bloc adresse en lignes puis en mots. Afin d'améliorer les performances de ce système, une étape de lecture du code postal et du nom de ville a été rajoutée au système. Ce type de système a permis de mettre en place un processus coopératif entre d'une part la phase de la segmentation, la phase de reconnaissance de mots et la LBA. Mais ce type de coopération est très consommatrice en temps de calculs en raison des allers-retours de validation et de vérification.

3.3.4 Bilan sur les méthodes de LBA

Nous avons présenté dans cette partie les méthodes essentielles de LBA : nous les avons classées selon leur modalité d'action répondant à différentes approches allant de l'émergence des blocs à la décision.

Suite à l'observation du comportement des différentes approches de LBA recensées dans cette section, nous pouvons affirmer que les méthodes qui s'appliquent sur des images binaires et qui utilisent une segmentation mixte (mi-ascendantes et mi- descendantes) sont plus robustes (meilleur capacité de localisation sur des images bruitées) que les approches exclusivement ascendantes ou descendantes. Nous pouvons également remarquer que les méthodes basées sur des règles déterministes nécessitent un nombre très élevé de critères. Le choix et la gestion de ces critères ainsi que l'ensemble des connaissances à prendre en considération deviennent

difficilement contrôlables face à la grande variabilité des enveloppes à trier.

Dans cette section, nous avons porté une attention toute particulière aux mécanismes d'apprentissage se présentant comme une solution alternative robuste pour résoudre la tâche de LBA. Ces approches permettent notamment au système de se libérer des tâches fastidieuses et coûteuses de description d'heuristiques concernant les courriers de structures variables et/ou complexes. Les méthodes les plus intéressantes sont celles qui s'entrainent à séparer et à classer directement les blocs quelle que soient leur nature.

La méthode de Wolf et John [RWO94], utilisant un réseau de neurones, donne un score de LBA (98,2%) à ce jour imbattable par rapport à toutes les autres méthodes. Cela montre que les réseaux de neurones sont également capables d'analyser automatiquement des relations spatiales et topologiques (Tableau 2). Par leur grande capacité à représenter n'importe quelle dépendance entre variables, les réseaux de neurones n'ont pas besoin de solliciter un modèle descriptif très complexe des contenus. Cependant, ils présentent un certain nombre d'inconvénients ou de contraintes : on peut constater qu'utiliser les réseaux de neurones ne dispense pas de bien connaître les problèmes liés à la LBA, ni de bien définir les classes avec pertinence, ou encore de bien définir les variables importantes. Mais la plus grande difficulté réside dans le fait qu'un réseau de neurones est une «boîte noire» qui n'explique pas ses décisions : la validation du modèle neuronal est donc parfois difficile à argumenter. Néanmoins, les réseaux de neurones ont une très bonne prédiction statistique (ayant la capacité de s'accommoder de valeurs très bruitées ou même manquantes), et la perte partielle de compréhension est rapidement compensée par la qualité des prédictions. C'est dans ce cadre que se situe notre proposition visant plus de robustesse vis-à-vis des contraintes présentées, plus de souplesse et des performances en temps et en précision accrues par rapport à l'existant.

Nous présenterons ainsi dans la partie suivante notre proposition d'architecture innovante de LBA issue de représentations pyramidales des images de documents (faisant appel à la multirésolution) et à la théorie des graphes. Les étapes de haut niveau reposent en partie sur la coloration hiérarchique des graphes (que l'on note CHG), permettant de synthétiser automatiquement, par l'intermédiaire d'une organisation pyramidale des données, la gestion des règles composées. Ces règles gèrent l'interprétation de la décomposition des images de courriers en composantes connexes, la formation et la reconnaissance des zones d'intérêt. A ce jour, aucun travail dans ce domaine ne s'est servi de la puissance de cet outil. A l'inverse des méthodes classiques qui utilisent bien souvent des architectures linéaires,

notre stratégie consiste à augmenter les performances de chaque module et leur cohérence avec les autres tâches du processus de localisation du bloc-adresse afin de réduire au maximum les rejets de courriers et les temps de traitement.

Méthode	Taille de la base d'images	Stratégie de LBA	Nature des courriers	Type de texte	Résolution (dpi)	Couleur	Taux de localisation et temps
Wang 1988	174	Base de connaissances	Courrier complexes	imprimé		NG	81%
Downton 1990		Règles	Courrier simple	imprimé			97%
Palumbo 1992	2000 images de test 63% imprimées Et 37% manuscrites	Règles	Lettres	Imprimé et manuscrit	100	NG	L=89% T=0.9 s
Lee 1994	1000 500 base de connaissances 500 base de test	Base de connaissances	courrier manuscrit Coréen	manuscrites de structures simples	200	NG	94,4%
R. Wolf 1994	800 images d'apprentissage 500 image de test	Réseau de neurones	Enveloppe Imprimé	Imprimé	300	NG	98,4 T= 0.3 à 0.5
Jain 1996	53 images	Règles	Magazine Lettres (simple, complexes)	Imprimé, manuscrit	40	RGB	72% T=0.3 s
Yu 1997	109 images de test 53 magazines 52 lettres simples	Réglés	Lettre, magazines, journaux	Imprimé et manuscrit	41	RGB	71,70% (Magazine) 92,86% (Lettres) Moy=82,57%
M. Wolf 1997	2000 images = 1400 imprimées et 600 manuscrites	Règles et mesure de similarité	enveloppes	Imprimé et manuscrit Bruité (publicité, logos et timbres)		NG	91,3% 98% BA est dans 3 premières réponses
Xue 1999		Règles	enveloppes manuscrites chinoises complexes		300		93,5%
Yonekura 2003	300	Règles					75%
Jeong 2004	1988= 1000 imprimées 988 manuscrites	Règles	Enveloppes coréennes	Imprimé Et manuscrit	200	NG	Totale= 96% 91,09% manuscrit 93,56% imprimé
EITERER 2004		Règles	complexe	Imprimé Et manuscrit	300		93,5

Tableau 3.2: Bilan sur les caractéristiques utilisées par les méthodes de LBA présentées dans la littérature.

Chapitre 4

Apport de la théorie des graphes

pour l'analyse de la structure physique et la reconnaissance des documents

4.1 Introduction

Les systèmes industriels de lecture et de reconnaissance automatique de documents sont par nature très exigeants en temps de traitement, en justesse et précision des résultats. Dans les chapitres précédents nous avons cherché à mettre en lumière la nécessité de recourir à des techniques d'analyse des images de documents plus adaptées et plus rapides. Nous avons vu notamment que les exigences des applications de tri automatique de documents étaient trop contraignantes pour que l'on puisse se satisfaire exclusivement d'approches existantes tant au niveau des prétraitements que de la reconnaissance. Nous avons souligné dans les chapitres précédents les limites de chacune d'elles en terme de performance et de temps de traitement. Cette étude nous a permis de déterminer les étapes clés dans une chaine de tri qui sont responsables de la plupart des cas de rejets, d'erreurs, de consommations mémoire et temps processeur excessifs. Nous sommes maintenant convaincus que toute amélioration du système de tri nécessite l'élaboration d'outils plus performants permettant d'améliorer les principaux modules de l'extraction de la structure physique des documents notamment par une meilleure localisation du bloc adresse sur le courrier postal qui soit robuste au bruit et qui dans le même temps soit capable de produire une reconnaissance automatique fiable du type de documents rencontrés. Nous nous sommes ainsi intéressés à une approche innovante de l'analyse des structures et de la reconnaissance du type de documents utilisant le concept de coloration de graphes jamais exploité dans un tel contexte.

Nous montrerons dans ce chapitre comment ce concept peut être mis au service de tâches de classification rendues plus efficaces et rapides et comment une approche non paramétrique et simple à mettre en œuvre (en terme d'implémentation) peut, en pratique, s'avérer très performante par sa capacité à fournir une très bonne sélection des représentants des classes, un bon contrôle des différentes phases de classification, et surtout l'obtention de résultats correspondant à l'information réellement présente dans la base d'apprentissage. Nous expliciterons notamment diverses adaptations d'algorithmes de coloration de graphes dans leur version non supervisée selon la nature des documents à trier. Les différentes adaptations des approches de coloration de graphes requises par le sujet de notre étude constituent la partie innovante de ce chapitre et le cœur même de notre contribution

La coloration de graphe constitue une branche très importante de la théorie de graphes. Ses applications sont nombreuses dans différents domaines scientifiques. Cela justifie une recherche importante en algorithmique. Les définitions de la coloration sont simples et de véritables pro-

blèmes de recherche peuvent être posés sous une forme bien structurée dont la formulation peut recouvrir de grandes difficultés pratiques. Il s'agit d'un modèle qui n'a jamais été utilisé dans le domaine du traitement d'image de document. Grâce à sa simplicité et son potentiel en matière de classification, nous avons pu imaginer des méthodes originales de segmentation, d'apprentissage, de reconnaissance et de localisation de régions d'intérêt dans les images de courriers d'entreprise. Sans écarter la contrainte du temps exigé par notre application, l'exploitation et l'adaptation d'un même modèle dans toutes les étapes d'analyse des documents (de la localisation à la reconnaissance) a permis de consolider la coopération et d'assouplir les échanges d'information entre les différents modules. Notre contribution s'inscrit donc à tous les niveaux de la chaîne d'analyse et de reconnaissance et repose différentes adaptations des modèles génériques de coloration et de b-coloration de graphes. Ces adaptations nous ont permis de mettre au point des algorithmes spécifiques de coloration visant à approximer le nombre chromatique minimal du graphe en complexité calculatoire linéaire.

Il faut avant tout savoir que le nombre chromatique d'un graphe est généralement déterminé en appliquant un algorithme exact sur un graphe présentant moins d'une centaine de sommets, dans un contexte où les limites de temps de calcul ne sont pas fixées. Dans des applications réelles contraintes par le temps (notre application), on doit se contenter d'estimer le nombre chromatique. Différents mécanismes de coloration sont alors possibles pour y parvenir : l'utilisation d'approches séquentielles ou heuristiques. Dans ce dernier cas, l'algorithme ne donne qu'une borne supérieure à la valeur du nombre chromatique mais produit un résultat dans des temps bien inférieurs à ceux des approches dites exactes. Partant de cette constatation et en nous inspirant conjointement des méthodes heuristiques et des méthodes séquentielles, nous avons développé un nouvel algorithme de coloration propre qui s'applique facilement sur n'importe quel graphe, quelle qu'en soit la taille. Nous avons destiné cet algorithme à la segmentation en temps réel de la structure physique des documents. Nous avons ensuite adapté un algorithme de b-coloration en deux étapes qui à partir de la détermination du nombre chromatique minimal produit un graphe de sommets dominants (sommets dont les voisins sont colorés par toutes les autres couleurs du graphe). Cette seconde proposition a été mise au point pour répondre aux problèmes de classification et de reconnaissance indispensable à l'application (classification des documents selon leur type et reconnaissance des blocs informants pour la localisation du bloc adresse).

Notre objectif de cette partie est donc de présenter les aspects théoriques de la coloration de graphe et de la b-coloration de sommets dans le cas général. Nous présentons ensuite en détails les algorithmes que nous avons produits et qui s'adaptent aux besoins de notre application temps réel en terme de :

- extraction de la structure physique des images
- localisation de blocs adresse et reconnaissance des familles de documents et de courriers d'entreprises.

L'apport de la coloration et de la b-coloration de graphes dans les phases de segmentation et de reconnaissance de documents est ensuite validé et discuté.

4.2 Les fondements théoriques de la coloration des graphes

4.2.1 Un aperçu historique de la coloration des graphes

La coloration de graphes est à l'origine un champ majeur et très actif de la théorie des graphes. Cette théorie s'est surtout développée depuis la deuxième moitié du XIX ème siècle et connaît une explosion depuis le début de ce siècle. Pour clarifier les idées, je partirai de la définition suivante: un graphe est une structure simple constituée d'un ensemble de points (appelés sommets ou nœuds), reliés entre eux par un ensemble de liens (appelés arêtes ou arcs). Chaque arête a pour extrémités deux points, éventuellement confondus.

La théorie des graphes sert avant tout à représenter et à organiser les tâches de façon optimale : après avoir traduit un problème sous forme de graphe, elle cherche à trouver la succession la plus rapide ou la moins coûteuse pour effectuer toutes les tâches. De fait, ses applications pratiques sont très diverses: optimisation dans les réseaux de transports, conception de réseaux électriques, de réseaux de communication, mécanique statistique, formules chimiques, sciences sociales, géographie, reconnaissance de formes et mise en correspondance entre images par isomorphisme de graphe…

La coloration est apparue depuis longtemps, comme en témoigne le fameux problème des 4 couleurs posé par Francis Guthrie en 1852 : est-il possible de colorier toute carte géographique avec au plus 4 couleurs de sorte que 2 régions qui ont une frontière en commun aient des couleurs différentes? Un coloriage d'un graphe est une fonction qui affecte une couleur à chaque sommet, et qui est telle que deux sommets voisins n'ont pas la même couleur. Comme nous allons le voir, la contrainte la plus courante est

celle de la propreté : deux éléments voisins doivent avoir des couleurs différentes. Les couleurs (ou entiers) attribuées aux éléments du graphe servent uniquement à regrouper les éléments en classes.

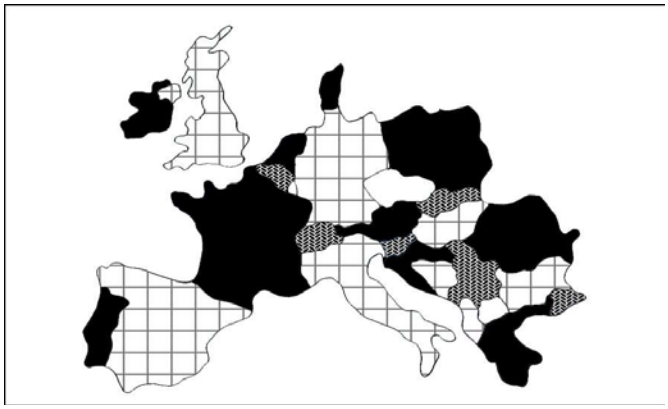


Figure 4. 1 : Carte de l'Europe colorié avec 4 couleurs, [LEV07].

Aujourd'hui, la coloration de graphes permet de modéliser de nombreux problèmes réels, depuis le placement de personnes autour d'une table ou de pièces sur un échiquier jusqu'aux différents problèmes d'ordonnancement et de planification de la vie de tous les jours (transport, logistique, réservation de ressources,...) et notamment dans le domaine des réseaux/télécom. A ce jour cet outils n'été jamais appliqué au domaine de la lecture et la reconnaissance automatique de documents.

4.2.2 Représentation des graphes et notations

Nous allons tout d'abord rappeler quelques notions communes utilisées dans les différentes parties. Les graphes considérés dans ce document sont en général non orientés et simples (sans boucle, ni arête multiple).

4.2.2.1 Définition d'un graphe

Un graphe fini, simple et non orienté $G = (V, E)$ est défini par l'ensemble fini $V = \{v_1, v_2, ..., v_n | V| = n\}$ dont les éléments sont appelés sommets, et par l'ensemble fini $E = \{e_1, e_2, ..., e_m / |E| = m\}$ dont les éléments sont appelés arêtes. Une arête e entre les sommets v_1 et v_2 sera notée (v_1, v_2) ou même simplement v_1v_2. Les sommets v_1 et v_2 sont appelés les extrémités de e. Si l'arête e relie les sommets v_1 et v_2, on dira que ces sommets sont adjacents ou incidents à e, ou encore que l'arête e est incidente aux sommets v_1 et v_2. Pour chaque sommet $v \in V$, nous définissons $N(v)$, comme l'ensemble des sommets qui lui sont adjacents.

4.2.2.2 Degré d'un sommet

Pour un graphe, on appelle degré du sommet v, et on note $deg(v)=|N(v_i)|$, le nombre d'arêtes incidentes avec ce sommet.

4.2.2.3 Degré et diamètre d'un graphe:

Le degré Δ d'un graphe est le degré maximum de tous ses sommets :

$$\Delta = \max \{\deg(v_i) \mid v_i \in V\} \qquad (4.\ 1)$$

Soient *x* et *y* deux sommets de *G*. La distance entre ces deux sommets dans *G* représente la longueur du plus court chemin entre *x* et *y*. Le diamètre d'un graphe *G*, *diam(G)*, est la distance maximum entre deux sommets dans le graphe *G*.

4.2.2.4 Les différentes représentations d'un graphe

On peut représenter un graphe par une matrice d'adjacences notée M_a. Celle-ci est alors une matrice carrée d'ordre *n*, de *n* lignes et *n* colonnes, obtenue en mettant 1 à l'intersection de la ligne *i* et de la colonne *j* lorsqu'il existe une arête reliant les sommets v_i et v_j (si les deux sommets sont adjacents) et en mettant 0 s'il n'existe pas d'arête.

$$M_a(i,j) = \begin{cases} 1 & \text{si } v_i \text{ est adjacent au } v_j \\ 0 & \text{sinon} \end{cases} \qquad (4.\ 2)$$

Cette matrice a d'autres caractéristiques: il n'y a que des zéros sur la diagonale, elle est symétrique: $M_a(i,j) = M_a(j,i)$.

On peut aussi représenter un graphe par un diagramme sagittal ou par un tableau d'adjacences en donnant pour chacun de ses sommets la liste des sommets auxquels il est adjacent.

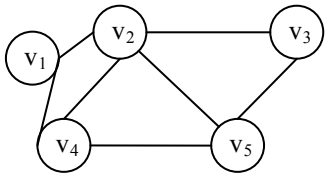


Diagramme sagittal

$$M_a(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ & 0 & 1 & 1 & 1 \\ & & 0 & 0 & 1 \\ & & & 0 & 1 \\ & & & & 0 \end{bmatrix} \end{array}$$

Matrice d'adjacences (Ma)

V	N(v)
V_1	V_2, V_4
V_2	$V_1, V_3,$
V_3	V_2, V_5
V_4	V_1, V_2, V_5
V_5	V_2, V_3, V_4

Tableau d'adjacences (Ta)

Figure 4. 2 : Exemple de trois représentations d'un graphe de 5 sommets *(n=5, Δ=deg(v₂)=4)*.

Mais, il ne faut pas confondre un graphe et son dessin : un même graphe peut être dessiné de plusieurs façons. La lisibilité de la visualisation est une question importante, et le dessin du graphe est à lui seul un domaine de recherche dont les applications sont nombreuses: fabrication de cartes routières, représentation d'interactions entre sous-systèmes, aide graphique à l'ordonnancement des tâches, pour ne prendre que quelques exemples.

4.2.3 Les aspects fondamentaux de la coloration des graphes

La coloration de graphes est un outil permettant de caractériser les graphes. Il existe ainsi plusieurs types de colorations : par exemple, la coloration de sommets à laquelle nous nous sommes intéressés [GCH98] [EMA99] [EFF03][EFF06], la coloration d'arêtes [BOJ01], la coloration par liste [HIL01]. La coloration des sommets d'un graphe *G(V,E)* consiste à affecter à tous ses sommets une couleur de telle sorte que deux sommets adjacents ou voisins (***reliés par une arête***) ne portent pas porter la même couleur.

4.2.3.1 Définitions

Définition 4. 1 : Un graphe est k-coloriable s'il peut être colorié au moyen d'un ensemble de *k* couleurs.

Définition 4. 2 : Une k-coloration d'un graphe *G* est définie comme une fonction *c* sur $V(G)=\{v_1, v_2,\ldots, v_n\}$ dans un ensemble de k couleurs (généralement, C = {1,2,…,k}), telle que pour tout sommet v_i, avec $1 < i < n$, nous avons $c(v_i) \in C$ et pour toute arête (v_iv_j) *de E(G),* $c(v_i) \neq c(v_j)$. L'ensemble de couleurs de sommets de $N(v_i)$ sera noté $Nc(v_i)$. Un graphe qui admet une k-coloration est dit k-coloriable.

Définition 4. 3 : Le nombre chromatique (*chromatic number*) $\chi(G)$ d'un graphe *G* est le plus petit entier *k* tel que *G* admet une k-coloration. Si $c(G) = k$, le graphe *G* est dit k-chromatique (k-chromatic).

Certains auteurs proposent de définir la coloration comme une partition des sommets en ensembles indépendants (on parle de stables). Les couleurs, servant juste à définir quels éléments font partie d'un même stable, sont en général représentées par un numéro (un entier). Une coloration pour laquelle deux sommets adjacents n'ont pas la même couleur est dite coloration propre. Sur le graphe *G* de la figure 4.3, dont l'ensemble de 11 formes différentes *V* représenté par les sommets $\{v_1,\ldots,v_{11}\}$, on a eu besoin de quatre couleurs pour colorer les *11* sommets de sorte que deux sommets adjacents (qui peuvent dans le cas de la figure 4.3 représenter deux formes dissemblables) ont des couleurs différentes.

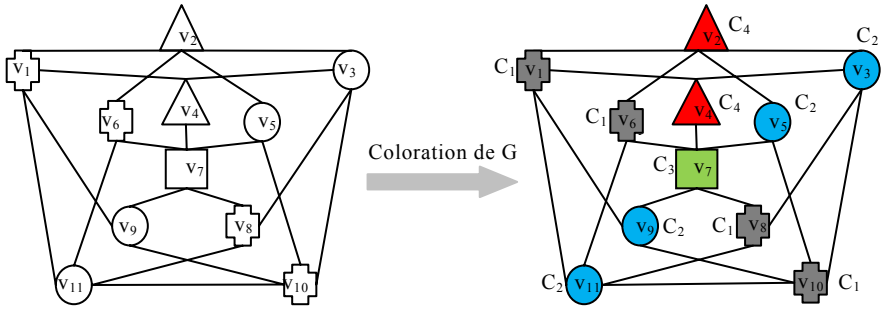


Figure 4. 3 : Coloration de graphe *G* de *11* sommets par *4* couleurs *(c_1, c_2, c_3 et c_4)*.

4.2.3.2 Encadrement du nombre chromatique $\chi(G)$

L'un des résultats connus concernant le nombre chromatique est donné par Brooks [BRO41] . On a l'encadrement suivant sur le nombre chromatique : pour tout graphe G

Théorème 4. 1 : La majoration de $\chi(G)$ est donnée par **:**

$$\chi(G) \leq \Delta(G)+1 \quad (4.\ 3)$$

Preuve: Soit un graphe et Δ le degré maximum de ses sommets. Donnons-nous une palette de $(\Delta + 1)$ couleurs. Pour chaque sommet du graphe on peut tenir le raisonnement suivant: ce sommet est adjacent à Δ sommets au plus, et le nombre de couleurs déjà utilisées pour colorer ces sommets est donc inférieur ou égal à Δ. Il reste donc au moins une couleur non utilisée dans la palette, avec laquelle nous pouvons colorer notre sommet.

Théorème 4. 2 : La minoration de $\chi(G)$ est donnée par :

$$\chi(G) \geq \omega(G) \quad (4.\ 4)$$

où $\omega(G)$ est la taille maximum des cliques du *G*.

Preuve: Puisque, par définition, dans une clique (nombre maximum de sommets 2-à-2 adjacents) d'ordre *m*, tous les sommets sont adjacents entre eux, il faut *m* couleurs. Le nombre chromatique du graphe est donc supérieur ou égal à l'ordre de sa plus grande clique.

La détermination du nombre chromatique d'un graphe est dans le cas général un problème difficile (NP-complet). De nombreux travaux ont donc été menés pour définir des bornes pour ce paramètre en fonction d'autres paramètres de graphe [CHE97] [KEM98] [CHE04] [WER03] [HEU03] [PAS07].

4.2.4 Le problème du choix de la meilleure coloration

La question essentielle soulevée en matière de coloration est de savoir quelle est la meilleure coloration possible du graphe. Dans de nom-

breuses situations il est essentiel de savoir si un graphe peut être colorié avec un nombre *fixe* de couleurs et, si tel est le cas, comment y parvenir.

4.2.4.1 Approches par minimisation du nombre chromatique

Déterminer le nombre chromatique d'un graphe est un problème central de l'optimisation combinatoire qui, par nature, est NP-complet dès lors que le nombre de couleurs est supérieur à 3. De nombreuses méthodes approchées ont été proposées pour résoudre le problème de coloration. Il existe assez peu de méthodes *exactes* qui s'avèrent efficaces : elles ne sont généralement applicables que sur des graphes de petite taille. On parle dans ce cas de coloration *exacte*, à la différence des approches de coloration approximée que l'on nomme approches *heuristiques* et *séquentielles*, elles sont généralement plus rapides et utilisables sur des graphes comportant un grand nombre de sommets. On se contente dans ce cas d'estimer une borne supérieure au nombre chromatique et non pas de le déterminer précisément.

Plus spécifiquement :

- Pour les approches exactes, le principe consiste à considérer tous les ordres possibles sur les nœuds et à appliquer une coloration pour chacun d'eux. Dans ce type d'approche, on est toujours certain d'obtenir une solution pour $\chi(G)$ qui est la meilleure, mais qui n'est pas aisée à obtenir en pratique (pour des graphes présentant plus de *85* sommets).

- Pour les approches séquentielles, le principe consiste à estimer une borne maximale du nombre chromatique assurant que le nombre total de couleurs utilisés ne dépasse pas $\Delta(G) + 1$. A chaque itération, le sommet à colorer reçoit la plus petite étiquette de couleur disponible Les algorithmes séquentiels ont l'avantage de la rapidité (quelques secondes de calcul pour des graphes de plusieurs centaines, voire milliers, de sommets), mais utilisent en général plus de couleurs que nécessaire. C'est pourquoi des algorithmes heuristiques plus efficaces ont été proposés, comme celui de Werra et Kobler dans [WER03]. Ils constituent généralement de bons compromis entre temps de calcul et qualité du résultat.

Une étude comparative dans [PAS07] a confirmé le fait que les algorithmes *séquentiels* ont l'avantage de la rapidité, mais utilisent en général plus de couleurs que nécessaire. A l'opposé, les algorithmes *exacts* permettent de déterminer avec une grande précision le nombre chromatique, mais nécessitent des temps de calculs très grands (une journée de calcul pour colorer un graphe d'une centaine de sommets). Si les algorithmes *séquentiels* ont l'avantage de la vitesse, ils fournissent en général une borne supérieure très large pour des graphes ayant plus de quelques centaines de sommets. C'est la raison pour laquelle les algorithmes *heuristiques*, comme ceux basés sur la méthode Tabou [GLO96], sont plus effi-

caces que la plupart des algorithmes séquentiels. Un avantage supplémentaire de l'algorithme Tabou provient du fait qu'il est relativement aisé de l'adapter pour répondre aux exigences de diverses applications.

Parmi les approches existantes au sein de ces trois catégories de méthodes de coloration, on peut citer les algorithmes séquentiels gloutons (Welsh et Powell), l'approche DSATUR de Brelaz dans [BRE79] basée sur le degré de saturation du graphe, l'algorithme BSC (Backtracking Sequential Coloring), les algorithmes basés sur l'arbre de Zykov, [COR73], [BRI81], les approches de coloration tabou et tabou renforcée (RTS), [GLO96], [DOR98] les stratégies d'énumération implicite, [SEW96], la génération de colonnes et programmation linéaire [MEH96], les approches hybrides évolutives [GAL99], le branch-and-bound [CAR02] et branch-and-cut [DIA02], les approches de décompositions linéaires du graphe d'entrée et la combinaison de solutions partielles calculées pour différents sous-graphes, [LUC04]. D'autres approches optimisant les méthodes citées ici ont encore vues le jour ces vingt dernières années.

4.2.4.2 Approches par maximisation du nombre chromatique

A l'inverse de tous ces paramètres qui tentent à minimiser le nombre de couleurs, il existe d'autres paramètres qui cherchent à maximiser le nombre de couleurs. Le nombre achromatique $\varphi(G)$, par exemple, est le nombre maximum de couleurs nécessaires à la coloration d'un graphe G pour que la coloration soit propre et que chaque paire de couleurs apparaisse au moins sur une des arêtes de G. Ce paramètre a été introduit en 1970 par Harary et Hedetuiemi [HAR70] et il fournit une borne supérieure au nombre chromatique: pour tout graphe G, $\chi(G) \leq \varphi(G)$.

Dans notre étude, nous sommes intéressés par un autre type de paramètre de coloration de sommets maximisant aussi le nombre de couleurs à utiliser, appelé nombre *b-chromatique*. Nous avons choisi l'algorithme itératif distribué de b-coloration (ou double coloration) proposé par Effantin [EFF06]. Pour appliquer une première coloration, Effantin propose un algorithme exact qui donne rapidement en temps linéaire $\Delta(G)+1$ couleurs (le nombre chromatique est maximisé par la borne supérieure). Cet algorithme exact s'applique sur n'importe quel type de graphe et il est beaucoup plus rapide que d'autres algorithmes exacts qui tentent à déterminer le nombre minimal de couleurs.

4.2.5 Les fondements théoriques de la b-coloration : un outil récent de grande performance

Le concept de la b-coloration a été introduit la première fois en 1999 par Irving et Manlove dans [IRV99]. Soit $G=(V, E)$ un graphe simple non orienté, où V est l'ensemble des sommets et E l'ensemble des arêtes. La coloration de G est appelée b-coloration, si pour chaque couleur C_i, il existe au moins un sommet v_j coloré par la couleur C_i dont le voisinage est coloré par toutes les autres couleurs. Le sommet v_j est dit sommet dominant pour la couleur C_i. Contrairement à la coloration minimale de sommets d'un graphe G, la b-coloration consiste à colorer les sommets du graphe avec un maximum de couleurs sous les contraintes de propretés et de domination (voir définition 4.5).

L'algorithme de b-coloration des sommets du graphe G s'éxécute en général en deux étapes : 1) génération d'une coloration propre des sommets de G avec un nombre maximum de couleurs

2) suppression, par une procédure gloutonne, de chacune des couleurs n'ayant pas de sommet dominant jusqu'à stabilité de la coloration (i.e. où toutes les couleurs du graphe G sont dominantes).

Définition 4. 4 : Une b-coloration d'un graphe G est définie comme une fonction c sur $V(G)$ $\{v_1, v_2,... ,v_n\}$ dans un ensemble de k_b couleurs (généralement, $C = \{1,2,...,k_b\}$), qui consiste à colorer tous les sommets de V à l'aide d'une coloration maximale de telle sorte que :

- pour tout sommet v_i, avec $1 < i < n$, nous avons $c(v_i) \in C$ et pour toute arête (v_iv_j) de $E(G)$, $c(v_i) \neq c(v_j)$.

- pour toute classe de sommets coloré par la couleur c, il existe au moins un sommet $v_i \in V$, coloré par cette couleur et adjacent à toutes les autres couleurs, appelé sommet dominant.

Définition 4. 5 : Une couleur avec un sommet dominant est dite couleur dominante (une couleur non dominante est une couleur ne contenant aucun sommet dominant).

Définition 4. 6 : Une coloration propre maximale est une coloration des sommets de G avec un nombre maximum de couleurs. Ce nombre est égal dans la plupart des cas à $\Delta(G)+1$.

L'exemple de la figure 4.4 présente la possibilité de b-colorer les sommets d'une classe de couleur à l'aide des autres couleurs. Si on applique la condition de domination sur les couleurs de G, chaque couleur doit être représentative d'au moins un sommet dominant qui est adjacent à toutes les autres couleurs. On remarque que la première coloration de G n'était pas suffisante pour vérifier cette condition importante et que toutes ses 5 couleurs sont non dominantes. Dans ce cas l'application de la b-coloration sur

le graphe n'a eu pour effet que de supprimer la couleur c_5 et de redistribuer les sommets de couleurs c_3 et c_4 pour rendre toutes les autres couleurs restantes dominantes et atteindre, finalement un état de stabilité. Le nombre final des couleurs $b(G)=4$ représente le nombre b-chromatique.

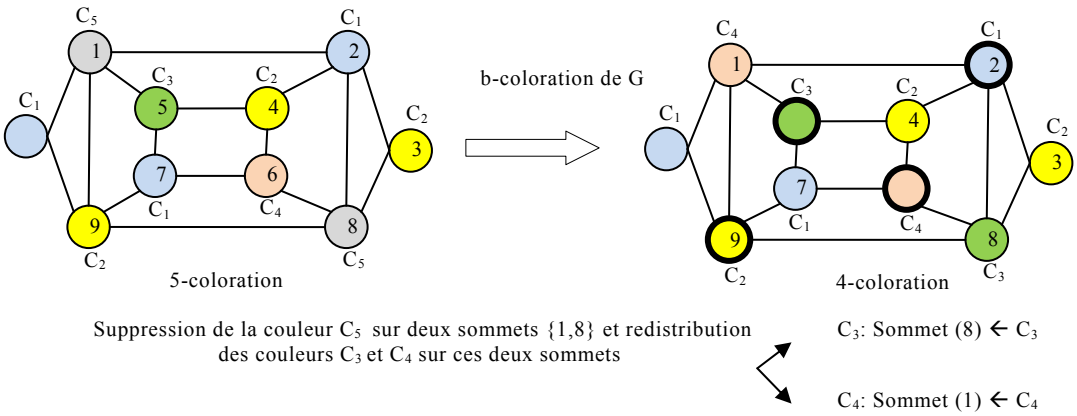


Figure 4. 4 : Exemple de b-coloration de G, **à gauche :** coloration avec un nombre maximum de couleur ($k=5$), **à droite :** suppression des couleurs non dominante jusqu'à stabilité. Les sommets 2, 5,6 et 9 sont les sommets de plus grande dominance.

Définition 4. 7 : Le nombre b-chromatique d'un graphe G, défini par $b(G)$, est le nombre entier maximal de couleurs k_b tel que G peut avoir une b-coloration par les k_b couleurs. Ce paramètre de coloration a été défini dans [IRV99]. Irving et Manlove ont montré dans [IRV99] les propriétés suivantes:

Proposition : Soit G un graphe et $\chi(G)$ son nombre chromatique, défini comme le nombre minimum de couleurs requises pour la coloration propre de G. $\Delta(G)$ est le degré maximum de G. Le nombre b-chromatique peut être encadré par la relation suivante :

$$\chi(G) \leq b(G) \leq \Delta(G)+1 \quad (4.\ 5)$$

Preuve : Comme la b-coloration de G est une coloration propre : $\chi(G) \leq b(G)$. Il est facile de voir que pour tout sommet de degré $\Delta(G)$ peut avoir au maximum $\Delta(G)$ couleurs voisines et prendre pour lui la couleur $\Delta(G)+1$. En conséquence, $b(G) \leq \Delta(G)+1$. Irving et Manlove ont également montré que la recherche du nombre b-chromatique $b(G)$ pour tout graphe G est un problème NP-difficile.

4.3 Quel algorithme faut-il choisir ?

4.3.1 Le choix du bon algorithme pour des applications temps réel

La plupart des évaluations de $\chi(G)$ et de $b(G)$ proviennent d'algorithmes de colorations portant sur des classes paticulières de graphes,

comme les arbres [IRV99], les graphes puissances [EFF03] et les produits cartésiens de graphes [KOU02]. Il en existe de nombreux, voilà pourquoi nous nous limiterons à citer l'étude comparative effectuée par Paschos dans [PAS07]. Plus de détails sur l'approximation du nombre b-chromatique ont été présentés par Corteel dans [COR05]. Plus récemment, Effantin et Kheddouci [EFF06] ont proposé un algorithme distribué permettant de partager la tâche de construction d'une b-coloration d'un graphe G quelconque sur plusieurs processus. Leur approche innovante d'approximation du nombre b-chromatique présente l'avantage d'une distribution parallèle de l'algorithme très utile pour traiter des problèmes très consommateurs de place mémoire et très coûteux en temps de calculs. Eghazel [ELG06] a utilisé cet algorithme de b-coloration dans une application de classification non supervisée des données médicales. La comparaison de la justesse de cette méthode avec celle de la méthode de classification hiérarchique agglomérative, de l'approche du Hansen et de la classification de DRG (Diagnosis Related Groups : classification des patients traités en hospitalier), a montré que cette technique offre une vraie représentation des classes par les individus dominants et garantit une meilleure disparité interclasses.

Dans le cadre applicatif du tri automatique de documents et de courriers d'entreprise, nous avons pensé que les propriétés de cette approche de b-coloration pouvaient être très efficacement utilisées pour la résolution des problèmes de segmentation et de classification de documents. Nous avons donc porté une attention toute particulière à l'adaptation de cette approche au cas de notre étude. Plus spécifiquement, nous avons constaté que les facilités offertes par l'exploitation d'algorithmes distribués de coloration et de b-coloration, comme ceux proposés par Effantin et Kheddouci dans [EFF06], répondent au mieux à la contrainte de temps réel imposé par les applications indutrielles, comme c'est le cas pour nous.

4.3.2 Une approche de b-coloration distribuée

Dans cette partie nous présentons en détail l'algorithme distribué proposé par Effantin et Kheddouci [EFF06] qui déterminer une décomposition ou (regroupement) des nœuds d'un graphe se basant sur le principe de b-coloration. Dans chaque groupe de sommets similaires nous choisissons un représentant de groupe (appelé sommet dominant) qui est le plus éloigné de tous les autres groupes de sommets (et qui possède donc une adjacence directe avec un représentant de chacun des autres groupes).

Le principe de cette approche consiste à chercher les couleurs non dominantes issues d'une coloration propre maximale. Si de telles couleurs

existent, elles doivent être enlevées une par une jusqu'à que toutes les couleurs du graphe *G* soient dominantes. Les sommets des couleurs non dominantes doivent donc soit appartenir à l'une des couleurs dominantes existantes, soit induire la création de nouvelles couleurs dominantes avec les nœuds des autres couleurs non dominantes. Ce processus donne à la fin un regroupement plus précis que celui donné par une seule étape de coloration. Cette méthode de re-coloration de sommets doit être initialisée en affectant aux sommets de degré maximum la *couleur 1*. Cette initialisation est suivie de l'étape de coloration décrite dans l'algorithme suivant, applicable à tout sommet *i* :

Procédure 1: Coloration_Initiale()

```
Début
  Si c(i) ≠ ∅ Alors
    Soit  M = Nc(i)⋃{c(i)}
    Soit    q = 0
    Pour chaque sommet j∈N(i) Sachant que c(j)=∅
    Faire
     q = min {k| k>q, k ∉M et k ∉c(j)}
     Si   q ≤ Δ + 1 Alors c(j) = q
     Sinon
     c(j) = min{k|k ∉Nc(j)}
     FinSi
    FinPour
  FinSi
Fin.
```

avec *c(i)* la couleur de sommet *i* , *N(i)* est l'ensemble de ses sommets adjacents au sommet *i* (*Nc(i)* est l'ensemble des couleurs des sommets *N(i)*), Les auteurs ont montré dans [EFF06] que cet algorithme s'exécute en *O(m)* avec $m = |E|$ (le nombre d'arrêtes dans *G*).

Cette procédure donne exactement $\Delta + 1$ couleurs en temps linéaire, par conséquent, elle ne peut pas être utilisée toute seule sans une deuxième coloration. C'est la raison pour la quelle nous avons développé un algorithme bien spécifique (procédure 7) dédié aux applications qui nécessitent une seule coloration (comme dans la tâche de l'extraction de la structure physique qui sera présentée dans le chapitre 5, section 5.2.2.

Pour calculer une b-coloration d'un graphe *G*, chaque sommet de *G* a besoin de quelques informations sur les sommets dominants et les couleurs dominantes. Nous fournissons ainsi pour chaque sommet *i*, la table $Dom_i\,[c(i)]$ contenant les étiquettes des sommets dominants de chaque couleur (initialisée à *0*). Si une couleur c n'a aucun sommet dominant nous avons $Dom_i[c]=0$ et si elle est retirée de la coloration $Dom_i[c]=\varnothing$. Avant

de retirer une couleur c, chaque sommet doit vérifier s'il n'est pas un sommet dominant pour cette couleur. S'il ne l'est pas, il envoie un message proposant de supprimer la couleur *c*.

Notons que les messages émis par les sommets du graphe sont représentés par l'ensemble $\{M_j(i, c(i)) / j \in \{1..4\}\}$ pour *i* sommet et *c(i)* sa couleur. Avant de les étudier dans le détail, nous présentons deux procédures *Traitement()* et *Attente()* qui interviennent dans leur émission.

La procédure *Traitement ()* applicable à tout sommet *i* évalue la domination du sommet *i* par la comparaison de l'ensemble des couleurs de son voisinage avec l'ensemble des couleurs existantes dans le graphe (chaque sommet maintient à jour la liste des couleurs de son voisinage *N(i))*. Nous choisissons ensuite parmi tous les sommets dominants dans une couleur *c*, le sommet qui possède la plus petite étiquette (qui correspond à la plus grande dominance, i.e. la plus grande distance par rapport aux autres couleurs). Si *i* n'est pas un sommet dominant et sa couleur *c(i)* ne contient aucun sommet dominant, alors on en déduit que la couleur de ce sommet est *non* dominante et on procède à l'attente.

Procédure 2 : Traitement()

```
Début
Soit Nc'=⋃q sachant que
 1 ≤ q≠ c(i) ≤ Δ + 1 et Domi[q] ≠ ∅
 Si Nc'= Nc(i) (i est un sommet dominant) Alors
   Si Domi [c(i)]>i ou Domi[c(i)]=0 Alors
      Domi[c(i)]=i
      Envoyer à chaque k∈N(i) le message M2(i,c(i))
   FinSi
 Sinon
   Si Domi[c(i)]=0 Alors
      Envoyer à chaque k∈N(i) le message M3(c(i))
      Exécuter Attente()
    FinSi
FinSi
Fin.
```

Où *Nc'* est l'ensemble de couleurs dominantes de sommets de *N(i)*.

La procédure *Attente()* donne quelques instructions de calcul durant la phase d'attente utilisée pour s'assurer que le message envoyé a bien atteint tous les sommets de *G*. Cette procédure est appliquée sur un sommet *i* seulement si sa couleur *c(i)* est non dominante. Puisque la méthode est distribuée, on peut appliquer cette procédure à plusieurs sommets en même temps. S'il existe un sommet dominant *j* dans la couleur *c(i)*, cette information sera distribuée sur tous les sommets de *G* en un temps *diam(G)*. Un sommet *i* peut donc arrêter l'exécution de la procédure *Attente()* dès la ré-

ception de cette information. Mais s'il ne reçoit rien après un temps d'attente *diam(G)+1, alors* il estime que sa couleur n'a aucun sommet dominant : il est donc obligé de changer sa couleur.

Procédure 3: Attente()

```
Début
 Après une durée diam(G)+1 faire
  Col = c(i)
  C(i=max{q | 1≤q≠c(i)≤Δ+1 ,q∉Nc(i) et Dom_i[q]≠∅}
  Envoyer à chaque k∈N(i) le message M1(i,c(i))
  Envoyer à chaque k∈N(i) le message M4(col)
Fin.
```

Voici ci-dessous la description de l'ensemble des messages échangés :

Aucun message n'est reçu : Dans ce cas, le sommet *i* applique la procédure 2 *Traitement()* pour évaluer sa dominance.

Message M_1(j,c) : *« le sommet j a la couleur c ».*

Dés qu'un sommet reçoit ce message, il met à jour la liste des couleurs de ses sommets voisins.

Message M_2(j, c) : *« le sommet j est un sommet dominant dans la couleur c ».*

Parmi tous les sommets vérifiant les conditions de domination pour la couleur *c*, le sommet dominant de *c* sera celui qui a la plus petite étiquette. Alors, si *j* est un sommet dominant pour la couleur *c(i) (c-à-d. c(i)=c)*, alors le sommet *i* interrompt la procédure *Attente()*. D'ailleurs, pour limiter le nombre de messages, le sommet *i* distribue cette information seulement si elle est nouvelle. Alors, si ce message est reçu, le noeud *i* applique la procédure suivante:

Procédure 4: $M_2(j,c)$

```
Début
Si  Dom_i[c]>j ou Dom_i[c]=0  Alors
     Dom_i[c]=j
     Envoyer à chaque k ∈ N(i) le message  M2(j,c)
     Si c = c(i)  Alors  interrompre l'Attente()
     FinSi
FinSi
Fin.
```

Message M_3(c) : *« la couleur c est nondominante ».*

Parmi toutes les couleurs non dominantes, la prochaine couleur à enlever sera la plus petite. Ce message est ainsi utilisé par n'importe quel sommet pour déterminer si sa couleur est la prochaine couleur à enlever. Ainsi, si la couleur reçue par le sommet *i* est plus petite que la couleur *c(i)*, alors il propage le message. D'ailleurs, si *i* est en cours d'exécution de la

procédure *Attente()*, alors il l'arrête dès que la prochaine couleur à enlever n'est pas *c(i)*.

Procédure 5 : $M_3(c)$

```
Début
Si c < c(i) ou Dom_i[c]=0 Alors
      Envoyer à chaque k∈N(i) : M3(c)
      Interrompre l'Attente()
FinSi
Fin.
```

Message $M_4(c)$: *« Supprimer la couleur c».*

Si *c(i)=c* alors le sommet i doit prendre une autre couleur existante, et il propage ce message pour enlever la couleur *c* du *G*. Par ailleur, puisque la couleur de sommet i doit être changée, il arrête sa procédure *Attente()* et lance à nouveau la procédure *Traitement()*.

Procédure 6: $M_4(c)$

```
Début
 Si c(i)=c Alors
    Dom_i[c]=∅
    C(i)=max{q|>1≤ q≤Δ+1,q ∉Nc(i)et Dom_i[q]≠∅}
    Envoyer à chaque k∈N(i):M1(i,c(i))
 FinSi
 Envoyer à chaque k∈N(i) : M4(c)
 Interrompre l'Attente()
 Exécuter Traitement()
Fin.
```

Effantin et Kheddouci dans [EFF06] ont montré que l'état stable de la b-coloration de *G* est atteint avec $O(n\Delta)$ échanges locaux, $O(m\Delta^2)$ messages échangés et une complexité en $O(\Delta diam(G))$. Cet algorithme peut être exécuté sur plusieurs machines aussi bien que sur une seule.

4.4 Notre contribution : Résolution des problèmes de segmentation et de classification par coloration de graphes

Dans les deux sections précédentes, nous avons décrit les aspects théoriques de la coloration de graphe et leur intérêt dans les processus de segmentation et de reconnaissance de documents dans un contexte

industriel. Notre proposition consiste à résoudre les questions essentielles liées à la segmentation et la localisation des régions d'intérêt (séparation de régions texte et non texte, localisation du bloc adresse et séparation de régions manuscrites et imprimées), et à la classification des documents selon une typologie précise liée à la nature des courriers à traiter, voir figure 4.5.

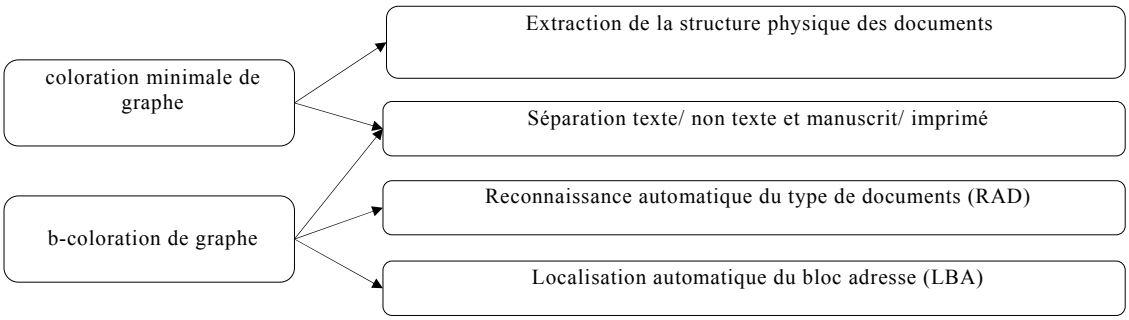


Figure 4. 5 : Contribution de la coloration de graphe à notre application de tri de courriers d'entreprise.

Notre proposition peut être définie comme la première tentative d'adapter et d'appliquer la coloration de graphe au tri de courrier. Plus précisément, nous allons montrer comment l'élaboration d'algorithmes spécifiques de coloration et de b-coloration contribue de façon très robuste à l'extraction de la structure physique des documents et à l'apprentissage pour la reconnaissance du type de documents et la localisation du bloc adresse. La figure 4.6 résume nos contributions. Sur cette figure, nous avons représenté deux parties :

a) Une partie portant sur la coloration propre (ou coloration minimale de graphe) permettant de produire de façon hiérarchique une segmentation physique des documents. Le terme hiérarchique est à considérer ici dans le sens où la segmentation physique se réalise de façon ascendante pyramidale à trois niveaux : le niveau des composantes connexes, le niveau des lignes puis le niveau des blocs. Durant la coloration hiérarchique, l'extraction des caractéristiques d'un niveau de la hiérarchie conduit à regrouper les éléments pour former les éléments de niveau supérieur. Plus spécifiquement, les éléments de la première segmentation (niveau 1) participent à former les sommets d'un second graphe pour la deuxième coloration qui a comme résultat une segmentation en lignes (niveau 2). Puis ces lignes participent à leur tour à la formation des sommets d'un 3ème graphe sur le quel on applique une 3ème coloration pour former la carte finale des blocs (niveau 3). En interaction avec chaque coloration on retrouve une phase d'extraction de caractéristiques des sommets. La segmentation texte / non texte (voir schéma figure 4.6) est le résultat de la première segmenta-

tion (niveau 1 de la hiérarchie). La décision d'étiquetage en texte et en non texte est le résultat d'une classification à partir de l'apprentissage par b-coloration.

b) Une partie portant sur la b-coloration propre (ou coloration en deux étapes : une coloration maximale et une recherche de sommets dominants) qui permet d'apprendre les configurations essentielles que les blocs peuvent prendre, soit pour permettre la localisation du bloc adresse, soit le classement des documents selon une typologie imposée par l'entreprise, soit encore décider si un bloc est imprimé, manuscrit, textuel ou non. L'étape de b-coloration portant sur une base d'apprentissage (ensemble d'exemples choisis pour constituer un ensemble de modèles) se fonde sur un premier graphe conçu à partir de l'extraction de la structure physique de chaque document de la base. Ce graphe subit alors deux étapes de colorations pour l'apprentissage. L'obtention d'un ensemble de sommets dominants conduit à la représentation d'un modèle pour chaque type de documents, modèle qui sera ensuite comparé avec chaque sommet du graphe à tester. Une étape de décision reposant sur un ensemble de critère est ensuite mise au point.

Cette section est dédiée à la présentation de l'apport de la théorie de graphe à ces applications de façon non formelle. La section suivante fera le formalisme complet.

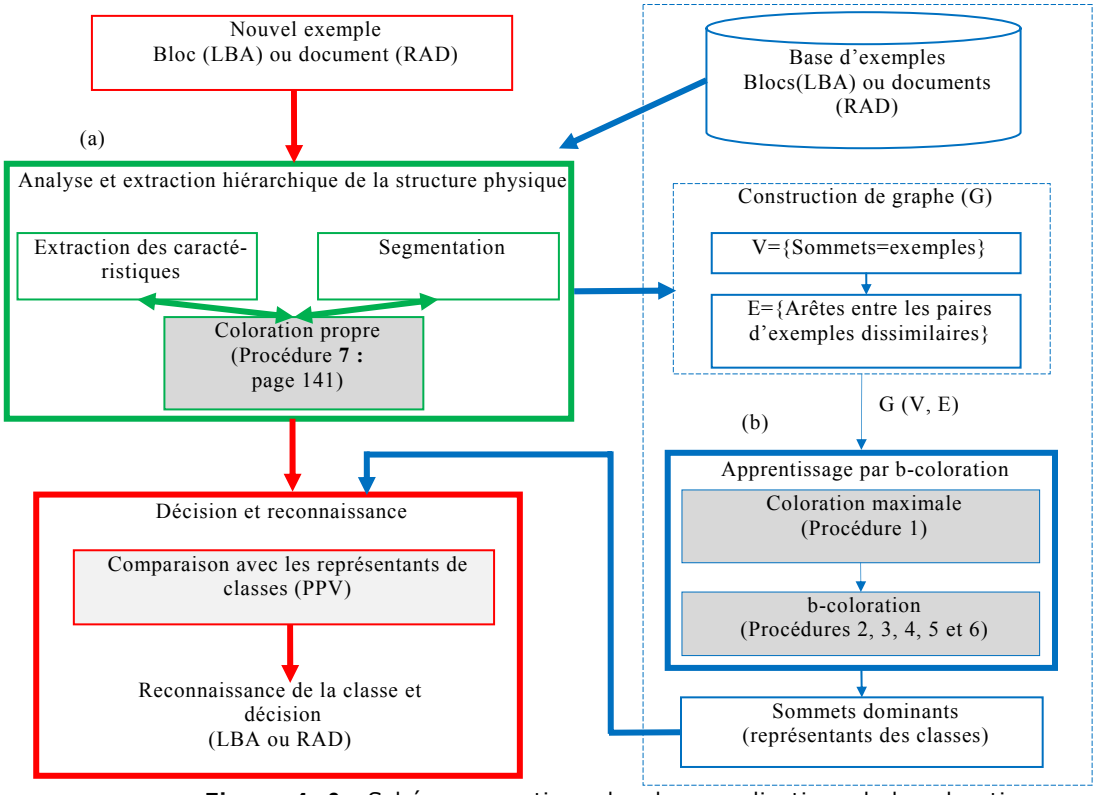


Figure 4. 6 : Schéma synoptique des deux applications de la coloration, **(a)** Extraction de la structure physique par coloration hiérarchique minimale propre, **(b)** apprentissage par b-coloration pour les applications (LBA et RAD).

Dans le chapitre suivant nous reviendrons dans le détail de l'extraction des caractéristiques pour la constitution des matrices d'adjacence nécessaires à la construction des graphes et l'analyse de l'ensemble des résultats produits pour les applications d'extraction de la structure physique, de localisation de blocs adresse et de classification de documents.

4.4.1 Contribution de la coloration minimale de graphe à l'extraction de la structure physique

Nous nous intéressons ici à l'extraction de la structure physique des images de documents et de courriers d'entreprises. Nous avons vu dans le chapitre 2, section 2.4, qu'il était difficile de reconstruire les blocs de texte d'un document de façon ascendante à l'aide de critères locaux. Nous avons également vu que les approches descendantes de segmentation de documents visent à localiser les séparations entre blocs de manière plus globale et moins précise exigeant un grand nombre de connaissances a priori sur la mise en forme de la page. Notre étude comparative a mis l'accent sur les limites des deux stratégies ainsi que les raisons de leur échec. C'est la

raison pour laquelle, nous avons mis en avant la nécessité de faire coopérer ces deux stratégies afin de gagner en temps et en précision.

Nous avons élaboré une coopération de deux approches pour bâtir un formalisme original de coloration minimale de graphe adaptée à la tâche de segmentation. Le principe consiste à regrouper de façon ascendante les éléments constitutifs d'un document (i.e. les CCs) en éléments homogènes de taille de plus en plus importante. Les critères utilisés pour la fusion dépendent de l'homogénéité, du voisinage des éléments de bas niveau du document mais également des propriétés inhérentes à la coloration. La construction de graphe à base de découpage descendant selon les adjacences (dissimilarités) entre sommets fournit une connaissance a priori très riche sur la structure globale de document. Cette connaissance permet de guider le processus de regroupement ascendant durant la coloration et conduit à des décisions de regroupement plus précises lorsque la connaissance locale toute seule ne peut pas être suffisante.

La segmentation d'un document en blocs homogènes consiste à faire apparaître correctement les différents blocs à partir d'un ensemble $X = \{x_1,.., x_n\}$ de composantes textuelles de l'image (caractères, lignes) et regrouper les autres composantes dans des blocs isolés formant les figures et les graphiques. Chaque bloc doit réunir le plus possible d'éléments similaires et voisins reposant respectivement sur deux critères de similarité et de voisinage. Ces deux critères (notés *SV*) spécifient que certaines paires d'éléments $\{x_i, x_j\}$ ne peuvent être fusionnés au sein d'un même groupe. Pour résoudre ce problème de partitionnement (ou de classification), on peut partir du point de vue inverse et formuler la question suivante, à savoir : « quel est le plus petit nombre de blocs homogènes que l'on peut former en respectant la contrainte *SV* ». L'intérêt de formuler le problème de cette manière, est qu'il est possible de la formuler en termes de coloration de graphe. Le positionnement du problème est alors le suivant : nous représentons chaque élément x_i par un sommet $v_i \in V$ d'un graphe simple G et nous ajoutons une arête $E(v_i,v_j)$ entre chaque paire d'éléments dissemblables (qui ne respectent pas la contrainte *SV*). La coloration des sommets du graphe $G(V,E)$ consiste alors à affecter à tous ses sommets une couleur de telle sorte que deux sommets adjacents (dissemblables) ne puissent pas porter la même couleur. Ces couleurs vont correspondre aux différents blocs homogènes qui constituent les différentes classes d'éléments. Dans ce problème de segmentation, la question de la détermination du plus petit nombre de blocs homogènes, revient à rechercher le plus petit k pour lequel le graphe G correspondant admet une *k-coloration* : c'est donc précisément le nombre chromatique $\chi(G)$ du graphe G qu'il faut déterminer.

Le même principe peut être appliqué à chaque niveau de la hiérarchie de décomposition de la structure d'un document séparant les caractères et les composantes non textuelles, puis créant les lignes de texte à partir des caractères, les blocs à partir des lignes et ainsi de suite.

Cette modélisation présente plusieurs avantages par rapport aux mécanismes de segmentation classiques :

- elle permet de s'affranchir des modèles classiques en ajoutant notamment à la coopération ascendant/descendant les propriétés intéressantes de la multi-résolution, de la hiérarchie de décomposition et des changements d'espace de représentation.

- elle permet de gérer facilement les ambiguïtés inhérentes aux éléments constitutifs de ce genre de document (la fusion de lignes de texte, la fusion de texte avec les graphiques et le bruit, les problèmes posé par l'inclinaison non uniforme des lignes de texte…

- elle permet d'exploiter conjointement les atouts des méthodes descendantes et des méthodes ascendantes de segmentation.

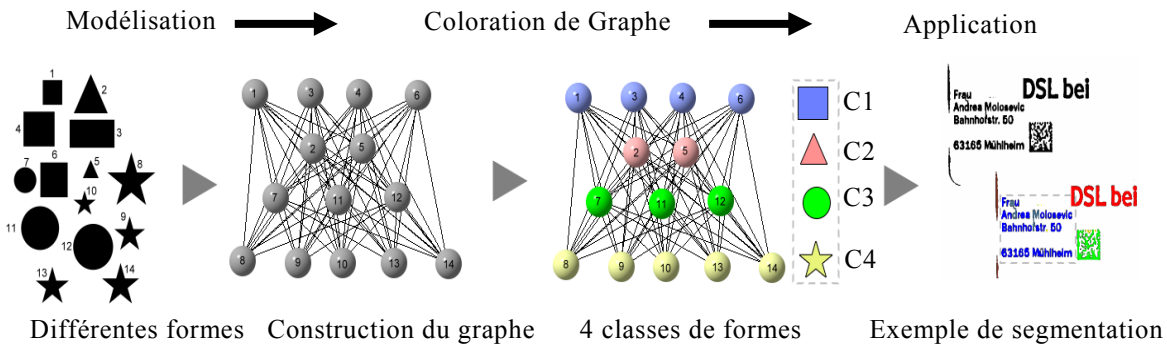


Figure 4. 7 : Exemple d'extraction de la structure physique d'un document type courrier d'entreprise par coloration.

Pour segmenter une image d'un document nous avons développé un nouvel algorithme de coloration minimale plus adapté. Il est possible de l'appliquer à plusieurs niveaux de la hiérarchie de la structure physique (niveau des CCs, niveau des lignes ou niveau des blocs). Notre algorithme est donné par la procédure suivante :

Procédure 7: Coloration_Minimale()

```
Début
Soit q=0
 Pour chaque sommet i∈V sachant que c(i)=∅ Faire
  q=q+1
  c(i)=q
  Pour chaque sommet j>i Sachant que
  j∉N(i)et c(j)=∅ Faire
    Soit q1=0 et q2=0
    Pour chaque sommet k<j Faire
     Sic(k)=q  Alors
      q1= q1+1
      Si k∉N(j) Alors q2=q2+1 FinSi
     FinSi
    FinPour
    Si q2=q1 ou q1=0 Alors c(j)=q
    FinSi
  FinPour
 FinPour
χ(G)=q
Fin.
```

Avec *c(i)* la couleur de sommet *i*, *N(i)* est l'ensemble de ses sommets adjacents au sommet *i*. La complexité de cet algorithme est inférieure à $O(n \times log(n))$.

Afin d'évaluer les temps de traitement, nous avons appliqué notre algorithme de coloration sur les cartes des composantes connexes (CCs) de *35* documents, et nous avons associé un sommet de *G* à chaque *CC*. Nous avons ensuite calculé le temps de la coloration des sommets de graphe *G* pour chaque type de document. La variation des temps de coloration en fonction de nombre de sommets est montrée sur la courbe suivante.

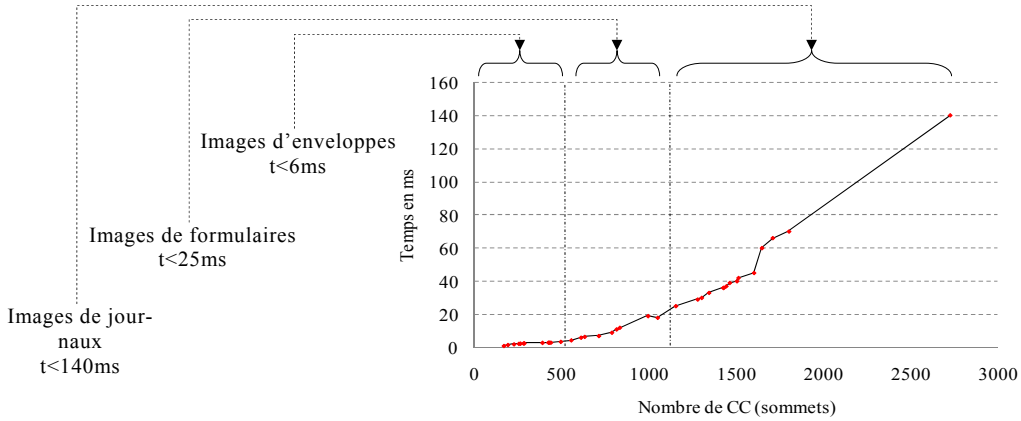


Figure 4. 8 : Temps de coloration en fonction du nombre de sommets : segmentation de la carte des composantes connexes (Processeur : Intel Pentium M 1,8 GHZ).

Nous remarquons que le temps moyen nécessaire pour colorer les CCs d'une enveloppe ne dépasse *6ms* et d'un formulaire ne dépasse *25ms*. Ceci montre que notre algorithme respecte parfaitement la contrainte de temps que nous impose l'application industrielle à laquelle se destine ce travail. Dans le chapitre suivant, nous présenterons dans le détail l'évaluation de cette approche tant au niveau des performances en terme de temps de calcul que de la précision dans les résultats de segmentation physique des documents.

4.4.2 Contribution de la b-coloration à la classification de documents et à la reconnaissance

Nous nous intéressons maintenant à la b-coloration de graphes afin de résoudre les problèmes de reconnaissance de documents industriels. Deux applications sont directement visées :

- la classification de documents (courriers en circulation dans l'entreprise) pour lesquels une typologie précise est définie.

- la localisation automatique de bloc adresse présent sur les documents courriers.

Ces deux applications, quoique très différentes, ont une caractéristique commune essentielle : elles ne peuvent pas, à l'inverse de la segmentation, être résolues directement à partir de connaissances existant a priori en raison de l'inexistence de modèles stables permettant de représenter efficacement l'organisation très hétérogène des contenus. En particulier, il n'existe pas de formalisme suffisamment précis pour prédire avec justesse la classe d'un document connaissant sa structure, de même, il n'existe pas d'équation qui décrive les propriétés topologiques des blocs sur une enveloppe. A ce stade il n'existe ni règles fiables ni propriétés stables permettant de modéliser convenablement les objets manipulés dans notre étude.

Pour parvenir à résoudre ce problème, nous sommes partis du constat que la coloration minimale à elle seule ne peut pas être un bon modèle pour la classification. En effet, elle ne permet pas de définir efficacement les représentants des classes qui sont indispensables à une reconnaissance avec des temps de réponse courts. C'est la raison pour laquelle nous avons formalisé les différents problèmes de reconnaissance induits dans notre application en termes de *b-coloration* de graphe. Cela permettra une bonne représentation des classes par les sommets dominants et une meilleure disparité inter-classes. L'apprentissage devient donc un élément central dans à cette partie.

Voici donc quelques éléments préliminaires de formalisation autour de la résolution des problèmes de classification de documents et de localisation du bloc adresse.

4.4.2.1 Formalisation du problème de classification des documents

Le processus de classification est appliqué à un corpus d'apprentissage V de n images de documents $d_i,...,d_n$. Le but est de regrouper les documents en classes homogènes. La nature des documents conduit à définir entre chaque paire de documents (d_i,d_j) une mesure de similarité qui traduit l'appartenance ou non des documents à la même classe. Deux questions inhérentes à la classification en découlent alors:

- quel est le nombre minimum de classes nécessaires pour regrouper les documents d'une manière sûre (en ignorant les contraintes de taille des diverses classes) ?
- quels sont les représentants des classes qui seront définis durant l'étape d'apprentissage et qui seront utilisés par la phase de reconnaissance ?

Nous reformulons ces deux questions centrales en termes de *b-coloration de graphe* et nous exposerons dans la section 5 les détails théoriques et d'implémentation de ces approches.

Définition 4. 8 : Soit $d_i \in \{1,...,n\}$ un document de la base d'apprentissage. Soit $I = \{1,..., k\}$ l'ensemble des couleurs du graphe.

Nous associons à chacun des n documents d_i un sommet v_i d'un graphe simple G, et à chaque paire (d_i, d_j) de documents qui ne peuvent être regroupés ensemble, nous faisons correspondre une arête (v_i , v_j) de ce graphe. Rappelons que cette arête exprime la dissimilarité entre deux sommets (donc pratiquement entre deux documents), notion qui sera définie en détail dans la section suivante.

Le problème se décompose en deux étapes :

1- La détermination du plus petit nombre de couleurs (classes de documents) c'est à dire du plus petit entier k qui permet d'attribuer à chaque sommet v_i (document d_i) une couleur (classe) $c(i)$ de l'ensemble I tout en ayant $c(i) \neq c(j)$ pour toute arête de G.

2- L'application d'une deuxième coloration permettant de faire ressortir les sommets dominants (les représentants des classes). Le principe étant que pour chaque couleur $c(i)$, il existe au moins un sommet v_i coloré $c(i)$ dont le voisinage est coloré par toutes les autres couleurs. Le sommet v_i est dit sommet *dominant* pour la couleur $c(i)$. On parle de b-coloration de graphe G si l'ensemble des couleurs I_b utilisables est de cardinalité k_b. Ceci représente les deux étapes de b-coloration des sommets du graphe G.

4.4.2.1 Formalisation du problème de localisation de bloc adresse

Nous avons vu dans le chapitre 3, section 3.3, que la localisation de l'adresse consiste à rechercher dans l'image de l'enveloppe des blocs de lignes d'écriture ou de caractères organisés en un ensemble présentant des caractéristiques topologiques proches et des relations spatiales spécifiques que l'on ne retrouve que dans l'écriture particulière des adresses postales. Après la formation de différents blocs candidats, le bloc adresse est sélectionné à l'aide de modèles obtenus par apprentissage statistique, i.e. à partir de prototypes de chacune des classes de blocs. L'apprentissage doit être effectué sur une base de n blocs $\{b_i, \ldots, b_n\}$ recueillis à partir de l'extraction de la structure physique des images d'enveloppes.

L'objectif consiste donc à résoudre un problème de classification où chaque bloc de la base doit appartenir à une classe parmi les k classes de blocs possibles (adresse, cachet de la poste, timbre, logos, ligne publicitaires, bruit). Sachant que le nombre de classes dépend de la nature des blocs présents sur l'enveloppe, il doit être considéré comme variable pour des enveloppes de nature différente.

En même temps, il est nécessaire de déterminer si l'adresse est imprimée ou manuscrite, et là encore il s'agit d'un problème de classification. Nous proposons formuler tous ces problèmes en terme de b-coloration de graphe. Pour cela, nous associons à chaque bloc b_i de la base d'apprentissage un sommet vi du graphe G. Sans prédéfinir le nombre de classes, on applique directement l'algorithme de b-coloration pour faire apparaître les différentes couleurs ou classes disponibles réellement. Cette stratégie, permet, d'isoler tous les blocs aberrants ou bruités dans des classes non significatives. Elle permet aussi, de distinguer en une fois non seulement les blocs graphiques des blocs textuels, mais aussi les blocs de texte publicitaire des blocs adresse. L'ensemble des blocs adresse est ainsi séparé en deux classes (classe des adresses imprimées et classe des adresses manuscrites). Cette partition automatique donne au système une grande souplesse pour s'adapter facilement au contenu de la base d'apprentissage. Les classes résultantes peuvent être validées et étiquetées par un superviseur.

4.5 Usage des graphes pour la segmentation par coloration et pour la classification par b-coloration

Le but de cette section est de présenter les étapes formalisées de la construction des graphes pour les applications (présentées dans la section précédente) de segmentation physique, de localisation de bloc adresse et de classification de documents.

4.5.1 Construction du graphe seuil de départ : notion de dissimilarité entre sommets et seuil d'adjacence

En général, la construction d'un graphe G à colorer ou à b-colorer à partir d'un ensemble $X=\{x_1,.. , x_n \}$ de n individus (qui selon le cas peuvent être soit des documents, des blocs, des lignes ou des CCs …), soumis à la segmentation ou à la classification, est principalement basé sur le calcul de la matrice de distances M_{Ds} qui traduit les dissimilarités existant entre les paires d'individus (i,j) donnée par la relation suivante. Nous faisons l'hypothèse que si x_i, x_j sont les descripteurs d'une paire d'individus (i,j), la distance $Ds(x_i, x_j)$ exprime la dissimilarité entre cette paire d'individus.

$$M_{Ds}[i,j]= Ds(x_i, x_j) \text{ avec } i \in [1,n] \text{et } j \in [1,n] / (i \neq j) \quad (4.6)$$

La mesure Ds peut être basée sur une métrique simple comme la distance euclidienne, la distance de Manhattan, la distance de Mahalanobis, la distance de Chebychev ou la distance binaire (de Hamming, de Jaccard ou de Tanimoto), ou bien elle peut être plus complexe et flexible utilisant la distance dynamique. Cette opération est importante dans la segmentation et la classification. Il est plus probable que deux vecteurs de caractéristiques semblables soient dans une même classe que deux vecteurs dissemblables. Les distances et les caractéristiques utilisées dans notre application sont décrites en détail lors de la présentation de notre nouvelle architecture de système de tri que nous présentons dans le chapitre suivant.

La construction d'un graphe repose alors sur cette matrice de dissimilarités M_{Ds} (X) : une fois les relations entre les individus de l'ensemble X décrites par la matrice M_{Ds} (X), nous associons à X un graphe seuil supérieur $G_{\geq S} =(V, E_{\geq S})$ décrit comme suit :

Définition 4. 9 : Un graphe *seuil supérieur*, pour un seuil S, est noté $G_{\geq S}$. Il représente une partie du graphe de départ $G= (V, E)$ liée à la contrainte de dissimilarité S. Il a pour sommets l'ensemble V de tous les éléments de X et pour arêtes l'ensemble $E_{\geq S}$ formant toutes les paires (x_i, x_j) dont la distance $Ds(x_i, x_j)$ est supérieure ou égale à S. Cet ensemble $E_{\geq S}$ peut être donné par la formule suivante :

$$E_{\geq S}\left[v_i, v_j\right] = \begin{cases} 1 & \text{si} \quad Ds(x_i, x_j) = Ds(v_i, v_j) \geq S \\ 0 & \text{sinon} \end{cases} \qquad (4.\ 7)$$

Pour ne pas confondre le terme *adjacence* (ou voisinage) avec le terme *similarité*, il faut noter que deux sommets sont adjacents s'ils ont un degré de *dissimilarité* supérieur au seuil *S*. Le seuil *S* est également nommé seuil d'adjacence.

Pour illustrer ce concept, nous proposons sur la figure suivante le diagramme sagittal du graphe seuil supérieur $G_{\geq 0.50}$ *(S = 0.50)* que l'on a associé à la matrice de dissimilarités M_{Ds}.

$M_{Ds}(X)=$

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	0.00	0.67	0.35	0.78	0.85	0.13
x_2		0.00	0.72	0.53	0.12	0.64
x_3			0.00	0.79	0.53	0.29
x_4				0.00	0.90	0.60
x_5					0.00	0.67
x_6						0.00

Matrice de dissimilarités

$M_a(G_{\geq 05})=$

	v_1	v_2	v_3	v_4	v_5	v_6
v_1	0	1	0	1	1	0
v_2		0	1	1	0	1
v_3			0	1	1	0
v_4				0	1	1
v_5					0	1
v_6						0

Matrice d'adjacences

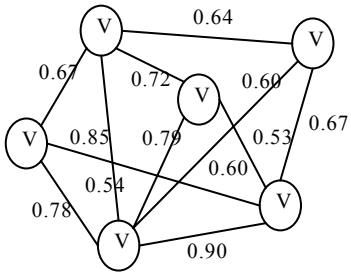
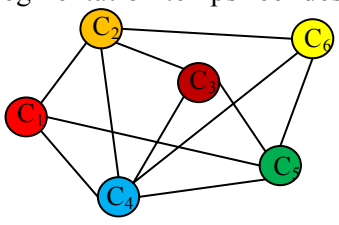


Diagramme sagittal de $G_{\geq 0.50}$

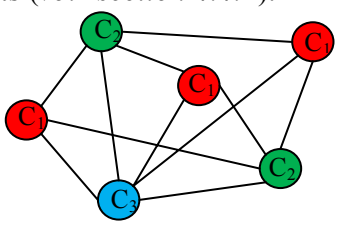
Figure 4. 9 : Construction d'un graphe seuil supérieur $G_{\geq 0.50}$ *(S = 0.50)* à partir de la matrice de dissimilarités M_{Ds}.

Une fois le graphe seuil construit, il est prêt à être envoyé à la phase de la coloration. La figure suivante montre deux exemples de coloration de graphe $G_{\geq 0.50}$. La figure de gauche montre le résultat de coloration par la *procédure 1 (voir section 4.3.2)* qui donne exactement *(Δ +1=6)* couleurs ; dans ce cas il est nécessaire de perfectionner le résultat par une deuxième coloration. La figure suivante droite montre un exemple de coloration obtenue par l'application de la *procédure 7* utilisée exclusivement pour la segmentation temps réel des documents (*voir section 4.4.1*).



(Δ +1=6)-Coloration

Coloration de $G_{\geq 0.50}$ par la procédure 1



3-Coloration

Coloration de $G_{\geq 0.50}$ par la procédure 7

Figure 4. 10 : Coloration de graphe seuil supérieur $G_{\geq 0.50}$ utilisant : **à gauche** la *procédure 1*, **à droite** notre algorithme de coloration, *procédure 7*.

Il est important de noter que le seuil *S* peut être un vecteur de plusieurs valeurs résumant l'ensemble des règles de séparation et correspondant à des valeurs seuils portant sur différentes caractéristiques décrivant les sommets *V*. Il peut être choisi de différentes manières selon l'application visée (figure 4.11).

- Dans le cas de l'extraction de la structure physique des images de documents, une des applications de notre étude, on doit lui attribuer manuellement une valeur résumant l'ensemble des connaissances a priori récoltées (i.e. la mise en forme des documents…).

- Dans le cas de la classification de documents on peut l'ajuster automatiquement et le valider par diverses approches d'évaluation de la qualité de classification. Nous présenterons notre approche dans la section suivante. Le meilleur seuil retenu correspond à celui qui permet d'assurer une qualité maximale de classification ψ qui peut être traduit par la formule suivante:

$$S^{Optimal} = \arg\max_{0 \le Si \le 1} (\psi(S_i)\} \qquad (4.8)$$

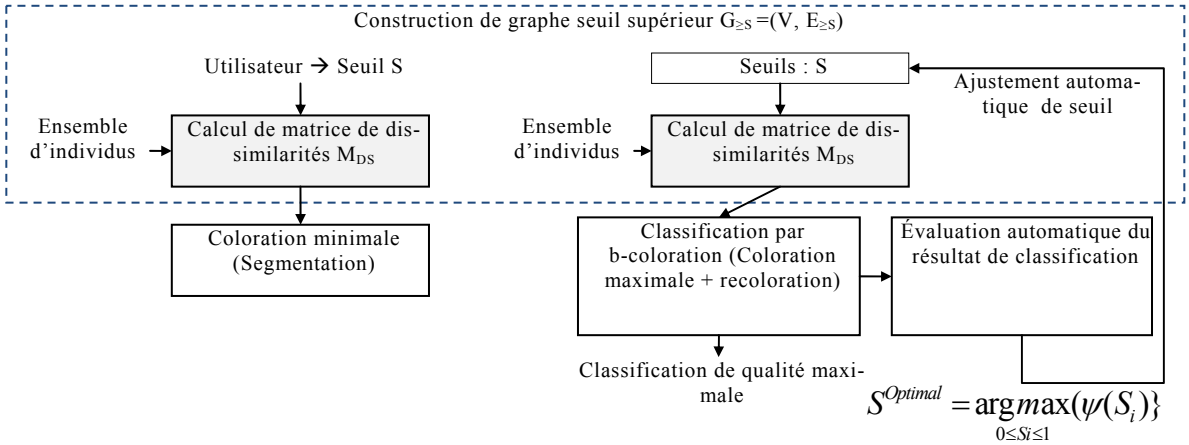


Figure 4. 11 : Construction de graphe et choix de seuil d'adjacence selon les deux applications visées (segmentation et classification).

4.5.2 Ajustement du seuil d'adjacence et évaluation de la qualité de la classification

Dans la construction du graphe, la notion de seuil d'adjacence *S* est centrale, car c'est sur lui que repose la pertinence de la coloration du graphe. Notre étude porte sur l'analyse de la validité de ce seuil à partir de l'évaluation du résultat de la classification. Nous venons de voir que chaque sommet dans *G* correspond à la description d'un individu dans *X*, et que les couleurs issues de la coloration de *G* correspondent aux différentes classes d'individus. Selon ce principe, la qualité d'une classification peut être donnée par la mesure de la qualité d'une coloration. L'évaluation de la qualité d'une coloration a un double objectif :

- elle sert d'outil de mesure à l'utilisateur afin d'ajuster au mieux le seuil d'adjacence qui dépend également d'un ensemble de paramètres, tels que la nature de l'ensemble des sommets à colorer (d'individus à classer), leur contenu et le but recherché.

- elle est nécessaire pour évaluer, tester ou comparer l'approche de classification proposée par rapport à d'autres approches préexistantes.

Afin de mieux répondre à ces deux objectifs, nous définissons deux types de critères quantitatifs d'évaluation, selon que l'on connaisse *a priori* ou non le nombre de classes : on parle alors d'évaluation supervisée et non supervisée.

Pour la présentation de ces différents critères, nous proposons d'expliciter quelques notations préliminaires.

Soit un graphe seuil supérieur $G_{\geq S}=(V=\{v1,...,vj\}, E_{\geq S})$ construit à partir d'un ensemble d'individus $X=\{x_1,...,x_j\}$. La coloration ou la b-coloration en k couleurs de l'ensemble de sommets V d'un graphe $G_{>S}$ donne un ensemble de couleurs (classes) $C=\{C_1,...,C_k\}$. On note $card(C_i)$ le nombre de sommets dans la couleur C_i qui correspond à l'aire de la classe C_i. On note n_i le nombre de sommets de G colorés par la couleur C_i et $n=|V|=|X|$ le nombre de sommets du graphe G. Enfin, $C(i)$ est définie comme la couleur du sommet i qui correspond à la classe de l'individu i.

4.5.2.1 Critère d'évaluation supervisée

Ce critère doit permettre de comparer le résultat d'une coloration (ou classification) C et la coloration (classification) de référence C_{ref} appelée vérité terrain (où on doit associer à chaque individu l'étiquette de sa classe désirée). Dans notre approche de classification par coloration de graphes, nous avons reformulé la mesure *Mg* proposée par Martin et al dans [MAR01] initialement mise au point pour valider la création d'une base de segmentations expertes sur des images naturelles. Cette mesure peut être donnée par la relation suivante :

$$\psi(s) = Mg\left(C(G_{\geq S}), C_{ref}(G_{\geq S})\right) = \frac{1}{n}\sum_{i=1}^{n} \min\left\{E_{RL}\left[c(i), c_{ref}(i)\right], E_{RL}\left[c_{ref}(i), c(i)\right]\right\} \quad (4.\ 9)$$

l'erreur de raffinement local E_{RL} étant définie comme suit :

$$E_{RL}\left[c(i), c_{ref}(i)\right] = \frac{card[L(c(v_i))] - card[L(c(v_i)) \cap L(c_{ref}(v_i))]}{card[L(c(v_i))]} \quad (4.\ 10)$$

C_{ref} désignant une classification (coloration) de référence, $L(c(v_i))$ l'ensemble des sommets de G qui ont la même couleur que le sommet v_i, $L(c_{ref}(v_i))$ l(ensemble des sommets de V (individus de X) qui ont la même couleur que le sommet i. $C_{ref}(i)$ est la couleur de référence de sommet i.

Le critère de qualité *Mg* sous sa forme finale tient compte des informations globales sur les sommets mal colorés ou confondus et permet de rendre compte classe par classe des erreurs de classification estimées par l'indicateur local E_{RL}.

4.5.2.2 Critère d'évaluation non supervisée

Ce critère d'évaluation ne requiert aucune connaissance sur les résultats de coloration (ou classification) à évaluer. Son principe consiste à estimer la qualité d'un résultat de classification à partir de statistiques calculées sur chaque classe formée. Cette mesure de qualité est établie en accord avec l'intuition humaine sur les conditions que devraient remplir une classification pour être considérée comme bonne. Pour un seuil donné, la qualité de classification ψ peut être calculée à partir de la combinaison de l'uniformité intra-classes avec la disparité inter-classes:

$$\psi(S_i) = M_{Inter\_Classes}(C(G_{\geq Si})) + M_{Intra\_Classes}(C(G_{\geq Si}) \quad (4.\ 11)$$

Ces deux critères sont inspirés des études de Levine et Nazif [LEV85] sur la segmentation automatique des images naturelles en régions. Ils vont servir de mesures pour ajuster automatiquement le seuil d'adjacence et évaluer la qualité la classification par b-coloration dans les applications de LBA et de RAD (voir le chapitre 5). Nous les présentons formellement dans les sections ci-dessous.

4.5.2.2.1 L'uniformité intra-classes

L'idée est ici de calculer l'uniformité d'une description d'un individu sur une classe en se basant sur la variance de cette description. La mesure d'uniformité $M_{Intra\text{-}Classes}$ issue d'une coloration C en K couleurs (classes) est la suivante:

$$M_{Intra-Classes}(C(G_{\geq S})) = 1 - \frac{1}{n}\sum_{i=1}^{K} \frac{\sum_{vj \in Ci}\left[v_j - \sum_{vl \in Ci} v_l\right]^2}{\left[\max_{vj \in Ci}(v_j) - \min_{vj \in Ci}(v_j)\right]^2} \quad (4.\ 12)$$

4.5.2.2.2 La disparité inter-interclasses

D'un point de vue complémentaire à l'uniformité intra-classes, un autre critère venant rapidement à l'esprit pour évaluer un résultat de classification est la disparité inter-classes. En effet, deux classes voisines sont supposées avoir un contenu différent. Il devrait donc y avoir une disparité décelable entre ces deux classes. La mesure de disparité *Disp* dans l'espace de caractéristiques entre deux classes (couleurs) C_1 et C_2 est donnée par la formule suivante :

$$Disp1\ (C_i, C_j) = \frac{\left| \frac{1}{n_i} \sum_{vx \in Ci} v_x - \frac{1}{n_j} \sum_{vx \in Cj} v_x \right|}{\frac{1}{n_i} \sum_{vj \in Ci} v_x + \frac{1}{n_j} \sum_{vx \in Cj} v_x} \quad (4.\ 13)$$

Dans le cas des classes non linéairement séparables il est plus judicieux de remplacer la disparité entre les barycentres des classes par la disparité entre tous les sommets des couleurs. La formule précédente devient donc :

$$\underset{\forall C_i, C_j \in C}{Disp1\ (C_i, C_j)} = \frac{1}{card(C_i) \cdot card(C_j)} \sum_{Vx \in Ci} \sum_{Vy \in Ci} Ds(v_x, v_y) \quad (4.\ 14)$$

où $Ds(v_x, v_y)$ représente une mesure de dissimilarité entre les deux sommets v_x, v_y. On peut ainsi définir la disparité d'une classe par rapport à toutes celles qui lui sont voisines par la formule suivante:

$$\underset{\forall C_i \in C}{Disp2\ (C_i)} = \sum_{Cj \in C | j \neq i} \frac{l_{ij} + 1}{l_i + 1} \cdot Disp1(C_i, C_j) \quad (4.\ 15)$$

Avec l_i le nombre de sommets non dominants dans la couleur C_i, l_{ij} le nombre de sommets de la couleur C_i qui n'on pas d'adjacence avec la couleur C_j. $(l_{ij} + 1) / (l_i + 1)$ correspond aussi au rapport longueur de la frontière commune entre la classe C_i et C_j sur le périmètre de C_i. La disparité globale est alors définie par la formule suivante :

$$M_{Inter-Classes}(C(G_{\geq S})) = \frac{\sum_{i=1}^{K} w_{ci} \cdot Disp2(C_i)}{\sum_{i=1}^{K} w_{ci}} \quad (4.\ 16)$$

où w_{ci} est un poids associé à chaque classe, qui peut être lié par exemple à l'aire de la classe. Dans ce cas, la manière dont le poids varie correspond à une gaussienne, soit :

$$w_{ci} = \frac{1}{\sqrt{2\pi\sigma}} \exp\left[- \frac{(card(C_i) - \mu)^2}{\sigma^2} \right] \quad (4.\ 17)$$

où μ et σ correspondent respectivement à la moyenne et à l'écart type de la courbe représentative du poids en fonction de l'aire des classes. Ce type de critère a l'avantage de pénaliser la sur-segmentation des classes.

Avant d'aborder dans le détail le principe de la reconnaissance par la b-coloration, nous soulignons que l'ensemble de ces définitions et notations seront reprises dans la partie d'évaluation de la contribution dans la partie suivante ainsi que dans le chapitre suivant.

4.6 Conception du système de reconnaissance par b-coloration: (de l'apprentissage à la reconnaissance)

Les aspects de reconnaissance interviennent essentiellement à deux niveaux de notre travail :

- pour la localisation du bloc adresse sur les images d'enveloppes
- pour la classification des documents selon des modèles de pages référentes

Nous présentons dans cette partie la conception complète de la partie apprentissage issue de l'adaptation de la b-coloration appliquée à ces deux cas.

Avant de présenter nos stratégies d'apprentissage, nous rappelons l'enchaînement des différentes étapes de conception d'un classifieur :

1) préparation d'une base d'exemples représentatifs,
2) description de chaque exemple par un ensemble de caractéristiques pertinentes, discriminantes et normalisées,
3) séparation des exemples en deux ensembles d'apprentissage et de test,
4) exécution de l'apprentissage par b-coloration sur la base d'apprentissage,
5) déduction de modèles à partir de sous-graphe qui contiennent l'ensemble des sommets dominants issus de la phase d'apprentissage,
6) test de modèle sur l'ensemble de tests,
7) évaluation des performances et validation de modèle,

Dans cette section, nous allons décrire les aspects formels de l'apprentissage à base de b-coloration ainsi qu'une approche de l'apprentissage incrémental gérant les flux entrants de documents inconnus permettant de créer à la « volée » de nouvelles classes.

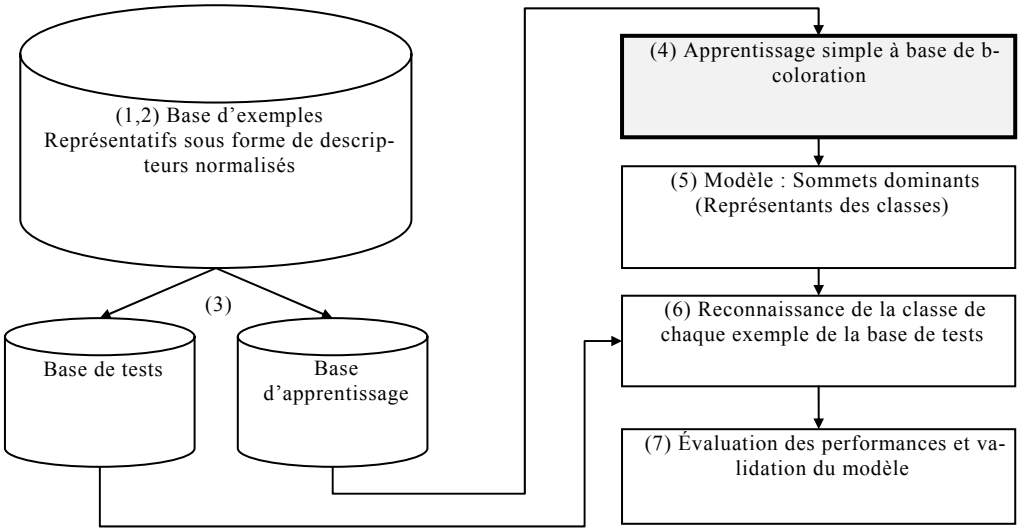


Figure 4. 12 : Enchaînement des différentes étapes de conception d'un classifieur.

4.6.1 Apprentissage simple à base de b-coloration

Nous nous intéressons dans cette partie à la phase d'apprentissage par b-coloration qui va nous permettre de résoudre les problèmes liés à la classification de documents et à la localisation du bloc adresse (il s'agit des étapes (4 et 5) sur le schéma ci-dessus).

Pour cela, on dispose d'un ensemble d'apprentissage constitué de n exemples (des documents pour un apprentissage de RAD ou des blocs pour un apprentissage de LBA). Dans le problème considéré ici, chaque exemple i est décrit par un vecteur de caractéristiques x_i. Notre principe de classification par b-coloration de graphes est itératif. Il consiste à faire évoluer de façon croissante la valeur du seuil de dissimilarités et à chaque itération t doit conduire à:

- la construction du graphe seuil supérieur $G_{\geq St}$.
- la classification des exemples de la base d'apprentissage par l'application de l'algorithme de la b-coloration décrit ci-dessus sur le graphe $G_{\geq St}$.
- une mesure de la qualité du résultat de la classification fournie par un des critères d'évaluation présentés ci-dessus.

À la fin des itérations, nous retenons la classification qui maximise l'un des critères d'évaluation (présentés dans la section 4.5.2) comme meilleure partition qui sera renvoyée à l'utilisateur. Les sommets dominants de cette b-coloration optimale constituent le modèle de reconnaissance. Nous pouvons utiliser ce concept en réalisant deux types de classification, supervisée ou non supervisée.

4.6.1.1 Classification supervisée

Cette classification peut être effectuée dans le cas où on possède suffisamment d'informations sur le nombre de classes k ou sur la réalité du terrain. Ceci nécessite l'affectation préalable de la réponse que devrait fournir le modèle à chaque exemple de la base d'apprentissage. Dans ce cadre la qualité de classification est mesurée à partir d'un critère d'évaluation supervisée (section 4.5.2.1). Soit $b(G_{\geq St})$ nombre b-chromatique de la b-coloration de graphe $G_{\geq St}$ à l'itération i. A chaque itération i on doit traiter les deux cas suivants :

- $b(G_{\geq St})=k$: cela signifie que toutes les couleurs doivent être prises en compte.

- $b(G_{\geq St})>k$: cela signifie qu'il y a $b(G_{\geq St})-k$ couleurs non pertinentes qui doivent être supprimées. Ces couleurs correspondent à quelques classes non denses (de faibles effectifs) qui contiennent souvent des exemples aberrants.

Ce genre de classification peut être appliqué à la reconnaissance automatique de type de documents lorsqu'on est limité à reconnaître un nombre fixe de catégories.

4.6.1.2 Classification non supervisée

On utilise une classification non supervisée lorsque :

- on ne dispose pas suffisamment d'informations sur les différentes classes en présence : cela résulte d'un manque d'informations ou d'une incertitude sur la réalité du terrain.

-ou le nombre exact de classes est en évolution, dans le cas d'une classification on line par exemple.

Dans ce cadre, la qualité de classification est mesurée automatiquement à partir d'un critère d'évaluation non-supervisée. Les couleurs résultantes de la b-coloration correspondent aux différentes classes ; l'interaction homme machine peut être allégée par cette classification non-supervisée. Ceci permet à un expert d'éviter l'étiquetage de tous les exemples de la base d'apprentissage en n'étiquetant que les sommets dominants. Ultérieurement, ceux-ci peuvent être étiquetés facilement par l'utilisateur afin que le modèle soit opérationnel lors de la phase de reconnaissance.

Ce genre de classification peut être utilisé efficacement dans le cadre de la localisation de bloc adresse où on peut ne pas savoir a priori la nature des blocs existants sur les enveloppes. Il peut aussi bien être utilisé dans la reconnaissance automatique de type de documents lorsqu'on ne dispose pas suffisamment d'information sur la nature des documents qu'on veut trier.

La réalisation d'un processus décisionnel avec des temps de réponse courts est capitale dans ces applications ayant des contraintes de temps réel élevées. De plus, les bases d'apprentissage servant à la réalisation de ces processus contiennent généralement des exemples redondants, non représentatifs, plus ou moins bruités.

4.6.2 Apprentissage incrémental par b-coloration

Nous avons expliqué comment obtenir une classification optimale par l'application de l'algorithme de b-coloration sur une base d'apprentissage restreinte. Il est toujours possible de réaménager cette base à l'aide de résultats de la b-coloration en supprimant les mauvais exemples isolés dans des classes non significatives. Dans le cadre de notre application de RAD (ou de LBA), il est très important de profiter du flux de documents entrant (ou de blocs d'enveloppes entrant), pour enrichir la base d'apprentissage. Ceci permet d'intégrer dans cette base des nouveaux documents qui sont reconnus ou rejetés et d'élargir les connaissances du système de lecture pour reconnaître des nouvelles classes. Ceci est d'autant plus important que nous avons remarqué que les taux de rejet et d'erreur baissaient sensiblement lorsque le nombre de prototypes par classe augmentait. Cela paraît naturel car plus la base d'apprentissage est grande et plus les intervalles vont bien modéliser les données, ce qui a pour conséquence de faire baisser les taux d'erreur et de rejet. Ce phénomène est observable pour la plupart des classifieurs. Cependant, il est tout aussi important de savoir arrêter l'intégration des connaissances liées au flux entrant dès que les taux de reconnaissance se stabilisent ou que le système commence à présenter des symptômes de sur-apprentissage.

Le problème à présent est de savoir comment gérer un flux séquentiel de données entrant de sorte à affecter une classe aux nouvelles données et à effectuer ainsi la mise à jour de la partition optimale obtenue. Pour une meilleure modélisation de ce problème, il nous semble opportun d'utiliser un algorithme d'apprentissage incrémental qui réponde aux critères suivants :

1) il doit être capable d'instruire des connaissances additionnelles à partir des nouvelles données,

2) il ne doit pas requérir l'accès aux données initiales (c'est-à-dire les données qui ont été utilisées pour apprendre le classifieur actuel),

3) il doit préserver les connaissances déjà acquises,

4) il doit être en mesure d'apprendre de nouvelles connaissances relatives à l'introduction de nouvelles classes.

Ces quatre points évoqués par Polikar et al. [POL01], pour résoudre le problème général de l'apprentissage incrémental, répondent pleinement à nos objectifs de conception d'un algorithme d'apprentissage rapide pour la RAD ou la LBA en temps réel. Dans la suite de cette partie nous proposons deux algorithmes d'apprentissage incrémental, actif et adaptatif qui s'appuient sur le principe de base de la b-coloration de graphe. Ces deux algorithmes profitent d'une part de la connaissance globale des adjacences entres les sommets de la base d'apprentissage actuelle et, d'autre part, ils tirent parti de la propriété de dominance de la b-coloration. De plus, ils respectent parfaitement les quatre critères évoqués ci-dessus.

Considérons à présent une base d'apprentissage déjà construite X^- de n exemples représentée par le graphe $G_{\geq S}=(V^-, E^-_{\geq S})$ et partitionnée par l'algorithme de b-coloration en k classes (couleurs) $C^-=\{C_1,...,C_k\}$. L'insertion d'un $(n+1)^{ème}$ exemple dans X^- correspond à l'ajout d'un sommet v_{n+1} au graphe $G^-_{\geq S}$ avec $V^+ = V^- \cup \{v_{n+1}\}$. Cette insertion consiste à ajouter des arrêtes entre ce sommet et les sommets dont la dissimilarité avec v_{n+1} est supérieure ou égale au seuil S.

Pour notre application de tri, le système d'apprentissage incrémental s'alimente selon deux manières à partir de données (documents ou enveloppes) :

- en considérant la nouvelle donnée comme étant bien reconnue et directement attribuable à une classe existante ;

- en considérant la nouvelle donnée en situation de *rejet* par le système.

Premier cas : on introduit un nouvel exemple v_{n+1} qui était bien reconnu dont la classe est $C_i \in C^-$. Dans ce cas, il faut juste mettre à jours le voisinage de tous les sommets et vérifier s'il y a des nouveaux sommets qui deviennent à leur tour dominants. Le déroulement de la mise à jour de l'apprentissage est donné par l'algorithme suivant :

Procédure 8: Insertion_sommet_reconnu(v_{n+1} ,C_i,C, $G_{\geq S}$)

```
Début
V+=V- ∪{v_{n+1}}
Pour chaque couleur C_j ∈C sachant que j≠i Faire
   Pour chaque sommet v_u∈ C_j  Faire
      Si  d(v_{n+1},v_u)≥S alors
           N(v_u)=N(v_u)  ∪  { v_{n+1}}
          N(v_{n+1})=N(v_{n+1})  ∪  { v_u  }
          Si Nc(v_u)∩C_i =∅ Alors Nc(v_u)=Nc(v_u) ∪ {C_i}
          FinSi
          Si Nc(v_{n+1})∩ C_j=∅ Alors
             Nc(v_{n+1})= Nc(v_{n+1})  ∪  { C_j  } FinSi
          Si v_u ∉ ESD(C_j ) et |Nc(v_u)| = k-1  Alors
             ESD(C_j) = ESD(C_j) ∪ {v_u} FinSi
     FinSi
   FinPour
Si v_{n+1} ∉ ESD(C_i) et |Nc(v_{n+1})| = k-1  Alors
   ESD(C_i) = ESD(C_i) ∪ {v_{n+1}} FinSi
n=n+1;
Fin
```

Avec $ESD(C_i)$ l'ensemble de sommets dominants de la couleur C_i.

Deuxième cas : on introduit un nouvel exemple v_{n+1} qui a été initialement rejeté par le système de reconnaissance. Dans ce cas, on est sur que le sommet v_{n+1} ne peut pas avoir une couleur de C^-. S'il existe déjà de nouvelles couleurs crées par la procédure d'apprentissage incrémental $(C^N=\{C^+-C^-\}$ *et* $C^N \neq \varnothing)$ et si la distance *dist* entre le sommet v_{n+1} et la couleur la plus proche $C_t \in C^N$ $(t=k+1,\ldots,k')$ est inférieure au seuil S, ce sommet est alors coloré par la couleur C_t. Une distance entre un sommet et une couleur peut être définie par :

$$dist(v_{n+1}, C_t) = \min_{\forall vj \in C_t} \{d(v_{n+1}, vj)\} \qquad (4.18)$$

Sinon, on affecte une nouvelle couleur C_{k+1} au sommet v_{n+1}. Dans tous les cas on doit mettre à jours la liste des sommets adjacents à chaque sommet et la liste des sommets dominants pour chaque couleur. L'algorithme suivant résume avec plus de détails tous ces cas :

Procédure 9: Insertion_sommet_rejeté(vn+1 , G≥S)

```
Début
  C^N={C^+-C^-}
  Soit dmin=∞
  Soit q=k+1
  Pour chaque couleur C_t∈ C^N avec t=k+1,…,k' Faire
    Si dist (v_{n+1}, C_t)<dmin   Alors
```

```
q=Ct;  dmin=dist(vn+1,Ct)
    FinSi ;
  FinPour
  Si C^N≠∅ et dmin≤S  alors
     Insertion_sommet_reconnu(vn+1, q, C+, G≥S)
  FinSi
  Sinon k'=k'+1 ; C+=C-∪{ Ck'};
     Insertion_sommet_reconnu(vn+1, Ck+1, C+, G≥S)
  FinSinon
Fin
```

Dès qu'une nouvelle classe crée devient importante (selon un critère fixer par le superviseur : comme le volume par exemple), elle est proposée à un superviseur de système de tri pour la valider et lui donner une étiquette en fonction de ses exemples constitutifs ou pour décider sa suppression. Après l'insertion d'une série d'exemples venant du flux de données entrant, le nouveau modèle de l'apprentissage incrémental est évalué.

4.6.3 Approche de la reconnaissance d'un exemple inconnu

L'apprentissage par b-coloration donne en sortie un ensemble de sommets dominants qui correspondent aux représentants des classes. La reconnaissance d'un exemple inconnu peut être effectuée selon trois scénarios :

Scénario 1 : *Distance minimale entre classes*

Toute classe doit être représentée par le sommet le plus dominant (le sommet le plus éloigné des autres couleurs ou le sommet qui a le plus grand nombre de sommets adjacents). Pour affecter un objet inconnu à une des classes on utilise un classificateur à distance minimum qui a la structure suivante:

Soit $k=b(G)$ classes issues de la b-coloration de G, avec $C=\{C_1,...,C_k\}$ l'ensembles de ces classes. Chacune des classes est représentée par le sommet le plus dominant, avec $V^* = \{v_1^*,...,v_k^*\}$ l'ensemble des sommets représentants des classes de l'ensemble C. Étant donné un objet inconnu x qui correspond aux sommets v_x, on affecte v_x à la classe C_i en appliquant la règle suivante :

$$\begin{cases} v_x \in C_i \quad si \quad \left(Ds(v_x, v_i^*) = \min_{j=1...k} \left[Ds(v_x, v_j^*) \right] \right) \leq S \\ v_x \in \text{ rejet de distance sinon} \end{cases} \quad (4.\ 19)$$

Où Ds est une distance entre deux sommets définie dans l'espace des caractéristiques et S est le seuil d'adjacence utilisé dans la phase d'apprentissage. Dans le cas où plusieurs sommets dominants ont une même distance par rapport à v_x, on applique un rejet de confusion sur le sommet v_x. Il est possible d'appliquer le même principe sur tous les som-

mets dominants sans avoir besoin de sélectionner le sommet le plus dominant pour chaque couleur. Mais ce principe risque d'être coûteux en temps de reconnaissance dès que le nombre de sommets dominants devient important.

Scénario 2 : *Approche barycentrique*

Toute classe C_i doit être représentée par le barycentre de ses N_i^* sommets dominants donné par la formule suivante :

$$\overline{v}_i^* = \frac{1}{N_i^*} \sum_{v_j^* \in C_i} v_j^* \qquad (4.\ 20)$$

Où N_i^* est le nombre de sommets dominants dans la classe C_i

Scénario 3 : *Choix d'une fonction de densité de voisinage*

Pour ce scénario, nous avons choisi de mettre en œuvre une approche non paramétrique de classification : la méthode des k-ppv. A la différence des approches par fenêtre de Parzen, qui nécessite de fixer la taille du volume de recherche des voisins puis de calculer le nombre d'échantillons dans le volume fixe, nous avons opté pour l'approche par K-plus proches voisins qui inversement fixe le nombre d'éléments du voisinage et fait varier la taille du volume de recherche. Ce choix a été guidé par le constat que les sommets dominants voisins au sommet à affecter peuvent être de distances très variables, l'approche par fenêtre de Parzen risque de conduire à des décisions de rejet plus fréquentes, comme cela est le cas pour le *scénario 1*.

Ainsi, au lieu d'utiliser le barycentre des dominants ou le sommet le plus dominant comme unique prototype d'une classe, la méthode du plus proche voisin fait intervenir les k_d sommets les plus dominants de chaque classe. Cette approche améliore le taux de reconnaissance et réduit l'erreur de confusion. Pour ce faire, la distance entre chacun des sommets dominants et celle de l'objet à classifier est calculée. La classe assignée à l'objet est alors celle du prototype le plus proche de celui-ci. Si le nombre de sommets dominants N_i^* dans une classe i est supérieur à k_d , alors la contribution de chacun des sommets de cette classe peut être pondérée par le poids $\rho = (k_d)^{-1}$, sinon cette contribution est donnée par le poids $\rho = (N_i^*)^{-1}$. Ceci garantit une forme de normalisation des distances entre sommets. Soit $V^* = \{v_1^*, ..., v_{N^*}^*\}$ l'ensemble de sommets dominants provenant de k classes $C=\{C_1, ..., C_k\}$ formées dans la phase d'apprentissage par b-coloration. Étant donné un objet inconnu représenté par le sommet v_x qui est indépendant de V^*, on recherche dans V^* les kp plus proches voisins (Kp-PPV) v_{kpj} de v_x en écrivant que :

$$Ds(v_x, v_{kpj}) < Ds(v_x, v_t) \leq S \quad \text{avec} \quad t \neq \{Kp_j,\ j = 1, ..., kp\} \qquad (4.\ 21)$$

On affecte ensuite v_x à la classe $C_{i=1,\ldots,k}$ en appliquant la relation suivante :

$$\begin{array}{l} v_x \in C_i \quad si \quad \mathrm{NC_i}=\mathrm{Max}\left[NC_{i'}\right] \text{ et } \mathrm{NC_i} \neq 0 \\ v_x \in \text{rejet de distance} \quad \text{sinon} \end{array} \qquad (4.\ 22)$$

$$NC_{i'} = \begin{cases} \sum_{v_{kpi} \in Ci'}^{kpi} \frac{1}{k_d} & si\ N^*_{i'} \geq k_d \\ \sum_{v_{kpi} \in Ci'}^{kpi} \frac{1}{N^*_{i'}} & \text{sinon} \end{cases} \qquad (4.\ 23)$$

En utilisant le classificateur des K plus proches voisins (des sommets dominants), le sommet v_x est affecté à la classe majoritaire dans l'ensemble des Kp-PPV de cette observation. Notons que plus k_p est grand, plus l'erreur de classification est petite mais plus le temps de classement devient important.

Ce classifieur permet de prendre une décision précise dans le cas d'ambiguïté (lorsque l'exemple à reconnaître se trouve dans la même distance entre plusieurs sommets dominants de couleurs différentes). Ce cas d'ambiguïté conduit le classifieur à distance minimale à un rejet de confusion.

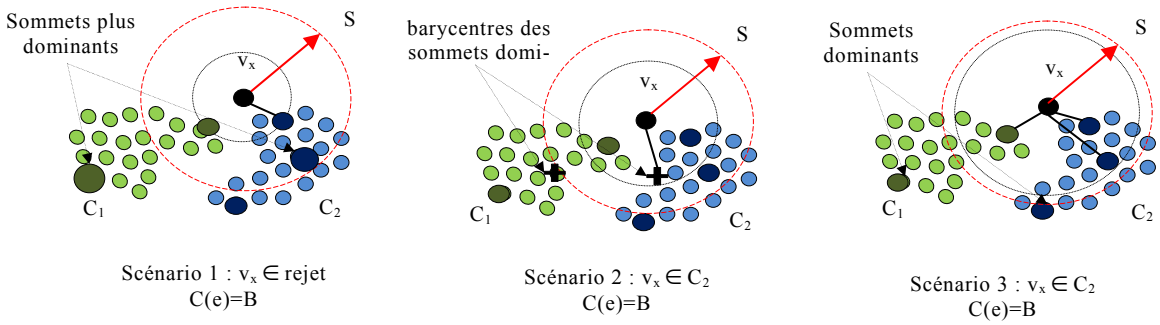


Figure 4. 13 : Exemple de différents scénarios de reconnaissance.

4.7. Conclusion

Nous avons présenté au début de ce chapitre les aspects théoriques de la coloration et de la b-coloration de graphe dans des contextes applicatifs très généraux. Nous avons sélectionné les algorithmes qui s'adaptent mieux aux exigences de notre application de tri automatique de documents et de courriers. Dans ce cadre nous avons proposé quelques algorithmes de coloration, d'apprentissage et plusieurs scénarios mettant en évidence, pour la première fois, la contribution de la coloration de graphe à la modélisa-

tion et à la résolution des problèmes de segmentation, d'apprentissage et de classification.

Dans un premier temps, nous avons montré que la tâche d'extraction de la structure physique des images de documents pouvait être formalisée à l'aide d'une coloration minimale de graphe qui procède au regroupement des éléments constitutifs du document (en fonction de critères d'homogénéité locale et de voisinage). Par ailleurs, la connaissance globale issue de l'analyse des adjacences entre sommets du graphe permet de prendre les décisions de séparation (partitionnement des sommets) dans les cas où la connaissance locale n'est pas suffisante. C'est donc la structure même du graphe qui renseigne sur ces connaissances globales et permet de séparer les données insuffisamment proches. Regroupement et partitionnement peuvent ainsi être considérés comme deux termes clés du processus de coloration.

Notre objectif ici est également de montrer que la coloration de graphe peut établir un cadre théorique unique pour la mise en œuvre d'un ensemble d'applications clés dans tout système de tri automatique de document. Pour cela, nous avons formulé notre problème d'extraction de la structure physique des images de documents en terme de coloration de graphe et notre problème d'apprentissage pour la RAD ou LBA en terme de b-coloration de graphe. Nous avons décrit les étapes essentielles de coloration minimale pour la segmentation hiérarchique bas niveau et de la b-coloration pour traiter les questions d'apprentissage artificiel, de classification automatique et de reconnaissance. Nous avons également vu comment la b-coloration de graphe pouvait permettre la recherche automatique du nombre de classes (à partir de la recherche des sommets dominants) et assurer tout à la fois une précision de séparation interclasse et une forte homogénéité intraclasse. De plus la b-coloration est un modèle idéal de représentation des classes par les sommets dominants.

L'extraction de la structure physique des images de courriers d'entreprise se réalise sans connaissance a priori du nombre de classes de textes (ou de graphiques). Le principe de coloration se charge de produire un nombre de couleurs proche de l'optimal. Pour la LBA et la RAD, compte tenu du nombre restreint et connu a priori de catégories de documents mis en circulation, nous avons opté pour une classification supervisée avec la connaissance a priori du nombre de classes.

Le processus de coloration de graphe peut être résolu de deux façons :

- soit par la recherche du nombre chromatique optimal minimial qui est excessivement gourmande en temps de calculs et en pratique irréalisable dans un cadre applicatif industriel

- soit par l'élaboration d'une stratégie heuristique qui se rapproche de la valeur optimale mais qui se fonde sur des indicateurs de convergence portés par des valeurs de seuils (que nous avons rassemblés sous le terme de dissimilarité dans cette thèse). Il est important de noter que la seule notion de dissimilarité revêt différents aspects : elle correspond à plusieurs valeurs résumant un ensemble des règles de séparation et correspondant à des valeurs seuils portant sur différentes caractéristiques décrivant les sommets. Elle peut ainsi être choisie de différentes manières selon l'application visée.

Nous avons choisi cette deuxième approche car elle est compatible avec une approche temps réel. Dans le cas de l'extraction de la structure physique des images de documents, le seuil de séparation de sommets résumant l'ensemble des connaissances a priori récoltées a été choisi manuellement. Dans le cas de la classification de documents ce seuil est ajusté automatiquement et est validé par plusieurs approches d'évaluation de la qualité de classification.

Finalement, nous avons proposé un nouveau concept d'apprentissage incrémental, élément essentiel pour mettre à jour la base d'apprentissage à partir de flux de documents et de courriers entrant. Cette propriété importante facilite l'adaptation du système de reconnaissance pour reconnaître de nouvelles catégories de documents et simplifier la tâche d'interaction d'un expert avec ce système.

Après avoir présenté tous ces éléments formels, nous présentons dans le chapitre suivant une proposition innovante de segmentation et de reconnaissance basée sur une nouvelle architecture pyramidale conçue à partir d'une coloration hiérarchique et d'une b-coloration pour l'apprentissage dont les fondements théoriques ont été explicités dans ce chapitre.

Chapitre 5

Proposition d'une nouvelle architecture pyramidale

À base de coloration de graphes

5.1 Introduction

Les architectures des systèmes de tri de courriers d'entreprise dont nous avons balayé les spécificités dans le premier chapitre de cette thèse présentent des faiblesses qui se traduisent par des taux d'erreurs de lecture et de rejet que l'on impute encore trop souvent aux OCR. Or, comme nous l'avons souligné, les étapes clés responsables des rejets et des erreurs de lecture sont les étapes fondamentales de segmentation et de localisation de zones d'intérêts. Ces deux étapes qui s'impliquent mutuellement jouent un grand rôle dans les performances des systèmes. Elles ont notamment une très grande influence sur le rendement d'une chaîne de tri automatique de documents et de courriers d'entreprises, en terme de vitesse de traitement et de taux de rejet.

Nous avons choisi de traiter le problème du tri de courrier en impliquant la coloration de graphes à toutes les étapes d'analyse de la structure des documents ainsi que dans la prise de décision pour la reconnaissance (reconnaissance de la nature du document à traiter et reconnaissance du bloc adresse). La partie de reconnaissance a été conçue autour d'un apprentissage traité à l'aide d'un modèle unique portant sur la b-coloration de graphe.

Les algorithmes impliqués dans le système ont été conçus pour leur rapidité d'exécution[5] (en adéquation avec les contraintes de temps réels), leur robustesse, et leur compatibilité.

Voici en détail la description du modèle retenu ainsi que les différents algorithmes développés pour résoudre les problèmes de segmentation et de reconnaissance des contenus

5.2 Description fonctionnelle du modèle pyramidal

Le schéma de la figure 5.1 donne un aperçu de l'ensemble des contributions que nous allons présenter dans ce chapitre. Notre architecture est conçue pour réaliser les deux étapes clés (décrite ci-dessous), en garantissant une réelle coopération entres les différents modules d'analyse et de décision. Elle s'articule autour de trois grandes parties (voir figure 5.1) :

- une partie de segmentation bas niveau (binarisation et recherche de connexités)
- une partie d'extraction de la structure physique par coloration hiérarchique de graphe

5. Les temps sont obtenus sur une machine Intel Pentium M Processeur 1.86 GHZ, RAM 1Go.

- une partie de localisation de blocs adresse et de classification de documents

En premier lieu, rappelons que la phase de segmentation en blocs des images de courriers (et plus généralement des images de documents d'entreprises au sens large) repose sur des modules de binarisation, de détection des composantes connexes et de séparation texte non texte (partie 5.1). L'ensemble de ces étapes participe à l'analyse de la structure physique permettant de faire apparaître les composantes élémentaires du texte : les mots, les lignes puis les blocs de texte (partie 5.2). A ce stade, une coopération efficace entre des informations de bas niveau (descripteurs de bas niveau des images) et la reconnaissance que l'on peut en faire doit permettre de produire des résultats de segmentation très nettement améliorés, par rapport aux approches linéaires typiquement ascendantes ou descendantes portant exclusivement sur des données brutes de bas niveau (partie 5.3). Dans ce type de configurations, il est important de rappeler que les temps de traitement, les taux de rejet et d'erreurs sont d'autant plus élevés que l'indépendance des processus engagés dans la reconnaissance est grande.

C'est là tout le sens que nous avons voulu donner à notre approche de la segmentation et qui se traduit sur le schéma de la figure 5.1 par des doubles fléchages systématiques entre de nombreux modules de traitements et d'analyse, notamment entre les modules d'« extraction de la structure physique » et d'« extraction hiérarchique de caractéristiques ». En particulier, nous avons voulu montrer que la localisation des zones d'intérêts des courriers d'entreprise (partie 5.3) était une étape de reconnaissance qui était indissociable d'une étape de caractérisation des contenus car elles exercent l'une sur l'autre une influence sur les décisions prises et les valeurs des primitives à calculer. Choisir de traiter ce problème par coloration de graphe nous a permis de produire une vision objective des contenus à différents niveaux de représentation (des connexités aux blocs informants). La coloration de graphe présente en effet de nombreux avantages pour la segmentation des images : elle permet notamment de mener une approche mixte en proposant une distribution des nœuds du graphe selon des critères de dissimilarité locale calculés entre chaque élément de contenu.

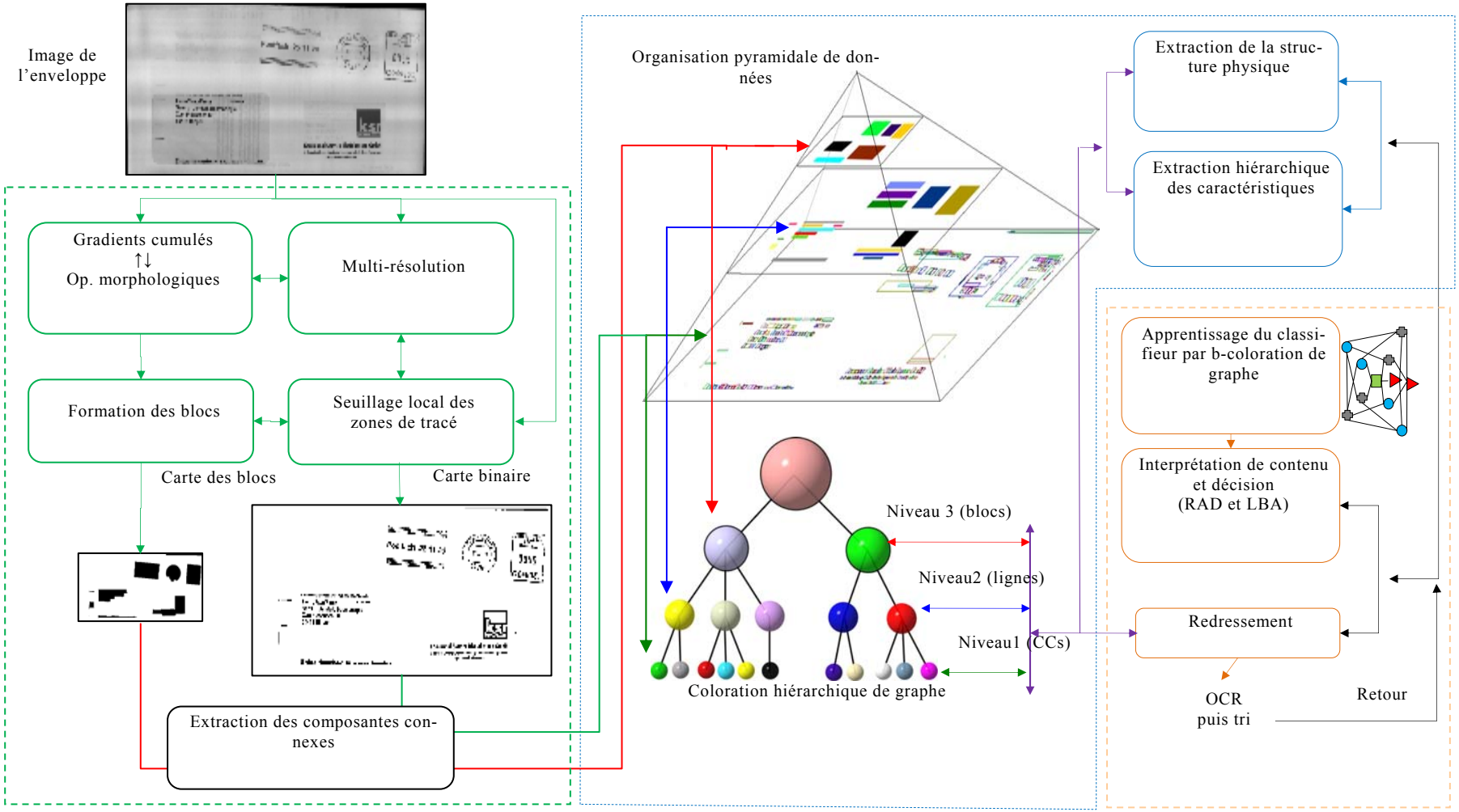


Figure 5. 1 : Schéma synoptique de la nouvelle architecture proposée en trois grandes étapes : segmentation – analyse de la structure physique – reconnaissance.

Elle donne ainsi une vision globale sur la structure du document en dirigeant un regroupement ascendant de ces éléments à partir des résultats de coloration (chapitre 2, partie 2.4).

Dans notre domaine d'étude, la localisation des zones d'intérêts va revêtir deux formes remarquables :

- d'une part nous allons nous intéresser aux documents de structure stable présentant des zones d'intérêt toujours situées aux mêmes endroits mais de complexité induite variable par le rajout d'annotations manuscrites de diverses natures. Dans ce premier cas d'étude, nous appliquerons notre contribution à la Reconnaissance automatique du type de documents liée à l'existence de connaissances a priori (connaissance de la position des zones d'intérêt, connaissance des algorithmes d'OCR à engager et des dictionnaires de mots à exploiter pour accomplir le tri)

- d'autre part nous allons nous intéresser aux documents de structures instables (courrier d'entreprises) parmi lesquels on trouvera les pages contenant des zones d'adresse se trouvant dans plusieurs positions selon des orientations variables également. Dans ce second cas, nous appliquerons notre contribution à la localisation automatique du bloc adresse sur les images de courrier.

5.2.1 Les étapes d'analyse exclusivement bas niveau

En premier lieu, nous présentons dans ce chapitre l'étape de binarisation préalable que toute image considérée par le système doit subir. En pratique, les images proviennent d'une acquisition par caméra et la binarisation qui porte sur elle procède en deux temps afin de réduire considérablement le nombre d'itérations généralement employées dans une approche de binarisation conventionnelle (globale ou adaptative). Elle débute par la localisation des zones de traits à partir de l'estimation des gradients cumulés et d'opérations morphologiques à basse résolution. Cette étape est suivie d'un seuillage local type Sauvola portant exclusivement sur ces zones à fort gradients dans la version des images à pleine résolution. Ce processus de binarisation peut être assimilé à une première étape de segmentation bas niveau brute, résultat des cartes binaires issues des images de gradients seuillés selon le critère de Fisher. Une première coopération entre l'étape de repérage des zones d'intérêt et l'étape de seuillage dirigée est ainsi proposée dans cette binarisation. Les temps de calculs sont globalement réduits par l'exploitation des gradients qui épargne l'analyse exhaustive des régions intégralement remplies (grandes plages noires très souvent visibles sur les logos apposés sur les enveloppes de courriers). L'étape de détection des composantes connexes se trouve ainsi considérablement accélérée.

Nous présentons ensuite l'étape d'extraction de composantes connexes utilisant conjointement le résultat de la carte binaire et le résultat de localisation des zones de traits.

La localisation des zones d'intérêt est ensuite réalisée sur chaque image : elle doit être vue comme un découpage brut non fonctionnel des régions textuelles des documents. Elle est issue de l'analyse de la carte des forts gradients et des résultats de la carte binaire à basse résolution. Le masque des blocs est considéré ici comme une première segmentation sans étiquetage formel qui donne uniquement une information sur l'existence des blocs bruts.

A ce stade la segmentation que l'on pourrait qualifier de « brute » et l'extraction de caractéristiques peuvent se mener conjointement.

A partir de ce schéma d'analyse exclusivement bas niveau et qui ne porte que sur des indices de niveaux de gris des images, nous avons engagé une approche complète de reconnaissance portant sur la coloration de graphe.

5.2.2 L'analyse de la structure physique par colorations hiérarchiques de graphe

Pour parvenir à une segmentation complète des images en différentes couches d'information (couches des connexités, des lignes, des blocs), nous avons choisi d'exploiter une coloration hiérarchique à trois niveaux

- Une première coloration de graphe est mise en œuvre pour séparer les composantes connexes en composantes de texte/ non texte. On utilise ici la position des blocs (issus du masque binaire) pour accélérer la coloration.

- Une deuxième coloration est ensuite mise au point : elle s'applique exclusivement sur les composantes des couleurs qui représentent le texte pour faire apparaître les lignes. L'estimation de l'inclinaison est intégrée à cette étape pour augmenter la précision de détection des lignes inclinées. Cette inclinaison est finalement utilisée pour redresser les lignes des zones d'intérêt.

- Une troisième coloration est finalement effectuée à partir des résultats de la deuxième coloration pour faire apparaître les blocs de texte homogène qui pourront finalement être reconnu à partir de l'apprentissage par b-coloration réalisé pour cela. Les applications RAD ou LBA exploite précisément ce processus. Elles seront intégralement explicitées dans ce chapitre.

En dernier lieu, il est important de noter que le système peut à ce stade exploiter les informations résiduelles issues de la différence entre les

blocs brutes (extraits de l'analyse bas niveau du masque binaire) et les blocs de texte extraits à partir des étapes de colorations. L'analyse des ces informations permet de récupérer les blocs non textuels dans une forme compacte où on retrouve des composantes hétérogènes fusionnées. Ces blocs non textuels sont ensuite utilisés pour effectuer une analyse de topologie des données non textuelles (logos, timbres et tampons) en fonction de l'adjacence des éléments et permettent ainsi de renforcer la prise de décision.

La segmentation complète des images par coloration hiérarchique de graphe (des connexités aux blocs de texte) repose sur une alternance « segmentation-caractérisation » menée de telle sorte qu'un niveau de la hiérarchie alimente le niveau suivant. Les colorations successives portent ainsi sur des nœuds représentant des entités différentes à chaque niveau. Au plus bas niveau ces entités représentent les connexités et, au niveau le plus haut, les blocs de texte. Les graphes successifs sont ainsi constitués de sommets colorés représentatifs d'un niveau de la hiérarchie et accompagnés d'un ensemble de caractéristiques propres à ce niveau.

5.2.3 La reconnaissance et apprentissage par b-coloration

Les étapes de LBA et RAD se fondent sur le paradigme : « segmenter pour reconnaître et reconnaître pour bien localiser ». En effet, grâce à la reconnaissance du type de documents, il est possible de cibler directement les zones informantes d'une page à analyser, de produire un retour sur la segmentation et de l'améliorer.

Dans cette partie de reconnaissance (partie 5.3 du schéma de la figure 5.1), l'étape de séparation imprimé-manuscrit a été intégrée à la LBA car la classification par b-coloration est un excellent support de décision permettant d'attribuer des couleurs différentes entre bloc adresse en texte imprimé et bloc adresse en texte manuscrit. En une seule étape, il est possible de localiser le bloc adresse et de séparer le texte manuscrit de l'imprimé.

Dans l'étape de reconnaissance de la classe d'un document, on connaît a priori la nature des contenus des régions textuelles (s'il s'agit d'imprimé ou de manuscrit). Dans quelques rares cas la zone d'intérêt peut être tantôt manuscrite tantôt imprimée dans des classes de documents identiques : dans ce cas, il est également possible d'appliquer la b-coloration sur les blocs textuels présentant ces éventuelles ambiguïtés afin d'apprendre au système à faire une séparation complète entre imprimé et manuscrit.

5.3 Les modules essentiels de segmentation « brute » et d'analyse bas niveau

5.3.1 Première étape du processus de segmentation : la binarisation

5.3.1.1 Une coopération nécessaire entre binarisation et localisation des régions d'intérêt

Nous avons montré dans le chapitre 2 (section 2.2) que la binarisation d'image est appliquée dans la première étape de notre chaine de traitement et d'analyse d'image et a un très fort impact sur les performances du système de tri. Nous avons déterminé les limites des méthodes globales et des méthodes locales par rapport à notre application de tri en temps réel. Nous avons vu que les méthodes globales ne peuvent pas fournir de bons résultats quand l'image en niveaux de gris a une luminosité non-uniforme ou un histogramme multimodal dû à la présence des images publicitaires ou des plis sur le papier. Les méthodes locales dépassent cette limite mais nécessitent plus de calcul et sont ainsi plus lentes. En effet, aucune méthode classique, ni globale ni locale ne remplit efficacement toutes les conditions de temps réel et de performances imposées par notre application. Nous avons fait ce constat lors de l'étude comparative de plusieurs méthodes de binarisation présentée dans le chapitre 2 (section 2.2).

À partir là, nous sommes convaincus que toute mise en œuvre de la tâche de binarisation sans coopération avec la tâche de localisation de tracé augmente les temps de calcul. De plus, la séparation entre ces deux tâches conduit souvent à une sur-segmentation du bruit et de la texture du papier très présents sur le fond (ou les zones vides) de l'image de document.

Face à ces défis, nous avons pu optimiser cette étape en appliquant un seuillage local uniquement à proximité des zones de texte (figure 5.2) que nous avons localisées par la méthode des gradients cumulés en employant conjointement la multi-résolution et la morphologie mathématique. L'idée fondamentale de notre approche est de détecter très rapidement les zones de texte sur une image représentée en basse résolution. Par la suite nous appliquons un seuillage local seulement sur les zones détectées. Ceci nous évite ainsi de binariser le fond qui représente la plus grande partie de l'image. Ce ciblage des zones de tracé nous permettra, d'une part, de réduire d'une façon considérable les temps et, d'autre part, d'améliorer la qualité de binarisation en réduisant les erreurs de sur-segmentation et de sous- segmentation de fond et des caractères de texte. Rappelons que toute amélioration de la qualité des caractères augmente naturellement les taux de reconnaissance par OCR.

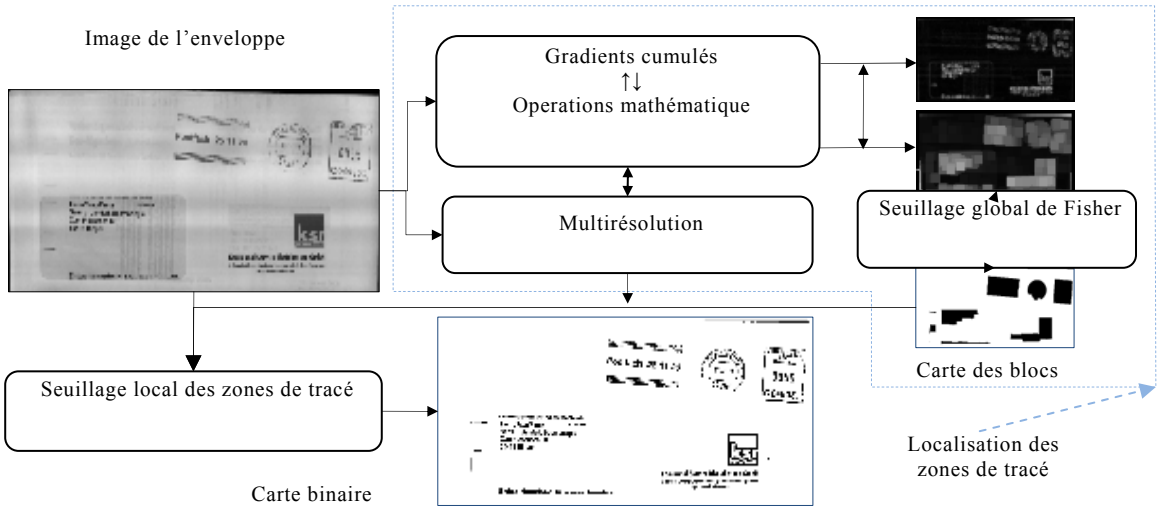


Figure 5. 2 : Schéma synoptique de notre méthode de binarisation (seuillage/localisation).

5.3.1.2 Notre proposition de localisation des zones de tracé par gradients cumulés

La localisation (ou la détection) des zones de texte doit être appliquée directement sur des images en niveaux de gris acquises par une caméra CCD linéaire de hautes vitesse et résolution. L'acquisition des images de documents ou d'enveloppes en mouvement dans une chaîne de tri génère un flou léger au niveau des caractères. L'encre utilisée lors de l'impression, la texture et la qualité du papier varient d'un document à un autre. De plus, la luminosité n'est pas toujours uniforme à cause de la présence de quelques plis. Tous ces paramètres rendent la tâche de localisation très difficile et coûteuse en temps de calcul. Pour mieux surmonter ces difficultés, nous avons utilisé la technique des gradients cumulés qui ne pose aucune contrainte sur l'éclairage ni sur la prise des images. Cette technique rapide consiste à accumuler les gradients mettant en évidence certaines régularités présentes au niveau des lignes de texte. Cette régularité est calculée à partir de séquences de pixels à gradients élevés. Le principe s'appuie sur le fait que les caractères du texte forment une texture régulière, les amplitudes de gradients les plus élevées correspondent donc aux fortes transitions lumineuses au niveau des contours intra ou inter caractères. Pour éviter d'introduire des nouveaux seuils ou perdre la pertinence de certains points lors de l'application de ce filtre, on effectue localement au voisinage Vs de chaque point (x_0,y_0) une simple sommation de gradients normalisée par le nombre N, nombre de pixels du voisinage $Vs(x_0,y_0)$:

$$Gr(x_0, y_0) = \frac{1}{N} \sum_{(x,y) \in Vs(x_0, y_0)} \frac{\partial f(x,y)}{\partial \vec{v}} \quad (5.1)$$

Ce filtre de « gradients cumulés », initialement développé pour la localisation de texte dans les images vidéo (images de petites tailles) [LEB97], a été utilisé notamment pour localiser des titres dans des vidéos non contraintes comme les archives télévisuelles [WOL02] et segmenter de l'imprimé composite couleur [LEB99]. Ce filtre dans sa version originale suppose que la direction du texte est à priori connue, les dérivées sont calculées dans la direction supposée du texte (souvent horizontale) et sommées dans cette même direction. Cette convention stricte pose inévitablement des problèmes sur des documents inclinés. Aussi, nous proposons d'améliorer et d'adapter ce filtre de la façon suivante, nous calculons les dérivées horizontales et verticales, puis nous les sommons dans les deux directions (5.2) pour rendre le filtrage insensible à la rotation des images des documents.

Pour s'adapter mieux à la grande taille de nos images ainsi qu'à la contrainte de temps réel, nous avons également mis en œuvre une approximation grossière mais rapide pour le calcul des dérivées (5.3). Le coût du calcul de la sommation en chaque point de l'image dans un voisinage *Vs* est trop élevé pour notre application. Nous allons donc réduire ce coût de calcul en effectuant la sommation par blocs en multi-résolution. Nous divisons l'image en blocs rectangulaires de taille $dx \times dy$ puis nous calculons dans chaque bloc la somme des gradients verticaux et horizontaux comme c'est écrit dans les formules suivantes :

$$Gr(x_0, y_0) = \frac{1}{dx\ dy} \sum_{i=1}^{dy} \sum_{j=1}^{dx} \left| \frac{\partial I(x_0 + j, y_0 + i)}{\partial x} \right| + \left| \frac{\partial I(x_0 + j, y_0 + i)}{\partial y} \right| \quad (5.2)$$

$$\begin{aligned} &\text{Avec} \quad \frac{\partial I}{\partial x}(u,v) = I(u-2,v) - I(u+2,v) \\ &\text{et} \quad \frac{\partial I}{\partial y}(u,v) = I(u,v-2) - I(u,v+2) \end{aligned} \quad (5.3)$$

L'accumulation des gradients par bloc donne rapidement une image *Gr* de basse résolution où les zones de texte représentent visiblement les zones les plus claires (voir figure 5.3). Afin d'obtenir un filtrage équivalent à celui de l'algorithme original et de rapprocher le résultat de la sommation par bloc à celui de la sommation par pixel, nous appliquons un lissage morphologique sur l'image *Gr*. Le surcoût de calcul de ce prétraitement en basse résolution est négligeable.

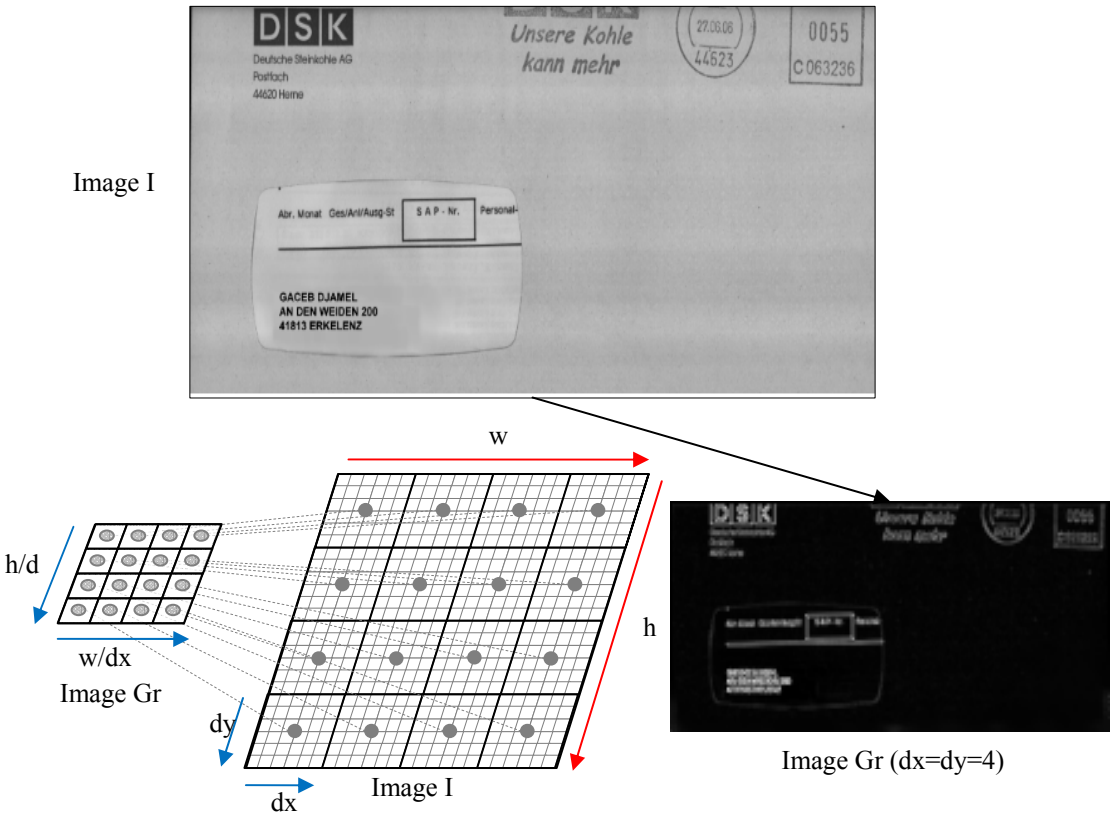


Figure 5. 3 : Accumulation des gradients de voisinage dans une basse résolution.

Pour cela, nous appliquons successivement sur l'image *Gr* en niveaux de gris, d_1 dilatation, e_1 érosions, d_2 dilatations et e_2 érosions. Ces opérations morphologiques peuvent être données pour tout élément structurant ξ par la formule suivante :

$$M(x,y) = \left[\left[\left[\underbrace{\left[(Gr(x,y) \oplus \xi) \oplus \ldots \oplus \xi \right]}_{d_1 \text{ dilatations}} \Theta \ldots \Theta \xi \right]_{e_1 \text{ érosions}} \oplus \ldots \oplus \xi \right]_{d_2 \text{ dilatations}} \Theta \ldots \Theta \xi \right]_{e_2 \text{ érosions}} \quad (5.4)$$

$$\text{Avec } (f(x,y) \oplus \xi) = \max_{(x_0,y_0) \in \xi} [f(x_0,y_0)] \quad \text{et} \quad (f(x,y) \Theta \xi) = \min_{(x_0,y_0) \in \xi} [f(x_0,y_0)] \quad (5.5)$$

Le résultat de d_k dilatations ou de e_k érosions peut être obtenu avec une seule itération en multipliant la taille de l'élément structurant par le nombre d'itérations. Ceci permet de réduire aussi les temps de traitement. La formule (5.4) devient donc :

$$M(x,y) = \left[\left[\left[\left[(Gr(x,y) \oplus \xi_{d1}) \right] \Theta \xi_{e1} \right] \oplus \xi_{d2} \right] \Theta \xi_{e2} \right] \quad \text{avec} \quad \begin{aligned} \xi_{d1} &= \xi \times d_1 \\ \xi_{e1} &= \xi \times e_1 \\ \xi_{d2} &= \xi \times d_2 \\ \xi_{e2} &= \xi \times e_2 \end{aligned} \quad (5.6)$$

L'application de ce traitement morphologique permet, d'une part, de re-densifier le texte et donc de l'agglomérer en blocs et, d'autre part, de prendre une marge suffisante autour du trait afin d'inclure l'information pertinente portée par l'arrière plan (la texture et la couleur) pour un meilleur seuillage. Ceci permet aussi d'améliorer la qualité des caractères binaires ce qui a pour conséquence d'augmenter le taux de l'OCR. Les paramètres d_1, e_1, d_2, e_2 du masque ainsi que la taille de la fenêtre $dx \times dy$ ont été fixés pour le moment empiriquement. On peut constater qu'une augmentation de dx et dy mène à une détection grossière et plus rapide du texte alors que l'augmentation de d_1 et e_1 détecte mieux les zones de textes agglomérées entre elles. L'augmentation de d_2 et e_2 renforce l'effet de lissage sur des images trop bruitées. Les expériences que nous avons effectuées sur des images de différents types de documents montrent que la stabilité du résultat peut être atteinte en respectant la convention suivante : $e_1=d_1=2$, pour détecter les zones de textes et $e_1=d_1=1$, pour détecter les mots. La succession ordonnée de ces opérations élémentaires donne des ouvertures et fermetures morphologiques filtrant toute sorte de réponses de gradient à la texture, au bruit de fond et même aux défauts de papier.

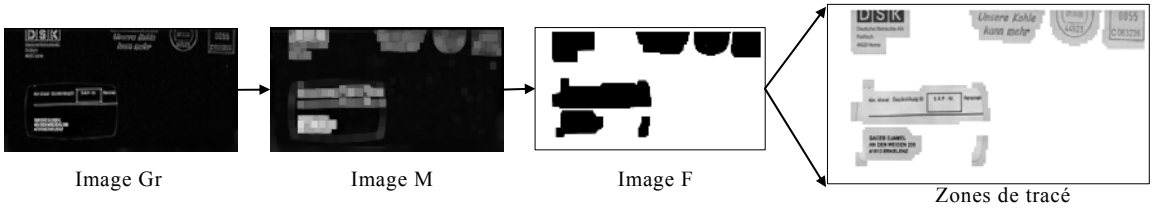


Figure 5. 4 : Traitement morphologue et obtention automatique d'un masque binaire *F* par la méthode de Fisher.

Le traitement morphologique est poursuivi par un seuillage global de Fisher (chapitre 2, section 2.2) donnant un masque binaire *F* qui contient les différents blocs de tracé. Cette méthode calcule très rapidement un seuil global à partir de l'histogramme de niveaux de gris de l'image *M*.

Ce masque binaire est utilisé pour diriger le seuillage local en pleine résolution vers les zones des tracés et peut être considéré comme une première segmentation en blocs de structure physique de l'image de document.

Cette mise en évidence rapide des blocs joue deux rôles très important en termes de temps de calcul : d'une part, elle permet de réduire efficacement le temps de seuillage local pour le rendre presque similaire à celui de seuillage global, d'autre part, elle permet d'accélérer la phase de l'extraction de la structure physique qu'on verra prochainement en détails.

5.3.1.3 Binarisation finale issue du seuillage local des régions d'intérêt

Pour obtenir la carte binaire *B* de premier plan en pleine résolution, nous avons choisi d'utiliser la méthode de Sauvola d'une part pour sa rapidité par rapport aux autres méthodes locales (tableau 5.1) et d'autre part pour ses performances (la méthode Wolf est spécifique aux images vidéo et ne convient pas pour notre application). Ce seuillage local est appliqué seulement sur les zones de tracé localisées dans le masque *F*. Le temps économisé a permis d'utiliser une grande taille de fenêtre (*21x21* par exemple) ce qui permet d'obtenir de très bons résultats sur les documents imprimés et manuscrits avec des tailles de caractères très variables.

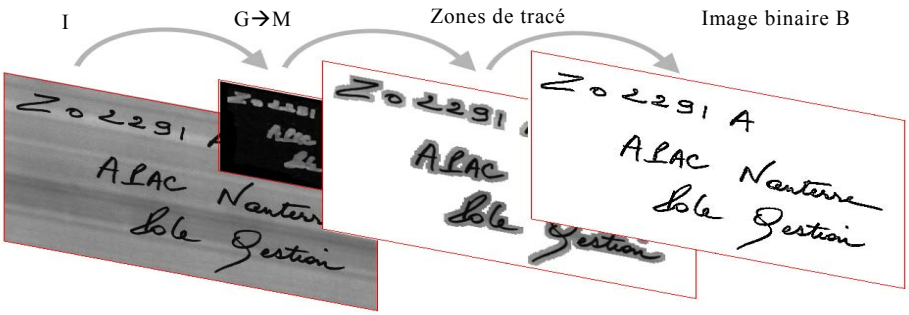
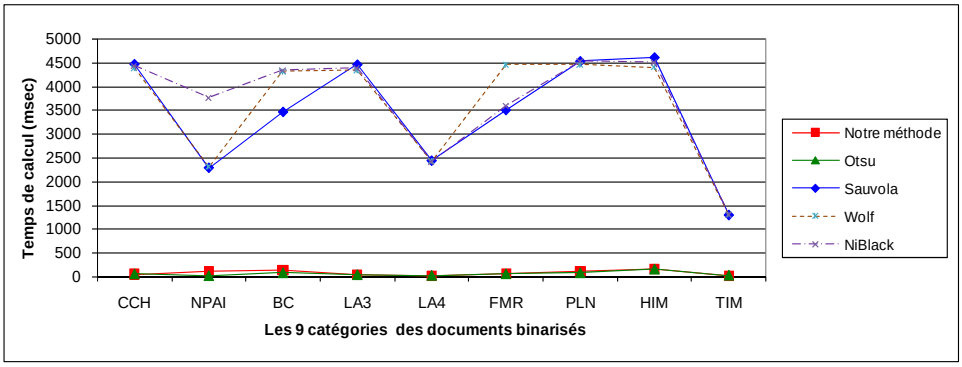


Figure 5. 5 : Étapes de la binarisation hybride (seuillage local / localisation des zones de tracé).

Le ciblage des zones de tracé a permis notamment de binariser l'image avec des temps réduits similaires à ceux des méthodes globales tout en profitant des performances des méthodes locales.

Afin de comparer les temps de binarisation de notre méthode avec ceux des méthodes globales (Otsu) et des méthodes locales (Sauvola, Niblack et Wolf), nous avons utilisé une base de *29225 images de documents* et de courrier interne d'entreprises. Cette base est répartie en *9* catégories (Chèques circulants : *CCH*, *NPAI*, Cartes Bleus : *CB*, Listings *A3* : *LA3*, Listings *A4* : *LA4*, Formulaires : *FMR*, Planus : *PLN*, Courrier interne manuscrit : *HIM*, Courrier interne dactylographique : *TIM*). Les courbes de la figure suivante montrent les temps moyens écoulés pour binariser les documents de chacune *9* catégories par notre méthode mixte puis par la mé-

thode globale d'Otsu et les méthodes locales de Sauvola, de Niblack et de Wolf.



Pour avoir plus de visibilité des courbes ci-dessus nous avons divisé les temps de la méthode Sauvola, NiBlack et Wolf par *10* (voir les courbes suivantes).

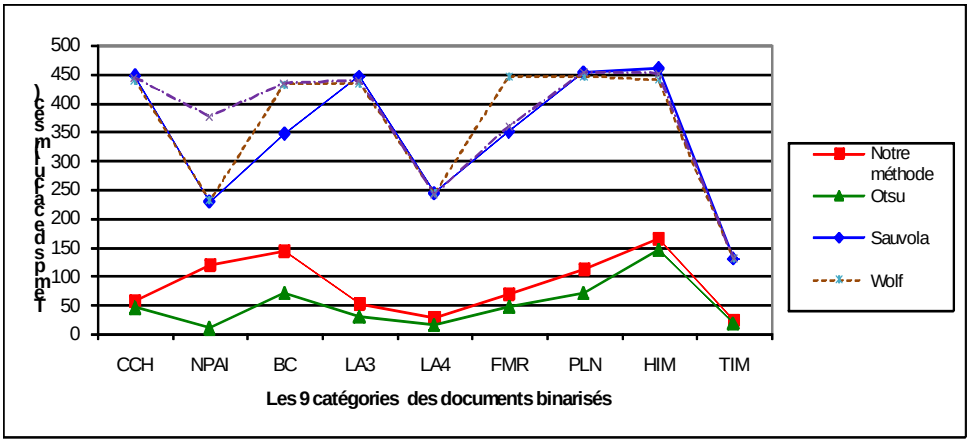


Figure 5. 6 : Comparatif des temps de binarisation de notre méthode par rapport aux différentes méthodes.

Le comparatif des temps de la figure ci-dessus montre que les temps de calcul moyens de notre méthode hybride (seuillage/ localisation) de binarisation sont similaires à ceux des méthodes globales et beaucoup moins petits que ceux des méthodes locales.

Pour évaluer la qualité des caractères binaires offerte par notre méthode mixte de binarisation, nous avons appliqué l'OCR sur les images de même base binarisées par notre méthode puis binarisées par la méthode classique de Sauvola. Le tableau suivant montre les augmentations des taux d'OCR par notre méthode mixte par rapport à la méthode de Sauvola. Ceci signifie que notre méthode de binarisation préserve parfaitement la qualité des caractères par rapport aux autres méthodes.

Categories	Augmentation des taux d'OCR
CCH	+2%
NPAI	+26%
CB	+13%
LA3	+11%
LA4	+11%
FMR	+23%
PLN	+20%
HIM	+76%
TIM	+16%

Tableau 5.1 : Augmentation des taux d'OCR par notre méthode mixte de binarisation (seuillage local/localisation) par rapport à la méthode classique de Sauvola.

La figure suivante montre comment notre méthode de seuillage mixte offre une binarisation de meilleure qualité par rapport aux méthodes globales et aux méthodes locales.

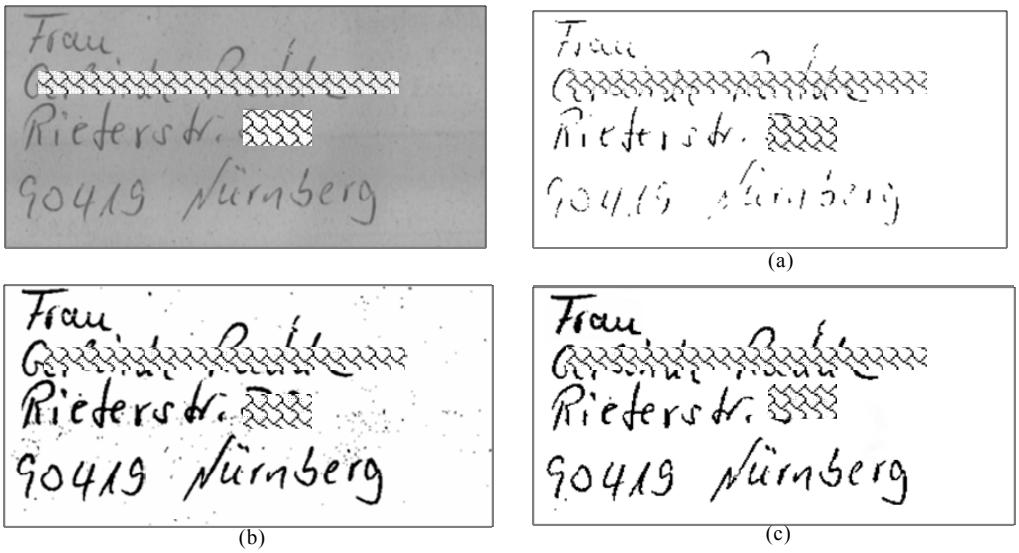


Figure 5. 7 : Résultat de binarisation d'une zone d'adresse, **(a)** par la méthode globale d'Otsu, **(b)** par la méthode locale classique de Sauvola, **(c)** par notre méthode de seuillage mixte.

5.3.1.4. Intérêt de l'approche de binarisation sur l'étape de détection des composantes connexes

En plus des avantages que ne venons d'expliquer, notre méthode hybride de seuillage a également permis de réduire les temps de calcul des composantes connexes par la diminution du nombre de pixels noirs à analyser dans des surfaces homogènes de grandes tailles qui correspondent souvent à des photos ou indications publicitaires. L'exemple de la figure 5.8

montre un exemple de résultat sur un cas concret. Nos tests ont montré que cette approche de binarisation a permis de diviser par trois le temps de détection des CCs par rapport à celui d'une détection des CCs obtenue par l'utilisation exclusive de l'algorithme de Sauvola.

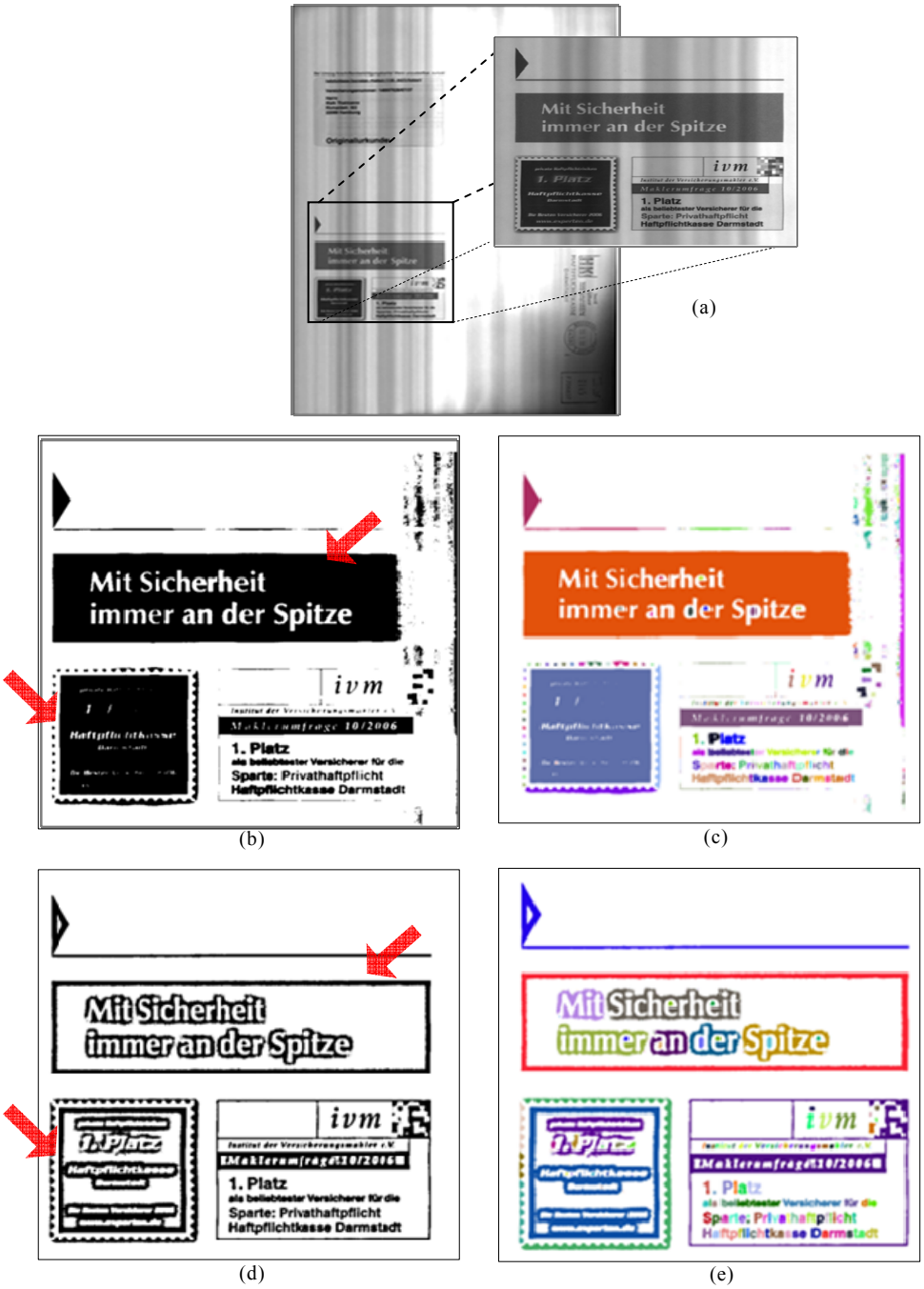


Figure 5. 8 : (a) Image d'une enveloppe en niveaux de gris, **(b)** seuillage par l'algorithme classique de Sauvola. **(d)** résultat de notre méthode de seuillage hybride. **(c)** et **(e)** cartes des composante connexes détectées sur les deux cartes binaires.

5.3.2 Étape d'extraction des composantes connexes

Nous détectons ainsi les composantes connexes (CCs) du masque binaire en basse résolution. Ces dernières sont utilisées ensuite pour guider une seconde extraction des CCs de carte binaire à pleine résolution (de premier plan proprement dit). Sur les deux cartes, l'extraction des connexités suit le même principe. Pour détecter ces CCs, nous avons besoin d'un algorithme rapide qui s'adapte aussi bien à notre architecture pyramidale qu'à notre application de tri contribuant ainsi à une extraction hiérarchique de certaines caractéristiques. Pour garantir en même temps une exécution des traitements dans des temps compatibles avec nos besoins de temps réel et permettre une extraction de caractéristiques utiles aux étapes d'analyse de structures, nous proposons une nouvelle méthode inspirée des travaux de Pavlidis [PAV77] sur la structure LAG (Line Adjacency Graph) décrivant les formes connexes présentes sur les images. L'idée de base consiste à adapter cette structure LAG à la méthode d'extraction de connexités développée par Lifeng [LIF07][LIF08] et que nous avons améliorée. Nous avons choisi cette méthode car elle fonctionne avec des temps bien meilleurs que les approches habituellement utilisées en recherche de connexités (le bilan est dans le chapitre 2).

5.3.2.1 L'algorithme de capture des connexités

Notre algorithme de capture des connexités s'exécute en un seul balayage de l'image utilisant la structure LAG simplifiée. Nous avons réduit à un seul scan l'algorithme de deux scans de Lifeng pour en augmenter la vitesse, [LIF07][LIF08]. Une partie du temps économisé par la suppression du deuxième scan est utilisée pour extraire quelques caractéristiques nécessaires à la détection de l'inclinaison des lignes. Si une séquence noire de la ligne d'image précédente est adjacente à une autre séquence noire de la ligne d'image suivante, alors on les met en relation en leur attribuant une étiquette commune. On propage ainsi les étiquettes de connexité en connexité avec un seul balayage avant de l'image (figures 5.9 et 5.10).

On suppose au début que tous les pixels du bord de l'image appartiennent au fond et que toutes les étiquettes temporaires c_t des séquences noires d'une composante connexes sont regroupées dans un ensemble $CC(c_r)$, où c_r est la plus petite étiquette qui est considérée comme représentante dans son ensemble CC. Pour sauvegarder le lien entre chaque étiquette temporaire avec son étiquette représentante, nous utilisons une table des équivalences (notée T_{eq}) de la façon suivante :

$$\forall c_t \in CC(c_r),\ \mathrm{T_{eq}}(c_t) = c_r \qquad (5.\ 7)$$

Dans un balayage avant de l'image binaire, notre algorithme parcours un par un les pixels des zones de tracé. Dès qu'une nouvelle séquence noire est trouvée, ces données seront enregistrées dans un vecteur *r*. En même temps, toutes les séquences noires de la ligne précédente connexes avec la séquence courante seront détectées. Pour optimiser l'espace mémoire nous utilisons en alternance deux buffers T_{pair}, T_{impair} appelés aussi « tampons » qui retiennent l'état des étiquettes des connexités des lignes d'image précédente et suivante. Pour chaque ligne de numéro impair, l'algorithme repère dans T_{impair} les séquences noires puis les compare avec les séquences adjacentes de la ligne précédente déjà étiquetées et décrites par le tampon T_{pair} et vice versa.

Si la nouvelle séquence *r* n'a aucune adjacence avec les séquences noires de la ligne précédente, alors elle forme une nouvelle connexité qui sera affectée d'une nouvelle étiquette *c* (*création*). Dans ce cas, l'ensemble d'étiquettes temporaires *CC* qui correspond à une nouvelle composante connexe reçoit la nouvelle étiquette comme premier élément : $CC(c)=\{c\}$. Cette étiquette temporaire unique représente elle-même dans la table d'équivalence : $T_{eq}[c]=c$. Les coordonnées du rectangle circonscrivant la composante connexe *CC* peuvent être données par :

$$x_d(CC) = x_d(CC),\ x_f(CC) = x_f(r),\ y_d(CC) = y_f(CC) = y(r) \quad (5.\ 8)$$

- La densité de cette composante connexe est initialisée par la formule suivante : $Dens(CC) = x_f(r) - x_d(r)$ (5. 9)

D'autre part, si la séquence r possède des adjacences avec certaines séquences noires de la ligne précédente (étiquetées de gauche vers la droite $c_1, \ldots,$ *et* c_n), toutes ces séquences seront fusionnées avec cette séquence dans la même composante connexe. Dans ce cas, $CC(c) = CC(c_1) \cup \cdots \cup CC(c_n)$ avec $c = \min\{c_1, \ldots, c_n\}$ et la séquence courante reçoit l'étiquette c_1 (étiquette de la séquence à l'extrême gauche). Le principe d'étiquetage de la séquence noire courante est illustré par un exemple simple de la figure suivante :

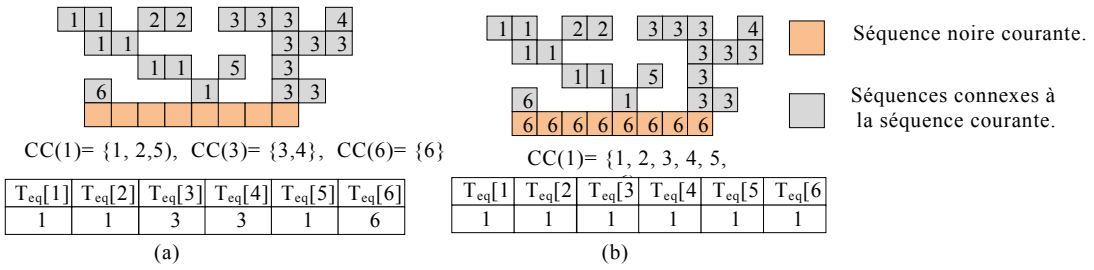


Figure 5. 9 : Exemple d'étiquetage d'une séquence noire courante, **(a)** cas avant étiquetage, **(b)** cas après étiquetage.

Dans ce cas, les coordonnées du rectangle circonscrivant la composante connexe *CC* ainsi que sa densité *Dens* peuvent être mises à jour par :

$$\begin{aligned} x_d(CC) &= \min\{\min_{i=1\ldots n}\{x_d(CC_{ci})\}, x_d(r)\} \\ x_f(CC) &= \max\{\max_{i=1\ldots n}\{x_f(CC_{ci})\}, x_f(r)\} \\ y_d(CC) &= \min_{i=1\ldots n}\{y_d(CC_{ci})\} \\ y_f(CC) &= y(r) \end{aligned} \qquad (5.\ 10)$$

$$Dens(CC) = \frac{1}{H \times W}\sum_{i=1}^{n} Dens(CC_{ci}) + x_f(r) - x_d(r) \qquad (5.\ 11)$$

La hauteur H et la largeur W de chaque composante connexe sont calculées directement à partir de ses coordonnées de la façon suivante :

$$W(CC) = x_f(CC) - x_d(CC) \text{ et } H(CC) = y_f(CC) - y_d(CC) \qquad (5.\ 12)$$

Quand le parcours de toute l'image est fini, chaque composante connexe a une étiquette représentante définitive. L'étiquette de n'importe quelle séquence noire peut être obtenue rapidement à l'aide de la table d'équivalences.

Afin de simplifier l'implémentation de notre algorithme, nous proposons d'utiliser le terme *«listes»* à la place de terme *«ensembles»* d'étiquettes temporaires. De plus nous avons utilisé trois tableaux pour gérer rapidement les opérations de création des nouvelles listes ou de fusion entre des listes.

Le premier tableau (noté *«Suivante»*) est utilisé pour conserver l'étiquette suivante de l'étiquette courante. *Suivante*$[c_i]=c_j$ signifie que l'étiquette c_j suit l'étiquette courante c_i, (*Suivante*$[c_i]=-1$ signifie que ci est la dernière étiquette de sa liste et n'a aucune étiquette suivante).

Le second tableau (noté *«Queue»*) est utilisé pour conserver la dernière étiquette de chaque liste. *Queue* $[c_i]= c_j$ représente la dernière étiquette c_j de la liste $CC(c_i)$.

Le troisième tableau (noté *«Equiv»*) est utilisé pour conserver les équivalences entres les étiquettes temporaires et les étiquettes représentantes. *Equiv*$[c_i]=c_j$ indique que c_j devient l'étiquette représente de l'étiquette temporaire c_i. De la même manière, on peut obtenir l'étiquette représentante de n'importe quelle étiquette temporaire par la relation suivante : $c_j=Equiv[c_i]$.

Durant le balayage de l'image binaire, la création d'une nouvelle composante déclenche une création d'une nouvelle liste d'étiquettes temporaires $CC(c)=\{c\}$. Cette opération peut être effectuée facilement par l'algorithme suivant :

Algorithme 5.1 : *Création(CC(c))*

```
Début :
        Equiv [c]= c
        Suivante [c]=˜1
        Queue [c]=c
Fin.
```

La fusion entre deux listes d'étiquettes temporaires, $CC(c_i)$ et $CC(c_j)$ avec $c_i < c_j$, consiste à relier la première étiquette de la liste $CC(c_j)$ à la dernière étiquette de la liste $CC(c_i)$. La liste temporaire résultante de cette fusion est donnée par la relation suivante : $CC(c_i) = CC(c_i) \cup CC(c_j)$. Les étapes de cette opération sont décrites dans l'algorithme suivant :

Algorithme 5.2 : *Fusion(c_i, c_j)*

```
Début :
        c = cj
        Tanque (c ≠1) Faire
         rtable[c] = ci
         c = Suivante[c]
        Fin de tant que
        Suivante[Queue [ci]] = cj
        Queue[ci] = Queue[cj]
Fin.
```

La figure 5.10 illustre un exemple d'application de notre algorithme d'extraction des composantes connexes sur la carte binaire *B* et le masque binaire *M*.

Les courbes de la figure 5.11 montrent les temps de capture des connexités de notre méthode comparés à la méthode LAG classique, la méthode de Lifeng et la méthode de Glassner. Les tests ont été effectués sur une base de *28 images* d'enveloppes et de formulaires de différentes tailles. Les courbes montrent que notre méthode consomme moins de temps de calcul que les trois autres méthodes.

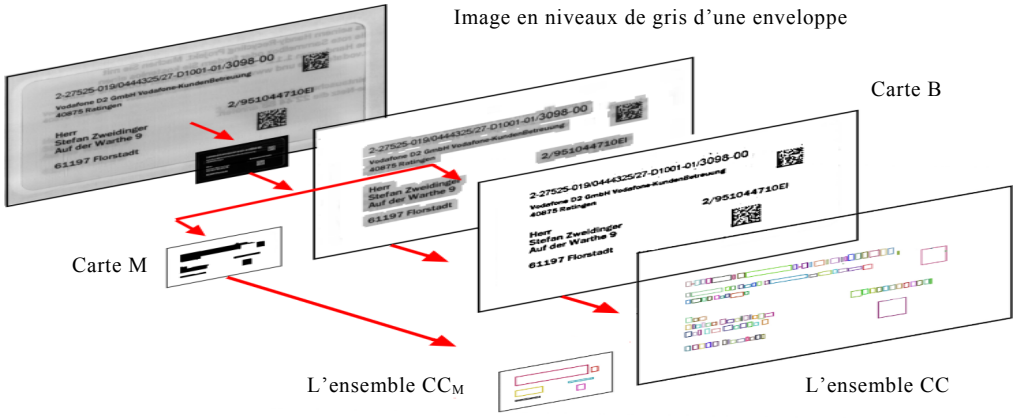


Figure 5. 10 : Seuillage mixte et détection des CCs sur les deux cartes (**à gauche carte M et à droite carte B**). L'ensemble $CC=\{cc_i\}$ représente

l'ensemble des *CCs* de la carte *B* et $CCM=\{FM_i\}$ représente l'ensemble des *CCs* des blocs de la carte *M*.

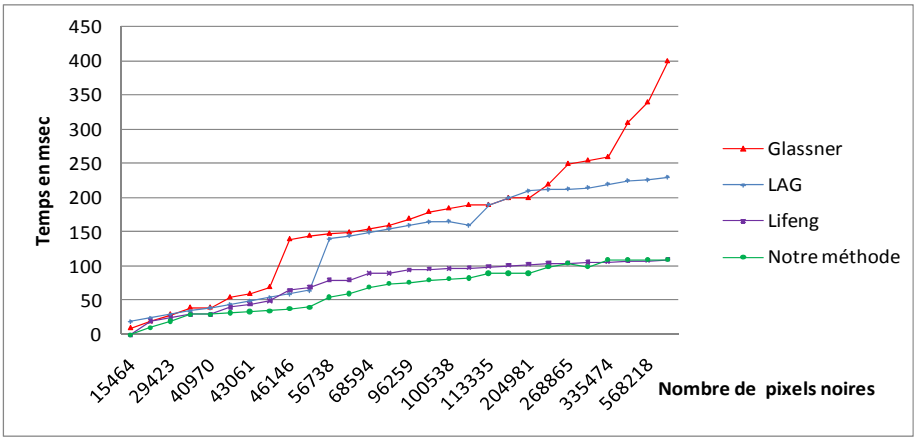


Figure 5. 11 : Comparatif des temps de capture des connexités par notre méthode par rapport aux autres méthodes.

5.3.2.2 Caractérisation des séquences noires des connexités

Comme nous l'avons présenté en introduction, durant l'étape de recherche de connexités, nous avons choisi d'utiliser une structure de graphe d'adjacence (nommé $LAG=(E_r,V_r)$, avec $V_r=\{r_i(x_d^i[y],x_f^i[y])\}$) pour représenter la carte des composantes connexes et l'adjacence locale des séquences noires qui les constituent.

Une séquence noire est définie comme une suite horizontale et contenue de pixels noirs (de premier plan). Le LAG est un cas particulier du graphe d'adjacence de régions, utilisé pour les traitements rapides lignes par lignes des images binaires [PAV77]. Dans ce graphe, chaque nœud r_i représente une séquence noire définie par sa position en abscisse (x_d, y_d), et doit avoir dans ce graphe au moins un nœud adjacent. Les arcs représentent les liens reliant les séquences noires connexes (adjacentes) (figure 5.12). L'adjacence entre deux séquences noires (r_y, r_{y-1}) de deux lignes consécutives (y-1 et y) peut être donnée par la relation suivante :

$$\forall e \in E_r,\ e(r_{y-1},\ r_y)=\begin{cases}1 & \text{si } x_d[y]\le x_f[y-1] \text{ et } x_d[y-1]\ \le x_f[y]\\ 0 & \text{sinon}\end{cases} \qquad (5.\ 13)$$

Ce type de représentation n'a pas les propriétés d'invariance à la rotation, mais il permet de caractériser les formes en terme de segments, de fermetures et d'ouvertures de boucles et d'intersections (figure 5.12). En effet, ce codage par adjacence présente l'avantage de conserver toute l'information des formes avec un encombrement raisonnable et offre un accès facile aux données pour les traitements séquentiels comme le suivi de contour, l'analyse des formes et l'estimation de l'italique et de l'inclinaison locale des composantes.

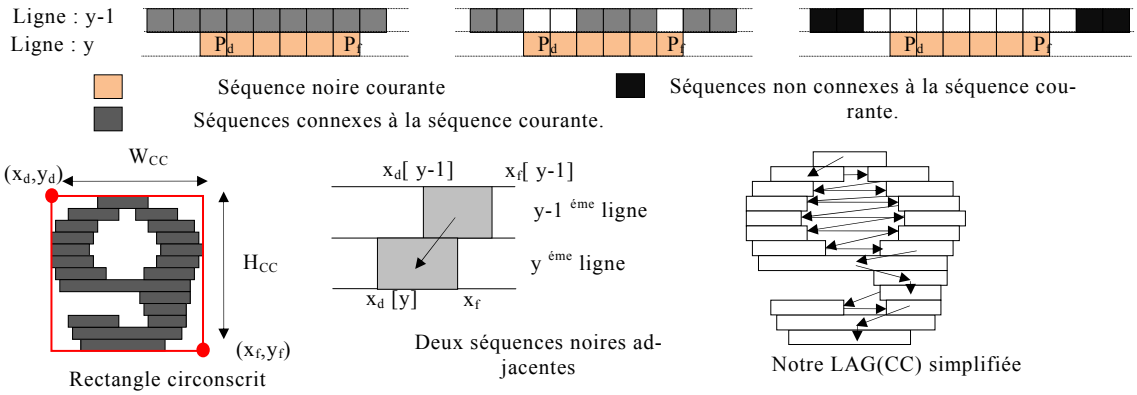


Figure 5. 12 : Notion de la connexité entre séquences noires de deux lignes consécutives, et exemple de LAG associé à une composante connexe.

Concernant notre application de tri, la construction du LAG simplifiée s'effectue simultanément avec notre algorithme de capture des connexités. Pour simplifier la structure d'origine, seules les coordonnées de rectangle circonscrit de chaque composante connexe seront retenues avec quelques caractéristiques nécessaires à la phase de redressement, d'analyse et d'analyse de la structure physique.

5.3.2.3 Bilan sur l'intérêt de notre approche de recherche de connexités

Par rapport à l'approche proposée par Lifeng dans [LIF07] et [LIF08], notre proposition de détection de connexités se base sur une seule utilisation de l'algorithme au lieu des deux passages nécessaires. Le temps économisé est alors utilisé pour extraire quelques caractéristiques utiles à la détection de l'inclinaison des lignes sur la base de la représentation par graphe d'adjacence LAG des séquences noires que nous présentons ci-dessous.

Notre algorithme de capture des connexités présente donc l'intérêt de pouvoir s'exécuter en un seul balayage en exploitant une structure LAG simplifiée de description des séquences noires des formes construite au fur et à mesure du balayage.

Si une séquence noire de la ligne d'image précédente est adjacente à une autre séquence noire de la ligne d'image suivante, alors on les met en relation en leur attribuant une étiquette commune. Ainsi on propage les étiquettes de connexité en connexité avec un seul balayage avant de l'image.

5.3.3 Redressement des lignes inclinées de texte et des caractères italiques

5.3.3.1 Redressement de l'inclinaison des lignes

5.3.3.1.1 Nécessité du redressement de lignes pour améliorer l'analyse

Comme nous l'avons vu au chapitre 1, les images de courriers d'entreprise présentent la particularité de contenir en de nombreux endroits des lignes d'inclinaisons variables. Ces inclinaisons (appelées en anglais skew) peuvent être dues à plusieurs facteurs (voir figure 5.13) :

- Certains documents sont scannés avec un angle de rotation (cas rare),
- Certaines informations sont écrites sur un papier adhésif (adresse par exemple) collé d'une façon inclinée,
- Une présence relativement fréquente d'annotations manuscrites dans des orientations non horizontales,
- Une rotation du support papier du bloc adresse à l'intérieure de l'enveloppe à fenêtre transparente plastifiée.

Figure 5. 13 : Exemples de blocs adresses postales inclinées dues à différents facteurs.

En observant ces images, on conçoit aisément, qu'en l'absence de redressement des lignes, on introduise des erreurs dans les analyses et traitements à venir. Si les outils d'analyse ne tiennent pas compte de ces diverses inclinaisons de lignes, on s'expose à des risques de rejet conséquents et des erreurs de lecture par OCR. En effet, on peut aisément concevoir que la présence de lignes de texte inclinées puisse réduire la précision de la segmentation, de la caractérisation des lignes et par conséquent de celle des blocs. Cela entraîne également des erreurs de localisation des blocs adresse.

On peut à titre d'exemple, citer certaines méthodes de segmentation en lignes comme la méthode de projection des profils et la méthode RLSA qui deviennent inopérantes sur des images dont le texte n'est pas redressé.

Un autre point important dans le redressement du texte est qu'il facilite la lecture optique (par l'OCR), car la lecture d'un texte bien aligné (dans le sens horizontal) conduit à des taux de reconnaissance de caractères bien meilleur.

Cela signifie que la segmentation en lignes d'un document et la reconnaissance optique des caractères des textes sont intimement liées à la détection de l'inclinaison du texte

En pratique, une fois l'emplacement des lignes de texte obtenu, il est relativement aisé d'en déduire leur angle d'inclinaison. De nombreuses approches d'estimation de l'inclinaison existent pour cela.

La plupart des algorithmes de détection d'inclinaison utilisent une information a priori sur la localisation des lignes du document, d'autres approches procèdent à une recherche d'inclinaison de ce qui est supposé correspondre aux composantes de lignes. Cette inclinaison permet ensuite une segmentation plus fine des lignes et des blocs.

Dans notre travail, nous avons choisi d'intégrer l'estimation de l'inclinaison dans la phase de l'extraction de la structure physique pour augmenter la précision de détection des lignes inclinées et réduire les temps de calcul. Cette inclinaison est finalement utilisée pour redresser uniquement les lignes des zones d'intérêt (sans redresser tout le document), ce qui conduit à un gain de temps de calcul conséquent.

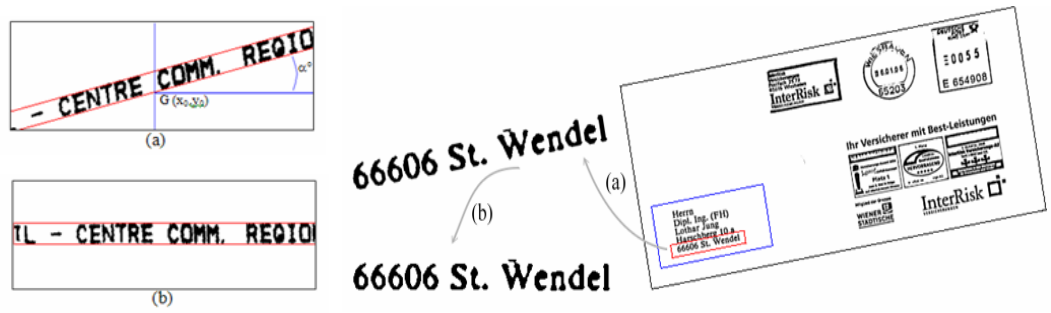


Figure 5. 14 : (a) ligne de texte binaire inclinée avec un angle α, **(b)** ligne texte binaire après rotation inverse.

5.2.3.1.2 Quelques approches usuelles de détection de l'inclinaison des lignes de texte

La première étape de redressement est très importante, car elle concerne l'estimation de l'angle d'inclinaison des lignes de texte (c'est une étape cruciale pour le redressement). On peut classer les méthodes présentées dans la littérature en plusieurs familles où chacune d'elles repose sur une des techniques suivantes :

- la projection des profils [SHI90], [BAG97], [COT98], [KAV99],
- la transformée de Hough [LED94], [SMI95], [GAT96][CHI97],
- les moindres carrés [CHI97] [BEL92], [ING99] ou le centre d'inertie [SAR07].

Plus rarement, on trouve d'autres techniques reposant sur l'analyse du spectre de Fourier [POS86], sur la transformée de Radon

[DON05], l'analyse du gradient ou encore sur les transformées morphologiques [BER98][XIA99][SAF00][LIO01][SHI03] [PAR05][LED94].

La méthode populaire de projection des profils consiste à effectuer plusieurs projections selon différentes orientations puis à valider l'angle qui correspond à l'entropie minimale [COT98] ou à une distribution de Wigner-Ville maximale [KAV99]. Ces méthodes présentent un grand avantage face aux dégradations dues à divers types de bruits (bruit réparti sur toute l'image, comme le bruit gaussien ou impulsionnel, mais aussi présence de petits éléments parasites localement). Elles reposent cependant sur de multiples projections très pénalisantes en temps de calcul. Ces méthodes à base de projection ne sont efficaces que sur des blocs contenant des lignes d'inclinaison uniforme. Dans le cas des documents de structure complexe contenant des graphiques, il est nécessaire d'isoler les lignes de texte avant de les projeter.

La transformée de Hough est aussi une technique très utilisée dans l'estimation de l'inclinaison. Elle repose sur la transformation de l'espace des coordonnées cartésiennes des pixels noirs de l'image à l'espace des coordonnées polaires. L'inclinaison correspond à l'angle qui à la plus grande accumulation de pixels noirs dans l'espace polaire. La précision de cette estimation est fortement liée à la résolution angulaire. Toute augmentation de cette résolution fait croitre très rapidement les temps du calcul, [SMI95]. L'économie en temps peut cependant être obtenue par une réduction de l'espace d'analyse uniquement autour des pixels noirs des lignes de texte, [SRI89]. [GAT96] [CHI97] ou encore uniquement sur les centres des CCs des lignes [YUB96] ou leurs contours.

Malgré toutes les tentatives faites pour réduire les temps d'exécution de la méthode de projection des profils ou de la transformée de Hough, nos évaluations expérimentales montrent que la méthode de moindre carrées conduit toujours aux meilleurs temps et à la plus grande précision. Nous avons choisi de porter notre attention sur cette méthode que nous avons améliorée.

5.3.3.1.3 Notre approche de l'estimation de l'inclinaison des lignes : Approche des moindres carrés

Notre approche de l'estimation de l'inclinaison des lignes repose sur le paradigme plusieurs fois repris dans cette thèse et adaptés, à chaque fois, à différents cas de figures, à savoir : « Segmenter pour reconnaître et reconnaître pour bien segmenter ». Dans ce cas précis, il s'agit de parvenir à bien segmenter initialement les composantes textuelles afin de produire une évaluation de l'inclinaison suffisamment précise pour contrôler alors une segmentation plus précise.

Nous avons intégré notre technique d'estimation de l'inclinaison des lignes de texte dans la phase d'extraction des connexités et de l'analyse de la structure physique. Durant ces deux phases, on mesure de façon incrémentale, à chaque niveau de l'analyse de la structure (niveau des séquences noires, niveau des CCs, niveau des lignes) les informations nécessaires à l'estimation de l'inclinaison.

Nous avons développé une méthode basée sur la technique des moindres carrés plus adaptée à la structure pyramidale que nous avons implémentée pour l'analyse de la structure des documents. Elle est conçue en interaction avec la segmentation hiérarchique. Elle garantit ainsi rapidité, précision et insensibilité au bruit. Elle s'est également montrée très appropriée à de nombreux types de documents de structures complexes incluant des graphiques.

Le principe de notre méthode consiste à calculer l'angle d'inclinaison d'une ligne L à partir de son nuage de points (de pixels noirs) regroupés en séquences noires, en CCs de texte puis en ligne de texte. On identifie par régression linéaire la meilleure droite passant à travers ce nuage de points incliné. Cette droite d'équation $y=A_Lx+B_L$, dite "Droite des Moindres Carrés" est caractérisée par sa pente A_L et une ordonnée à l'origine B_L [BEL92], [CHI97], [ING99], [SAR07], voir figure 5.15.

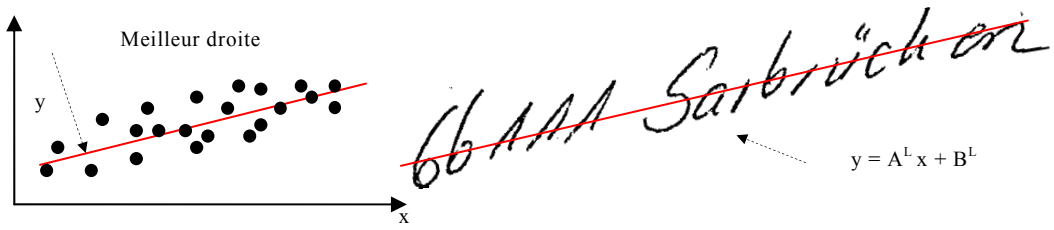


Figure 5. 15 : Exemple de droite des moindres carrés.

Soit une ligne L(i) de $n_{L(i)}$ CCs avec $L(i)=\{cc(j)\in L(i) \mid j=1...n_{L(i)}\}$, chaque composante *cc(j)* se compose de $n_{cc(i)}$ séquences noires avec $cc(j)=\{r(t)\in cc(j) \mid t=1...n_{cc(j)}\}$. Les coefficients $A_{L(i)}$ et $B_{L(i)}$ de la droite des moindres carrés de cette ligne peuvent être calculés à partir des formules suivantes :

La pente $A_{L(i)}$ a est donnée par :

$$A_{L(i)} = \frac{\sum_{j=1}^{n_{L(i)}} Sxy_{cc(j)} - \sum_{j=1}^{n_{L(i)}} Sx_{cc(j)} \times \sum_{j=1}^{n_{L(i)}} Sy_{cc(j)}}{\sum_{j=1}^{n_{L(i)}} Sxx - \left[\sum_{j=1}^{n_{L(i)}} Sx\right]^2} \qquad (5.\ 14)$$

Les valeurs $Sxy_{cc(j)}, Sx_{cc(j)}, Sy_{cc(j)}, Sxx_{cc(j)}$ sont calculées à partir des séquences noires de la façon suivante :

$$Sxy_{cc(j)} = n_{cc(j)} \sum_{t=1}^{n_{cc(j)}} x_G^{r(t)} y_G^{r(t)}, \quad Sx_{cc(j)} = \sum_{t=1}^{n_{cc(j)}} x_G^{r(t)}$$
$$Sxx_{cc(j)} = n_{cc(j)} \sum_{t=1}^{n_{cc(j)}} \left[x_G^{r(t)} \right]^2, \quad Sy_{cc(j)} = \sum_{t=1}^{n_{cc(j)}} y_G^{r(t)} \qquad (5.\ 15)$$

avec $x_G^{r(t)} = \frac{1}{2}(x_d^{r(t)} + x_f^{r(t)}), \quad y_G^{r(t)} = \frac{1}{2}(y_d^{r(t)} + y_f^{r(t)})$ (5. 16)

La translation B_L est donnée par :

$$B_{L(i)} = \frac{1}{\sum_{j=1}^{n_{L(i)}} n_{cc(j)}} \sum_{j=1}^{n_{L(i)}} Sy_{cc(j)} - A_{L(i)} \times \sum_{j=1}^{n_{L(i)}} Sx_{cc(j)} \qquad (5.\ 17)$$

Finalement, l'angle d'inclinaison de la ligne est donné par la formule suivante :

$$\theta_{L(i)} = \tan^{-1}(A_{L(i)}) \qquad (5.\ 18)$$

5.3.3.1.4 Résultats et comparatifs d'approches

La courbe suivante représente la succession des valeurs d'angle d'inclinaison estimées par différentes méthodes sur *21* lignes de texte tournées d'un pas de *0.5°* dans l'intervalle angulaire *[0°,10°]*.

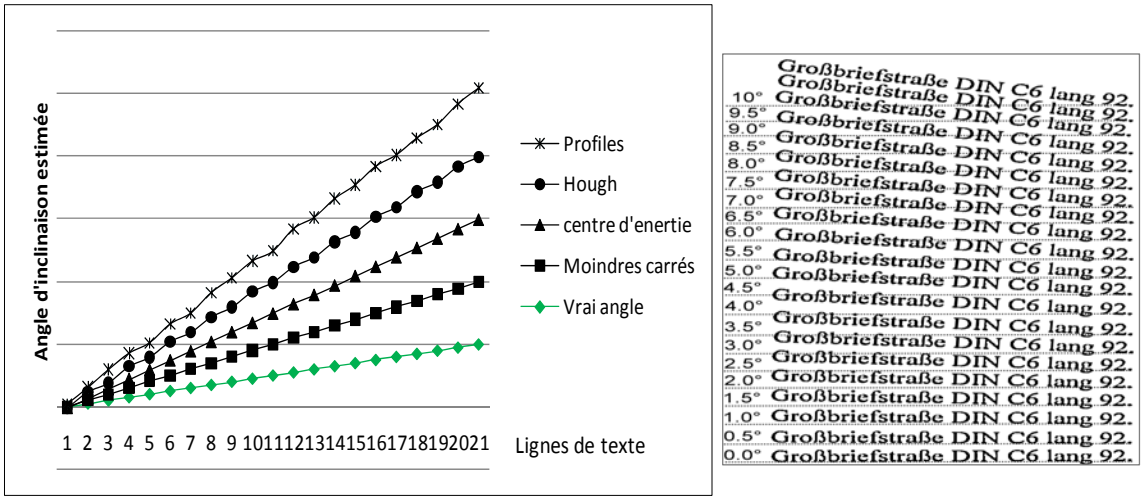


Figure 5. 16 : Comparaison de notre méthode basée sur la technique de moindres carrés avec plusieurs méthodes d'estimation de l'inclinaison par rapport à *21* inclinaisons de référence.

Le temps estimé pour l'exécution de chacune de ces méthodes et l'erreur angulaire moyenne sont représentés dans le tableau suivant :

Méthode	Erreur moyenne (Δθ°)	Temps de calcul (ms)
Moindres carrés	-0.060	0.3
Centre d'inertie	+0.075	0.55
Hough	+0.240	238.60
Projection de profils	+0.327	500.57

Tableau 5.2 : Erreurs moyennes et temps de calcul des trois méthodes comparées à notre méthode.

Ces résultats montrent que notre méthode basée sur la technique des moindres carrées donne une meilleure précision, avec des temps globalement plus faibles que les trois autres approches testées : méthode du centre d'inertie, méthode de Hough et méthode de projection des profils. De plus, les tests que nous avons réalisés ont été élargis avec succès sur une base de 1635 lignes d'adresses imprimées avec un taux de bonne estimation égale à 99,98%. Sur une autre base de 90 lignes d'adresses manuscrites, elle a fourni un taux de bonne estimation égale à 99,95%.

5.3.3.1.5 Redressement des lignes par rotation inverse en trois passes

Immédiatement après l'étape de localisation des zones d'intérêt, on utilise l'angle d'inclinaison calculé par la méthode des moindres carrés pour redresser les lignes par rotation inverse.

Pour garantir un traitement temps réel, nous devons choisir un algorithme de rotation discrète performant et rapide.

Les techniques de redressement basées sur la rotation euclidienne classique à une seule passe, transformation basée sur la matrice suivante :

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \quad (5.\ 19)$$

présentent un inconvénient majeur dû à la non bijectivité de rotation sur des images discrètes. Le calcul de rotation inverse réalisé sur les caractères pour les redresser conduit à des dégradations sous forme des trous, ou à l'apparition de phénomène de crénelage. Or, il est important de noter que la qualité d'une méthode de rotation se mesure directement à partir de la qualité de l'image redressée et sur le fait qu'elle doit comporter un minimum de distorsions et de trous.

Les limites des approches procédant en une opération unique de rotation inverse sur les images et qui sont productrices de phénomènes indésirables (tels que les trous, les effets de crénelage) ont été à l'origine l'objectif de nombreux travaux de recherche dans ce domaine.

Parmi eux, on peut citer les approches alternatives proposées par implémentation hardware. On en distingue trois principales.

- Une première approche de rotation, dite classique, utilise des multiplicateurs et une table de recherche (look-up) contenant des fonctions trigonométriques. Sa réalisation en *hardware* nécessite un matériel spécifique pour garantir vitesse et précision, ce qui en pratique n'est pas réalisable.

- Une version plus efficace remplaçant les multiplications des réels par des additions d'entiers a été proposée dans [BHA96]. La deuxième approche repose sur l'usage de l'algorithme CORDIC[6] mis au point Volder [VOL59] pour approximer des fonctions trigonométriques par des opérations élémentaires (additions, soustractions et multiplications). Cet algorithme a été utilisé dans des architectures parallèles et linéaires intégrant la rotation on ligne des images numériques dans des cartes de type ASIC [GHO95]. Dans cette carte, l'image est divisée en petites fenêtres qui subissent à des rotations parallèles.

- Un autre concept de rotation à haut débit a été proposé dans [SUC04], utilisant cette fois-ci une division hiérarchique de l'image pour augmenter la précision de rotation et la symétrie entre pixels pour réduire les temps. La méthode CORDIC a été simplifiée dans [JIA05] pour s'adapter aux systèmes de ressources limitées. Malgré toutes ces tentatives d'amélioration, on reproche encore à ces méthodes la décomposition en fenêtres qui conduit à des distorsions relativement importantes. La troisième approche introduit une factorisation de la matrice de rotation classique, de sorte que la rotation est effectuée par la mise en œuvre d'une série de translations, [BER98].

Enfin, d'un point de vue implémentation soft, on a pu observer des améliorations des approches à une passe procédant à des rotations composées de plusieurs cisaillements (comme les méthodes de rotation à deux et trois passes). Pour en savoir plus sur ces techniques, on peut citer en références les travaux proposés dans [WOL90], [YAN93], [FRA94], [UNS95], [FLE97], [PAR97], [BER98], [GIB00], [WIL01], [BER02], [SHA03] et [CHE01].

Pour redresser une zone d'intérêt avant de l'envoyer à l'OCR, nous avons choisi d'utiliser la méthode de rotation *en trois passes*. Le principe de cette technique est d'appliquer une rotation en appliquant trois cisaillements à l'image : un cisaillement horizontal, un cisaillement vertical puis à nouveau un cisaillement horizontal. Cette approche permet de réduire considérablement les trous et l'effet de crénelage, elle augmente par conséquent les taux de lecture optique, voir figure 5.17.

6. CORDIC : COordinate Rotation DIgital Computer.

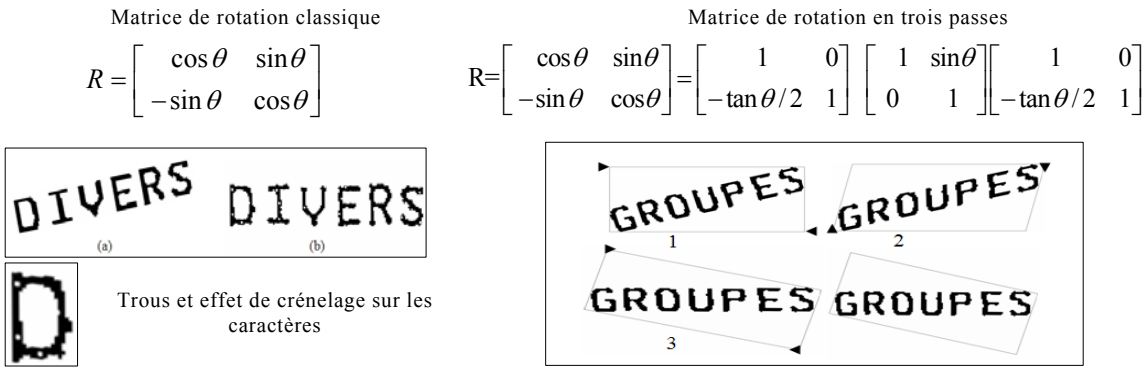


Figure 5. 17 : Exemple de rotation d'image, à gauche rotation classique, à droite rotation en trois passes (cisaillements).

5.3.3.1.5 Résultats de notre approche de redressements des lignes en trois étapes

Pour améliorer la qualité des caractères redressés, nous avons utilisé une interpolation des pixels en fonction de leurs voisins. L'élaboration de plusieurs méthodes d'interpolation (linéaire, b-linaire et PPV) a montré que la technique du plus proche voisin (PPV) est la plus adaptée aux images binaires et à la transformée de type cisaillement.

Le triplet de méthodes :

- les moindres carrés pour déterminer l'angle d'inclinaison
- la rotation en trois passes pour limiter les effets de crénelage et de trous
- l'interpolation par plus le proche voisin qui permet d'améliorer l'apparence des caractères après le redressement.

a été intégré efficacement à l'architecture générale de notre système. Les courbes suivantes présentent les scores de reconnaissances obtenues avant redressement et après redressement. Les tests de notre approche en trois étapes ont été effectués sur une base de *238 lignes* d'adresses (imprimées et manuscrites). Les courbes montrent une augmentation importante des scores de lecture optique sur les *238 lignes*.

Plus précisément, ces courbes montrent des scores très largement améliorés dans les cas où l'inclinaison détectée et corrigée est importante et quasiment identiques lorsque celle-ci est faible. Cela justifie la présence de cisaillement dans l'allure de la courbe. Sur cette figure, les lignes ont été triées par ordre croissant des scores d'OCR sur leur état avant redressement.

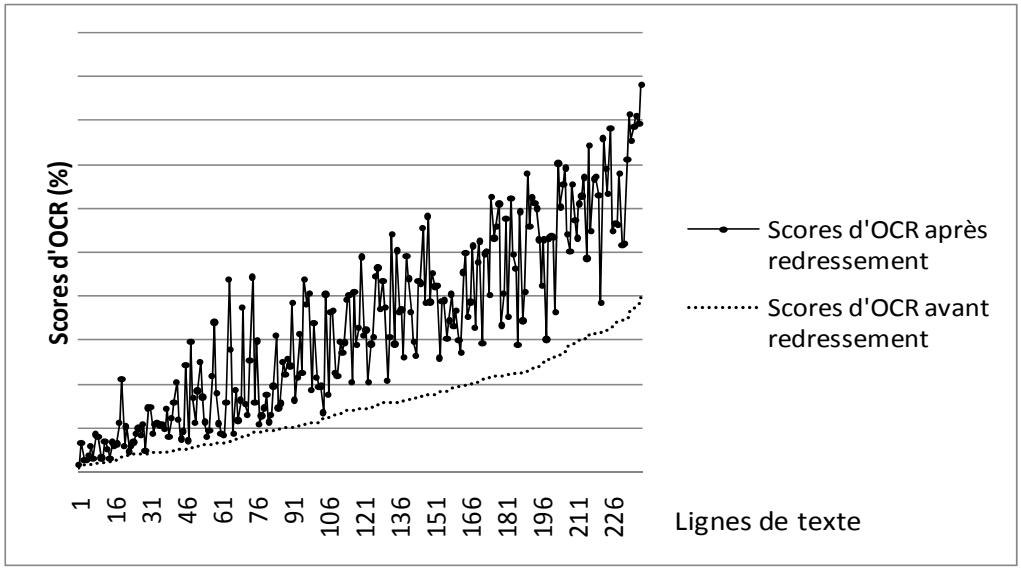


Figure 5. 18 : Scores d'OCR avant et après redressement par notre méthode.

Nous montrons à la figure suivante quelques exemples des lignes d'adresses redressées.

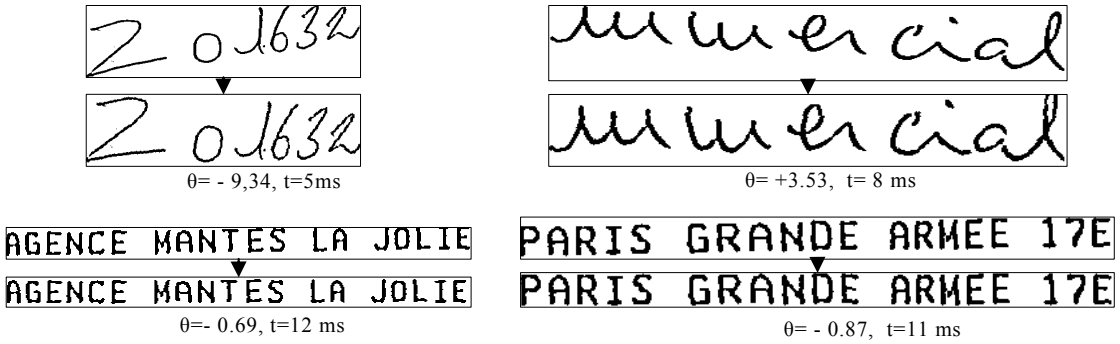


Figure 5. 19 : Exemple de redressement de quelques lignes d'adresse inclinées par notre triplet de méthodes et temps de redressement.

5.3.3.2 Redressement de l'inclinaison des caractères (mode italique)

A l'issue des étapes de traitement de bas niveau, le redressement des lignes de texte inclinées des zones d'intérêt n'est pas le seul type de redressement à envisager pour améliorer la lecture optique des caractères. Un autre type de redressement complémentaire s'avère indispensable : il s'agit du redressement des textes manuscrits penchés ou des textes imprimés italiques où les caractères sont inclinés par rapport à la normale verticale. Nos expérimentations montrent un peu plus loin que cette mise en forme spécifique fait sensiblement diminuer les taux de lecture de l'OCR (par rejet ou

mauvaise décision). Dans le cas du redressement de l'italique, on procède généralement en deux étapes :

- une étape de vérification de l'angle d'inclinaison verticale des caractères qui correspond au biais de l'écriture (appelé slant en anglais) pour décider si le texte est penché ou pas.
- une étape de redressement utilisant une transformation géométrique de type cisaillement par rapport à l'axe des x de la ligne de texte (figure 5.20.c).

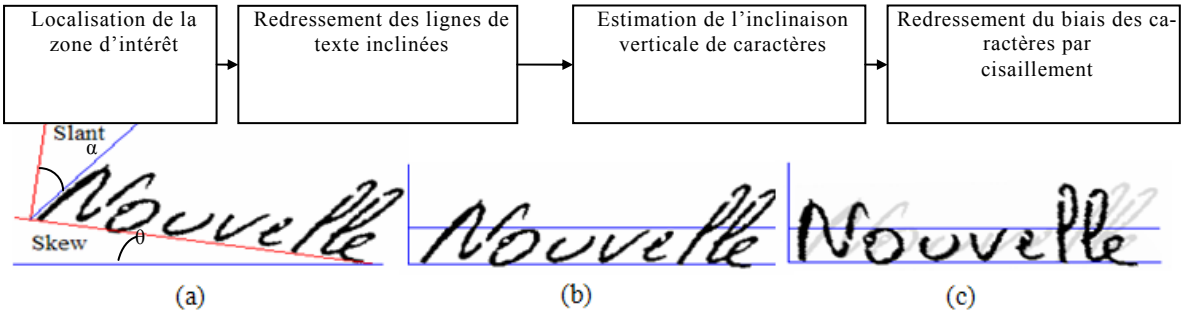


Figure 5. 20 : Étapes de redressement complet de l'inclinaison des lignes de texte (inclinaison globale et biais). **(a)** ligne manuscrite inclinée et penchée. **(b)** premier redressement : estimation et correction de l'inclinaison (par rapport à l'horizontale). **(c)** deuxième redressement : estimation et correction du biais (par rapport à la verticale).

5.3.3.2.1 Estimation du biais des caractères (slant) : une approche structurelle par codage des chaînes de contours

Codage en 4 directions : L'estimation du Slant est une problématique sur laquelle de nombreux auteurs ont déjà travaillé depuis plusieurs années. Cependant, la majorité des approches est fortement basée sur trois techniques :

- la projection de profils [VIN00][VIN02][KAV03],
- le codage des chaînes de contours selon ses directions [DIN00][DIN04][ALC00].
- la densité des orientations du gradient [SUN97].

Très peu de travaux ont, à ce jour, exploité la densité des orientations du gradient. Ces approches s'appliquent uniquement sur des images en niveaux de gris et sont très sensibles aux variations de gradients dues au bruit, aux dégradations et aux changements d'éclairage. Les méthodes basées sur la projection des profils binaires sont bien robustes mais elles nécessitent beaucoup de temps de calcul pour produire les séquences de pro-

jections selon plusieurs directions. Les méthodes basées sur le codage des chaînes respectent mieux le compromis temps/précision : elles semblent ainsi plus adaptées à notre application de tri.

Notre méthode d'estimation du biais des caractères consiste à coder les chaînes de contours des caractères de texte selon quatre directions *(0°, 45°, 90° et 135°)*. Ce codage est utilisé pour calculer l'angle d'inclinaison par rapport à la verticale de chacune des chaînes. Les segments horizontaux de code n_0 ne participent pas au calcul de l'orientation moyenne de chaque chaîne donnée par α_4 (le *-4-* rappelant la prise en compte de *4* directions uniquement) :

$$\alpha_4 = \arctan\left(\frac{n_1 - n_3}{n_1 + n_2 + n_3}\right) \quad (5.\ 20)$$

où n_1, n_2 et n_3 sont les nombres de points de contours correspondent respectivement aux directions *45°, 90° et 135°*.

Les chaînes des contours d'une composante connexe sont formées à partir des points de début et de fin de toutes séquences noires de la structure LAG simplifiée (voir figure 5.21.a et b). Les points de début sont représentés en gris et les points de fin en noir. Les points qui n'ont aucun voisin ne participent pas au calcul des entités n_1, n_2 et n_3. Sur l'exemple de la figure 5.21, on obtient ainsi 7 chaînes contours connexes. Leur codage suit l'ordre de stockage des structure LAG, c'est-à-dire qu'on code chaque chaîne en commençant du haut vers le bas et de gauche à droite. Cela permet de justifier la numérotation produite par le codage de la figure 5.21.c.

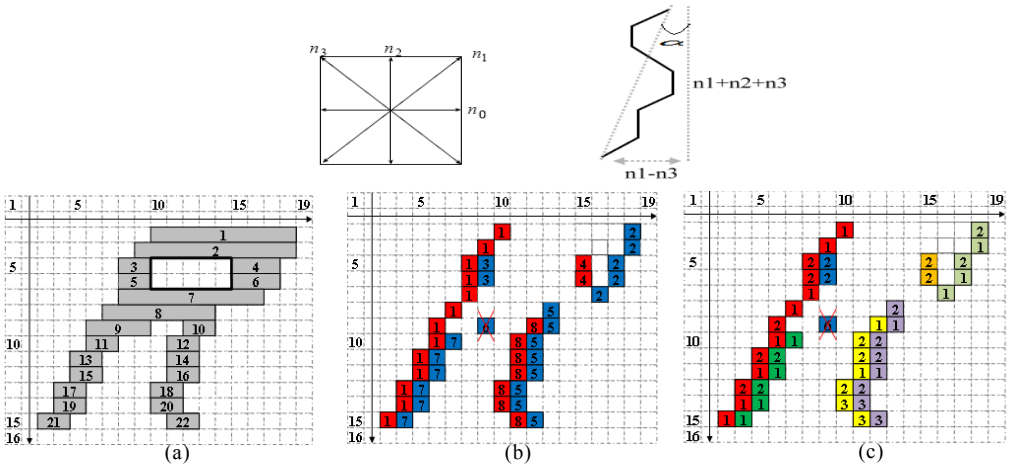


Figure 5. 21 : (a) Structure LAG simplifiée, **(b)** extraction des 7 chaînes de contours à partir de notre structure LAG simplifiée (points de début des séquences noires en rouge et les points de fin en bleu), **(c)** codage des chaînes de contours selon les 4 directions et l'angle d'inclinaison moyenne.

On estime la pente globale (notée α_{cc}) d'une composante connexe définie par la moyenne des angles du biais de toutes ses chaînes, chacune

des chaînes i est pondérée par un poids représenté par sa longueur l_i (cette longueur est égale au nombre de points appartenant à cette chaîne). L'utilisation de ces poids nous évite des distorsions qui peuvent être provoquées par la présence de petites chaînes. Cette pente moyenne peut être estimée par l'expression suivante :

$$\alpha_{cc} = \frac{\sum_{i=1}^{nc} l_i \times \alpha_i}{\sum_{i=1}^{nc} l_i} \quad \text{avec} \quad \alpha_i = \arctan\left(\frac{n_{i1} - n_{i3}}{n_{i1} + n_{i2} + n_{i3}}\right) \quad (5.\ 21)$$

où n_c désigne le nombre de chaînes dans la composante connexe CC. D'une manière similaire, on estime la pente d'une ligne par la moyenne des pentes des CCs qui lui appartiennent.

Généralisation du codage en 8, 12 et 16 directions : En général, le biais varie toujours dans l'intervalle *[-45, +45°]*, dans ce cas le codage en *4* directions donne une très bonne estimation. S'il y a une grande quantité de chaînes qui possèdent une direction en dehors de cet intervalle (cas rares en pratique et jamais rencontrés dans nos tests), alors la valeur du biais moyen estimé à partir d'un ensemble de petites chaînes demeure inférieur à ce qu'il est en réalité [DIN04]. L'augmentation de la résolution angulaire en *8*, en *12* ou en *16* directions peut être la solution à envisager (voir figure 5.22).

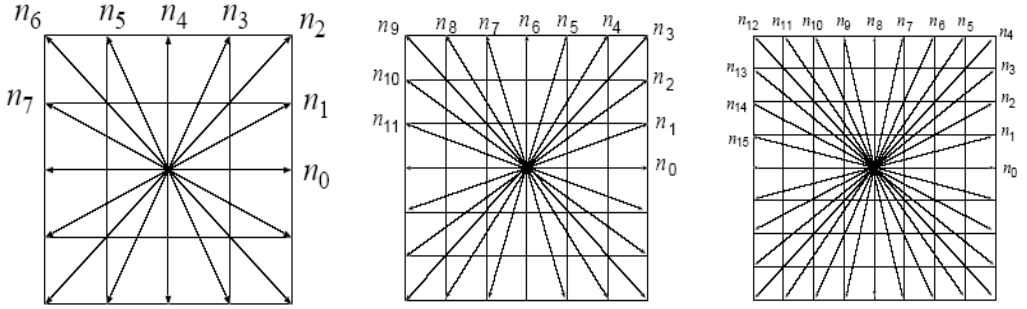


Figure 5. 22 : Multiplication de secteurs angulaires envisagés pour le codage des contours des caractères manuscrits.

Les angles d'inclinaison pour les différentes résolutions sont donnés par les expressions suivantes :

$$\alpha_8 = \arctan\left(\frac{(2n_1 - 2n_2 + n_3) - (n_5 + 2n_6 + 2n_7)}{(n_1 + 2n_2 + 2n_3) + 2n_4 + (2n_5 + 2n_6 + n_7)}\right) \quad (5.\ 22)$$

$$\alpha_{12} = \arctan\left(\frac{(3n_1 + 3n_2 + 3n_3 + 2n_4 + n_5) - (n_7 + 2n_8 + 3n_9 + 3n_{10} + 3n_{11})}{(n_1 + 2n_2 + 3n_3 + 3n_4 + 3n_5) + 3n_6 + (3n_7 + 3n_8 + 3n_9 + 2n_{10} + n_{11})}\right) \quad (5.\ 23)$$

$$\alpha_{16} = \arctan\left(\frac{(4n_2 + 4n_3 + 4n_4 + 3n_5 + 2n_6 + n_7)(n_9 + 2n_{10} + 3n_{11} + 4n_{12} + 4n_{13} + 4n_{14})}{(2n_2 + 3n_3 + 4n_4 + 4n_5 + 4n_6 + 4n_7) + 4n_8 + (4n_9 + 4n_{10} + 4n_{11} + 4n_{12} + 3n_{13} + 2n_{14})}\right) \quad (5.\ 24)$$

Dans la méthode « des 16 directions », les directions n_1 et n_{15} n'ont pas été prises en compte pour éviter toute sur-estimation du biais. Dans chacune de ces méthodes, toute chaîne de longueur inférieure à *5* ne participe pas au calcul du biais. Cette longueur est limitée à *3* pour la méthode « des *8* directions » et à *4* pour la méthode « des *12* directions ».

Nous avons testé notre méthode d'estimation du biais des caractères selon l'algorithme des « *4* directions » sur une base de *87 lignes* de texte. Cette base regroupe des lignes de texte imprimé en italique et d'autres de texte manuscrit penché. Chaque ligne de texte est lue par l'OCR avant et après sa correction. Les deux courbes de la figure suivante montrent les scores de reconnaissance obtenus sur les lignes de texte avant et après redressement de biais des caractères. Comme dans le cas de l'inclinaison des lignes complètes, ces courbes montrent que ce type de redressement augmente sensiblement les scores de l'OCR. Plus précisément, ces courbes montrent des scores très largement améliorés dans les cas où le biais détecté et corrigé est important et quasiment identiques lorsque celui-ci est faible. Cela justifie l'allure très ciselée de la courbe. Sur cette figure, les lignes ont été triées par ordre croissant des scores d'OCR sur leur état avant redressement.

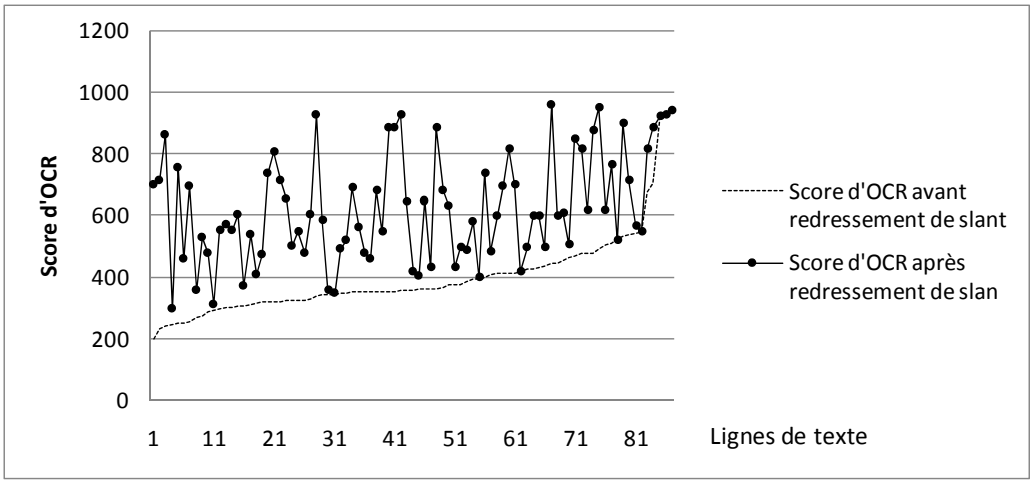


Figure 5. 23 : Comparatif de score de l'OCR avant et après redressement de l'inclinaison des caractères (penchés ou italiques) sur une base de 87 lignes de texte manuscrit penché et imprimé en italique.

5.3.3.2.2 Redressement des lignes de texte italique ou penché par cisaillement

Après l'estimation de l'angle de biais α_4 d'une ligne de texte, on regarde s'il faut appliquer une transformation horizontale inverse de type cisaillement d'angle $-\alpha_4$ pour rendre aux caractères leur verticalité. Par cette transformation linéaire, chaque séquence noire r_i est translatée sur l'axe des x par la distance :

$$d = Entier\{[y(ri) - y_c] \times \tan(\alpha) + 0.5\} \quad (5.\ 25)$$

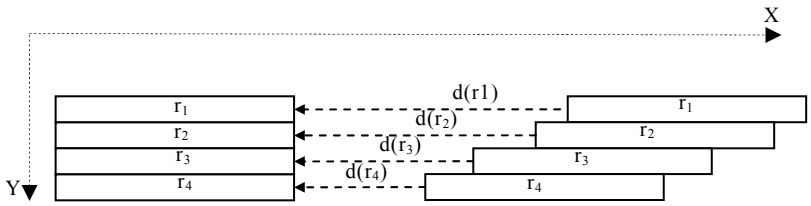


Figure 5. 24 : Déplacement des séquences noires par cisaillement.

Ceci signifie que chaque pixel noir de position *(x,y)* est translaté vers la nouvelle position *(x', y)* par l'équation suivante :

$$x' = x + d \quad (5.\ 26)$$

où y_c est l'ordonné de centre de gravité de la ligne de texte à redresser. La figure suivante montre quelques lignes d'exemple de texte italique redressées par cisaillement après l'estimation de leur angle de biais par la méthode de *4* directions.

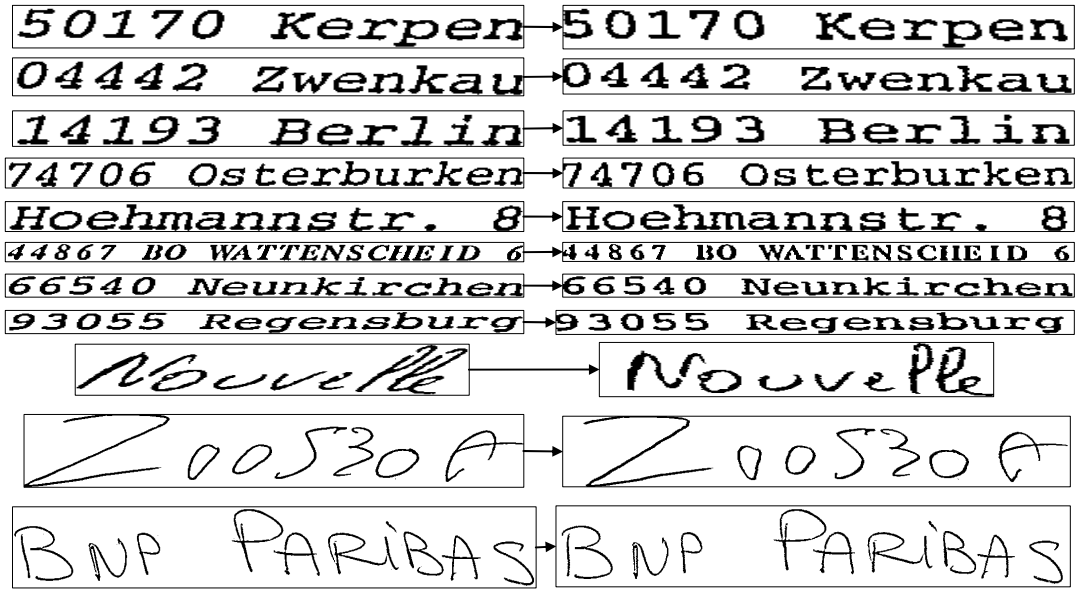


Figure 5. 25 : Résultat de redressement du biais des caractères, **à gauche** : lignes de texte penché, **à droite** : lignes de texte redressées par notre méthode.

5.4. Analyse de la structure physique par coloration hiérarchique de graphes

L'analyse de la structure physique des images de courriers repose sur une coloration à trois niveaux de la pyramide des données

- au niveau des connexités pour permettre une séparation et un marquage en régions textuelles et graphiques (première coloration). Cette coloration s'applique directement sur les connexités et le premier graphe coloré se construit à partir de la quantification de dissimilarités morphologiques et de différences de densités entre composantes.

- au niveau des connexités marquées par le premier niveau de coloration pour permettre l'émergence des lignes séparées les unes des autres selon des critères de position, d'orientation et selon un ensemble de caractéristiques morphologiques et géométriques extraites pour chaque connexité.

- au niveau des lignes pour permettre la formation de blocs de textes homogènes, incluant des lignes d'orientation similaire et tenant compte de la proximité spatiale et de la similarité morphologique des lignes.

La deuxième coloration constitue une étape pivot essentielle, car c'est à ce stade que l'on peut affiner le redressement d'orientations des lignes par une localisation précise des emplacements que seule la deuxième coloration peut apporter.

Cette stratégie pyramidale apporte à la segmentation toute la puissance d'un classifieur par coloration des graphes permettant de distinguer les éléments pertinents et de les regrouper en ensembles homogènes tout en écartant les composantes parasites présents dans les images difficiles (en général là où les éléments graphique peuvent être très proches du texte, ou encore là où certaines annotations peuvent chevaucher le texte…). Dans cette section nous présentons en détails notre nouvelle architecture pyramidale de l'extraction de la structure physique des documents, que nous nommons également segmentation en couches des documents. En plus de l'optimisation en temps apportée à chacune de ces étapes de segmentation par la coloration de graphes, nous avons conçu cette architecture de façon à éviter de faire des tâches redondantes et à réduire ainsi, au maximum, les temps de calcul.

5.4.1 Les composants du système nécessaires à l'analyse de la structure physique

Nous avons vu dans le chapitre *2* que l'extraction de la structure physique est une étape incontournable qui doit précéder les phases de reconnaissance et d'interprétation dans un système de tri automatique de documents et de courrier. Son rôle consiste à découper une image de document en blocs homogènes au sens d'un critère donné.

Nous avons vu que la qualité d'une structuration en blocs d'un document dépend fortement de la complexité de disposition des objets et de la présence d'objets parasites au voisinage des blocs de texte. Ceci signifie que les connaissances issues des primitives de description des blocs ne peuvent pas satisfaire la description de blocs hétérogènes contenant des éléments parasites ou des blocs mal découpés. Nous avons expliqué au chapitre 2 en quoi les méthodes mixtes (mi-ascendantes mi-descendantes) donnaient les meilleurs compromis temps-précision. Cependant, elles restent insuffisantes lorsque les images à analyser présentent de nombreuses irrégularités de composition. Notre architecture vise à compenser les insuffisances de ces approches conventionnelles en proposant une méthodologie d'analyse et de structuration des contenus des documents selon un mécanisme original de regroupement hiérarchique des connexités basés sur la coloration de graphe. Nous allons notamment montrer que la qualité de la segmentation peut être très largement améliorée par rapport aux approches conventionnelles à la fois en précision de découpage et en robustesse face à la complexité et l'irrégularité des mises en page.

La figure 5.26 illustre les différentes étapes du processus de structuration des images de courriers d'entreprise que nous proposons et qui correspondent à trois niveaux de coloration des graphes. Nous avons également montré le lien qu'entretient ce processus avec la segmentation «brute» des contenus procédant par binarisation des régions d'intérêt et extraction des formes connexes.

La séparation des couches d'informations textuelles et non textuelles intervient en premier lieu au niveau de la première coloration de graphe. Au fur et à mesure des colorations on passe successivement d'une information brute de connexités simplement caractérisés par des primitives morphologiques et géométriques à une information structurée de contenus regroupés en lignes et en blocs dont les contenus textuels sont complètement caractérisés (zones de textes manuscrits, imprimés, graphiques et logos). Les phases successives d'analyse jouent un rôle de filtrage et de fusion et séparation des différents éléments participant à la description de la structure du document.

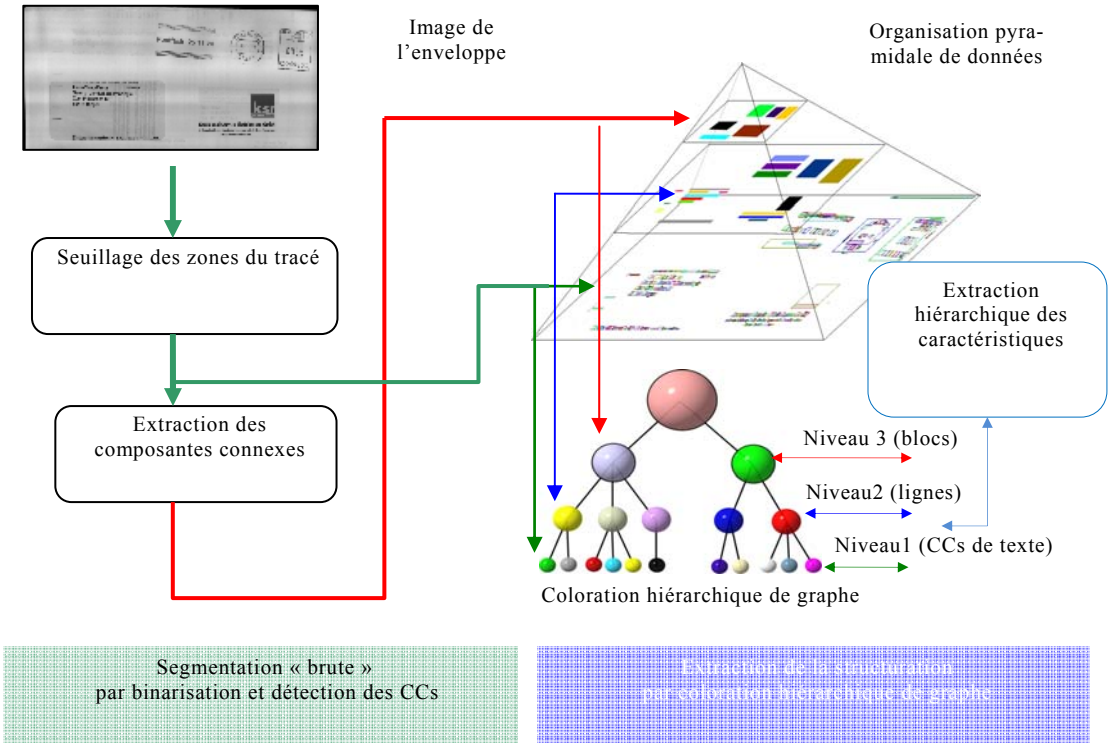


Figure 5. 26 : Schéma fonctionnel des différentes étapes de structuration des images de documents mettant en évidence les trois niveaux de colorations de graphe.

Rappelons ici que la segmentation « brute » telle qu'elle a été présentée à la section précédente permet d'obtenir deux cartes issues de l'image d'origine : l'image binaire et la carte des blocs. Cette partie se poursuit avec une étape de détection de connexités sur les deux cartes par une méthode que nous avons présentée précédemment. Cette étape conduit à la formation de deux ensembles de composantes connexes avec leurs caractéristiques : l'ensemble CC_B des CCs de l'image binaire B avec $CC=\{cc_i,\ i=1...n_{cc}\}$ et l'ensemble CC_M des CCs de la carte des blocs FM avec $CC_M=\{FM_i,\ i=1...n_M\}$. La carte M n'est pas la carte des blocs définitifs, car elle contient des blocs séparés selon leurs dispositions spatiales sans aucune garantie d'homogénéité. Cependant, cette carte sera utilisée pour accélérer la séparation texte-non texte par une première coloration de graphe qui servira par ailleurs de repères pour renforcer la décision de LBA.

La partie de structuration proprement dite se distingue de la segmentation « brute » par le fait qu'on cherche à induire des connaissances plus évoluées sur la nature des contenus conduisant à un découpage du document en régions homogènes de lignes et de blocs structurés informants. Cette partie que nous détaillons dans cette section repose sur une structure

pyramidale de données et une coloration hiérarchique de graphe. La pyramide des données se compose de trois niveaux de fin à grossier : le niveau des connexités, le niveau des lignes, le niveau des blocs, voir figure 5.26.

5.4.2 Les différents niveaux de coloration et de structures

La construction des trois niveaux de la pyramide (connexités, lignes de texte, blocs) est gérée et contrôlée par une coloration hiérarchique des graphes. Pour cela on utilise trois colorations en cascade où chacune d'elle reprend la formalisation et les notions présentées dans le chapitre 4 portant sur la coloration. A chaque coloration on applique alors notre algorithme de coloration minimale (présentée à la section 4.3.1) qui détermine à chaque fois un nombre de couleurs non fixé a priori. Le principe de coloration se charge de produire un nombre de couleurs proche de l'optimal. Rappelons que le processus de coloration de graphe que nous avons choisi est résolu par l'élaboration de stratégies heuristiques basées sur des valeurs de seuils (que nous avons rassemblés sous le terme de dissimilarité) portant sur différentes caractéristiques décrivant les sommets.

5.4.2.1 La première coloration réalisée pour la séparation texte/non texte

Dans le chapitre précédent, nous avons présenté séparément les problèmes de séparation des blocs en texte/non texte et de séparation de texte en imprimé / manuscrit, puis comparé pour chaque problème plusieurs méthodes de la littérature afin de mettre en évidence les méthodes les plus simples et les plus efficaces en temps et en performance. A partir de cette étude bibliographique nous avons mis en évidence les caractéristiques communes régissant les deux types de séparation. Nous avons également vu que les méthodes de séparation en trois classes (texte imprimé, texte manuscrit, objets non textuels) étaient très prometteuses, voir figure 5.27. En ce sens, nous avons mis en évidence la nécessité de traiter les deux problèmes de séparation conjointement avec le même jeu de caractéristiques permettant de s'adapter aux contraintes de temps réel de notre application.

Notons que la séparation des parties *texte manuscrit / texte imprimé* peut être améliorée et pleinement étiquetée par l'utilisation du mécanisme d'apprentissage par b-coloration que nous aborderons à la dernière section de ce chapitre.

Nous allons présenter dans cette section les différentes caractéristiques morphologiques que nous pouvons exploiter pour permettre une telle séparation d'information au premier niveau de la hiérarchie. Cette extraction de caractéristiques offre l'avantage de pouvoir se calculer durant l'étape de capture des connexités. Ces caractéristiques sont basées sur des mesures quantifiant la régularité des dispositions des composantes.

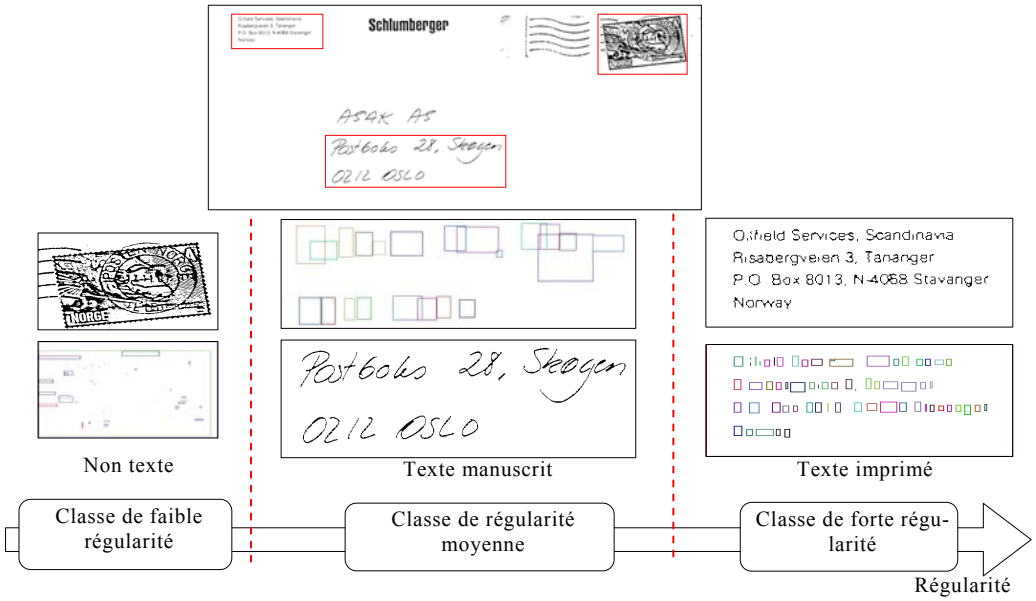


Figure 5. 27 : Principe de séparation en trois classes (non texte, texte manuscrit et texte imprimé) selon la régularité des blocs.

5.4.2.1.1 Choix des caractéristiques de coloration de premier niveau

Les caractéristiques morphologiques se calculent pour chaque groupe (couleur issue de la première coloration) de composantes connexes C_{cc}. Nous avons choisi de retenir les primitives suivantes :

- les densités moyennes d'un groupe de CCs.
- la moyenne et la variance des largeurs (μ_W, σ_W) des CCs, des hauteurs des CCs et (μ_H, σ_H), des excentricités $(\mu_\zeta, \sigma_\zeta \,|\zeta=W/H)$ des CCs, et des surfaces des CCs $(\mu_{Ar}, \sigma_{Ar} \,|Ar=H\times W)$ des CCs, où H et W désignent la hauteur et largeur respective de chaque composante connexe.

La densité des connexités est donnée par la formule suivante :

$$DT(C_{cc}) = \frac{1}{n}\sum_{i=1}^{n} Dens(CC_i) \qquad (5.\ 27)$$

Où CC_i est la $i^{éme}$ composante connexe de groupe C_{cc}.

Les moyennes sont données par les équations suivantes :

$$\mu_W = \frac{1}{n}\sum_{i=1}^{n} W(CC_i),\ \mu_H = \frac{1}{n}\sum_{i=1}^{n} H(CC_i),$$
$$\mu_\zeta = \frac{1}{n}\sum_{i=1}^{n} \frac{W(CC_i)}{H(CC_i)},\ \mu_{Ar} = \frac{1}{n}\sum_{i=1}^{n} W(CC_i)\times H(CC_i) \qquad (5.\ 28)$$

Et les écarts types sont donnés par les équations suivantes :

$$\sigma_W = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(W(CC_i)-\mu_W\right)^2},\ \sigma_H = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(H(CC_i)-\mu_H\right)^2},$$
$$\sigma_\zeta = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(\zeta(CC_i)-\mu_\zeta\right)^2},\ \sigma_{Ar} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(S(CC_i)-\mu_{Ar}\right)^2} \quad (5.\ 29)$$

L'observation des images de courriers que nous avons étudiées nous a conduit à constater que les tailles des CCs de texte imprimé étaient plus régulières que celles de texte manuscrit et les écarts σ_W et σ_H étaient plus petits.

5.4.2.1.2 Construction du premier graphe et définition de l'indice de dissimilarité

La première coloration est appliquée sur la carte de n_{cc} composantes connexes brutes (niveau des éléments symboliques) pour réaliser une séparation *texte / non texte* (graphique, bruit, tableau, photos,...). Elle permet de regrouper les CCs de manière à isoler les CCs textuelles dans des couleurs de plus forte densité qui contiennent un grand nombre de composantes par rapport aux autres selon un seuil de densité S_{cc} défini ci-dessous. Cela suppose l'élimination préalable des composantes graphiques apparaissant près du texte ou superposées au texte. Le vecteur descripteur de chaque CC est basé tout simplement sur cinq caractéristiques morphologiques (sa densité, sa hauteur, sa largeur, son excentricité et sa surface) estimées lors de détection des connexités. Ces caractéristiques permettent de distinguer les CCs uniquement selon leur forme. A ce stade, ni la position, ni les relations de voisinage spatial des CCs ne sont encore prises en compte. Après la normalisation de ces caractéristiques, le calcul de la matrice de dissemblance M_d^{cc} entre les CCs repose sur la distance Manhattan (aussi appelée distance « city-block » ou métrique absolue notée d_{cb}), avec :

$$M_d^{cc}[i,j] = d_{cb}(\ cc_i,\ cc_j\ ) = \sum_{p=1}^{5}\left|cc_i(p)-cc_j(p)\right| \ avec\ (j>i)\ \in [1,n_{cc}] \quad (5.\ 30)$$

Chaque *(cc_i, cc_j)* constitue une paire de composantes connexes, et *p* représente une des cinq caractéristiques. Pour évaluer les adjacences entre connexités nous avons choisi cette distance Manhattan car elle évite de nombreux calculs (multiplications et racines carrés exigées dans la distance euclidienne par exemple). Le choix des seuils de dissimilarité qui sont ensuite utilisés ne dépend pas de la mesure choisie.

Les cinq caractéristiques ont des dispersions différentes, l'utilisation des variables telles quelles dans le calcul de la distance donnera de façon implicite plus de poids aux variables de plus forte dispersion, annihilant presque complètement l'effet des autres. Une approche classique pour obtenir une distance ne privilégiant pas uniquement les variables de forte dispersion est de standardiser les composantes caractéristiques par ca-

ractéristique : chaque caractéristique p doivent être centrées par la moyenne μp^p (ou encore la médiane) et réduite (ou normalisée) par l'écart-type σp^p calculées indépendamment de notre processus de segmentation sur base représentative de composantes connexes. Or lorsque on utilise la distance euclidienne ou de Manhattan il devient inutile de faire le centrage des caractéristiques. La dissimilarité d_{cb} devient :

$$d_{cb}(\, cc_i,\, cc_j\,) = \sum_{p=1}^{5} \frac{1}{\sigma P} \left| cc_i(p) - cc_j(p) \right| \; avec \; (j > i) \in [1, n_{cc}] \qquad (5.\,31)$$

Chaque caractéristique est normalisée de la façon suivante :

$$Dens = \frac{1}{\sigma 1} Dens, H = \frac{1}{\sigma 2} H, W = \frac{1}{\sigma 3} W, \zeta = \frac{1}{\sigma 4} \zeta, Ar = \frac{1}{\sigma 5} \qquad (5.\,32)$$

Le premier graphe $G_{cc}(V_{cc}, E_{>Scc})$ est construit de la façon suivante :

a) l'ensemble des sommets V_{cc} correspond à l'ensemble des descripteurs des n_{cc} CCs avec :

$$V_{cc} = \{ v_i^{cc} \Leftrightarrow cc_i(Dens, H, W, \zeta, Ar) \,|\, i = 1 \ldots n_{cc} \} \qquad (5.\,33)$$

b) l'ensemble des arêtes $E_{>Scc}$ est déduite de la matrice de dissemblance M_d^{cc} par la relation suivante :

$$E_{\geq Scc}\left[v_i^{cc}, v_j^{cc} \right] = \begin{cases} 1 & \text{si } \mathrm{M}_d^{cc}(i,j) \geq S_{cc} \\ 0 & \text{sinon} \end{cases} \qquad (5.\,34)$$

Le seuil S_{cc} est ajusté de façon expérimentale à partir des connaissances extraites sur la mise en forme des textes dans nos documents (comme le type et la taille de la police de texte imprimé, ou encore la hauteur des connexités des textes manuscrits). Il est également possible d'ajuster ce seuil automatiquement à partir de la base de composantes étiquetées *texte / non-texte* reprenant le mécanisme décrit dans le chapitre 4 (section 4.4.2.1).

5.4.2.1.3 Application du premier niveau de coloration

Une fois le premier graphe construit, on applique l'algorithme de coloration propre donné dans la procédure 7 du chapitre 4 pour obtenir un ensemble C_{cc} de k_{cc} couleurs. Rappelons ici que le premier niveau de coloration se fonde sur un nombre de couleurs non fixé à priori et qu'il se base sur des contraintes morphologiques exclusivement (sans aucune prise en compte d'information de voisinage à ce stade). En se basant sur des valeurs de densité (portant sur l'élaboration de la valeur de seuil S_{Dens}) et sur des critères de dispersion (portant sur l'estimation d'un seuil unique $S_{\sigma t}$), on parvient à réaliser la séparation entre les éléments exclusivement textuels et graphiques. La valeur de k_{cc} *d*épendra donc de la nature du contenu de

chaque image de documents : les composants graphiques (ou textuelles) peuvent donc apparaître de couleurs différentes, voir figure 5.28.

L'algorithme de coloration utilisé fournit une coloration propre (c'est-à-dire que deux couleurs adjacentes n'ont pas la même couleur). Cette contrainte se focalise sur les dissimilarités pour mettre en évidence des groupes de composantes parfaitement homogènes d'une manière automatique, souple et efficace. Cet avantage exclusivement intrinsèque à la coloration est lié au découpage précis selon les dissimilarités entre CCs puis au regroupement reposant sur la contrainte de propreté de coloration. Cette dernière assure qu'aucune des couleurs ne puisse correspondre à une composante parasite. En effet, tout regroupement local des sommets du graphe en une même couleur est soumis à une expertise globale au sens des adjacences du graphe. En pratique cette analyse globale des adjacences de tout nouveau sommet à colorer garantit que la coloration ne puisse pas laisser de place à une composante intruse (non similaire) au sein d'un groupe de composantes de propriétés morphologiques différentes. Plus précisément, rappelons qu'avant d'affecter la couleur à un sommet *i* au sommet *j*, il faut vérifier si globalement il n'y a pas de conflit d'adjacence ce qui revient à analyser les couleurs de toutes leurs composantes dissimilaires (sommets adjacents) au sommet *j* et vérifier qu'aucune d'elles n'est égale à la couleur *i*.

Ainsi toute décision de fusion locale d'un sommet d'une couleur en une autre est conditionnée par l'analyse complète globale des couleurs des sommets qui lui sont directement adjacents.

Ceci signifie que chaque couleur représente des CCs de même taille et de même densité. Par exemple, deux CCs de même taille mais d'épaisseur différente ne peuvent pas être regroupées dans la même couleur. Cette propriété importante pour toute segmentation permet de mettre facilement et automatiquement en évidence les différentes structures présentes sur la carte des symboles quelque soit la complexité de cette dernière et indépendamment de la résolution de l'image (voir la figure 5.28).

L'ensemble des couleurs de texte, noté C_{cct}, est ensuite très facilement déduit à partir de l'ensemble C_{cc} par la relation suivante :

$$C_{cct} = C_{cct} \cup \{c(i) \in C_{cc} \mid i = 1...k_{cc}\} \text{ si } \begin{cases} \sigma_T[c(i)] < S_{\sigma T} \\ et\ Dens[c(i)] > S_{Dens} \\ et\ |c(i)| > S_N \end{cases} \quad (5.\ 35)$$

S_N est un seuil sur le nombre de composantes connexes (utilisé pour exclure de C_{cct} toute couleur qui ne contient pas un nombre de CCs suffisant). Avec :

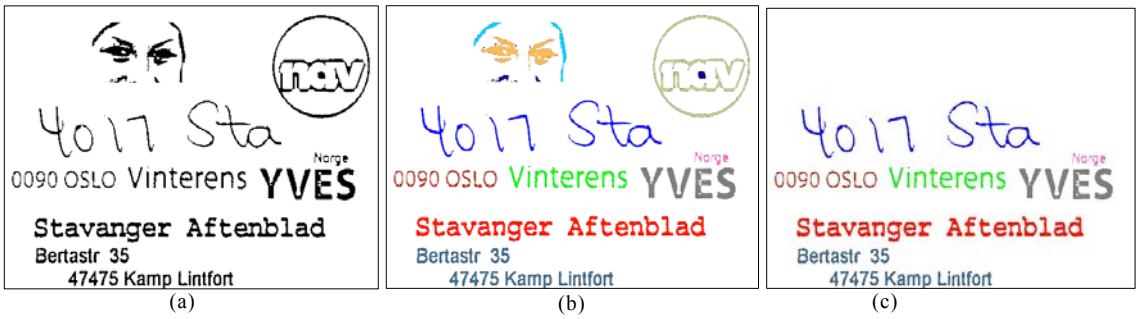
$$\sigma_T[c(i)] = \sigma_H(i) + \sigma_W(i) + \sigma_\zeta(i) + \sigma_{Ar}(i) \mid i = 1...k_{cc} \quad (5.\ 36)$$

Ceci devient :

$$\sigma_T[c(i)] = \frac{1}{|c(i)|}\sqrt{\sum_{k=1,\ cc_k \in c(i)}^{|c(i)|}\left(\left(W(cc_k)-\mu_W\right)^2+\left(H(cc_k)-\mu_H\right)^2+\left(\zeta(cc_k)-\mu_\zeta\right)^2+\left(Ar(cc_k)-\mu_{Ar}\right)\right)^2} \quad (5.37)$$

Où $|c(i)|$ indique le nombre de CCs qui ont la couleur *c(i)*, σ_W, σ_H, σ_ζ et σ_{Ar} sont les écarts types qui représentent la dispersion des CCs de la couleur c(i) autour de leur moyenne par rapport aux caractéristiques normalisées $(Dens, H, W, \zeta, Ar, \sigma_W, \sigma_H, \sigma_\zeta$ et $\sigma_{Ar})$. $Dens[c(i)]$ est la densité de la couleur *c(i)* qui peut être calculée de la façon suivante :

$$Dens[c(i)] = \frac{\sum_{k=1,\ cc_k \in c(i)}^{|c(i)|} Dens(cc_k)}{\sum_{k=1,\ cc_k \in c(i)}^{|c(i)|} H(cc_k) \times \sum_{k=1,\ cc_k \in c(i)}^{|c(i)|} W(cc_k)} \quad (5.38)$$



(a) (b) (c)

Figure 5. 28 : (a) Image binaire, **(b)** Exemple de séparation de différentes structures et résultat de la première coloration (10 couleurs) sur la carte des composantes connexes basée sur des caractéristiques morphologiques sans aucune contrainte de voisinage, **(c)** sélection des couleurs de texte avec un simple critère (S_{ot} =*80* et S_{Dens}=*0.20*, S_n=5).

Les n_{cct} composantes (ou caractères) qui appartiennent aux couleurs de texte seront préservées dans l'ensemble *CCt* formant le premier niveau de la pyramide de données. Ces composantes de texte représentent les sommets de deuxième graphe noté G_{cct} .

Remarque 1 : Afin de réduire les temps de traitement d'une façon importante, nous avons remplacé la coloration globale par plusieurs colorations locales en appliquant indépendamment et localement la procédure *7* sur les CCs dans chaque bloc FM_i de masque *M* représenté par l'ensemble $CC_M=\{FM_i,\ i=1...n_M\}$. Les FM_i sont les CCs de la carte *FM* définie dans la section 5.3.2. Le résultat global final C_{cc} est égal à l'union de toutes les couleurs locales. Le seul inconvénient de la coloration locale est qu'elle n'intègre que des connaissances portées par les adjacences locales et aucune visibilité réellement globale n'est apportée par une telle approche.

Cependant à ce niveau, ce n'est pas pénalisant car cette approche de la coloration n'intervient qu'au niveau de la séparation texte/ non texte et qu'au niveau local elle évite la fusion de connexités proches de nature différente. Elle contribue donc finalement à une réelle amélioration de l'étiquetage des connexités.

Le principe de la coloration locale est de garantir une séparation *texte/ non texte* localement pour éviter de comparer les composantes non textuelles avec les composantes de tous les autres blocs (ce qui se produit lors de la coloration globale). On gagne ainsi en temps de calcul et à l'issue de la coloration du graphe on fusionne les sommets de couleurs « textuelles ». Ainsi, au lieu de colorer toutes les composantes connexes de l'image (représentées par un seul graphe) on va appliquer une coloration locale (dans chaque bloc de masque *M*). On aura ainsi autant de colorations locales que de blocs du masque *M*.

Remarque 2 : Les deux seuils $S_{\sigma t}$ et S_{Dens} peuvent être calculés automatiquement par la méthode d'évaluation non supervisée qu'on a présentée dans le chapitre 4 portant sur la coloration. Dans ce travail appliqué aux images de courriers de l'entreprise, nous avons choisi les valeurs de seuils empiriquement car nous disposons d'un ensemble de connaissances sur les mises en forme de ces images de courriers. Cependant, nous avons montré qu'il était tout à fait envisageable d'automatiser les valeurs de seuils. Pour généraliser la méthode, nous allons précisément exploiter les résultats d'apprentissage par b-coloration sur les couleurs d'une grande base d'images afin de séparer automatiquement les couleurs en trois classes : deux classes de texte (*imprimé ou manuscrit*) et une classe couleur non texte. Le seuil S_N est en général supérieur à *20* (pour le courrier).

5.4.2.2 La deuxième coloration pour la formation des lignes de texte

5.4.2.2.1 Choix des caractéristiques

Ce niveau se caractérise par l'introduction de l'information de position des composantes de texte. Celles-ci, issues de la première coloration deviennent les briques de base pour construire les lignes de texte (constituant le deuxième niveau de la pyramide). Cette construction se base sur une deuxième coloration de graphe. Ce graphe est noté G_{cct} (cct pour composante connexe de texte). Pour cela, nous construisons un deuxième graphe noté $G_{cct}(V_{cct}, E_{>Scct})$ à partir de toutes les composantes de texte extraites du premier niveau de la pyramide de la façon suivante : on représente chaque composante de texte $cct(i) \in CCt \,|\, i = 1...n_{cct}$ par un sommet de G_{cct}. Avec :

$$V_{cct} = \{v_i^{cct} \Leftrightarrow cct_i(Dens, H, W, \zeta, Ar, x_d, y_d, x_f, y_f) \,|\, i = 1...n_{cct}\} \quad (5.\ 39)$$

Cette fois-ci, chaque composante (donc chaque sommet du nouveau graphe) est décrite par neuf caractéristiques. En plus des cinq caractéristiques morphologiques utilisées dans la première coloration, on utilise quatre coordonnées du rectangle englobant pour mesurer le voisinage spatial entre les composantes (cette mesure est un élément indispensable pour la formation des lignes). Ces coordonnées sont notées x_d, y_d, x_f et y_f.

La dissemblance morphologique (liée aux cinq caractéristiques calculées sur chaque composante de texte au niveau précédent) entre deux composantes de texte peut être déduite directement de la matrice M_d^{cc} construite au niveau *1* de la coloration.

A ce stade, il est essentiel d'introduire des informations topologiques permettant de mesurer avec précision l'éloignement spatial entre connexités. Le voisinage spatial entre deux composantes de texte repose sur un mélange entre la distance horizontale D_x (espace inter-caractères) et la distance verticale (ou alignement vertical) D_y. Ces deux distances sont déduites à partir des coordonnées des rectangles englobant les CCs à comparer. Elles sont données par la relation suivante :

$$\begin{cases} D_x[cct(i), cct(j)] = Max\left[x_d^{cct(i)}, x_d^{cct(j)}\right] - Min\left[x_f^{cct(i)}, x_f^{cct(j)}\right] \\ D_y[cct(i), cct(j)] = Max\left[y_d^{cct(i)}, y_d^{cct(j)}\right] - Min\left[y_f^{cct(i)}, y_f^{cct(j)}\right] \end{cases} \quad (5.\ 40)$$

$D_x[cct(i), cct(j)] < 0$ (ou $D_y[cct(i), cct(j)] < 0$) signifie que la composante *cct(i)* chevauche horizontalement (ou verticalement) la composante *cct(j)*.

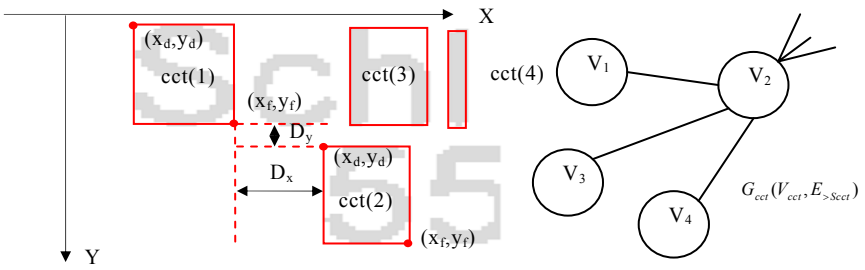


Figure 5. 29 : exemple de distances spatiales (horizontale et verticale) entre deux composantes connexes.

5.4.2.2.2 Construction de la matrice d'adjacence par combinaison de critères morphologiques et topologiques

Le voisinage spatial est donc basé sur la distribution horizontale d'un index de gauche vers la droite. Chaque connexité possède une valeur d'index propre. Si la composante connexe *cct(j)* est suffisamment proche de la composante *cct(i)* alors elle reçoit l'index de la composante *i*, c'est-à-

dire si $\{D_x < S_{Dx}\ et\ D_y < S_{Dy}\}$ alors $Index[cct(i)] = Index[cct(j)]$. Les seuils S_{Dx} et S_{Dy} gèrent respectivement l'espace inter-caractères et interlignes.

Entre deux sommets du graphe on évalue l'adjacence qui sera répercutée dans la matrice M_d^{cc} qui servira de base à la deuxième coloration. L'adjacence entre deux connexités à ce stade est notée E_{SP1} en rapport avec l'adjacence spatiale introduite pour la première fois (niveau ligne). On définira plus tard une adjacence E_{SP2} au niveau des blocs. Cette adjacence est donnée par la relation suivante :

$$E_{SP1}[cct(i),cct(j)] = \begin{cases} 1 \text{ si } \{(D_x \geq S_{Dx} \text{ ou } D_y \geq S_{Dy})\ et\ Index[cct(i)] \neq Index[cct(j)]\} \\ 0 \text{ sinon} \end{cases} \quad (5.41)$$

Cette adjacence est donc maximale (et vaut 1) si les deux connexités considérées ne sont pas dans un voisinage proche, limitées par les valeurs de seuils S_{Dx} et S_{Dy}. Ces valeurs de seuils sont choisies empiriquement.

Parallèlement à l'estimation de l'adjacence spatiale entre connexité, on définit, comme lors de la première étape de coloration une adjacence morphologique permettant de contrôler les fusions de connexités en fonction de la nature du contenu : type de police, imprimé, manuscrit que les caractéristiques de niveau 1 permettent de bien différencier. Le seuil S_{cct} utilisée pour l'adjacence $E_{>Scct}$ est donc un seuil global tenant compte du seuil S_{cc} de la première coloration et des seuils S_{Dx} et S_{Dy} présentés ci-dessus.

Une arête $E_{>Scct}$ est attribuée à chaque paire de sommets de G_{cct} selon la relation suivante :

$$E_{\geq Scct}\left[v_i^{cct}, v_j^{cct}\right] = \begin{cases} 1 \quad \text{si} \quad \{E_{\geq Scc}[cct(i),cct(j)] = 1 \text{ ou } E_{SP1}[cct(i),cct(j)] = 1\} \\ 0 \qquad\qquad \text{sinon} \end{cases} \quad (5.42)$$

Avec : $S_{cct} = \{S_{cc}, S_{Dx}, S_{Dy}\}$.

La construction du graphe $G_{cct}(V_{cct},E_{>Scct})$ donne une connaissance riche sur la structure globale du document exprimée sous forme d'un graphe composée d'un ensemble de sommets dont les arêtes traduisent une dissimilarité composée portant à la fois sur des indices géométriques des connexités et des indices topologiques traduisant leur éloignement.

La figure 5.30 résume l'ensemble des caractéristiques nécessaires à la construction des arêtes du graphe. Elle présente le résultat de la coloration sur un bloc adresse en fonction de différents choix de caractéristiques retenues qui conditionnent l'utilisation d'un indice de dissimilarité à chaque fois différent également. Selon le jeu de caractéristiques retenu, l'étiquetage des lignes est à chaque fois différent. La prise en compte de l'ensemble des caractéristiques produit une séparation en ligne idéale (voir figue 5.30.1)

S_{cc}	Désactivé	Activé	Désactivé	Activé	Désactivé	Activé	Désactivé	Activé
S_{Dx}	Désactivé	Désactivé	Activé	Activé	Désactivé	Désactivé	Activé	Activé
S_{Dy}	Désactivé	Désactivé	Désactivé	Désactivé	Activé	Activé	Activé	Activé
Figure	5	3	4	2	4	1	4	1

Figure 5. 30 : Synthèse des caractéristiques entrant dans la construction du graphe G_{cct}.

La connaissance globale ainsi obtenue pour chacune des connexités combinées à la contrainte de propreté lors de l'étape de coloration du graphe permet de gérer parfaitement le rangement local des composantes en lignes (par couleur ou groupe de composantes homogènes) écartant ainsi tout risque d'inclure dans une ligne une composante parasite (car dissimilaire au sens de la relation $E_{>Scct}$ que nous proposons).

5.4.2.2.3 Influence de la détection de l'inclinaison pour améliorer la précision et les temps de détection des lignes

Une fois le deuxième graphe construit, on applique à nouveau l'algorithme de coloration propre donné dans la *procédure 7* sur le graphe $G_{cct}(V_{cct},E_{>Scct})$ pour obtenir un ensemble C_L de n_L couleurs. Le résultat de cette coloration conduit à la formation des lignes de texte représentant le deuxième niveau de notre pyramide de données. Chaque couleur représente donc une ligne de texte avec $C_L = \{L_i(x_d^i, y_d^i, x_f^i, y_f^i) | i = 1...n_L\}$. Les points de coordonnées x_d, y_d, x_f *et* y_f correspondent à une simple représentation spatiale d'une ligne. Naturellement, la structure réelle et complète de chaque ligne est décrite par la liste des composantes de texte qui appartiennent à cette ligne.

L'inclinaison de chaque ligne (correspondant donc à une couleur différente des autres) est calculée progressivement durant la coloration. La méthode utilisée pour l'estimation de cette inclinaison est basée sur la

technique des moindres carrés que nous avons décrite en détail dans la section portant sur le redressement de ligne.

Dans le cas de faibles inclinaisons (θ <5°), les coordonnées de chaque ligne *L(i)* sont données tout simplement par la relation suivante :

$$\left(x_d^{L(i)} = \underset{\forall cct_j \in L_i}{Min} \left\{ x_d^{cct(j)} \right\},\ y_d^{L(i)} = \underset{\forall cct_j \in L_i}{Min} \left\{ y_d^{cct(j)} \right\} \right)$$
$$\left(x_f^{L(i)} = \underset{\forall cct_j \in L_i}{Max} \left\{ x_f^{cct(j)} \right\},\ y_f^{L(i)} = \underset{\forall cct_j \in L_i}{Max} \left\{ y_f^{cct(j)} \right\} \right) \quad (5.43)$$

Dans les autres cas (cas de forte inclinaison (θ >5°) des lignes), il est indispensable de tenir compte de la valeur d'inclinaison des lignes afin d'augmenter la robustesse dans la formation des blocs quelle que soit l'orientation des lignes qui les constituent. L'idée générale ici consiste à déplacer les sommets des composantes de texte par rotation inverse et à récupérer les sommets des rectangles englobant les lignes dans leur direction redressée horizontale. Ceci permet d'éviter le chevauchement entre les lignes et de calculer avec précision leurs relations spatiales comme leur hauteur, largeur, surface et excentricité. La prise en compte de l'inclinaison globale des lignes joue un grand rôle dans l'amélioration de l'étape de formation des blocs, car elle permet d'éviter toutes sortes d'erreurs de sous ou de sur-segmentation liées aux grandes inclinaisons des lignes (voir figure 5.31). Pour plus de détail sur l'estimation de l'orientation de chaque ligne, il est nécessaire de se reporter à la section 5.3.3.1 Sur le redressement de lignes.

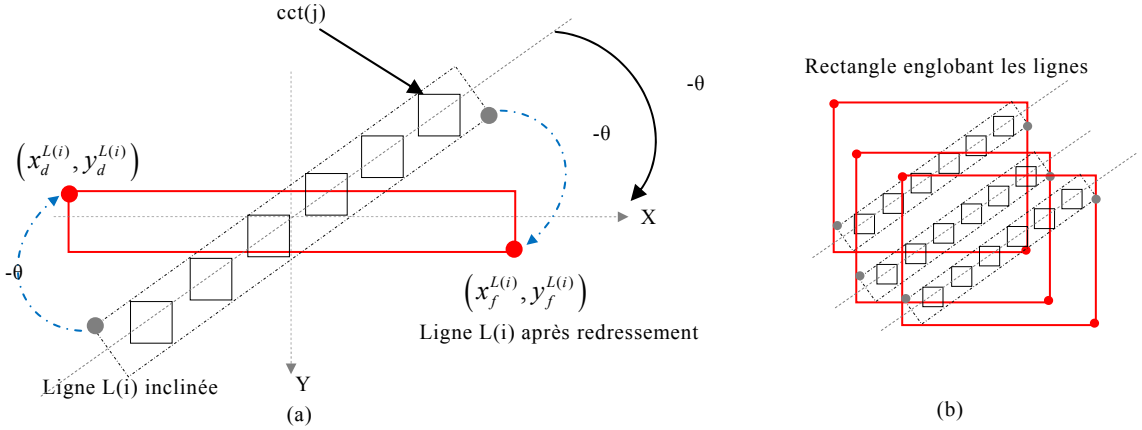


Figure 5. 31 : (a) Précision du découpage des lignes inclinées par rotation inverse, **(b)** imprécision du découpage et chevauchement des boîtes englobantes des lignes sans rotation inverse.

Les coordonnées de chaque ligne *L(i)* sont donc données par les équations suivantes :

$$
\begin{aligned}
x_d^{L(i)} &= \underset{\forall cct_j \in L_i}{Min} \left\{ x_d^{cct(j)} \cos(\theta) - y_d^{cct(j)} \sin(\theta) \right\} + x_G^{L(i)} \\
y_d^{L(i)} &= \underset{\forall cct_j \in L_i}{Min} \left\{ y_d^{cct(j)} \cos(\theta) + x_d^{cct(j)} \sin(\theta) \right\} + y_G^{L(i)} \\
x_f^{L(i)} &= \underset{\forall cct_j \in L_i}{Max} \left\{ x_f^{cct(j)} \cos(\theta) - y_f^{cct(j)} \sin(\theta) \right\} + x_G^{L(i)} \\
y_f^{L(i)} &= \underset{\forall cct_j \in L_i}{Max} \left\{ y_f^{cct(j)} \cos(\theta) + x_f^{cct(j)} \sin(\theta) \right\} + y_G^{L(i)}
\end{aligned}
\qquad (5.\ 44)
$$

Avec : $x_G^{L(i)} = \frac{1}{2}\left(x_d^{L(i)} + x_f^{L(i)} \right)$ et $y_G^{L(i)} = \frac{1}{2}\left(y_d^{L(i)} + y_f^{L(i)} \right)$

Par remplacement de x_G, le système d'équations devient donc :

$$
\begin{aligned}
x_d^{L(i)} &= \frac{3}{2} \times \underset{\forall cct_j \in L_i}{Min} \left\{ x_d^{cct(j)} \cos(\theta) - y_d^{cct(j)} \sin(\theta) \right\} + \frac{1}{2} \times \underset{\forall cct_j \in L_i}{Max} \left\{ x_f^{cct(j)} \cos(\theta) - y_f^{cct(j)} \sin(\theta) \right\} \\
y_d^{L(i)} &= \frac{3}{2} \times \underset{\forall cct_j \in L_i}{Min} \left\{ y_d^{cct(j)} \cos(\theta) + x_d^{cct(j)} \sin(\theta) \right\} + \frac{1}{2} \times \underset{\forall cct_j \in L_i}{Max} \left\{ y_f^{cct(j)} \cos(\theta) + x_f^{cct(j)} \sin(\theta) \right\} \\
x_f^{L(i)} &= \frac{3}{2} \times \underset{\forall cct_j \in L_i}{Max} \left\{ x_f^{cct(j)} \cos(\theta) - y_f^{cct(j)} \sin(\theta) \right\} + \frac{1}{2} \times \underset{\forall cct_j \in L_i}{Min} \left\{ x_d^{cct(j)} \cos(\theta) - y_d^{cct(j)} \sin(\theta) \right\} \\
y_f^{L(i)} &= \frac{3}{2} \times \underset{\forall cct_j \in L_i}{Max} \left\{ y_f^{cct(j)} \cos(\theta) + x_f^{cct(j)} \sin(\theta) \right\} + \frac{1}{2} \times \underset{\forall cct_j \in L_i}{Min} \left\{ y_d^{cct(j)} \cos(\theta) + x_d^{cct(j)} \sin(\theta) \right\}
\end{aligned}
\qquad (5.\ 45)
$$

et permet en un seul balayage de résoudre l'ensemble du système de coordonnées x_d, y_d, x_f, y_f.

L'angle d'inclinaison de chaque ligne est calculé lors de la deuxième coloration, c'est-à-dire durant la formation des lignes par la technique proposée dans la partie portant sur le redressement de ligne, section *5.3.3.1* Voici figure *5.32*, quelques exemples de formation de lignes d'inclinaisons variables obtenues après la coloration de niveau 2.

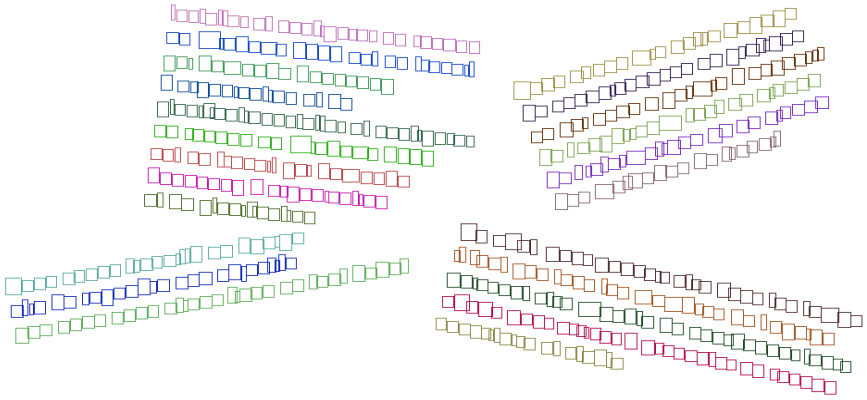


Figure 5. 32 : Exemples de formation des lignes à partir de l'estimation de l'inclinaison globale (définie à la section *5.3.3.1*).

5.4.2.3 La troisième coloration pour la formation des blocs finaux

5.4.2.3.1 Choix des caractéristiques

La troisième coloration permet de regrouper les lignes (issues de la deuxième coloration) en blocs de texte homogènes et polygonaux. Pour effectuer cette coloration nous construisons le troisième graphe (noté $G_L(V_L, E_{>SL})$) à partir des lignes de la façon suivante :

- On représente chaque ligne de texte $L(i) \in L \,|\, i = 1...n_L$ par un sommet de G_L. Chacune des lignes est décrite par dix caractéristiques de la façon suivante :

$$V_L = \{v_i^L \Leftrightarrow L_i(Dens, H, \mu_H, \sigma_H, Ar, \theta, x_d, y_d, x_f, y_f) \,|\, i = 1...n_L\} \quad (5.\ 46)$$

Avec θ angle d'inclinaison de la ligne, μ_H et σ_H sont la moyenne et l'écart type des hauteurs de CCs de la ligne.

Par hypothèse, deux lignes de texte ne peuvent pas être regroupées ensemble si elles ont une inclinaison, une forme ou un alignement dissimilaire ou si elles sont très éloignées (espacement interligne très grand).

La dissemblance morphologique entre deux lignes permet de distinguer les lignes de texte selon leurs formes et leurs textures. En pratique, nous avons utilisé comme caractéristiques, la densité, la hauteur, la moyenne et l'écart type des hauteurs et la surface ($L = \{Dens, H, \mu_H, \sigma_H, Ar\}$)pour assurer l'homogénéité de chaque bloc. Se basant sur ces cinq propriétés, les lignes de texte imprimé ne doivent donc pas

être regroupées dans un même bloc de lignes manuscrites ou bloc de texte en gras ou bloc de police très différente. Les dissemblances entres les lignes de texte sont données par la matrice M_d^L traduisant la distance de Manhattan exprimée entre chaque ensemble de cinq caractéristiques (chacune des caractéristiques est normalisée par son écart type mesuré sur une base de lignes indépendantes).

$$M_d^L[L(i),L(j)] = d_L(\mathrm{L}_i, \mathrm{L}_j) = \sum_{p=1}^{5} \left| L_i(p) - L_j(p) \right| \; avec \; (j>i) \in [1, n_L] \; et \; L(p) \in \{Dens, H, \mu_H, \sigma_H, \zeta, Ar\} \quad (5.47)$$

Avec :

$$Dens\left[L(i)\right] = \frac{1}{\left|L(i)\right|} \sum_{k=1,\, cct_k \in L(i)}^{|L(i)|} Dens(cct_k) \quad (5.48)$$

Où $|L(i)|$ est le nombre de composantes connexes qui appartient à la ligne *i*.

L'écart type des hauteurs σ_H est donné par :

$$\sigma_H = \frac{1}{\left|L(i)\right|} \sqrt{\sum_{i=1}^{n} \left(H(cct_i) - \mu_H\right)^2} \quad (5.49)$$

5.4.2.3.2 Construction de la matrice d'adjacence par combinaison de critères morphologiques et topologiques

A partir de la matrice de dissemblances morphologique M_d^L on obtient la matrice d'adjacence entre sommets suivante :

$$E_{\geq SFL}\left[v_i^L, v_j^L\right] = \begin{cases} 1 & \text{si} \quad M_d^L(i,j) \geq S_{FL} \\ 0 & \text{sinon} \end{cases} \quad (5.50)$$

L'alignement spatial entre deux lignes de texte est donné par la relation suivante :

$$D_x\left[L(i), L(j)\right] = \left| x_d^{L(i)} - x_d^{L(j)} \right| \quad (5.51)$$

L'espacement interligne est donné par :

$$D_y\left[L(i), L(j)\right] = Max\left[y_d^{L(i)}, y_d^{L(j)}\right] - Min\left[y_f^{L(i)}, y_f^{L(j)}\right] \quad (5.52)$$

Le voisinage spatial est donc basé sur la distribution verticale d'un index de haut vers le bas. Si la ligne *L(j)* est suffisamment proche (au sens du seuil S_{Dy}) et bien alignée avec la ligne *L(i)* (au sens du seuil S_{Dx}) alors elle reçoit l'index de la ligne *i*.

C'est-à-dire si :

$$\left\{ D_y\left[L(i), L(j)\right] < S_{Dy} \text{ et } D_x\left[L(i), L(j)\right] < S_{Dx} \text{ avec } (y_d^{L(i)} < y_d^{L(j)} \text{ et } y_f^{L(i)} < y_f^{L(j)}) \right\} \quad (5.53)$$

alors $Index\left[L(j)\right] = Index\left[L(i)\right]$.

L'adjacence E_{SP2} entre deux lignes (et donc entre deux sommets du graphe) peut être, selon la notion de coloration, donnée par la relation suivante :

$$E_{SP2}\left[L(i),L(j)\right]=\begin{cases}1 \text{ si } \left\{(D_y \geq S_{Dy} \text{ ou } D_x \geq S_{Dx}) \text{ et } Index\left[L(i)\right] \neq Index\left[L(j)\right]\right\}\\ 0 \text{ sinon}\end{cases} \quad (5.\ 54)$$

On relie entre eux deux sommets (lignes) par une arête $E_{>SL}$ de GL si et seulement si les deux sommets ont des formes, des inclinaisons différentes (en relation avec les valeurs de seuils S_{FL} et S_θ) ou ne sont pas spatialement voisins (en relation avec les seuils S_{Dx} et S_{Dy}). Les adjacences totales (selon les trois critères) sont données par relation suivante :

$$E_{\geq SL}\left[v_i^L, v_j^L\right]=\begin{cases}1 \quad \text{si} \quad \left\{E_{\geq SFL}\left[v_i^L, v_j^L\right]=1 \text{ ou } E_{SP2}\left[L(i),L(j)\right]=1 \text{ ou } \left|\theta^{L(i)}-\theta^{L(j)}\right| \geq S_\theta\right\}\\ 0 \quad \text{sinon}\end{cases} \quad (5.\ 55)$$

$$\text{Avec : } S_L=\left\{S_{FL}, S_\theta, S_{Dx}, S_{Dy}\right\}$$

À l'issue de l'étape de construction du graphe, on applique la *procédure 7* de coloration sur le graphe $G_L(V_L, E_{>SL})$ pour obtenir l'ensemble C_B les n_B couleurs. Chacune de ces couleurs représente la structure définitive d'un bloc de texte B_i. On peut déduire les coordonnées de chaque bloc à partir des coordonnées des lignes qui lui appartiennent de la façon suivante :

$$\left(x_d^{B(i)}=\underset{\forall L_j \in B_i}{Min}\left\{x_d^{L(j)}\right\},\ y_d^{B(i)}=\underset{\forall L_j \in B_i}{Min}\left\{y_d^{L(j)}\right\}\right)$$
$$\left(x_f^{B(i)}=\underset{\forall L_j \in B_i}{Max}\left\{x_f^{L(j)}\right\},\ y_f^{B(i)}=\underset{\forall L_j \in B_i}{Max}\left\{y_f^{L(j)}\right\}\right) \quad (5.\ 56)$$

Pour offrir plus de précision à la localisation de blocs, on a préféré garder toute la structure polygonale des blocs en préservant la liste des lignes. On représente donc tout bloc *B(i)* par une séquence de lignes qui lui appartient triées du haut vers le bas. Avec :

$$\underset{i=1\ldots n_B}{B(i)}=\left\{\underset{j=1\ldots|B(i)|}{L(j)} \in B(i)\right\} \quad (5.\ 57)$$

où $|B(i)|$ représente le nombre des lignes du bloc B(i).

La description de chaque bloc est la synthèse hiérarchique de toutes les descriptions obtenues au niveau des lignes, des composantes de texte, des composantes connexes brutes et des séquences noires (plages noires).

5.4.2.3.3 Algorithme et bilan de l'analyse hiérarchique de la structure d'un document

L'algorithme suivant résume toutes les étapes de l'extraction de la structure physique à travers les trois niveaux de coloration successifs.

Algorithme 5.3 : Analyse- Structure-Physique()

Début

Première coloration : Séparation texte/non texte

Construire $G_{cc}(V_{cc}, E_{>Scc})$

Sommets :

Pour chaque $cc(i) \in CC \,|\, i = 1 \ldots n_{cc}$ faire $v_i^{cc} \Leftrightarrow cc_i(Dens, H, W, \zeta, Ar)\}$

FinPour

Pour chaque paire de sommets dissimilaires v_i^{cc}, v_j^{cc}

Associer une arête $E_{\geq Scc}\left[v_i^{cc}, v_j^{cc}\right] = 1$,

Appliquer la *Procédure* $7(G_{cc}(V_{cc}, E_{>Scc}))$,

Isoler les couleurs de forte régularité (texte),

et charger les n_{cct} composantes de couleurs de texte dans l'ensemble C_{cct}.

Deuxième coloration : Formation des lignes de texte

Construire $G_{cct}(V_{cct}, E_{>Scct})$ à partir de l'ensemble C_{cct}

Sommets :

Pour chaque $cct(i) \in C_{cct} \,|\, i = 1 \ldots n_{cct}$ faire

$$v_i^{cct} \Leftrightarrow cct_i(Dens, H, W, \zeta, Ar, x_d, y_d, x_f, y_f)$$

FinPour

Pour chaque paire de sommets dissimilaires ou non voisins (par indexation) v_i^{cct}, v_j^{cct} Associer une arête $E_{\geq Scct}\left[v_i^{cct}, v_j^{cct}\right] = 1$

Appliquer la *Procédure* $7(G_{cct}(V_{cct}, E_{>Scct}))$

Chaque couleur de sortie représente une ligne de texte $L(i)$ dans l'ensemble C_L.

Troisième coloration : **Formation des blocs de texte**

Construire $G_L(V_L, E_{>SL})$

Sommets :

Pour chaque $L(i) \in L \,|\, i = 1 \ldots n_L$ faire

$$v_i^L \Leftrightarrow L_i(Dens, H, \mu_H, \sigma_H, Ar, \theta, x_d, y_d, x_f, y_f)$$

FinPour

Pour chaque paire de sommets v_i^L, v_j^L de Formes, d'inclinaisons ou d'alignements dissimilaires ou qui ne sont pas verticalement voisins

Associer une arête $E_{\geq SL}\left[v_i^L, v_j^L\right] = 1$

Appliquer la *Procédure* $7(G_L(V_L, E_{>SL}))$

Chaque couleur représente un bloc de texte polygonal $B(i)$ dans l'ensemble C_B.

Fin

La figure ci-dessous illustre le résultat de l'application des trois niveaux de coloration sur un document contenant à la fois des parties textuelles et non textuelles, et des lignes d'orientations variables.

Figure 5. 33 : (a) zone d'adresse en niveau de gris, (b) image binaire, (c) première coloration, (d) séparation texte/non texte, (e) deuxième coloration : extraction des lignes, (f) troisième coloration : extraction des blocs.

Nous avons évalué l'efficacité de notre méthode d'extraction de la structure physique sur une base de *10000* images de courriers de structures complexes (i.e présentant des difficultés diverses de segmentation). Plus de *95 %* des blocs adresse ont été correctement segmentés par notre méthode, contre *60%* par la méthode de RLSA et à peine *30%* par la méthode de projection des profils.

L'analyse de ces résultats prouve que plusieurs erreurs de sous ou sur segmentation commises par la méthode RLSA ou par la méthode de projection des profils peuvent être considérablement réduites en impliquant la coloration hiérarchique à toutes les phases de l'extraction de la structure physique.

A l'observation des résultats produits sur les images de courriers, nous pouvons faire le double constat suivant :

- notre approche de coloration hiérarchique produit un découpage (en connexités, lignes et blocs) très précisément étiqueté en composantes textuelles et graphiques.

- pour les composantes textuelles, l'approche hiérarchique de regroupement par coloration est très performante sur des images de mises en page complexes et bruitées (structure peu stable d'un courrier à l'autre, présence de bruits graphiques très fréquents …). Le système est capable de rejeter facilement les composantes parasites situées au voisinage des zones de texte.

On peut voir sur les courbes suivantes, que notre proposition de segmentation permet de réduire considérablement les temps de traitement par rapport aux méthodes de segmentation plus conventionnelles tout en offrant une caractérisation complémentaire des composantes textuelles sur les trois niveaux de la hiérarchie.

Il faut également préciser que nous avons choisi de montrer les résultats de notre approche de coloration en relation avec des approches de segmentation de complexité très faible (RLSA et projections de profils) qui procèdent par balayages simples de l'image sans heuristiques lourdes de décision. Pour l'approche RLSA, il est nécessaire de réaliser un double lissage unidirectionnel de l'image à segmenter selon deux seuils (lissage horizontal et vertical). La segmentation est obtenue en appliquant l'opérateur logique "and" sur les deux images.

L'analyse de ces techniques (RLSA ou projections de profils) montre plusieurs insuffisances. La première insuffisance est liée au choix arbitraire des seuils de lissage (RLSA), à la sensibilité aux inclinaisons et à leur inadaptation à la segmentation des blocs graphiques, et des images contenant des structures de type tableaux (essentiellement pour les projections de profils). La deuxième insuffisance est liée à la difficulté de segmenter les documents à structures complexes sans l'intervention d'un utilisateur humain pour fixer les valeurs de seuils et garantir l'horizontalité des lignes d'écritures.

D'autres approches comme la recherche par pavage de Fond de Pavlidis [PAV92] ou les approches de segmentation fondée sur l'analyse des espaces inoccupés et la fusion des segments d'espaces blancs adjacents comme celle d'Akindele dans [AKI93] permettent un partitionnement de l'image mais ne donnent aucune indication sur la nature des données (textuelle ou graphique). Des travaux précurseurs ont été proposés par Lee dans [LEE98] qui élabore une approche innovante d'analyse hiérarchique multiéchelle des images exploitant les décompositions en ondelettes et l'étiquetage des régions selon leur niveau d'homogénéité locale. La nature multiéchelle de ce travail présente un grand intérêt pour l'analyse de la structure des documents, cependant le passage au domaine fréquentiel et l'utilisation d'un support de décomposition spectral en fait une approche

très coûteuse en temps de calcul et impossible à mettre eu œuvre dans notre contexte industriel.

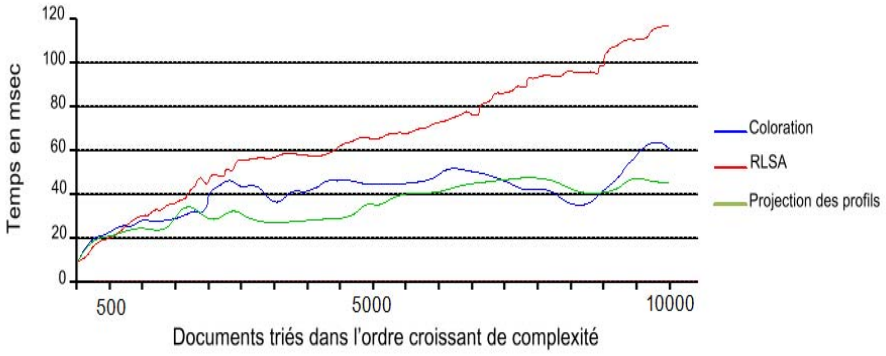


Figure 5. 34 : Comparatif des temps d'exécution (temps de binarisation et de détection des CCs n'est pas compris).

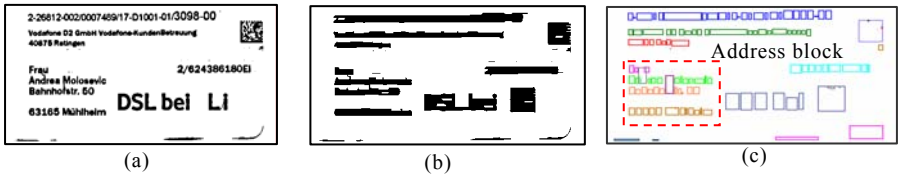


Figure 5. 35 : (a) Image binaire, **(b)** résultat de RLSA (les trois première lignes d'adresse sont fusionnées), **(c)** segmentation en ligne réussie (isolation des composantes qui cause la fusion des lignes).

Figure 5. 36 : (a) La projection fusionne des lignes d'adresse inclinées, **(b)** résultat de seconde coloration (segmentation robuste à l'inclinaison des lignes).

5.5 Application de la théorie des graphes à la classification de documents

5.5.1 Rappel du principe général de la RAD

Cette application est dédiée aux documents de structure régulière. Nous avons expliqué dans le troisième chapitre comment l'intégration de la reconnaissance automatique du type de document permettait de réduire de façon significative les taux de rejets et les erreurs de lecture. Nous avons ainsi décidé d'appliquer cette opération importante comme étape préliminaire de la chaîne de traitement dédiée au tri de documents hétérogènes (i.e. répartis en différentes classes homogènes de structures internes stables). Cette reconnaissance préalable de la nature de document doit diriger les autres étapes du processus de tri automatique de documents. Une fois le document identifié, le module de lecture possède la connaissance de la position de la zone d'intérêt (la zone à lire) et la nature des traitements à effectuer pour extraire que les données nécessaires à la reconnaissance des contenus, et à la prise de décision.

Par exemple, si le système de tri automatique détecte :

- un courrier manuscrit, il appliquera le processus localisation de bloc adresse adéquat, et les modules de reconnaissance de texte dédiés à de tels documents ;
- une enquête, il ira lire les cases à cocher et le texte pré-casé ;
- un RIB, il ira lire le numéro de compte ;
- une facture, il réalisera des contrôles sur le numéro de facture.

Dans le chapitre 4 nous avons présenté notre modélisation de la RAD en termes de b-coloration de graphe. Nous allons présenter dans cette partie les étapes détaillées de cette modélisation par graphe pour la reconnaissance du type de documents de courriers d'entreprise. Nous utiliserons une description issue de l'analyse de la structure physique de documents exploitant notre méthode de segmentation basée sur la coloration de graphe que nous avons présenté dans la section précédente. La classification des documents est basée sur la b-coloration.

Les étapes de notre méthode de RAD sont présentées dans le schéma de la figure 5.37.

Notre méthode de binarisation est appliquée dans la première étape du processus de RAD. Nous utilisons ensuite notre méthode de détection des CCs avant de procéder à une extraction de la structure physique par coloration hiérarchique des composantes (figure 5.38).

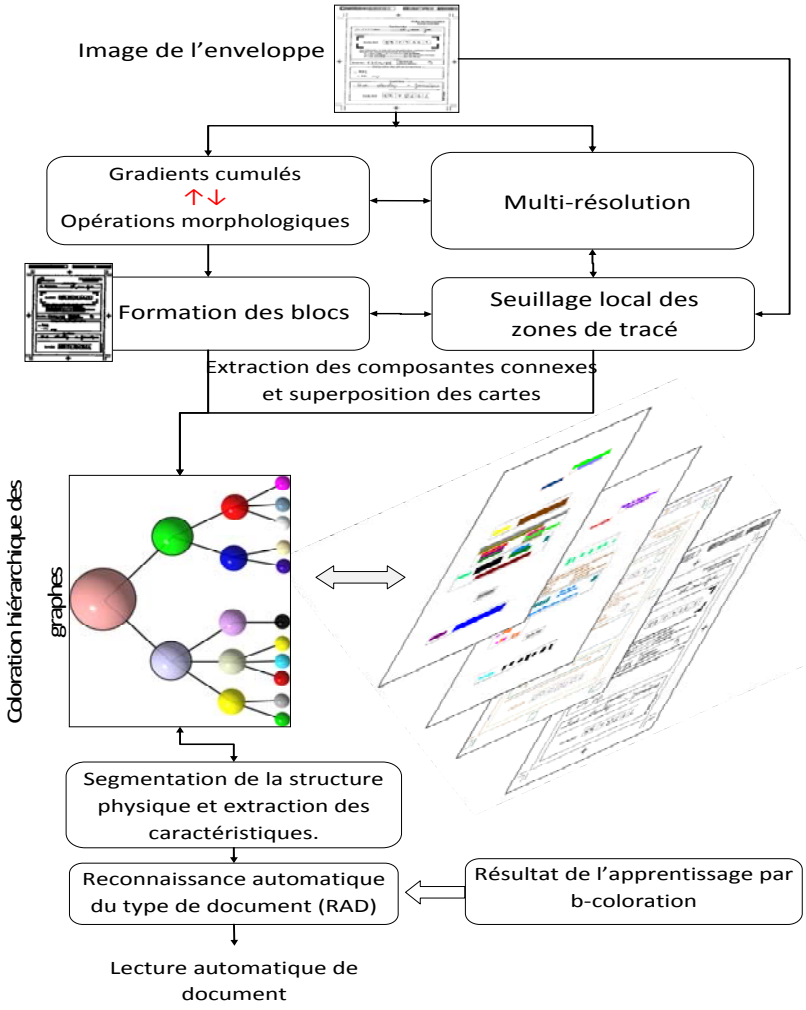


Figure 5. 37 : Diagramme fonctionnel de la Reconnaissance automatique de Documents.

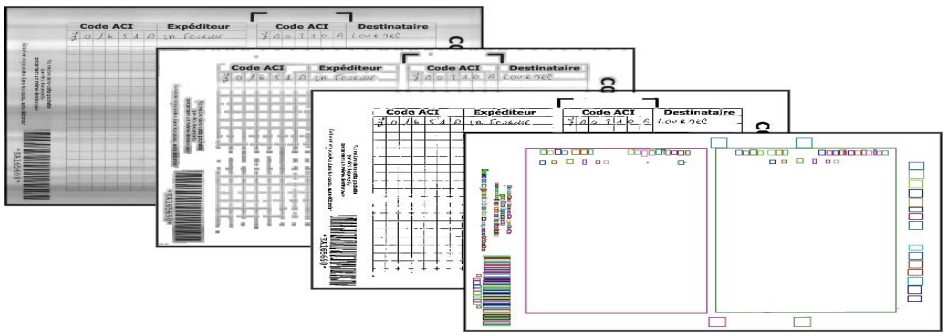


Figure 5. 38 : Seuillage local à proximité des zones de texte et détection des composantes connexes.

A partir de là, nous appliquons notre méthode d'extraction de la structure physique de document qui porte principalement sur l'exploitation

des bonnes propriétés de partitionnement de la coloration de graphes qui permettent de bien séparer les éléments pertinents puis de les regrouper en ensembles homogènes. Dans ce processus de partitionnement / regroupement, les éléments parasites sont automatiquement rejetés.

Rappelons brièvement à ce stade le principe de l'extraction de la structure physique qui entre en jeu pour la reconnaissance de documents (RAD). Ce principe consiste à distinguer, par coloration des CCs (vues comme sommets du graphe), toutes les zones textuelles des zones non textuelles, puis à regrouper les CCs des zones textuelles en lignes. Les lignes les plus régulières correspondent aux lignes de texte imprimé (les lignes de régularité moyenne correspondent aux lignes manuscrites). L'angle d'inclinaison calculé dans cette phase est utilisé dans la description des documents comme paramètre global de l'inclinaison de document. Parallèlement à l'extraction de la structure physique, on procède à une extraction de caractéristiques. Voici son principe en détails.

5.5.2 Extraction des caractéristiques de documents

L'extraction des caractéristiques a pour objectif de minimiser la quantité d'informations nécessaire à la séparation des documents. Elle est appliquée progressivement en parallèle à l'extraction de la structure physique. Pour représenter un document nous déterminons à partir de sa structure physique un certain nombre de caractéristiques : *20 caractéristiques locales*, propres à chaque ligne de texte, et *15 caractéristiques globales*, relatives au document dans son ensemble.

5.5.2.1 Les 20 caractéristiques locales

Elles sont extraites à partir de chaque ligne de texte imprimé Li (elles regroupent les caractéristiques de forme incluant les relations spatiales de chaque ligne par rapport aux autres lignes de documents). Ces caractéristiques sont :

1) $n_{cct}(L_i) = card(cct_k \in L_i)$: nombre de composantes connexes de texte d'$i^{ème}$ ligne.

2) $\mu_{Hcct}(L_i)$: hauteur moyenne des composantes connexes de la ligne L_i.

3) $\mu_{Wcct}(L_i)$: largeur moyenne des composantes connexes de la ligne avec L_i

4)W(Li) : largeur d'une ligne,

5) $Ar(L_i) = H(L_i) \times W(L_i)$: aire de la ligne L_i.

6) $\zeta(L_i) = W(L_i) / H(L_i)$: excentricité de la ligne L_i.

7)*$Xg(L_i)$:* position horizontale centrée par rapport à l'abscisse du centre de gravité de toutes les lignes de texte présentes sur le document. *Cette position est invariante à la translation verticale des lignes de texte sur l'image*.

$$Xg(L_i) = \frac{1}{2}\left[\left(x_d^{L(i)} + x_f^{L(i)}\right) - \frac{1}{n_L}\sum_{j=1}^{nL}\left(x_d^{L(j)} + x_f^{L(j)}\right)\right] \quad (5.\ 58)$$

Où n_L est le nombre de lignes de texte imprimé dans le document.

8)$Yg(L_i)$: position verticale centrée par rapport à l'ordonnée du centre de gravité de toutes les lignes de texte présentes sur le document. Cette position est invariante à la translation horizontale des lignes de texte sur l'image.

$$Yg(L_i) = \frac{1}{2}\left[\left(y_d^{L(i)} + y_f^{L(i)}\right) - \frac{1}{n_L}\sum_{j=1}^{nL}\left(y_d^{L(j)} + y_f^{L(j)}\right)\right] \quad (5.\ 59)$$

9)$O_{LH}(L_i)$: ordre de la ligne de texte selon le tri des lignes (noté T) dans l'ordre croissant de leur position verticale (égale aussi au nombre de lignes situées « au-dessus » de la ligne L_i, dans le sens vertical).

10)$O_{LB}(L_i)$: nombre de lignes qui se trouvent « en dessous » de la ligne L_i selon le tri T, avec $O_{LB}(L_i) = n_L - O_{LH}(L_i)$.

11)$D_{PX}(L_i)$: distance ***verticale*** de la ligne L_i à la ligne ***précédente*** selon le tri T ;

12)$D_{PY}(L_i)$: distance ***horizontale*** de la ligne L_i par rapport à la ligne ***précédente*** selon le tri T ;

13)$D_{SX}(L_i)$: distance ***verticale*** de la ligne L_i à la ligne ***suivante*** selon le tri T ;

14)$D_{SY}(L_i)$: distance ***horizontale*** de la ligne L_i à la ligne ***suivante*** selon le tri T ;

Avec :

$$\begin{aligned}
D_{SX}(L_i) &= \underset{\text{Selon l'ordre de tri T}}{D_{SX}(L(j), L(j+1))} = Max\left(x_d^{L(j)}, x_d^{L(j+1)}\right) - Min\left(x_f^{L(j)}, x_f^{L(j+1)}\right) \\
D_{SY}(L_i) &= \underset{\text{Selon l'ordre de tri T}}{D_{SY}(L(j), L(j+1))} = Max\left(y_d^{L(j)}, y_d^{L(j+1)}\right) - Min\left(y_f^{L(j)}, y_f^{L(j+1)}\right) \\
D_{PX}(L_i) &= \underset{\text{j selon l'ordre de tri T}}{D_{PX}(L(j), L(j-1))} = Max\left(x_d^{L(j)}, x_d^{L(j-1)}\right) - Min\left(x_f^{L(j)}, x_f^{L(j-1)}\right) \\
D_{PY}(L_i) &= \underset{\text{Selon l'ordre de tri T}}{D_{PY}(L(j), L(j-1))} = Max\left(y_d^{L(j)}, y_d^{L(j-1)}\right) - Min\left(y_f^{L(j)}, y_f^{L(j-1)}\right)
\end{aligned} \quad (5.\ 60)$$

15)$D_{GPX}(L_i)$: distance entre l'abscisse du centre de gravité de la ligne L_i et l'abscisse du centre de gravité de la ligne ***précédente*** selon le tri T ;

16)$D_{GPY}(L_i)$: distance entre l'ordonnée du centre de gravité de la ligne L_i et l'ordonnée du centre de gravité de la ligne ***précédente*** selon le tri T ;

17)$D_{GSX}(L_i)$: distance entre l'abscisse du centre de gravité de la ligne L_i et l'abscisse du centre de gravité de la ligne ***suivante*** selon le tri T ;

18)$D_{GPY}(L_i)$: distance entre l'ordonnée du centre de gravité de la ligne L_i et l'ordonnée du centre de gravité de la ligne ***suivante*** selon le tri T ;

19)P : pente entre la ligne L_i et le centre de gravité de toutes les lignes de documents.

20) $\theta(L_i)$: Angle d'inclinaison de la ligne L_i.

Remarque : l'angle de l'inclinaison de la ligne $\theta(L_i)$ est utilisé pour appliquer une rotation inverse des coordonnées de la ligne lors du calcul des caractéristiques spatiales. Il rend cette description plus robuste à l'inclinaison des documents.

5.5.2.2 Les 15 caractéristiques globales

Sont extraites à partir de la structure physique sur tout le document.

1) n_{cc} : nombre de composantes connexes dans tout le document ;
2) n_L : nombre de lignes de texte imprimé sur tout le document ;
3) $AlH_G = \sigma(x_d^{L(i)})_{i=1...nL}$: écart type de l'alignement horizontal gauche des lignes de texte de document ;
4) $AlH_D = \sigma(x_f^{L(i)})_{i=1...nL}$: écart type de l'alignement horizontal droit des lignes de texte de document
5) $AlHG = \sigma(x_G^{L(i)})_{i=1...nL}$: écart type de l'alignement horizontal mesuré sur les abscisses des centres de gravités des lignes de texte ;
6) $AlV = \sigma(y_d^{L(i)})_{i=1...nL}$: écart type de l'alignement vertical des lignes de texte de document.
7) σ_{HL} : écart type des hauteurs des lignes
8) σ_{WL} : écart type des largeurs des lignes
9) PL_{Max} : Ordre de la ligne qui contient le plus grand nombre de CCs.
10) *SDX* : mesure de régularité des profils ***horizontaux***.
11) SDY : mesure de régularité des profils ***verticaux***.

Avec :

$$SDX(\theta_\mu) = \sum_x h(x_{\theta\mu}) - h(x_{\theta\mu} - 1) \text{ et } SDY(\theta_\mu) = \sum_x h(y_{\theta\mu}) - h(y_{\theta\mu} - 1)$$
$$x_{\theta\mu} = x\cos(\theta_\mu) - y\sin(\theta_\mu) \text{ et } y_{\theta\mu} = y\sin(\theta_\mu) + y\cos(\theta_\mu),\ \theta_\mu = \frac{1}{n_L}\sum_{k=1}^{nL}\theta(L_k). \quad (5.\ 61)$$

où θ_μ est l'angle d'inclinaison de document (égale à l'inclinaison moyenne des lignes de texte) utilisée pour rendre *SDX* et *SDY* robuste à l'inclinaison des documents.

h est l'histogramme des projections des profils.

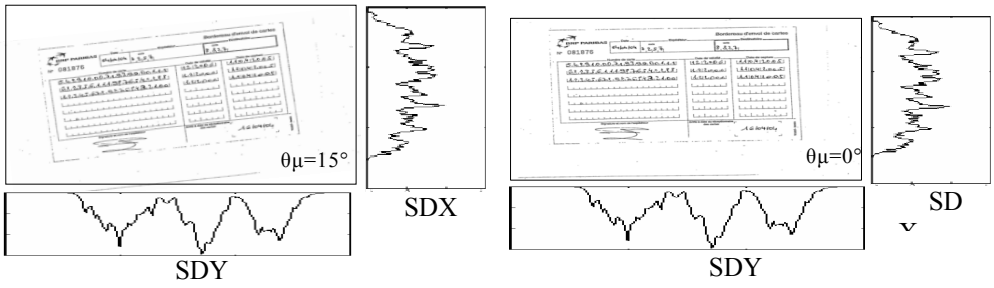


Figure 5. 39 : Mesures de régularité des profils horizontaux et verticaux.

12) *Dens* : Densité totale de documents (rapport entre le nombre de pixels noirs et la surface de document).
13) $Dens_{CCmax}$: Densité de la plus grande composante connexe (elle correspond par exemple au plus grand tableau sur un formulaire).
14) H_{ccmax} *:* la hauteur de la plus grande CC dans le document.

15)H_{ccmax} *:* la largeur de la plus grande CC dans le document.

Remarque : Durant l'apprentissage ou la reconnaissance, chaque caractéristique doit être normalisée par son écart type sur tous les documents de la base d'apprentissage afin de pouvoir les comparer aux autres variables numériques présentant des unités de mesures différentes. Cette normalisation en variables centrées réduites est importante car elle permet de ramener des distributions caractérisées par des moyennes, des écarts-types et des unités de mesure différents à un seul et même modèle théorique de référence : la distribution normale centrée réduite précisément. Il est alors possible d'effectuer des opérations qui seraient impossibles sur les échelles d'origine de chaque variable prise indépendamment des autres.

5.5.3 Les représentations des documents utilisées

La représentation de chaque type de document est basée sur la description de la structure physique uniquement.

Nous utilisons ainsi deux types de représentations portant sur une :

1) Représentation structurelle : chaque document *j* est représenté dans l'espace Rs^n par une séquence ordonnée de *n* lignes de texte : $Rs(j)=\left(L_1^j,L_2^j,...,L_n^j\right)$ où la $t^{ème}$ ligne L_t est représentée par un vecteur de *p=20* caractéristiques locales avec $L_t=\left(x_1^t,x_2^t,...,x_p^t\right)$.

2) Représentation vectorielle globale : chaque document *j* est représenté dans l'espace Rv^{m} par un vecteur de *m* = 15 caractéristiques globales, avec $Rv(j)=\left(y_1^j,y_2^j,...,y_m^j\right)$.

1.

5.5.4 Mesures de dissimilarité entre documents

Pour comparer deux documents, on utilise la combinaison de deux distances (D_{Rv} dans l'espace Rv^{m} et D_{Rs} dans l'espace Rs^n) donnée par la formule suivante :

$$DT=\gamma D_{Rv}+(1-\gamma)D_{Rs} \qquad (5.\ 62)$$

Expérimentalement, la valeur de γ est ajustée à *0.45* de manière à maximiser la qualité de classification Ψ [GAC08], avec :

$$\gamma=\{k=\arg\max_{0\le k\le 1}(\psi_k)\} \qquad (5.\ 63)$$

Si deux documents sont séparés par une faible distance *DT* alors ils se ressemblent.

La distance euclidienne normalisée D_{Rv} entre deux documents (i et j), représentés respectivement par les descripteurs *Rv(i)* et *Rv(j),* se calcule facilement de la façon suivante :

$$D_{Rv}\left[Rv(i),Rv(j)\right]=\frac{1}{m}\left[\sum_{k=1}^{m}\left|y_k^i-y_k^j\right|^{\alpha}\right]^{\frac{1}{\alpha}} \text{ avec } \alpha=2 \quad (5.64)$$

La distance D_{RS} réalise un mapping spatial entre les séquences *Rs(i)* de *ni* lignes et *Rs(j) de nj* lignes, elle est appelée la *Warping Function*. L'ajustement non-linéaire entre *Rs(i)* et *Rs(j)* peut être représenté par un chemin : $C=c_1,c_2,...,c$ avec $c_k=(i_k,j_k)$ (figure 5.40).

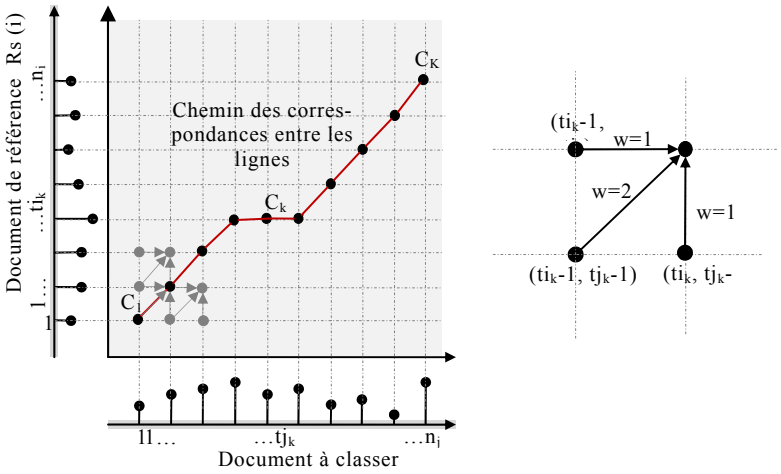


Figure 5. 40 : Principe de la fonction de déformation.

La somme pondérée des erreurs le long de la Warping Function C est :

$$D(c)=\frac{\sum_{k=1}^{K}d(c_k)\cdot w_k}{\sum_{k=1}^{K}w_k} \text{ avec } d(c_k)=d(L_t^{ik},L_t^{jk})=\sqrt{\sum_{l=1}^{p}\left[x_l^t(ti_k)-x_l^t(tj_k)\right]^2} \quad (5.65)$$

avec w_k un coefficient de pondération non-négatif, utilisé en dénominateur pour compenser l'effet de K (le nombre de point dans la Warping Function). Les fonctions ti_k et tj_k doivent être croissantes et respecter certaines conditions de continuité comme :

- La monotonie : $ti_k \geq ti_{k-1}$ *et* $tj_k \geq tj_{k-1}$
- La continuité : $ti_k - ti_{k-1} \leq 1$ *et* $tj_k - tj_{k-1} \leq 1$
- Les limites : $ti_1 = 1,\ tj_1 = 1,\ ti_K = n_i$ *et* $tj_K = n_j$

Le problème à résoudre devient :

$$D_{Rs}\left[Rs(i),RS(j)\right]=\frac{1}{n_i+n_j}\min_{C}\sum_{k=1}^{K}d(c_k)\cdot w_k \quad (5.66)$$

Les coefficients de pondération :

$$w_k = ti_k - ti_{k-1}+tj_k - tj_{k-1} \text{ et donc } \sum_{k=1}^{K}w_k = n_i + n_j \quad (5.67)$$

Ce qui revient à chercher parmi tous les chemins possibles, le chemin qui minimise la dissemblance entre la séquence *Rs(i)* et la séquence *Rs(j)*. Ce problème peut se résoudre simplement en explorant tous les chemins possibles.

Malheureusement, ce nombre de chemins possibles croit exponentiellement avec les nombre de lignes dans les documents à comparer et le calcul de la distance D_{Rs} devient très coûteux. Ce problème peut être résolu de manière efficace par un algorithme de comparaison dynamique qui va rapidement mettre en correspondance optimale les lignes de deux documents. Le principe est qu'au lieu d'étudier tous les chemins possibles, il est possible de trouver la solution optimale en étudiant le problème localement (figures 5.40 et 5.41).

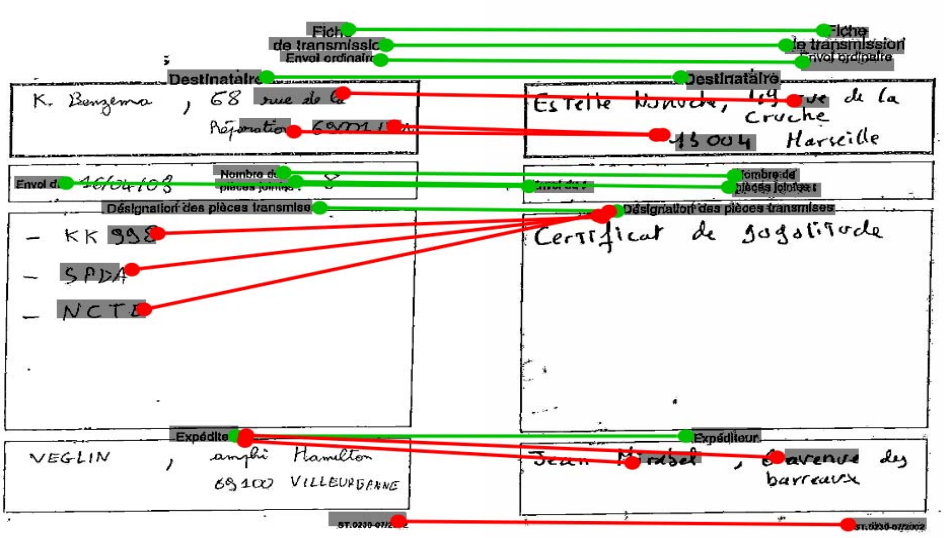


Figure 5. 41 : Comparaison dynamique de deux documents de même classe contenant des annotations manuscrites différentes.

Il suffit alors, pour chaque point de l'espace $Rs^{ni} \times Rs^{nj}$, de trouver le chemin qui répond mieux aux critères de continuité tout en minimisant la contribution à l'accumulation de la distance globale. Il suffit donc d'étudier les transitions autorisées et d'appliquer la relation récursive locale :

$$f(1,1) = 2 \times d(L_t^1, L_t^1) \quad (5.\ 68)$$

$$f(ti_k, tj_k) = \left\{ \begin{array}{l} f(ti_k - 1, tj_k) + d(L_t^{ik}, L_t^{jk}) \\ f(ti_k - 1, tj_k - 1) + 2 \times d(L_t^{ik}, L_t^{jk}) \\ f(ti_k, tj_k - 1) + d(L_t^{ik}, L_t^{jk}) \end{array} \right\} \text{ avec } \left| \begin{array}{l} ti_k = 1 \ldots n_i \\ tj_k = 1 \ldots n_j \end{array} \right. \quad (5.\ 69)$$

$$D_{Rs}\left[Rs(i), RS(j)\right] = \frac{1}{n_i + n_j} f(n_i, n_j) \quad (5.\ 70)$$

Où $f(n_i, n_j)$ est la distance cumulée le long de chemin optimal allant du point $(1,1)$ au point (n_i, n_j). f est évaluée sur tout le domaine par un parcours colonne par colonne ou ligne par lignes en partant du point $(1,1)$.

5.5.5 Principe de la classification automatique des documents

Nous représentons un ensemble R de N documents dans un graphe $G_{\geq SDT}=(V=\{v1,...,vj\}, E_{\geq SDT})$ où chaque sommet correspond à un document. Deux sommets v_i et v_j sont alors adjacents si et seulement si la distance DT entre les documents i et j est supérieure strictement à un seuil S_{DT}. Le mécanisme d'optimisation de ce seuil est détaillé dans le chapitre 4 (section 4.5.2.1). L'adjacence entre les sommets peut être donnée par :

$$E\left[v_i, v_j\right]=\begin{cases}1 & \text{si } DT(v_i,v_j)>S_{DT}\\ 0 & \text{sinon}\end{cases} \quad (5.\ 71)$$

Afin de ranger les éléments de l'ensemble de documents R dans des classes homogènes, on applique l'algorithme de b-coloration décrit dans le chapitre 4 (Procédures 2, 3, 4, 5 et 6) sur le graphe seuil $G_{\geq SDT}$. Cette b-coloration permet d'affecter à chaque sommet de $G_{\geq SDT}$ une couleur de telle sorte que deux sommets adjacents (paire de documents dont la dissimilarité est supérieure au seuil S_{DT}) ne doivent pas avoir la même couleur, et que pour chaque classe de couleur, il doit exister au moins un sommet dominant (sommet qui est adjacent à au moins un sommet dans chacune des autres couleurs). Une classification associée à chaque valeur de seuil S_{DT} est alors retournée avec un critère d'évaluation supervisé de la qualité de cette classification. La meilleure classification retenue correspond au seuil qui permet d'assurer une qualité maximale de classification ψ traduite par la formule suivante :

$$S_{DT}^{Optimal} = \underset{Si\in[\text{Smin, Smax}]}{\arg max}\ (\psi(S_i)\} \quad (5.\ 72)$$

Le critère ψ permet de comparer localement et globalement le résultat d'une coloration (ou classification) C avec la coloration de référence C_{ref} appelée vérité terrain. Cette vérité est constituée par association manuelle d'une étiquette de classe à chaque sommet (document). Le détail de ce mécanisme est exposé au chapitre 4, section 4.2.1. Nous adaptons donc à tous ces objectifs le critère de Martin et al dans [MAR01] de la façon suivante :

$$\psi(s_{DT}) = Mg\left(C(G_{\geq SDT}), C_{ref}(G_{\geq SDT})\right) = \frac{1}{n}\sum_{i=1}^{n}\min\left\{E_{RL}\left[c(i), c_{ref}(i)\right], E_{RL}\left[c_{ref}(i), c(i)\right]\right\} \quad (5.\ 73)$$

avec E_{RL} erreur de raffinement local définie comme suit :

$$E_{RL}\left[c(i), c_{ref}(i)\right] = \frac{card[L(c(v_i))] - card[L(c(v_i)) \cap L(c_{ref}(v_i))]}{card[L(c(v_i))]} \quad (5.\ 74)$$

avec $L(c(v_i))$ ensemble des sommets de G qui ont la même couleur que le sommet v_i, $L(c_{ref}(v_i))$ l'ensemble des sommets de V qui ont la même couleur que le sommet i. $C_{ref}(i)$ est la couleur de référence de sommet i.

Le critère de qualité Mg sous sa forme finale tient compte des informations globales sur les sommets mal colorés ou confondus et permet de rendre compte, classe par classe, des erreurs de classification estimées par l'indicateur local E_{RL}.

5.5.6 Mécanismes d'apprentissage embarqués

Dans cette étape on fournit à la machine d'apprentissage une base R de $N=512$ documents répartis en 14 classes.

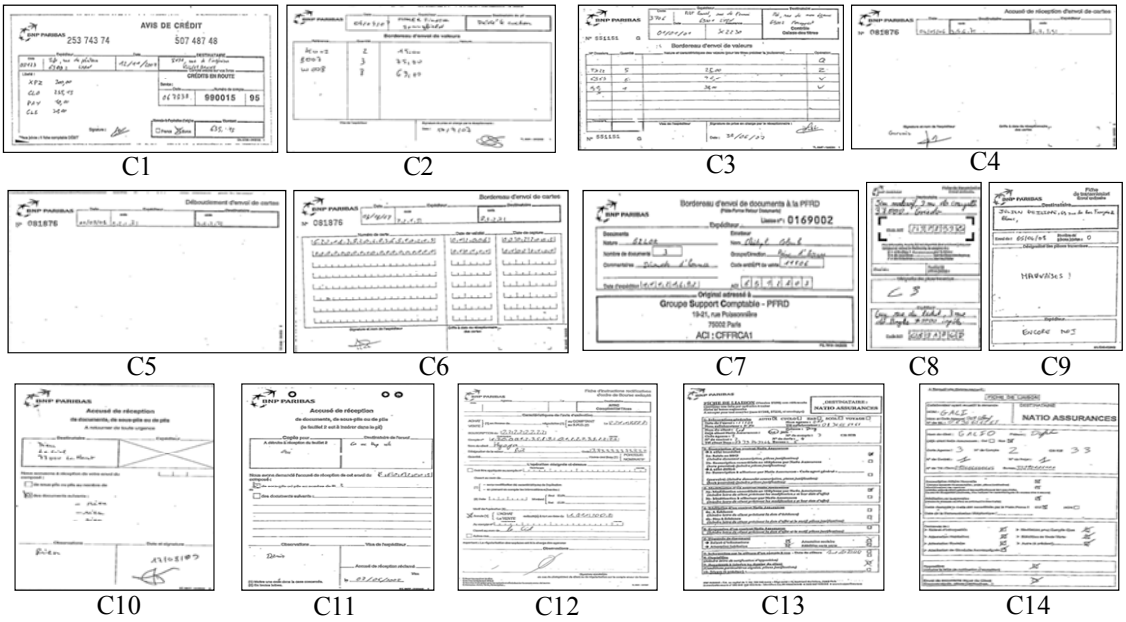


Figure 5. 42 : Exemple de différentes classes de documents.

La répartition (étiquetage) préalable des documents de la base d'apprentissage sur les 14 classes est donnée dans le tableau suivant :

Classes	Documents		Classes	Documents
C1	1 - 64		C8	289 - 320
C2	65 - 128		C9	321 - 352
C3	129 - 160		C10	353 - 384
C4	161 - 192		C11	385 - 416
C5	193 - 224		C12	417 - 448
C6	225 - 256		C13	449 - 480
C7	257 - 288		C14	481 – 51 2

Tableau 5.3 : Répartition des documents de la base d'apprentissage.

L'algorithme d'apprentissage utilise donc d'une façon itérative la technique de classification automatique exposée à la section 6. L'objectif

est de ranger par b-coloration de graphe les documents de la base d'apprentissage dans des classes homogènes.

Dans cette application, nous disposons de 14 classes précisément.

La courbe suivante montre les mesures ψ de la qualité de classification (par évaluation supervisée) pour chaque valeur de seuil d'adjacence qui varie dans l'intervalle] 0, 1[avec un pas 0.02.

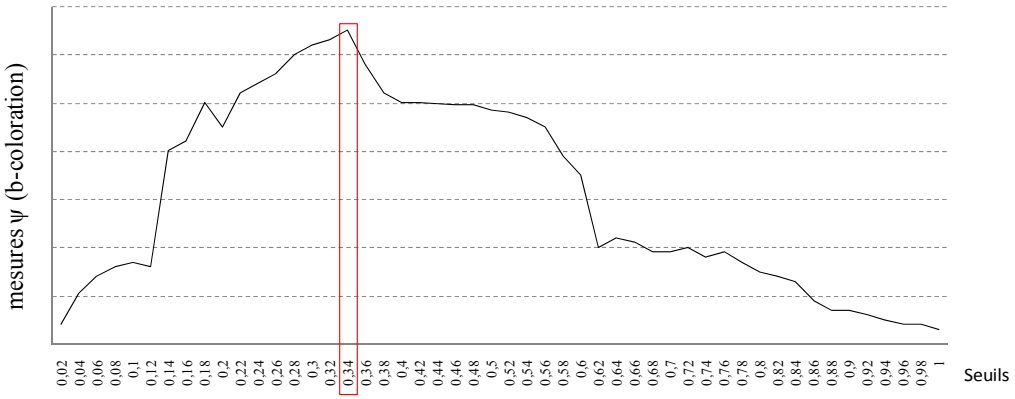


Figure 5. 43 : La qualité de la classification associée à chaque seuil, le pic dans la courbe correspond au seuil qui offre une qualité de classification optimale ($S_{DT}=0.34$).

Le résultat de la b-coloration de qualité maximale associé au seuil $S_{DT}=0.34$ représente le résultat final de la classification des documents de la base d'apprentissage finale (c'est la classification qui offre une erreur de classification minimale).

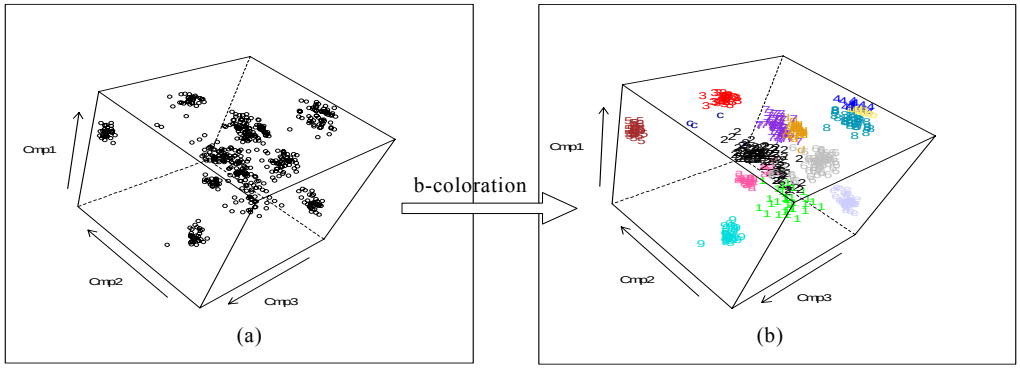


Figure 5. 44 : (**a**) Représentation de 512 documents dans l'espace de caractéristiques, (**b**) émergence des 14 classes par b-coloration ($S_{DT}=0.34$).

À l'issue de l'étape de b-coloration associée au seuil optimal ($S_{DT}=0.34$), on récupère automatiquement un jeu de N^* sommets dominants

(représentants des classes) $R^* = \{R_1^*,...,R_{N^*}^*\}$ qui seront utilisés pour reconnaître le type du document en temps réel.

5.5.7 Comparaison de la pertinence de l'approche de classification par b-coloration :

Nous avons comparé les performances de notre méthode de classification par b-coloration de graphe avec d'autres méthodes de classification sur la même base d'apprentissage. Il existe une quantité importante de méthodes pour la classification de documents. Toutes sont issues des recherches sur l'apprentissage. L'étape de représentation des documents est essentielle quelle que soit l'approche choisie : la plupart des méthodes nécessitent de représenter chaque document sous la forme d'un vecteur (type attribut/valeur). Aussi, appliquées à la classification de documents, les méthodes peuvent se révéler très lentes puisqu'il est courant de traiter plusieurs centaines (voire milliers) de composantes connexes pour chaque document.

Nous avons choisi la méthode K-means très couramment utilisée dans ce type d'applications et la méthode SVM (non linéaire basée sur un noyau gaussienne) très performante sur de grandes bases d'images (voir chapitre 3, section 3.2).

Pourquoi le k-means ?

Les approches par k-means sont simples à mettre en œuvre et facilement compréhensibles. Elles sont de complexité relativement faible par rapport à d'autres méthodes de classification (complexité en *O(k.n)*, où *n* et *k* sont respectivement le nombre d'objets à classer et le nombre de classes). Dans ce type de méthodes le nombre de classes doit être fixé au début : elles ont une très mauvaise capacité à catégoriser des données bruitées ou proches de plusieurs classes en même temps. Le résultat dépend fortement du tirage initial des points représentant les centres des classes.

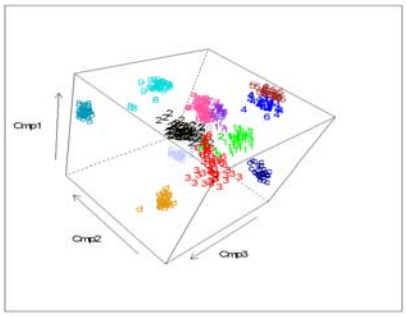


Figure 5. 45 : Représentation des 14 classes formées par Kmeans sur la base d'apprentissage.

Pourquoi le SVM?

Les SVM sont plus évoluées par rapport aux Kmeans, lorsque les classes son non linéairement séparables, elles consistent à projeter les données dans un espace de grande dimension par une transformation basée sur une fonction noyau gaussien. Dans cet espace transformé, les classes sont séparées par des classifieurs linéaires qui maximisent la marge. La complexité d'un classifieur SVM va donc dépendre non pas de la dimension de l'espace des données, mais du nombre de vecteurs supports nécessaires pour réaliser la séparation, donc de la taille de l'ensemble d'apprentissage. Par ailleurs, ces méthodes exigent une étape fastidieuse d'étiquetage de tous les documents de la base d'apprentissage, procédure qui devient encore très difficile lorsqu'on veut effectuer un apprentissage incrémental.

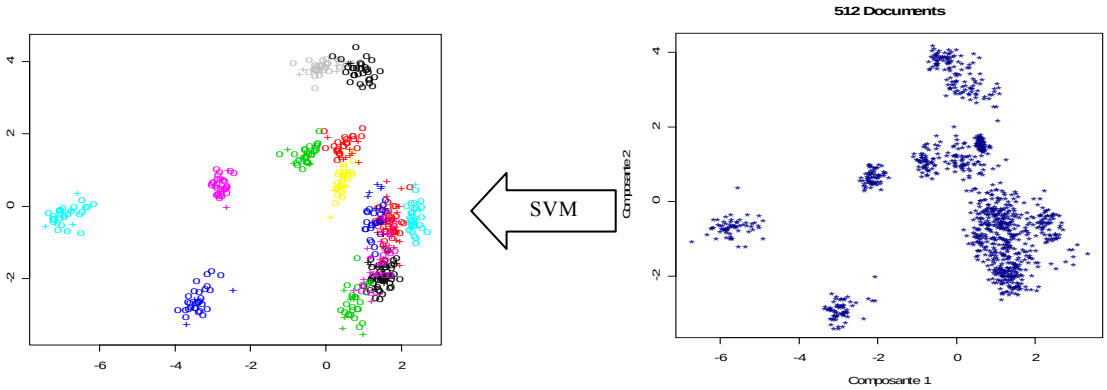


Figure 5. 46 : Projection des documents sur les axes principaux 1, 2. 14 classes ont été formées par SVM à partir de la base d'apprentissage (les 112 vecteurs supports sont représentés avec le signe « + »).

Nous avons utilisé la mesure ψ pour évaluer le taux de confusion, la pertinence et la précision de la classification de 512 documents de la base d'apprentissage obtenue par chacune des trois méthodes (KMeans, SVM et b-coloration). Plus cet indice est proche de 100%, plus la classification est correcte. L'histogramme suivant montre que la b-coloration donne une meilleure classification par rapport aux deux autres méthodes.

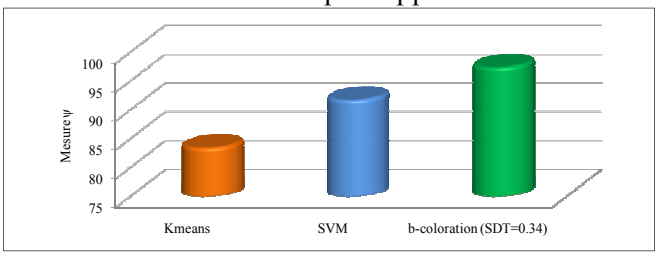


Figure 5. 47 : Comparatif des méthodes de classification en utilisant la mesure de qualité ψ pour chaque méthode (Ψ(Kmeans)=83.32%, ψ(SVM)=91.60%, ψ(b-coloration, S_{DT}=0.34)=97.32%)

5.5.8 Reconnaissance du type de document

5.5.8.1 Rappel des scénarios testés

La phase de reconnaissance en temps réel du type d'un document passant dans une chaîne de tri exploite le résultat de l'apprentissage par b-coloration sous forme de représentants de classes (sommets dominants). Pour effectuer cette reconnaissance, nous comparons les résultats de reconnaissance obtenus selon les trois scenarios que nous avons présentés en détails dans le chapitre 4 (section 5.3), avec :

Scénario 1 : distance minimale entre classes (utiliser le sommet de la plus grande dominance comme représentant des classes).

Scénario 2 : approche barycentrique (chaque classe est représentée par barycentre de ses sommets dominants).

Pour les deux premiers scénarios la fonction de décision est presque la même. Étant donné un document d'entrée *T(i),* l'objectif du système de reconnaissance est de comparer sa description avec celles de tous les représentants des classes (sommets de plus grande dominances ou les barycentres des sommets dominants de chaque classe) de R^* issues de la phase d'apprentissage. L'algorithme d'appariement reconnaît en temps réel le type de document *T(i)* à partir du type le plus proche dans R^* de la façon suivante :

$$Type\left[T(i)\right] = \begin{cases} \text{Rejet si } \underset{k=1\ldots N*}{\arg\min}\left(DT\left[T(i), R_k^*\right]\right) > S_{DT} \\ Type(R_k^* \mid \underset{k=1\ldots N*}{\arg\min}\left(DT\left[T(i), R_k^*\right]\right) \text{ sinon} \end{cases} \quad (5.\ 75)$$

Le seuil d'adjacence S_{DT} permet aussi de délimiter les connaissances du classifieur pour rejeter les documents qu'il n'a pas appris à reconnaître.

A titre d'illustration, l'exemple de la figure suivante montre deux documents (*T1* et *T2*) à reconnaitre en utilisant les sommets dominants numérotés de *1 à 14* (représentants de *14* classes) obtenus durant l'apprentissage par b-coloration. Le document *T1* est plus proche du sommets dominant 8 avec une distance inférieure à S_{DT} : il est donc reconnu comme un document appartenant à la $8^{ème}$ classe. La distance du document *T2* par rapport aux sommets dominants les plus proches est supérieure à S_{DT} : le document *T2* doit donc être rejeté par le système.

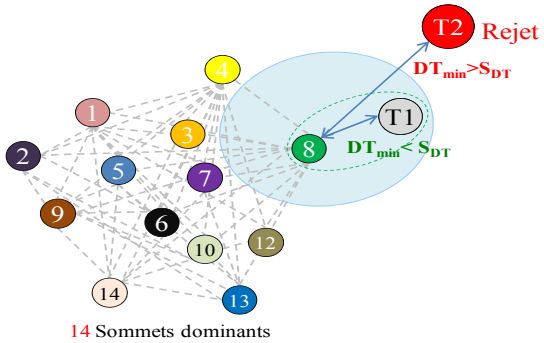


Figure 5. 48 : Exemple de reconnaissance ou de rejet de documents en utilisant les représentants des classes issues de l'apprentissage par b-coloration.

Scénario 3 : choix d'une fonction de densité de voisinage. Au lieu d'utiliser le barycentre des dominants ou le sommet le plus dominant comme unique prototype d'une classe, la méthode du plus proche voisin fait intervenir les k_d sommets les plus dominants de chaque classe (Expérimentalement k_d=5).

5.5.8.2 Évaluation de la reconnaissance et du rejet

Nous avons testé les trois scénarios avec une base de test de 576 documents répartis en 14 classes dont le type a été appris et 2 classes dont le type n'a pas été appris (tableau 5.4).

Classe	N° de document		Classe	N° de document
C1	1 – 64		C8	289 – 320
C2	65 – 128		C9	321 – 352
C3	129 - 160		C10	353 - 384
C4	161 - 192		C11	385 - 416
C5	193 - 224		C12	417 - 448
C6	225 - 256		C13	449 - 480
C7	257 - 288		C14	481 - 512
C15	513-544		C16	545-576

Tableau 5.4 : Répartition des documents de la base de test sur les 16 classes de 576 documents. 14 classes apprises (**1-14**) et 2 classes de rejet non apprises (**15 et 16**).

Les courbes de la figure suivante montrent les taux de reconnaissance sur les 14 classes connues et leurs taux de rejet sur les 2 classes inconnues selon les trois scénarios. Les courbes montrent que le troisième scénario améliore le taux de reconnaissance par rapport aux deux premiers en réduisant notamment les erreurs d'affectation et offrant une meilleure décision de rejet sur les documents dont le type n'a pas été appris lors de l'étape d'apprentissage.

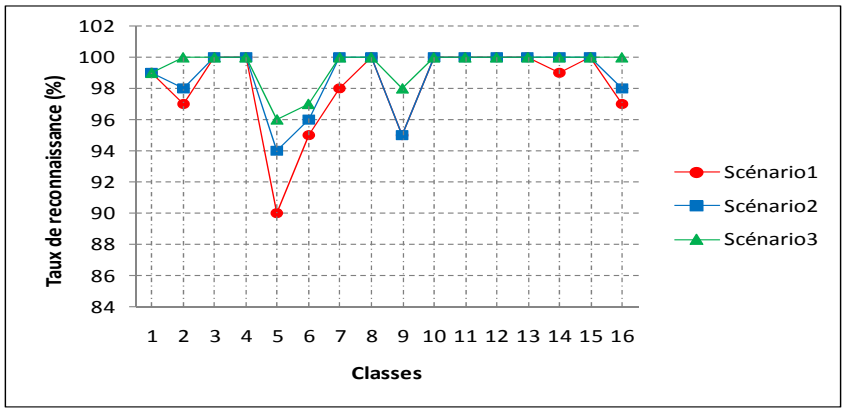


Figure 5. 49 : Comparatif de performances de reconnaissance des trois scénarios.

Nous avons finalement comparé les performances de reconnaissances par notre méthode (qui utilise le scénario 3) par rapport aux méthodes basées sur les Kmeans et les SVM. Les courbes suivantes montrent leurs taux de reconnaissance sur les 14 classes connues et leurs taux de rejet sur les 2 classes inconnues. La b-coloration donne de meilleures performances aussi bien au niveau de la reconnaissance qu'au niveau des rejets. On remarque que le système de reconnaissance basée sur Kmeans n'arrive pas à reconnaître les classes C_6 et C_{10}. Ceci revient à la confusion de la classe C_6 avec la classe C_3 et de la classe C_{10} avec la classe C_{11} à cause de similitude de leurs structures physiques. La reconnaissance basée sur les SVM présente quelques confusions remarquables au niveau de la classe 6 alors que la b-coloration montre une grande fiabilité.

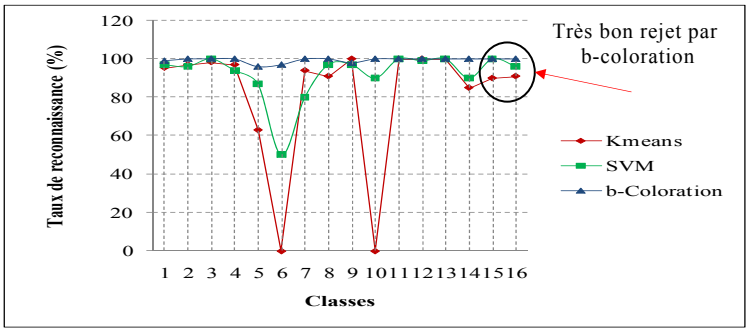


Figure 5. 50 : Comparatif des trois classifieurs.

La courbe suivante montre les temps moyens nécessaire pour binariser, extraire la structure physique et reconnaitre la nature des documents de chacune de classes. Pour des documents de grande complexité (classes C_{13}) le temps ne dépasse pas les *480 ms* sur une machine de *1Go* de *RAM* de vitesse *1.6 GHZ*. Sur des machines plus récentes ce temps peut être divisé par quatre.

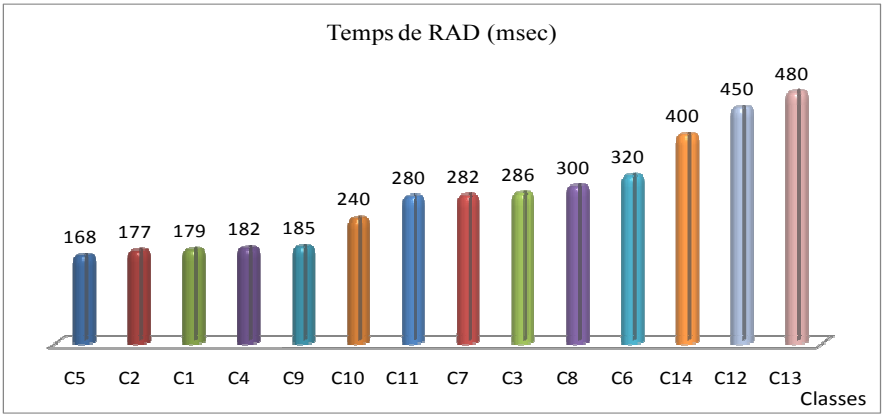


Figure 5. 51 : Temps nécessaire à toutes les étapes de reconnaissance automatique du type de documents.

5 .5.9 Apprentissage incrémental

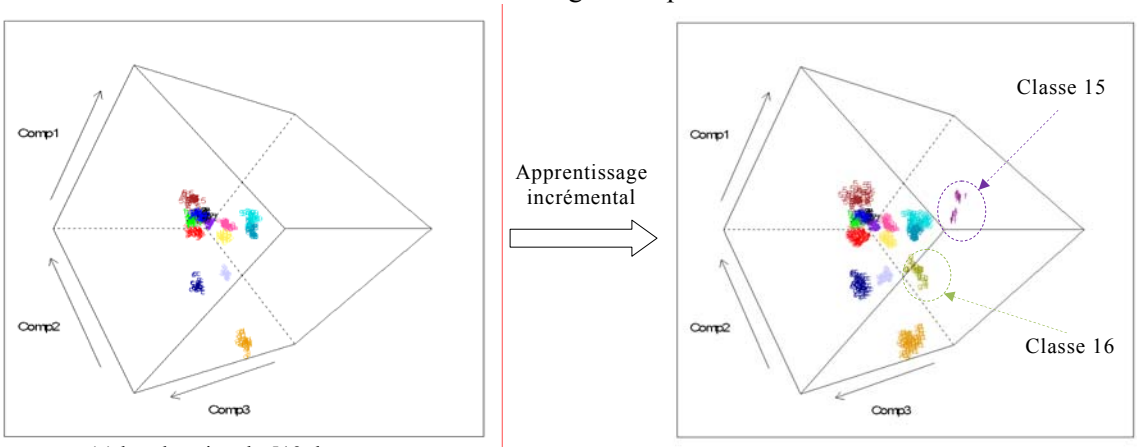
L'apprentissage incrémental qui est intégré à notre système repose lui aussi sur le principe de b-coloration. Il vise à profiter du flux de documents entrants (passant dans la chaîne de tri) pour enrichir la base d'apprentissage existante. Ceci permet d'intégrer dans cette base des nouveaux documents qui auraient naturellement été rejetés et d'élargir les connaissances du système de lecture pour reconnaître des nouvelles classes. Notre approche de mise à jour de l'apprentissage repose sur un processus incrémental, respectant les critères imposés par la b-coloration. Ce mécanisme d'apprentissage incrémental est détaillé au chapitre 4 (Procédures 8 et 9, section 5.2). Pour évaluer cette approche en terme de séparabilité entre classes, nous alimentons de deux manières différentes le système d'apprentissage incrémental par 576 documents de la base de test :

- tout document (représenté par le sommet v_{n+1}) qui était bien *reconnu* est considéré comme étant directement *attribuable à une classe existante* $C_i \in C$ (formée durant le premier apprentissage). On applique sur le sommet v_{n+1} la procédure 8 (Insertion_sommet_reconnu(v_{n+1} , C_i, C , $G_{\geq SDT}$)).

- tout document (représenté par le sommet v_{n+1}) *rejeté* par le système est considéré comme étant *attribuable à une nouvelle classe créée* respectant le critère de dominance entre les classes. On applique sur le sommet v_{n+1} la procédure 9 (Insertion_sommet_rejeté (v_{n+1} , $G_{\geq S}$)).

La figure suivante montre la représentation des 576 documents de la base test « b-colorés » et associés par apprentissage incrémental au résultat de premier apprentissage (section 5). L'apprentissage incrémental permet une adaptation automatique avec les nouveaux documents injectés et une mise à jour autonome des classes existantes (C_1-C_{14}). Il a permis de

créer deux nouvelles classes (C_{15} et C_{16}) à partir d'éléments rejetés durant la reconnaissance basée sur le premier apprentissage. À l'issue de cet apprentissage, le système envoie un message au superviseur lui signalant la présence de deux nouvelles classes de documents de volume important. Ce message permet au superviseur d'accepter l'intégration de ces deux nouvelles classes dans le processus de reconnaissance et de leur associer un ensemble de critères nécessaires au tri. Cette étape va permettre au système de tri de classer désormais une gamme plus étendue de documents entrants.



(a) b-coloration de 512 documents

(b) b-coloration de 1088 documents (512+576)

Figure 5. 52 : (a) premier apprentissage sur la base de 512 documents (résultat de b-coloration de graphe), **(b)** résultat de l'apprentissage incrémental par insertion de 576 documents de la base de test. Les classes 15 et 16 sont les deux nouvelles classes crées automatiquement à partir des documents rejetés.

5.5.10 Conclusion

Nous avons présenté une nouvelle méthode de reconnaissance du type de documents basée sur la coloration hiérarchique des graphes. Notre méthode utilise une représentation issue de la description de la structure physique des documents. La coloration hiérarchique de graphe a été introduite dans la phase de segmentation pour augmenter la robustesse aux composantes parasites considérées comme facteurs d'erreur des méthodes classiques de segmentation. La b-coloration a été introduite dans la phase d'apprentissage pour garantir un excellent partitionnement entre catégories de documents. Grâce au nombre restreint de règles dont elle dispose, cette nouvelle technique répond à une large variété de documents, offre une vraie représentation des classes par documents dominants et garantit une meilleure disparité interclasses. De plus, nous avons pu augmenter la cohérence entre les différentes phases de la RAD par l'exploitation de la b-coloration et réduire les temps de calcul.

Nous avons également présenté un nouveau modèle d'apprentissage incrémental basé sur la b-coloration. Grâce à ce modèle, le

système de la RAD est capable d'apprendre de nouveaux types de documents à partir de très peu d'exemples. Il peut s'adapter et s'améliorer à partir de chaque nouveau document passant dans la chaîne de tri. Durant la phase d'apprentissage, nous avons intégré une approche de l'évaluation de la classification afin d'améliorer la classification elle-même et conduire ainsi à de meilleurs taux de reconnaissance de type de document. Notre processus d'optimisation du seuil d'adjacence portant sur la maximisation de la qualité de la classification peut être considéré comme similaire à l'optimisation des poids synaptiques d'un réseau de neurones pour minimiser l'erreur de l'apprentissage. Chaque neurone formel possède des caractéristiques propres, en particulier un seuil de déclenchement, assimilable à un poids synaptique dont le dépassement implique la décharge du neurone, c'est-à-dire la transmission d'une information de sortie. La fonction seuil interne à chaque neurone formel assure que la valeur de sommation des potentiels pré-synaptiques ne dépassera pas certaines limites raisonnables.

De façon similaire, il est tout à fait envisageable de comparer le processus de b-coloration du graphe avec le principe d'apprentissage des SVM : les sommets dominants de la b-coloration jouant un rôle relativement comparable à celui des vecteurs de supports des SVM.

Le constat que nous pouvons faire de la grande généricité et de la grande simplicité de notre approche de coloration (et de b-coloration) de graphe à toutes les étapes de reconnaissance et d'apprentissage fait de notre méthode de RAD un outil réellement performant.

5.6 Application de la b-coloration de graphes au service de la localisation du bloc adresse

Nous travaillons ici sur un cas particulier de documents : il s'agit de courriers d'entreprises de structure irrégulière qui font l'objet d'une reconnaissance du bloc contenant l'adresse de destination, qu'elle soit imprimée ou manuscrite, en vue de son tri. Cette application concerne aussi bien des lettres que des colis ou des plis (lettres de grand format, magazine, revues, journaux, documents publicitaires). La localisation automatique des adresses consiste à rechercher dans chaque bloc des groupes de lignes d'écriture ou de caractères organisés en un ensemble présentant les caractéristiques typiques d'une adresse (position sur l'enveloppe, nombre de lignes, taille des lignes, espacement, alignement). Nous avons présenté dans le chapitre 3 (section 3.3) les différentes contraintes qui accompagnent généralement les mécanismes de tri de courriers ainsi que les méthodes es-

sentielles de LBA. Nous les avons classées selon leur modalité d'action à travers différentes approches allant de l'émergence des blocs à la décision. Nous avons également expliqué, su des images de courriers de structures complexes, comment les méthodes basées sur l'apprentissage se présentent comme des solutions alternatives robustes permettant au système de se libérer des tâches fastidieuses et coûteuses de réglage manuel de paramètres ou de règles à appliquer.. La complexité de ces documents est souvent liée à la présence de figures, des logos, de lignes de texte publicitaires des codes à barres et de tableaux très proches à la zone d'adresse. Après l'extraction et la description des blocs présents sur une image de courrier, la phase de reconnaissance qui utilise le résultat de l'apprentissage repose sur une prise de décision importante portant sur l'inspection de l'ensemble des données obtenues pour identifier le bloc adresse parmi plusieurs candidats. C'est dans ce cadre que nous avons orienté notre proposition visant plus de robustesse, de souplesse et de performances en temps et en précision par rapport à l'existant.

Nous avons présenté dans le chapitre 4 (section 4.4.2.2) notre formulation de la problématique de LBA portant sur le concept de b-coloration de graphes jamais exploité dans un tel contexte. Nous avons présenté également dans le même chapitre les fondements de la conception de la partie apprentissage issue de l'adaptation de la b-coloration appliquée à la LBA ainsi que les différents scénarios de d'identification du bloc adresse. Nous détaillons dans cette partie les différentes étapes de conception complète du système de localisation de bloc adresse qui procède aux enchaînements suivants :

- Analyse hiérarchique de la structure physique (formation et stratégie de description des blocs par la méthode décrite aux sections 5.2 et 5.3).
- Apprentissage pour la localisation de bloc adresse.
- Reconnaissance (identification) du bloc adresse parmi plusieurs candidats.

5.6.1 Analyse hiérarchique de la structure physique

5.6.1.1 Rappel de l'approche pyramidale de caractérisation et de formation des blocs

L'extraction de la structure physique des documents consiste à segmenter l'image de courrier en blocs de texte par coloration hiérarchique de graphe et à déterminer les propriétés caractéristiques de chacun des blocs afin de distinguer la plus justement possible un bloc adresse parmi un ensemble de blocs candidats. Reconnaître ensuite en temps réel un bloc inconnu consiste à déterminer ses propriétés, à les comparer à celles des

blocs de référence (représentants des classes issus de la phase d'apprentissage par b-coloration de graphe sur une base représentative de blocs) et à prendre une décision de reconnaissance.

Nous avons présenté dans la section précédente les différentes étapes de segmentation des images de courrier en blocs de texte homogène.

Nous avons mis en avant le principe de *coopération hiérarchique* entre les phases de description et de segmentation des blocs. La description représente l'ensemble des mesures (caractéristiques) effectuées sur trois niveaux (les CCs, les lignes et les blocs de texte) reflétant la position et l'identité de la forme générale de chaque bloc. La progression dans la hiérarchie permet à chaque niveau d'utiliser toutes les informations exprimées dans les autres niveaux. On tire ainsi partie des avantages des deux phases, et on acquiert des connaissances plus précises sur le contenu de l'image jusqu'à l'obtention d'une description globale de tous les blocs. Chaque jeu de caractéristiques peut être visible à différents niveaux de perception. Par exemple, l'alignement des lignes de texte n'est pas perçu au même niveau que l'espacement des caractères, ni celui de la position des blocs sur l'enveloppe (voir la figure 5.53).

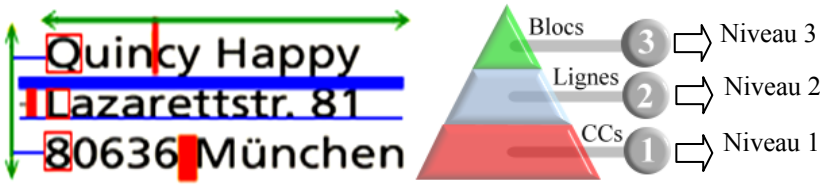


Figure 5. 53 : Extraction hiérarchique des caractéristiques et perception des caractéristiques aux différents niveaux de la pyramide.

5.6.1.2 Caractérisation hiérarchique complète des contenus

Une fois toutes les propriétés extraites de chaque bloc à partir des trois niveaux de la hiérarchie, on les exprime sous la forme d'une représentation vectorielle qui servira de base aux étapes ultérieures d'apprentissage et de reconnaissance de bloc adresse. La description complète de chaque bloc de texte (candidat au label « bloc adresse ») noté $B_i \in C_B$ avec $i = 1...n_B$ est basée sur l'ensemble de 21 caractéristiques suivantes. Elles sont rassemblées en trois grands groupes : les caractéristiques morphologiques traduisant les propriétés de nature plus topologiques des blocs, les caractéristiques d'homogénéité des blocs permettant d'informer sur les densités moyennes des contenus, les propriétés d'alignement renseignant sur la localisation du bloc sur la page.

Les caractéristiques morphologiques

1) Nombre de composantes connexes de texte d'$i^{ème}$ bloc $n_{cct}(B_i) = card(cct_k \in B_i)$

2) Nombre de lignes de $i^{ème}$ bloc $n_L(B_i) = card(L_j \in B_i)$

3) L'aire $Ar(B_i) = H(B_i) \times W(B_i)$

4) L'excentricité $\zeta(B_i) = W(B_i) / H(B_i)$

5) La densité du $i^{ème}$ bloc $Dens(B_i) = \frac{1}{Ar(B_i)} \sum_{j=1}^{nL(Bi)} Dens(L_j) \times Ar(L_j)$ avec $L_j \in B_i$

6) La hauteur du $i^{ème}$ bloc $H(B_i) = y_f^{B(i)} - y_d^{B(i)}$.

7) La largeur du $i^{ème}$ bloc $W(B_i) = x_f^{B(i)} - x_d^{B(i)}$

8) La hauteur moyenne des CCs de bloc $\mu_{Hcct}(B_i)$

9) La largeur moyenne des CCs $\mu_{Wcct}(B_i)$.

10) La largeur moyenne des lignes $\mu_{WL}(B_i)$

Les caractéristiques d'espacement

11) L'espace interlignes moyenne

12) Le plus grand espace interlignes

13) Le plus petit espace interlignes

Les caractéristiques d'homogénéité (distinction de texte manuscrit et de texte imprimé)

14) L'écart type des aires $\sigma_{Arcc}(B_i)$ des composantes

15) L'écart type des excentricités $\sigma_{\zeta cct}(B_i)$ des composantes.

16) L'écart type des hauteurs des Ccs $\sigma_{Hcct}(B_i)$

17) L'écart type des largeurs $\sigma_{Wcct}(B_i)$

18) L'écart type des hauteurs des lignes $\sigma_{HL}(B_i)$

$$\sigma_{Hcct}(B_i) = \sqrt{\frac{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} W(cct_j)\left[H(cct_j) - \bar{H}_{cct}(B_i)\right]^2}{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} W(cct_j)}} \text{ avec } \bar{H}_{cct}(B_i) = \frac{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} W(cct_j) \times H(cct_j)}{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} W(cct_j)} \quad (5.76)$$

$$\sigma_{Wcct}(B_i) = \sqrt{\frac{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} \left[W(cct_i) - \bar{W}_{cct}(B_i)\right]^2}{n_{cct}(B_i) - 1}} \quad \text{et} \quad \bar{W}_{cct}(B_i) = \frac{\sum_{j=1,cctj\in Bi}^{ncct(Bi)} W(cct_j)}{n_{cct}(B_i)} \quad (5.77)$$

$$\sigma_{HL}(B_i) = \sqrt{\frac{\sum_{j=1,Lj\in Bi}^{nL(Bi)} W(L_j)\left[H(L_j) - \bar{H}_L(B_i)\right]^2}{\sum_{j=1,Lj\in Bi}^{nL(Bi)} W(L_j)}} \quad \text{avec} \quad \bar{H}_L(B_i) = \frac{\sum_{j=1,Lj\in Bi}^{nL(Bi)} W(L_j) \times H(L_j)}{\sum_{j=1,Lj\in Bi}^{nL(Bi)} W(L_j)} \quad (5.78)$$

Les caractéristiques de position du $i^{ème}$ bloc sur l'enveloppe de largeur W et de hauteur H :

19) La position verticale de haut : $P_{Vd} = \frac{y_d(B_i)}{H}$

20) La position verticale de haut : $P_{Vf} = \frac{y_f(B_i)}{H}$

Les caractéristiques de l'alignement vertical des lignes dans le $i^{ème}$ bloc :

21) Alignement vertical des lignes de bloc (B_i).

$$Al(B_i) = \sqrt{\frac{\sum_{j=1, Lj \in Bi}^{nL(Bi)} H(L_j)\left[X_d(L_j) - \bar{X}_d(B_i)\right]^2}{\sum_{j=1, Lj \in Bi}^{nL(Bi)} H(L_j)}} \text{ avec } \bar{X}_d(B_i) = \frac{\sum_{j=1, Lj \in Bi}^{nL(Bi)} H(L_j) \times X_d(L_j)}{\sum_{j=1, Lj \in Bi}^{nL(Bi)} H(L_j)} \quad (5.\ 79)$$

L'étiquetage logique de bas niveau (en blocs de texte et non texte) issu de l'étape d'extraction de la structure physique permet d'orienter le système à chercher le bloc adresse uniquement dans l'ensemble des blocs textuels. Dans certains cas ambigu, lorsque deux blocs de texte sont identifiés comme blocs adresse avec un même score de reconnaissance, les blocs non textuels (comme les logos, des figures, le cachet) peuvent servirent de repères pour une exploitation des relations spatiales des blocs afin de choisir au final le bloc qui se positionne le mieux pour présenter la zone d'adresse. Pour cette raison, nous avons constitué notre base d'apprentissage de telle sorte que l'on dispose à la fois de blocs textuels et de blocs non textuels.

Rappelons à ce stade de l'analyse, que les blocs non textuels sont déduits par comparaison entre la carte composantes connexes $CC_M=\{FM_i\}$ des blocs bruts formée par les gradients cumulées lors de la binarisation et la carte des blocs textuels obtenue durant l'extraction de la structure physique par coloration.

Cette déduction donne un ensemble de blocs $B^{nt} = CC_M \cap C_B = \left\{B_i^{nt} \middle| i = 1...n_{Bnt}\right\}$. Chaque bloc non textuel est décrit par les caractéristiques décrites par les équations 1,3,4, 5, 6,7,8,9,14,15,16,17,18 et 20 (c-à-d : même caractéristiques que les blocs textuels mais sans celles qui sont relatives au lignes qui correspondent aux équations 2, 10,11, 12, 13,19 et 21), Et en plus en utilise les écarts types de positions des composantes connexes dans le bloc non textuel $\sigma_{xd}(B_i^{nt}), \sigma_{yd}(B_i^{nt}), \sigma_{xf}(B_i^{nt}), \sigma_{yf}(B_i^{nt})$.

5.6.2 La reconnaissance du bloc adresse

La reconnaissance du bloc adresse regroupe les deux tâches d'apprentissage et de décision qui jouent des rôles tout à fait complémen-

taires dans le processus d'identification de la nature des blocs. En effet, elles tentent, toutes les deux, d'utiliser la même description des blocs et la même mesure de distance, soit pour ranger ces blocs dans des classes homogènes lors de l'apprentissage (il s'agit là d'un problème de classification*),* soit pour attribuer chaque bloc à un représentant d'une classe lors de la décision (il s'agit d'un problème de classement).

L'ensemble des représentants des classes est le résultat de l'apprentissage par b-coloration, il représente les meilleurs modèles à utiliser pour identifier en temps réel le bloc adresse d'une enveloppe inconnue à trier. Le résultat de la décision est donc un avis sur l'appartenance ou non d'un bloc inconnu aux modèles de l'apprentissage. A partir de représentants de chacune des classes, chaque bloc inconnu doit être attribué à une classe parmi les N classes possibles issues de l'apprentissage (ou être attribué à une classe dite « de rejet » si le bloc adresse est trop éloigné pour être identifié : l'objet postal doit alors être envoyé à la station de vidéo codage et l'adresse doit être saisie manuellement). Dans la suite, nous allons présenter ces tâches en détaillant les approches qui les réalisent.

5.6.2.1 Apprentissage pour la localisation du bloc adresse

A fin de préparer une base d'apprentissage représentative nous avons sélectionné 400 blocs de plusieurs catégories (blocs adresse, timbre, logos...), issus de l'extraction de la structure physique d'une grande variété d'images de 250 enveloppes. Les blocs de la base sont répartis selon la façon suivante : 150 blocs-adresse imprimée de différents mise en formes, 100 blocs-adresse manuscrite et le reste représentant des timbres, des cachets, logos et autres blocs graphiques.

Nous construisons notre graphe d'apprentissage que l'on note $G_{>S}$ juste après la description de chaque bloc de la base par un vecteur de caractéristiques discriminant (sections 5.2, 5.3 et 5.4). Chaque sommet de $G_{>S}$ est associé à un bloc. Pour effectuer une première coloration, nous appliquons la *procédure 1* sur le graphe $G_{>S}$, quelques couleurs qui en résultent ne possèdent aucun sommet dominant. Nous nous servons, par la suite, des *procédures* (2, 3, 4, 5 et 6) pour b-colorer les couleurs non dominantes de $G_{>S}$ (voir le chapitre 4 section 4.5.2.2).

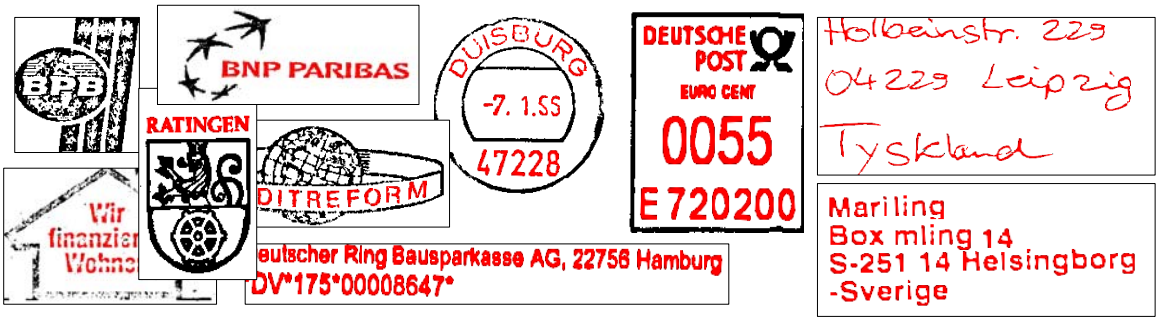


Figure 5. 54 : Exemples de différents blocs de la base (les composantes de texte sont marquées en rouge).

La dissemblance entre deux blocs de même nature (texte ou non texte) représentés par les sommets v_i et v_j est une distance euclidienne qui compare les caractéristiques de ces blocs deux à deux de la façon suivante :

$$d(v_i, v_j) = \sqrt{\sum_{k=1}^{P} \left(v_i^k - v_j^k\right)^2} \qquad (5.\ 80)$$

P est le nombre de caractéristiques utilisées pour décrire chaque bloc : il est égal à 21 pour les blocs B_i textuels et à 18 pour les blocs B_i^{nt} non textuels.

Les adjacences entre sommets qui correspondent aux blocs de la base d'apprentissage sont données par les relations suivantes :

- Si les deux blocs sont textuels :

$$E_{\geq S}\left[B_i, B_j\right] = E_{\geq S}\left[v_i, v_j\right] = \begin{cases} 1 & \text{si } d(v_i, v_j) \geq S \\ 0 & \text{sinon} \end{cases} \qquad (5.\ 81)$$

- Si les deux blocs sont non textuels :

$$E_{\geq S}\left[B_i^{nt}, B_j^{nt}\right] = E_{\geq S}\left[v_i^{nt}, v_j^{nt}\right] = \begin{cases} 1 & \text{si } d(v_i^{nt}, v_j^{nt}) \geq S \\ 0 & \text{sinon} \end{cases} \qquad (5.\ 82)$$

- Si les deux blocs sont de nature différente (on doit toujours leur associer une arrête pour qu'ils n'aient jamais la même couleur) :

$$E\left[B_i, B_j^{nt}\right] = E\left[v_i, v_j^{nt}\right] = 1 \qquad (5.\ 83)$$

Nous pouvons remarquer ici que l'étiquetage des blocs en texte ou non texte durant l'extraction de la structure physique contribue à la construction du graphe pour l'apprentissage.

L'optimisation automatique du seuil d'adjacence S est basée sur un critère d'évaluation non supervisée de la qualité de la classification que nous avons détaillée dans le chapitre 4, section 4.5.2.2. Ce critère est basé sur la combinaison des disparités intra-classes et inter-classes de chaque classification et correspond à un seuil S. Le seuil $S_{Optimal}$ qui offre une meil-

leure classification par b-coloration est celui qui maximise la qualité de classification ψ donnée par la fonction suivante :

$$S_{Optimal} = \arg\max_{0 \le Si \le 1}(\psi_{LBA}(S_i) = M_{Inter\_Classes}(C(G_{\ge Si})) + M_{Intra\_Classes}(C(G_{\ge Si})\} \quad (5.\ 84)$$

Cet auto-paramétrage de seuil permet de réaliser une classification automatique non supervisée : c-à-d sans aucune intervention préalable du superviseur de système de tri.

Ceci permet au superviseur :

a) d'éviter l'introduction préalable du nombre de classes, sachant que l'imprécision de ce nombre pourrait facilement forcer le classifieur à commettre des erreurs de classification (sur une base d'apprentissage qui possède certains blocs trop déformés).

b) d'éviter de réaliser un étiquetage fastidieux de tous les blocs de la base d'apprentissage où chaque bloc doit être associé à l'un des classes connues.

La b-coloration associée à ce mécanisme automatique d'optimisation de seuil s'adapte parfaitement au contenu de la base d'apprentissage. Sans aucune limitation préalable du nombre de classes, la b-coloration de $G_{>Soptimal}$ produit suffisamment de classes (couleurs) uniformes pour que la partition finale soit de bonne qualité.

En effet, cette classification par b-coloration intègre également les logos dans plusieurs classes avec une grande souplesse face à l'hétérogénéité des blocs. Elle permet ainsi la création d'autant de classes que nécessaire : la reconnaissance des logos de BNP Paribas par exemple permet de renforcer la localisation du bloc adresse en apportant des critères supplémentaires. Cela rend la localisation plus précise sur toutes les enveloppes qui contiennent ce logos. Les blocs adresses manuscrites se regroupent dans des classes bien séparées de celles des blocs adresses imprimées. Quelques blocs adresses de style rare comme ceux écrits en gras forment aussi leurs propres classes.

Les blocs qui sont trop déformés (coupés ou mal alignés) s'isolent dans des classes de faible effectif. Cette souplesse dans la séparation des contenus de la base d'apprentissage rend notre méthode de localisation de bloc adresse plus générique. À la fin de l'apprentissage les classes de faibles effectifs ne vont pas participer à la reconnaissance mais elles peuvent évoluer par un apprentissage incrémental. Les classes de forts effectifs (représentés avec un nombre restreint de sommets dominants) peuvent ainsi être représentées par le bloc correspondant au sommet le plus dominant de la classe : elles seront présentées au superviseur du système de tri qui leur affectera une étiquette logique (adresse imprimée, adresse manuscrite, montant d'affranchissement, texte publicitaire, timbre, logos BNP, logo

ColiPost, code à barres ou autres…) afin que le modèle soit opérationnel lors de la phase de reconnaissance, voir figure 5.55.

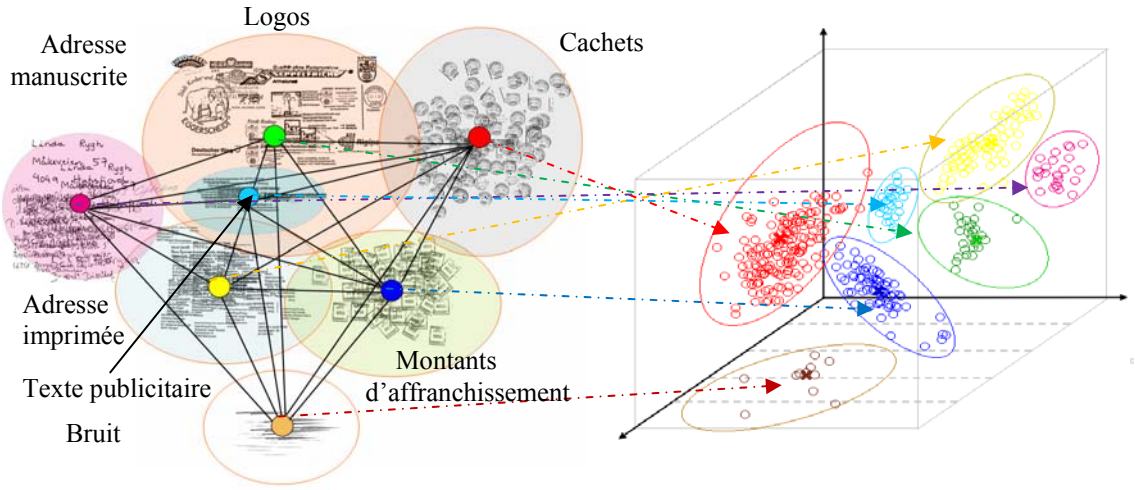


Figure 5. 55 : Séparabilité des caractéristiques, et apprentissage : classification des blocs par la b-coloration de $G_{>SOptimal}$ et détections des représentants des classes.

La courbe suivante présente la qualité ψ_{LBA} calculée en fonction des seuils S_i sur le résultat de classification par b-coloration de 400 blocs de la base d'apprentissage.

Une valeur très petite de seuil d'adjacence conduit à associer des arêtes à des blocs qui sont similaires et à générer plus de couleurs qu'il faut (cas de sur classification des blocs). Une valeur très grande de ce seuil conduit fusionné des blocs dissimilaires en générant peu de couleurs (cas de sous-classification).

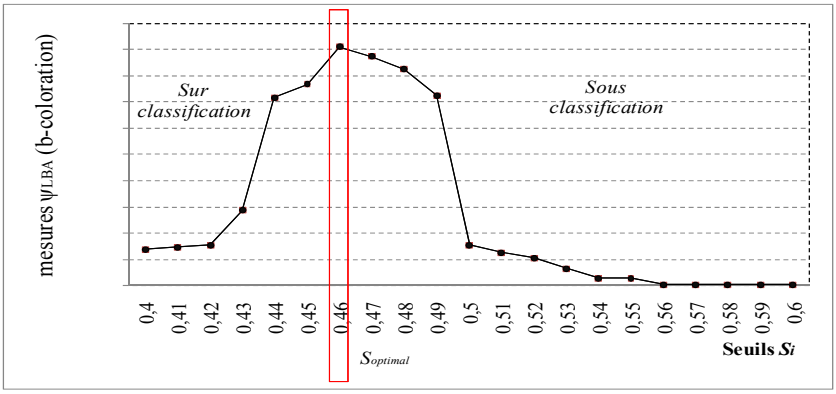


Figure 5. 56 : La qualité de la classification par b-coloration associée à chaque seuil d'adjacence $s_i \in [0.4,\ 0.6]$, le pic dans la courbe correspond au seuil qui offre une qualité de classification optimale ($S_{optimal}$=0,46).

Les représentants (sommets dominants) des classes étiquetées seront utilisés dans la phase de LBA pour identifier en temps réel le bloc adresse parmi plusieurs candidats. On remarque à la figure 5.55 que les couleurs des blocs adresse, de blocs timbre et de blocs cachet sont les couleurs les plus émergentes par rapport à la couleur de bruit et de texte publicitaire par exemple (les couleurs des blocs déformés ont un effectif très faible, elles ne sont pas présenté dans la figure 5.55).

5.6.2.2 Identification en temps réel du bloc-adresse basée sur la b-coloration des graphes

Pour sélectionner le BA dans une liste des blocs textuels candidats issus de la segmentation de la structure physique représentés par les sommets $B_i = v_i \in C_B$ avec $i = 1...n_B$ nous utilisons le troisième scénario de reconnaissance présenté dans le chapitre 4, section 4.6.3. Ce scénario consiste à faire intervenir les $k_d = 10$ sommets les plus dominants de chaque classe de blocs textuels. Cette approche améliore le taux de reconnaissance et réduit l'erreur de confusion. Pour ce faire, la distance entre chacun des sommets dominants et celle du bloc à localiser est calculée. La classe assignée au bloc est alors celle du prototype le plus proche. Les équations de reconnaissance et de rejet utilisée sont données dans les équations (4.22 et 4.23) de chapitre 4 avec $S=S_{optimal}$.

Cas de confusion entre deux ou plusieurs bloc textuels (cas rare pour le scénario 3) : Dans le cas où deux (ou plusieurs) classes ont la même densité, le système cherche à reconnaître la nature des blocs non textuelles de l'enveloppe en repérant les éventuels logos, cachets, montants d'affranchissement et figures. Ce repérage qui utilise des connaissances a priori donne une idée du type de l'enveloppe et permet d'intégrer les relations spatiales entre ces blocs émergeants et le bloc adresse. Il permet de conclure rapidement sur la nature du bloc à reconnaître par sa probabilité d'être une adresse.

5.6.3 Évaluation de la méthode

5.6.3.1 Premier test : test sur des enveloppes (complexes) rejetées par un système d'architecture standard.

Ce test a été effectué sur une base de 100 images d'enveloppes complexes rejetées par un système dont les spécificités sont détaillées ci-dessous. Les causes de rejets sont liées à la présence de :

- caractères qui se chevauchent
- lignes de texte inclinées
- textes publicitaires et de graphiques proches de la zone d'adresse.

Le module de localisation mis en œuvre dans cette architecture standard n'a pas réussi à localiser convenablement le bloc adresse ou a conclu à un rejet immédiat du document. Voici les éléments à prendre en considération et qui caractérisent ce système (pour une comparaison avec le notre) :

1) Phase 1 : La binarisation est basée sur la méthode adaptative de Sauvola : on observe le rejet de 18 enveloppes du fait d'une binarisation mal adaptée,

2) Phase 2 : L'extraction de la structure physique qui est basée sur la méthode RLSA appliquée à l'image sur plusieurs niveaux de résolution afin de faire apparaître les mots, les lignes puis les blocs : on observe le rejet de 53 enveloppes du fait d'une approche de la segmentation pas adaptées aux inclinaisons des lignes.

3) Phase 3 : L'identification du bloc adresse utilise les barycentres issus de la classification supervisée par la méthode de K-means durant l'apprentissage : on observe ici un rejet de *29* enveloppes. A ce stade, c'est le module d'identification qui est en cause.

De façon plus générale, nous avons pris pour chaque étape d'une architecture de localisation de bloc adresse les approches les plus couramment utilisées, voir figure 5.57. Pour chacune des étapes, nous avons dressé un comparatif des résultats entre les mécanismes dits « classiques » et nos propositions.

Nous pouvons voir dans les courbes de la figure suivante de quelles façons ces taux de rejets évoluaient en fonction de l'utilisation de différentes méthodes liées aux trois étapes : binarisation, extraction de la structure physique et localisation du bloc adresse.

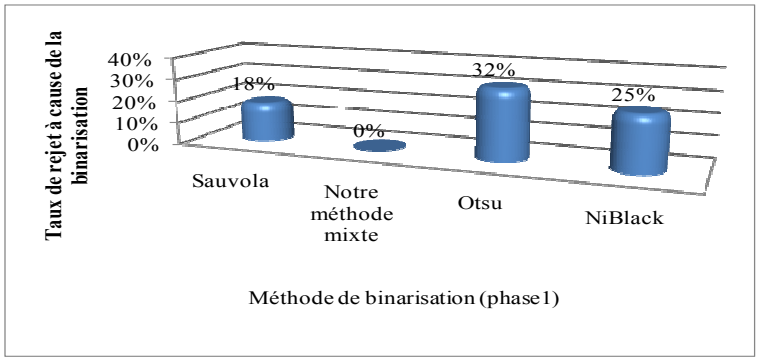
La courbe (a) montre que notre méthode de binarisation a permis d'éviter tout rejet lié à la binarisation, par rapport à une binarisation de type Sauvola, Niblack ou Otsu.

La courbe (b) montre de quelle façon notre méthode d'extraction de la structure physique par coloration hiérarchique de graphe permet de réduire le taux de rejet à 5% sur des images complexes. Le taux de rejet issu de notre approche est le plus faible par rapport à ceux associés aux trois autres méthodes : RLSA, regroupement de CCs et projection de profils.

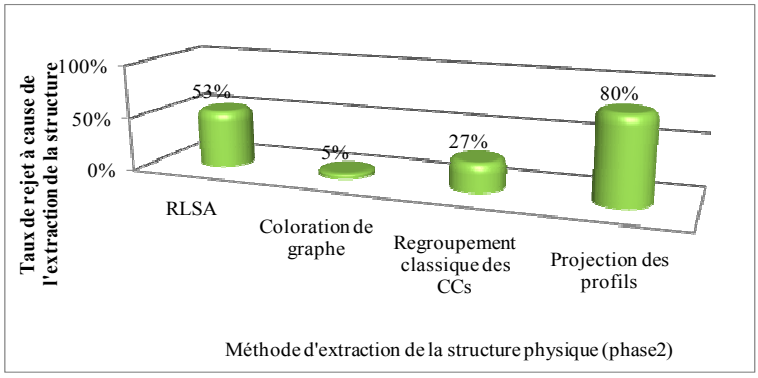
La courbe (c) enfin montre l'évolution des taux de rejet selon les mécanismes d'apprentissage utilisés. Notre méthode de b-coloration a permis de réduire considérablement ce taux par rapport aux trois autres méthodes : K-means, SVM , MLP

Ces comparaisons montrent que le triplet « Binarisation par gradients cumulés – Coloration hiérarchique de graphe – B-coloration pour

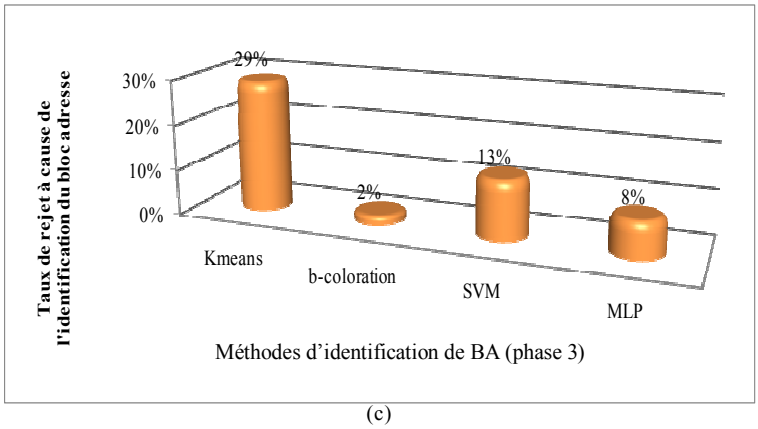
l'apprentissage » augmente de façon significative les performances d'un système de tri. La robustesse de notre proposition vis-à-vis d'un grand nombre d'irrégularités de mise en page, de dégradations diverses et de bruit montre la très bonne adéquation de nos solutions vis-à-vis d'un problème de tri. Nous avons réussi à faire passer un taux de rejet de 100% à seulement 7% avec notre approche.



(a)



(b)



(c)

Figure 5. 57 : Taux de rejets au niveau de chaque phase de LBA par différentes méthodes.

5.6.3.2 Deuxième test

Nous avons testé également notre méthode de localisation sur une base de 11500 images d'enveloppes de différents formats réparties de la façon suivante :

Bases	Base 1	Base 2	Base 3	Base 4	Base 5	Base 6	Base 7	Base 8
Tailles (enveloppes)	1500	2000	1500	1000	2000	2000	800	700
Type de texte de l'adresse	Imprimé	Imprimé	Imprimé	Imprimé	Imprimé	Imprimé	mixte	Manuscrit
Taux de bonne Localisation de l'adresse	98,93%	99,33%	99,53%	99,86%	100%	100%	98,53%	97,60%

Tableaux 5.5 : Taux de bonne LBA sur les 8 bases d'images d'enveloppes.

La figure suivante montre quelques exemples de bonne localisation de bloc adresse dans des cas difficiles :

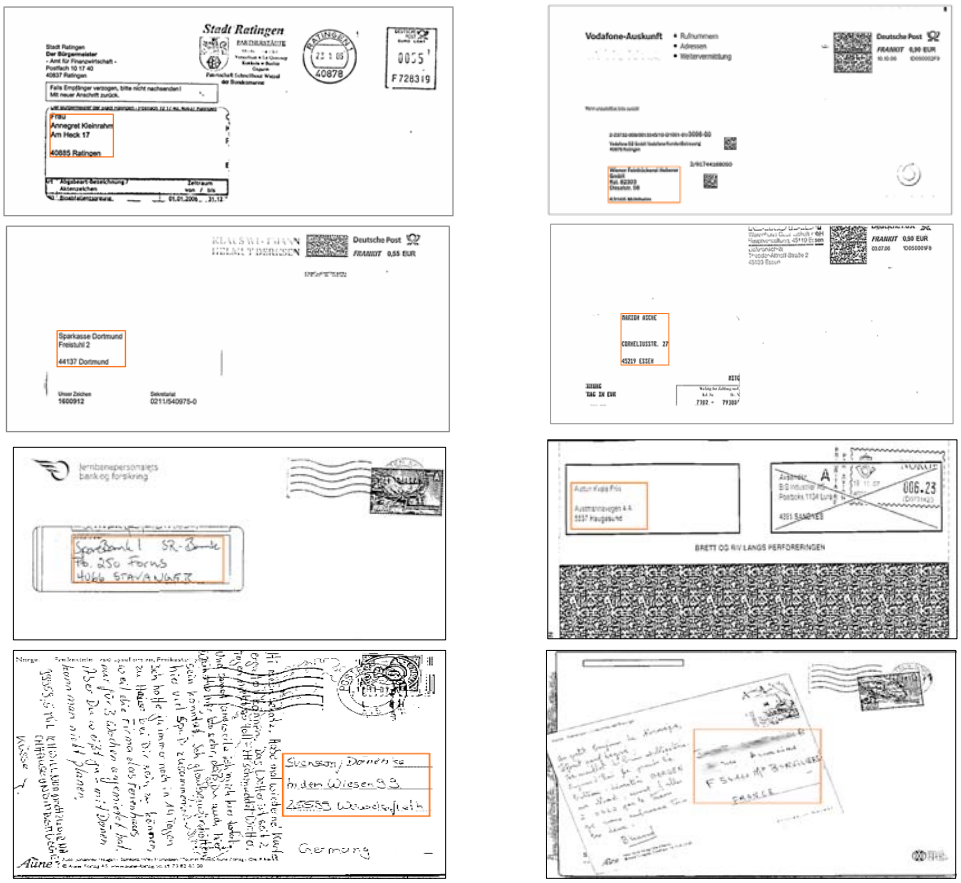


Figure 5. 58 : quelques résultats de bonne LBA par notre méthode dans des situations ambiguës, rejet sûr pour les trois méthodes classiques.

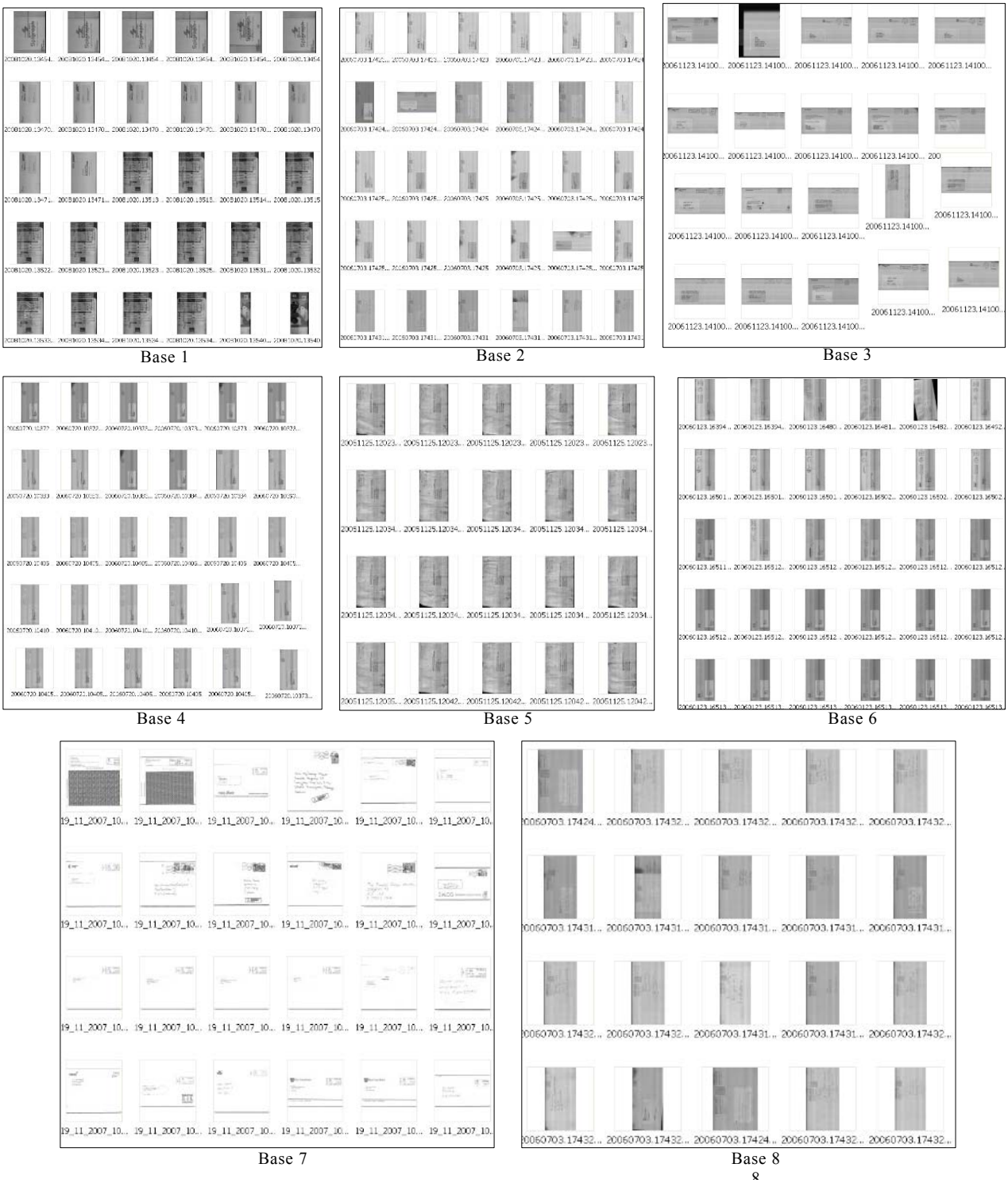


Figure 5. 59 : Échantillons de 8 bases de test de LBA.

5.6.3.3 Troisième test

Les courbes de la figure suivante représentent les temps moyens (en milliseconde) des trois phases de LBA calculées sur chacune des 8 bases.

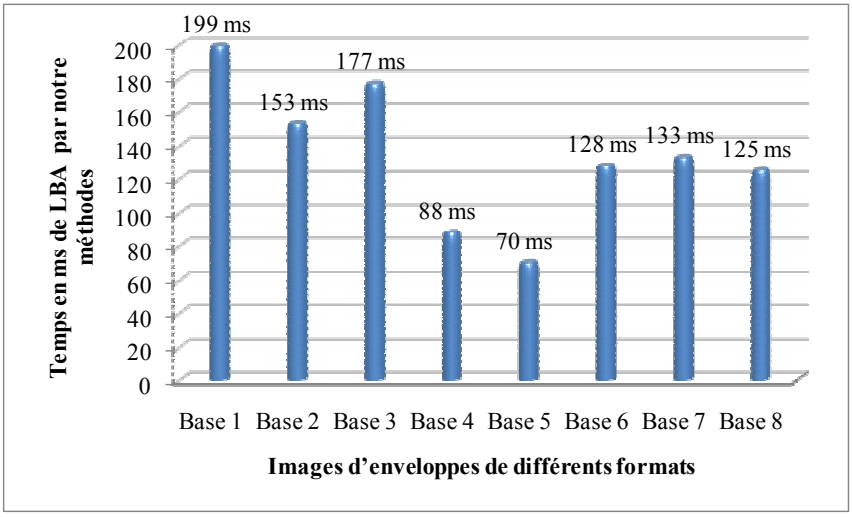


Figure 5. 60 : Temps moyens de LBA sur les 8 bases d'images d'enveloppes.

Les résultats obtenus confirment la grande cohérence entre les différentes phases engagées dans la LBA et la faisabilité d'une solution temps réel. L'implantation de nos algorithmes a été étudiée également.

5.6.4 Conclusion

Nous avons présenté une nouvelle approche de localisation du bloc-adresse basée sur une coloration hiérarchique de graphe et une organisation pyramidale des données. Nous avons expliqué comment la coloration hiérarchique de graphe offre une grande robustesse aux objets parasites considérés comme facteurs d'erreur de l'extraction de la structure des blocs de texte, et comment la b-coloration s'adapte parfaitement au contenu de la base d'apprentissage apportant des améliorations remarquables au niveau de l'interprétation des blocs. Cette interprétation en temps réel offre une localisation automatique robuste du bloc adresse et des informations très riches qui contribuent à l'automatisation de choix de l'OCR (texte manuscrit/ imprimé), à la lecture du montant d'affranchissement, la reconnaissance du format de courrier et même de nom de la société expéditrice. Enfin, les fondements de la b-coloration nous ont permis de concevoir une nouvelle méthode localisation de bloc adresse générique qui peut offrir également à ce domaine d'autres applications comme la reconnaissance de la police de caractères des styles de texte manuscrit et même des OCR.

Conclusion et perspectives

Ces travaux de thèse nous ont donné l'occasion de démontrer la faisabilité de la théorie des graphes pour les tâches essentielles de segmentation et de reconnaissance des documents dans un contexte industriel très contraint. Dans cette partie, nous allons ainsi dresser le bilan de nos contributions. La première section récapitule les objectifs fixés en ciblant précisément les spécificités propres à un système de tri de courriers. Nous commenterons ainsi notre apport dans ce contexte. La seconde partie de la conclusion est dédiée aux perspectives directement envisageables de nos travaux dans des applications plus étendues d'analyse de documents.

1. Bilan du travail effectué

Nous avons vu que l'efficacité du tri automatique dépendait pour une grande part du repérage des zones d'intérêt sur l'image. Celles-ci se résument essentiellement au repérage du bloc adresse qui contient l'information pertinente au tri. Après avoir relevé les insuffisances des méthodes actuelles de localisation de régions d'intérêt, nous avons pu faire le constat que la plupart d'entre elle est encore régie par un fonctionnement très linéaire et séquentiel et n'exploite que très rarement les coopérations interprocessus (prétraitement, analyse, décision) pourtant très enrichissantes. Nous avons montré comment les performances du système sont directement liées à la combinaison et la complémentarité d'approches et de ressources logicielles, du niveau le plus bas (i.e. extraction de la structure physique et extraction de caractéristiques de présentation) aux niveaux les plus élevés (analyse et interprétation du contenu, reconnaissance de type de documents, repérage des zones d'intérêt…). Les conditions d'exécution et la grande vitesse imposées par les systèmes de tri automatique de documents exigent des algorithmes optimaux qui doivent être non seulement efficaces et robustes, mais également très rapides.

La prise de décision est bien souvent issue d'une très grande diversité de technologies présentes d'un bout à l'autre de la chaîne (de la binarisation à la reconnaissance). Celles-ci doivent être optimisées et doivent pouvoir coopérer afin de répondre au plus près aux exigences des applications de vision industrielle. C'est la voie que nous avons choisi de suivre.

En ce sens, nous avons porté plusieurs niveaux de contribution répondant pour certaines à des optimisations algorithmiques aux plus bas niveaux de l'analyse (prétraitement, binarisation, redressement…) et pour d'autres à une véritable reformulation des problèmes d'analyse et de décision par l'introduction de la théorie des graphes inédite au domaine (seg-

mentation, analyse de la structure des documents, localisation du bloc adresse, catégorisation de documents) et qui aura satisfait à toutes les contraintes industrielles. Précisément, nous avons proposé de nouveaux algorithmes de coloration, d'apprentissage et plusieurs scénarios mettant en évidence, pour la première fois, la contribution de la coloration de graphe à la modélisation et à la résolution des problèmes de segmentation, d'apprentissage et de classification.

Dans ces travaux, *nous avons formalisé la tâche d'extraction de la structure physique des images de documents à l'aide d'une coloration minimale de graphe qui procède au regroupement des éléments constitutifs du document* (en fonction de critères d'homogénéité locale et de voisinage). Par ailleurs, la connaissance globale issue de l'analyse des adjacences entre sommets du graphe a permis de prendre les décisions de séparation (partitionnement des sommets) dans les cas où la connaissance locale n'était pas suffisante. C'est donc la structure même du graphe qui a renseigné sur ces connaissances globales et a permis de séparer les données insuffisamment proches. Regroupement et partitionnement ont été considérés comme deux termes clés du processus de coloration hiérarchique. Cette nouvelle stratégie mixte d'extraction de la structure physique a permis d'optimiser les temps de traitement et de réduire les erreurs de segmentation que nous avons présentées dans le chapitre 2 (section 2.4), comme en témoignent les résultats présentés dans le chapitre 5 (sections 5.2, 5.3 et 5.4).

Nous avons également montré comment la coloration de graphe a établi un cadre théorique solide et unique pour l'optimisation et la mise en œuvre des applications clés des systèmes de tri automatique de documents.

Dans un second temps*, nous avons présenté une nouvelle méthode de reconnaissance automatique de type de documents basée sur la coloration hiérarchique des graphes utilisant une représentation issue de la description de la structure physique des documents*. La coloration hiérarchique de graphe introduite dans la phase de segmentation a permis d'augmenter la robustesse aux composantes parasites considérées comme facteurs d'erreur des méthodes classiques de segmentation. La b-coloration introduite dans la phase d'apprentissage a garanti un excellent partitionnement entre catégories de documents. Grâce au nombre restreint de règles dont elle dispose, cette nouvelle technique a pu répondre à une large variété de documents, offrir une vraie représentation des classes par documents dominants et garantir une meilleure disparité interclasses. De plus, nous avons pu augmenter la cohérence entre les différentes phases de la RAD par l'exploitation de la b-coloration et réduire les temps de calcul. *Nous avons également pré-*

senté un nouveau modèle d'apprentissage incrémental basé sur la b-coloration. Grâce à ce modèle, le système de RAD a été capable d'apprendre de nouveaux types de documents à partir de très peu d'exemples. Il a montré sa capacité à s'adapter et à s'améliorer à partir de chaque nouveau document passant dans la chaîne de tri. Durant la phase d'apprentissage, nous avons intégré une approche de l'évaluation de la classification afin d'améliorer la classification elle-même et de conduire à de meilleurs taux de reconnaissance.

Le constat de la grande généricité et de la grande simplicité de notre approche de coloration (et de b-coloration) de graphe à toutes les étapes de reconnaissance et d'apprentissage fait de notre méthode de RAD un outil réellement performant.

Enfin, *nous avons présenté une nouvelle approche de localisation du bloc-adresse* basée sur une coloration hiérarchique de graphe faisant apparaître la structure des blocs et agissant sur une organisation pyramidale des données pour une analyse progressive (des composantes les plus fines aux plus grossières). A ce stade, nous avons montré en quoi la coloration hiérarchique de graphe offrait une grande robustesse aux objets parasites considérés comme facteurs d'erreur dans le processus d'extraction de la structure des blocs de texte, et comment la b-coloration pouvait parfaitement s'adapter au contenu de la base d'apprentissage en apportant des améliorations remarquables au niveau de l'interprétation des blocs. Cette interprétation en temps réel offre une localisation automatique robuste du bloc adresse et contribue ainsi à l'automatisation du choix de l'OCR à mettre en œuvre (en permettant une séparation optimale entre texte manuscrit et texte imprimé). Elle conduit ainsi directement à la lecture du montant d'affranchissement, la reconnaissance du format de courrier et même l'identification du nom de la société expéditrice porté sur le logo de l'enveloppe.

A travers ces deux applications, nous avons montré comment la b-coloration de graphe offrait une capacité exceptionnelle de recherche automatique du nombre de classes (à partir de la recherche des sommets dominants) en assurant tout à la fois, une excellente précision de séparation interclasses et une forte homogénéité intra-classe. Ces propriétés essentielles à tout bon classifieur confirment que la b-coloration est un modèle idéal de représentation des classes enrichie par la notion des sommets dominants.

Rappelons également que nous avons proposé un nouveau concept d'apprentissage incrémental basé sur la b-coloration permettant de mettre à jour la base d'apprentissage à partir d'un flux entrant de documents et de courriers dans la chaîne de tri. Cette propriété importante a facilité

l'adaptation du système de reconnaissance en reconnaissant de nouvelles catégories de documents et grandement simplifié la tâche d'interaction d'un expert avec ce système.

Enfin, l'approche de b-coloration de graphe nous a également permis de concevoir des nouvelles méthodes de reconnaissance et de localisation générique comme la reconnaissance de la police de caractères, des styles de textes manuscrits.

2. Perspectives et extensions envisagées

Nous envisageons à l'issue de ce travail deux types de prolongement :

- d'une part les perspectives induites par de nouvelles améliorations supplémentaires du système de tri de courriers existant, directement exploitables sur site

- et les extensions de ce travail à plus long terme permettant une exploitation des aspects de coloration de graphes dans d'avantage de situations. La grande généricité de l'approche de catégorisation par coloration et b-coloration que nous avons développée nous permet d'envisager de telles extensions pour optimiser certains processus non abordés dans cette thèse (OCR, identité virtuelle des enveloppes). D'autres applications liées à l'analyse et la reconnaissance de documents au sens large et plus généralement à l'analyse d'images de traits et de symboles sont également envisageables dans cette partie. Nos différentes propositions visent deux types d'extensions liées à la coloration et à la b-coloration.

2.1 Applications de la Coloration de graphe

2.1.1 Approche collaborative de la lecture optique

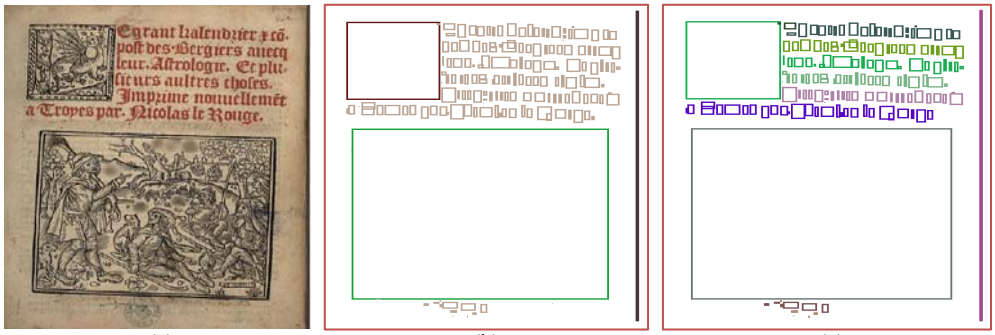
Tout d'abord, rappelons que le système de tri de notre étude fonctionne avec un module de lecture optique contrôlé par une société privée et qu'il n'est pas possible d'agir directement sur lui. En revanche, nous envisageons de proposer en perspectives directe de ce travail, une approche collaborative de la lecture optique faisant coopérer la lecture optique (OCR, ICR…) avec l'analyse de la structure physique, grâce à laquelle nous pouvons extraire un grand nombre de caractéristiques qui permettraient de guider et de contrôler la lecture par OCR. Ceci éviterait notamment un traitement redondant de l'information (en entrée de l'OCR et durant la phase de lecture par l'OCR actuel) et de fait une réduction considérable des temps de calculs.

2.1.2 Extraction d'une identité virtuelle (IDV) de l'image de l'enveloppe

L'IDV est une signature (ou empreinte) propre à l'image de chaque enveloppe, différente pour deux enveloppes distinctes. Elle permettrait l'association de l'adresse reconnue à l'enveloppe physique papier circulant sur la chaîne de tri (afin de l'envoyer vers le bon casier de destination) sans avoir besoin de l'impression des codes barres actuellement en vigueur. Cette signature unique pourrait être construite à partir de l'information riche fournie par la phase de l'extraction et la description de la structure physique par coloration hiérarchique de graphe. L'analyse pyramidale de notre approche devrait permettre de distinguer des enveloppes similaires ne présentant que de légères différences (enveloppes identiques et adresses identiques, mais avec noms de destinataires différents). Cette application de la coloration de graphe doit permettre d'accélérer le tri, de le rendre plus économique et écologique (ne nécessitant aucune impression de codes à barres, aucune encre, et par conséquent pas de machine de lecture de code à barres).

2.1.3 Application à l'analyse de structures des documents anciens du patrimoine

L'analyse des documents anciens requiert des approches très souples d'analyse des contenus hétérogènes contenant tout à la fois des motifs, des composants textuels ou graphiques en tous genres et souvent très irrégulièrement positionnés. Sur la figure ci-dessous, nous avons illustré le résultat de la séparation texte/graphique basée sur une première coloration des connexités et l'extraction de la structure des lignes par une seconde coloration prenant compte des relations de voisinage spatial entre connexités. Les premiers résultats de mise en œuvre nous semblent déjà très prometteurs. Des améliorations portant sur d'éventuels prétraitements des images s'avèreront sans doute nécessaires avant l'application de notre approche.

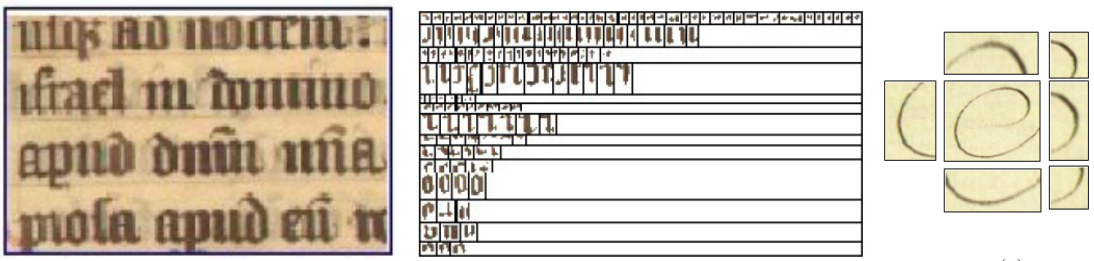


(a) (b) (c)

Figure 6.1 : Exemple d'application de notre méthode d'extraction de la structure physique sur des documents anciens, **(a)** image de documents anciens, **(b)** séparation texte / non texte (première coloration), **(c)** extraction de la structure des lignes par une deuxième coloration.

2.1.4 Création d'un code book génératif de l'écriture

La création d'un code book (liste de graphèmes similaires catégorisés) est une étape importante de l'analyse des écritures qui peut être directement exploitable pour l'identification de scripteurs, la recherche d'occurrences de termes ou de motifs similaires ou la compression des traits d'écritures par couches (fond/forme) sans perte. La coloration de graphe peut être utilisée d'une façon très efficace pour la création d'une liste de graphèmes similaires où chaque couleur correspond à une forme particulière. Le principe de catégorisation non supervisée porté par la coloration de graphes nous semble très approprié dans ce contexte. Les groupes de graphèmes triés par ordre décroissant d'effectifs sont présentés à la figure 6.2 (b).



(a) (b) (c)

Figure 6.2 : (a) Images d'un manuscrit ancien médiéval, [Source IRHT-Paris], **(b)** regroupement des graphèmes dans des ensembles homogènes et création d'un code Book, [Ali 2007] **(c)** exemple de décomposition de l'écriture en graphèmes.

2.1.5 Généralisation à la recherche d'occurrence de mots en mode image

Sans passer par une étape de décomposition en graphèmes, nous projetons d'appliquer une coloration de graphes à des ensembles de mots dont la segmentation nécessite une adaptation spécifique de notre approche de coloration de graphe (chapitre 5 – algorithme 5.3). Il devient alors possible de grouper automatiquement par colorations ces mots en différentes ensembles homogènes, sans connaissance préalable de leur nombre. Le vecteur de caractéristiques de chaque mot correspond donc à un unique sommet du graphe à colorer. Ce regroupement est appliqué une fois pour tout sur les mots d'une page ou d'un livre en entier. En conservant l'approche de reconnaissance développée pour la RAD, la recherche d'occurrences mots à partir d'un mot requête introduit par l'utilisateur en mode image pourra s'effectuer en temps réel. Au lieu de faire des comparaisons avec tous les mots existants (méthodes classiques), il suffit de chercher le groupe qui possède la plus grande similarité par rapport au mot requête. Ce mode

d'interrogation est très utile à la fois aux codicologues qui s'intéressent à la mise en forme des documents manuscrits très anciens pour l'authentification et la datation, qu'aux paléographes qui cherchent toutes les occurrences d'un même mot avec exactement la même graphie, et aux historiens qui cherchent à composer des lexiques de mots fréquemment utilisés dans certains textes ou encore à retrouver par appariement, les mots d'une même famille (même racine lexicale), les mots illisibles rencontrés et difficiles à identifier. Dans ce dernier cas, un outil de recherche de mots peut être envisagé à plus long terme comme une aide à la transcription en cas de difficulté de lecture provoquant la recherche du mot difficile à lire dans un ensemble des pages d'un même corpus. Dans ce cas, l'appariement avec des occurrences possibles de ce même mot peut lever l'ambiguïté de lecture par une compréhension liée au contexte.

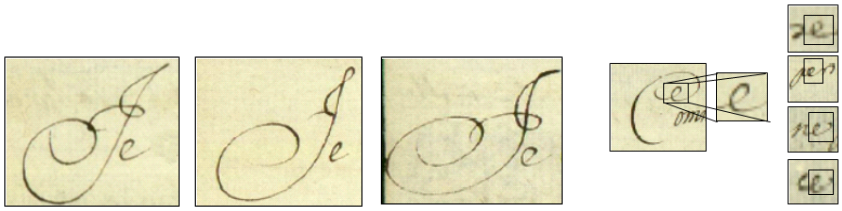


Figure 6.3 : Exemple de similarité des fragments de mots.

2.2. Application de la b-coloration de graphe

Comme nous l'avons démontré dans nos travaux de thèse, la b-coloration de graphe est très utile à la mise en place de mécanisme d'apprentissage et de scénarios de reconnaissance. Nous proposons ici quelques pistes possibles pour l'application de la b-coloration à la reconnaissance des fontes et l'identification des scripteurs.

2.2.1 La reconnaissance de la fonte

La b-coloration peut être mise au service de la reconnaissance automatique de fontes des textes imprimés et permettre l'ajustement et la sélection automatique des OCR pour le tri automatique (ou plus généralement la reconnaissance de caractères). Cette reconnaissance de fonte pourra utiliser les sommets dominants issus de la classification par b-coloration des caractères de fontes différentes de la base d'apprentissage, et permettre ainsi de faire des distinctions entre alphabets (de langues arabes et de langues latines par exemple) ou entre polices de caractères.

2.2.2 Identification de scripteurs

La b-coloration peut aussi répondre au besoin de l'identification automatique de scripteurs. Le principe retenu pourra être le suivant : « deux documents de même scripteur ne doivent pas être classés dans deux ensembles (couleurs) différents ».

Les caractéristiques choisies devront également être discriminantes et pertinentes dans le sens où elles ne devront pas être partagées par tous les scripteurs. La b-coloration peut offrir ici une très bonne modélisation et représentation de classes par sommets dominants. Le concept peut être supervisé si on dispose des connaissances a priori sur l'affectation des documents de la base de scripteurs ou non supervisée lorsqu'on ne peut pas disposer de cette information à l'avance. Pour l'étape d'identification de scripteur, l'utilisateur dispose d'un seul document extérieur à la base d'apprentissage. Ce document constitue la requête. On cherche alors à identifier son auteur parmi un ensemble de N scripteurs connus. Bien que l'identification du scripteur s'inscrive dans la même problématique que la reconnaissance de l'écriture, elle ne semble pas, poser le même type de difficultés. En effet, la tâche d'identification peut tirer profit de la variabilité des écritures afin de les discriminer, tandis que la tâche de reconnaissance doit au contraire parvenir à s'affranchir de la variabilité entre les scripteurs pour identifier le message textuel quel qu'en soit le scripteur. Ces spécificités propres à chacune de ces tâches doivent être prisent en compte très tôt dans la conception des modèles. La b-coloration de graphes est un outil qui peut parfaitement convenir aux deux.

2.2.3 Extension aux images naturelles et médicales

Enfin, la b-coloration peut être introduite et utilisée dans n'importe quel domaine de vision par ordinateur lorsqu'il s'agit de résoudre et d'optimiser un problème d'apprentissage ou de classification comme la détection des visage, la reconnaissance des émotions, le suivi de la main, la localisation des objets (voitures, textes), la séparation des organes anatomiques, la localisation et caractérisation des lésions sur des images médicales (IRM, rayon X), le repérage des structures sur des images satellitaires, etc.

Elle peut également être utilisée pour construire automatiquement une base d'apprentissage représentative à partir d'une base brute en isolant toutes les couleurs de faibles effectifs (qui correspondent à des classes d'exemples bruités ou déformés) ou pour comprimer et représenter des bases d'apprentissage très volumineuses par un nombre restreint de sommets dominants (correspondant aux exemples représentatifs).

Comme nous venons de le voir, les applications de la coloration et de la b-coloration sont très nombreuses en raison de la grande généricité des approches engagées tant aux plus bas niveaux (physiques) qu'aux niveaux plus élevés (reconnaissance et décision). J'envisage de poursuivre la valorisation de cet outil à des domaines plus ouverts en analyse d'images, vision par ordinateur et bien sûr en analyse des documents.

Bibliographie

[AGN00] Agne S., Rogger M., Benchmarking of Document Page Segmentation, Part of the IS&T/SPIE Conference on Document Recognition and Retrieval VII, San Jose, California, January 2000, pp. 165-171.

[AKI86] Akiyama T., Masuda I., A Method of Document-Image Segmentation Based on Projection Profiles, Stroke Densities and Circuniscribed Rectangles, Trans. IEICE (in Japanese), 1986, vol.J69-D N°08, pp.1187-1195.

[AKI93] O. Akindele, A. Belaid, Page Segmentation by Segment Tracing. In: proceedings of Second Int'l Conf. Document Analysis and Recognition, 1993. pp. 341-344.

[ALC00] S. ALCEUDE, JR. BRITTO, Improvement in Handwritten Numeral String. Recognition by Slant Normalization and Contextual. Information, Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, Amsterdam, ISBN 90-76942-01-3, Nijmegen: International Unipen Foundation, 2000, pp. 323-332.

[ANI92] Anigbogu J.C., Reconnaissance de caractères imprimés multifontes à l'aide de modèles stochastiques et métriques, Thèse de doctorat, Université de Nancy I, 1992, 156 p.

[ANT88] T. Antognini, L. Turnbaugh, An expert System for Location of Destination Address in Mail of Arbitrary Complexity , USPS Advanc, Techno , Conf. #3, 1988, pp . 279-292.

[ANT94] Antonacopoulos A., Ritchings R., Flexible Page Segmentation Using the Background. In: Proceedings of 12 th Int'l Conf. Pattern Recognition, 1994, pp. 339–344.

[APP01] Appiani E., Cesarini F., Colla A.M., Diligenti M., Gori M., Marinai S., Soda G., Automatic document classification and indexing in high-volume applications, International Journal on Document Analysis and Recognition, 2001, vol.4, n° 2, pp. 69-83.

[APP89] Appiani E., Conterno B., Luperini V., Roncarolo L., EMMA2: a high-performance hierarchical multiprocessor Micro, IEEE, Feb. 1989, vol.9, Issue. 1, pp.42 - 56.

[BAB90] N. Babaguchi and al, Connectionist model binarisation, Pattern Recognition, ICPR 1990. Proceedings, 10th International Conference, 16-21 June 1990, vol.2, pp. 51-56.

[BAD03] Badekas E., Papamarkos N., A system for document binarisation Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the 3rd International Symposium, 18-20 Sept. 2003, vol.2, pp.909-914.

[BAG03] BAGDANOV A.D. and Worring M.: First order Gaussian graphs for efficient structure classification, Pattern Recognit. vol. 36, Issue 6, 2003, pp.1311–1324.

[BAG97] Bagdanov A., Kanai J., Projection Profile Based Skew Estimation Algorithm for JBIG Compressed Images, 1997, ICDAR, pp. 401- 405.

[BAI90] Baird H S., Jones S .E., Fortune S.J., Image segmentation by shape-directed covers, International Conf. On Document Analysis and Recognition, 1990, pp.820-825.

[BAI92] H .S . Baird, H . Bunke, K. Yamamoto, «Structured document analysis» ,Springer, 1992.

[BAL03] BALDI S. and al, Using tree-grammars for training set expansion in page classification, the 7th International Conference on Document Analysis and Recognition, Scotland, 2003, pp. 829-833.

[BEL92] BELAID A., BELAID Y., Reconnaissance des formes : méthodes et application, IA-InterEditions, Paris, 1992.

[BEL03] BELLILI A., GILLOUX M., GALLINARI P., An MLP-SVM combination architecture for offline handwritten digit recognition. Reduction of recognition errors by Support Vector Machine rejection mechanisms, International Journal on Document Analysis and Recognition, 2003, vol. 5, pp. 244-252.

[BEL93] Belaïd A., Akindele O., A labeling approach for mixed document blocks, document Analysis and Recognition, ICDAR'93, Proceedings of the second International Conference, 1993, pp. 749-752.

[BEL94] Belaïd A., Analyse et reconnaissance de documents, Cours INRIA: le Traitement électronique de Documents, Collection ADBS, 3-7 octobre, Aix-en-Provence, 1994, 34 p. Numéro de rapport : CRIN - 94-R-068.

[BEN99] Ben Amara N., Utilisation des modèles de Markov cachés planaires en reconnaissance de l'écriture arabe imprimée. Thèse de doctorat, Université Tunis II, 1999, 55.p.

[BER02] Bernd Jähne, Digital Image Processing, 5th revised and extended edition, ed. Berlin: Springer, 2002, Meas. Sci. Technol, 585 p. ISBN 3 540 67754 2.

[BER98] Berthaud C., Bourennane E., Paindavoine M., Milan C., Implementation of a real time image rotation using B-spline interpolation on FPGA's board Image Processing, ICIP 98, Proceedings, International Conference, 1998, vol.3, pp. 995-999.

[BHA96] Bhandarkar S.M., Yu H., VLSI Implementation of Real-time Image Rotation, Proceedings of International Conference on Image Processing, 1996, vol.2, pp.1015-1018.

[BIS95] Bishopp C., Neural Networks for Pattern Recognition. Oxford Press Libri, nov 1995, 504p. ISBN. 0198538642.

[BOJ01] Bojarshinov V., Edge and total coloring of interval graphs. Discrete Applied Mathematics, Publisher: Elsevier, 30 October 2001, vol.114, n°1, pp. 23-28.

[BOT00] BOTTOU L., HAFFNER P., LE CON Y., HOWARD P., VINCENT PP., DjVu : Un Système de Compression d'Images pour la Distribution Réticulaire de Documents Numérisés , Actes de la Conférence Internationale francophone sur l'Ecrit et le Document, Lyon, France, 2000.

[BRE79] Brelaz D. New methods to color the vertices of a graph. Communications of the ACM, 1979, vol.22, n°4, pp.251-256.

[BRE84] Breiman L., Friedman J., Olshen R., Stone C., Classification and Regression Trees. Wadsworth International Group, 1984.

[BRI81] Brigham R.D., Dutton R.D., A new graph coloring algorithm, The. Computer Journal 24

1981, pp. 85-86.B

[BRO41] R. Brooks. On coloring the nodes of a network. Proc. Camb. Philos. Soc., vol.37, pp.194-197, 1941.

[BRU97] Brugger R., Zramdini A., Ingold R., Modeling Documents for Structure Recognition Using Generalized N-Grams, 4'th International Conference on Document Analysis and Recognition, ICDAR'97, 1997, vol.1, pp 56-60.

[CAR04] CARMAGNAC F. and al, Une stratégie originale de classiffication basée sur le calcul de distances avec sélection de caractéristiques, application à la classiffcation d'images de document, Actes du 14ème congrès francophone AFRIF-AFIA, 2004.

[CAR02] Caramia M., Dell'Olmo PP., Vertex coloring by multistage branch-and-bound. Proceedings of the Computational Symposium on Graph Coloring and its Generalizations, 2002, pp. 40-47.

[CES01] CESARINI F. and al, Encoding of modified X–Y trees for document classification, the 6th International Conference on Document Analysis and Recognition, Seattle, USA, 2001, pp. 1131-1136.

[CES99] Cesarini F., Gori M., Marinai S., Soda G., Structured document segmentation and representation by the modified X-Y tree, Document Analysis and Recognition, ICDAR '99. Proceedings of the Fifth International Conference on 20-22 Sept. 1999, pp. 563-566.

[CHA04] Chang F., Chen C. J., Lu C. J., A lineartime component-labeling algorithm using contour tracing technique," Computer Vision and Image Understanding, 2004, vol. 93, pp.206-220.

[CHA95] Moon-Soo Chang and al, Improved binarization algorithm for document image by histogram and edge detection, Document Analysis and Recognition, ICDAR. 1995, Proceedings of the Third International Conference on 14-16 Aug. 1995, vol.2, pp.636-639.

[CHE07] Nawei Chen and Dorothea Blostein, A survey of document image classification: problem statement, classifier architecture and performance evaluation, Revue International Journal on Document Analysis and Recognition, Éditeur Springer Berlin / Heidelberg, juin 2007, vol.10, n°1, pp. 1-16. ISSN 1433-2833.

[CHE01]. Baoquan Chen, Arie E. Kaufman, Two-Pass Image and Volume Rotation (ST). Volume Graphics 2001.

[CHE04] Chellabi M., Volkmann L., Relations between the lower domination parameters and the chromatic number of a graph. Discrete Mathematics, 6 January 2004, vol.274, Issues1-3, pp.1-8.

[CHE94] Cheng Y., Jensen J. R., Huntsberger T., Huntsberger B., Hypercube Algorithm for Image Component Labeling, Proc. of the Scalable High-Performance Computing Conference 1994. IEEE, CA, USA. pp. 259-262.

[CHE97] Chen J.-J., Chang G. J., Huang K.-C. Integral distance graphs. Jonrnal of Graph Theory, 1997, vol. 25, pp.287-294.

[CHI01] Chi Z., Wong K.W., A two-stage binarization approach for document images , Intelligent

Multimedia, Video and Speech Processing, ISIMP . Proceedings of International Symposium on 2-4 May 2001, pp.275-278.

[CHI97] Chin.W, Harvey.A, Jennings.A, Skew detection in handwritten scripts TENCON '97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications, Proceedings of IEEE, 1997, vol.1, pp. 319- 322.

[CHI92] Y. Chigusa and al, An image binarization system for composite pictures Circuits and Systems, ISCAS '92. Proceedings, IEEE International Symposium, 3-6 May 1992, vol.5, pp.2292-2295.

[CIN03] Cinque L., Levialdi S., Malizia A., A system for the automatic layout segmentation and classification of digital documents Image Analysis and Processing, Proceedings. 12th International Conference on 17-19 Sept. 2003, pp.201-206.

[COC95] Cocquerez J.P, Philipp S., Analyse d'images: Filtrage et Segmentation, Edition Masson, 1995. 450 p.

[COR05] Corteel S., Valencia-Pabon M., Vera J.C, On approximating the b-chromatic number, Discrete Applied Mathematics archive, 2005, vol.146, pp. 106-110.

[COR73] Derek G. Corneil, Bruce Graham, An Algorithm for Determining the Chromatic Number of a Graph. SIAM J. Comput. 1973, vol.2, n°4, pp. 311-318.

[COT98] Côté M., Lecolinet E., Cheriet M., Suen C., Automatic reading of cursive scripts using a reading model and perceptual concepts, International Journal on Document Analysis and Recognition, 1998, vol.1, n°1, pp. 3-17.

[DEF94] Déforges O., Barba D., A fast multiresolution text-line and non text line structures extraction and discrimination scheme for document image analysis, ICPR 94, pp. 134-138.

[DEF95] DEFORGE O., PIQUIN PP., VIARD-GAUDIN C., BARBA D., Segmentation d'images de documents par une approche multirésolution. Extraction précise des lignes de texte, Traitement du Signal ,GRETSI, Saint Martin d'Hères, France, 1995, vol.12, n° 6-NS, pp. 527-539.

[DEN96] Dengel A., Dubiel F., Computer understanding of document structure. International Journal of Imaging Systems and Technology, 1996, n°7, pp.271-278.

[DIA02] Diaz, I. M., Zabala, PP., A branch-and-cut algorithm for graph coloring. In Computational Symposium on Graph Coloring and its Generalizations, COLOR02, 2002, Ithaca, N-Y.

[DIL03] DILIGENTI M. and al, Hidden Tree Markov Models for document image classification, IEEE Trans. Pattern Anal. Mach. Intell. 2003, vol.25, n°4, pp. 519-523.

[DIN00] YIMEI DING, FUMITAKA KIMURA, YASUJI MIYAKE, Slant estimation for handwritten words by directionally refined chain code, Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, Amsterdam, , Nijmegen: International TJnipen Foundation, 2000, pp. 53-61. ISBN 90-76942-01-3

[DIN04] Yimei Ding, W.Ohyama, F. Kimura, M. Shridhar, Local slant estimation for handwritten English words, Frontiers in Handwriting Recognition, WFHR'04. Ninth International Workshop, 26-29 Oct 2004, pp.328-333. DOI.10.1109/IWFHR.2004.64.

[DOE98] Doermann D., Rivlin E., Rosenfeld A., The Function of Documents, IJCV, 1998, vol.16, pp. 799-814.

[DON05] Jian-xiong Dong, Dominique P., Krzyyzak A., Suen C.Y., Cursive word skew/slant corrections based on Radon transform Document Analysis and Recognition, Proceedings. Eighth International Conference on 29 Aug.-1 Sept. 2005, pp. 478-483.

[DOR98] Dorne R., Hao J.K.. Tabu search for graph coloring, T-coloring and set T-colorings. Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization, 1998, pp. 77-92.

[DOW90] Downton A.C., Leedham C. G., Preprocessing and presorting of envelope images for automatic sorting using OCR, Pattern Recognition, 1990, vol.23, pp. 347-362.

[DRI95] Drivas D., Amin A., Page segmentation and classification utilising a bottom-up approach, Document Analysis and Recognition, ICDAR, 1995, vol.2, pp. 610-614.

[DUY02] Pınar Duygulu, Volkan Atalay, A hierarchical representation of form documents for identification and retrieval International Journal on Document Analysis and Recognition,IJDAR 2002, Springer Berlin / Heidelberg, vol. 5, N° 1, pp.17-27.

[EFF03] Brice Effantin and Hamamache Kheddouci, The b-chromatic number of power graphs, DMTCS 2003, vol.6, pp. 45-54.

[EFF06] Brice Effantin and Hamamache Kheddouci, a distributed algorithm for a b-coloring of a graph, IEEE, ISPA'2006, Serrento, Italy.

[EGL03] V. EGLIN and S. BRES, Document page similarity based on layout visual saliency: application to query by example and document classification, the 7th International Conference on Document Analysis and Recognition, Scotland, 2003, pp. 1208-1212.

[EGL04] V. Eglin, S. Bres, Analysis and interpretation of visual saliency for document functional labeling. Int. J. Doc. Anal. Recognit, 2004, vol7, n°1, pp.28-43.

[EGL99] V.Eglin, S. Bres et H. Emptoz, Structuration de documents par repérage de zones d'intérêt , Traitement du Signal, 1999, vol. 16, n°3, pp. 219-239.

[EIT04] L.F.Eiterer, J.Facon, D.Menoti, Postal envelope address block location by fractal-based approach, Computer Graphics and Image Processing, 17th Brazilian Symposium, IEEE, 2004, pp. 90-97.

[ELG06] Haytham Elghazel, Mohand-Said Hacid, Hamamache Khedddouci1 and Alain Dussauchoy, A New Clustering Approach for Symbolic Data: Algorithms and Application to Healthcare Data, BDA 2006, Lille, France.

[ELM96] ELMS A.J., The Representation and Recognition of Text Using Hidden Markov Models, Thèse de doctorat, Université de Surrey, Angleterre, 1996.

[EMA99] Mahmoodian E., Mendelsohn E., On defining numbers of vertex colouring of regular graphs. Discrete Mathematics, 1999, n197/198, pp.543-554.

[ESP00] ESPOSITO F. and al, Machine learning for intelligent processing of printed documents. J. Intell. Inf. Syst, 2000, pp. 175-198.

[ESP95] F. Esposito, D. Malerba and G. Semeraro, A Know-ledge-Based Approach to the Layout Analysis, Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, Canada, 1995, pp.466-471.

[ESQ 02] I. Esquef, M. Albuquerque, M. Abuquerque, Nonextensive entropic image thresholding , Computer Graphics and Image Processing, SIBGRA. 2002. Proceedings. XV Brazilian Symposium on 7-10 Oct. 2002, 402 pp.

[ETE97] Etemad K., Doerinam U., Clielinpa R., Multiscale Segmentation of Unstructured Document Pp. Using Soft Decision Integration, IEEE Tmns. Pattern Analysis and Machine Intelligence, Jan 1997, vol.19, n°.1, pp. 92-96

[FAN98] Fan K., Wang L., Tu Y., Classification of Machine- Printed and Hand-Written Texts Using Character Block Layout Variance, Pattern Recognition, 1998, vol.31, n°9, pp.1275-1284.

[FAR06] Farooq F., Sridharan K., Govindaraju V., Identifying Handwritten Text in Mixed Documents Pattern Recognition, ICPR 2006. 18th International Conference, 2006, vol.2, pp.1142 -1145.

[FAU94] Fausett L., Fundamentals of Neural Networks, Englewood Cliffs, NJ: Prentice Hall, 1994.

[FEN04] Meng-Ling Feng, Yap-Peng Tan, Adaptive binarization method for document image analysis Multimedia and Expo, ICME '04. IEEE International Conference, 27-30 June 2004, vol.1, pp.339-342.

[FIS58] Fisher W.D., on grouping for maximum homogeneity, JASA, vol.53, 789-798, 1958.

[FLE88] Fletcher L. A., Kasturi R., A Robust Algorithm for Text String Separation from : Mixed Text/Graphics Image. PAM, 1988, vol.6, n°10, pp. 910-918.

[FLE97] Martin Fleury and Adrian F. Clark, Sampling Concerns in Scanline Algorithms IEEE TRANSACTIONS ON MEDICAL IMAGING, 1997, vol.16, n°3, pp.349-361.

[FON93] Fontanot PP. and Ramponi G., A polynomial filter for the preprocessing of mail address images, in Proc. IEEE Winter Workshop on Nonlinear Digital Signal Processing, Tampere, Finland, Jan. 1993, pp. 2.1-2.6.

[FRA93] Franke J., Oberlander M., Writing Style Detection by Statistical Combination of Classifiers in Form Reader Applications, Proceedings of 2nd ICDAR, 1993, pp. 581-584.

[FRA94] Donald Fraser, and Robert A. Schowengerdt, Avoid-ance of Additional Aliasingin Multipass Image Rotations, 1994, IEEE TRANSACTIONS ON IMAGE PROCESSING, vol.3, n°6, pp.721-735.

[GAC08] Gaceb DJ., Eglin V., Lebourgeois F., Emptoz H.. Implication de la b-coloration de graphes pour la reconnaissance automatique du type de document. Dans Colloque International Francophone sur l'Ecrit et le Document(CIFED), France, 2008, pp.31-36.

[GAL99] Galinier P., Hao J., Hybrid evolutionary algorithms for graph coloring. Journal of Combinatorial Optimization, 1999, vol.3, n°4, pp.379-397.

[GAT96] Gatos B., Perantonis S. J., Papamarkos.N: Accelerated Hough transform using rectangular image decomposition, Electronics Letters, 1996, vol.32, Issue 8, pp.730-732.

[GCH98] Chen G., Gyarfas A., Schelpp R.. Vertex colorings with a distance restriction. Discrete Mathematics, 1998, n.191, pp.65–82.

[GEN99] Gene Myers, A fast bit-vector algorithm for approximate string matching based on dynamic programming, Journal of the ACM (JACM), May 1999, vol.46, Issue 3, pp. 395 - 415 , ISSN:0004-5411.

[GHO95] Ghosh, I., and Majumdar, B., VLSI Implementation of An Efficient ASIC Architecture for Real-Time Rotation of Digital Images, International Journal of Pattern Recognition and Artificial Intelligence, 1995, pp. 449-462.

[GIB00]. Jerry D. Gibson, Hand Book of Image and Video Processing, 2000.

[GIL95] GILLOUX M., LEROUX M., BERTILLE J.-M., Strategies for handwritten Word Recognition Using Hidden Markov Modcls, Machine Vision and Applications, 1995, vol.8, Issue 4, pp. 197-205.

[GLA01] Glassner A., Fill 'Er Up!, IEEE Computer Graphics and Applications, Jan.-Feb. 2001, vol. 21, n°21.1 pp.78-85.

[GLO96] Glover, M. Parker, J. Ryan. Coloring by tabu branch and bound. vol. 26 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, Providence, RI, USA, 1996, pp. 285-307.

[GOR93] O'Gorman L., The document spectrum for page layout analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1993, vol.15, n°11, pp. 1162-1173.

[GOR98] N. Gorski, and al., A new A2iA bankcheck recognition system, Handwriting Analysis and Recognition, IEEE Third European Workshop on 14-15 July 1998, pp.1-6.

[GOR99] GORSKl N. et al., A2iA Check Reader: A Family of Bank Check Recognition Systems , Fijih International Conference on Document Analysis and Recognition, 1999, pp. 523.

[GOV03] Govindaraju V., Tulyakov S., Postal address block location by contour clustering, Document Analysis and Recognition, Proceedings. Seventh International Conference on 3-6 Aug. 2003, vol.1, pp.429 - 432.

[GRO76] Grossberg S., Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions. Biological Cybernetics, 1976, n°23, pp.187-202.

[GUO01] J.K. Guo, M.Y. Ma, Separating handwritten material from machine printed text using hidden Markov models Document Analysis and Recognition, Proceedings. Sixth International Conference on 10-13 Sept. 2001, pp.439-443.

[HAJ95] Ha J., Haralick R., Phillips I., Document Page Decomposition by the Bounding- Box Projection Technique. In: Proceedings of Third Int'l Conf. Document Analysis and Recognition, 1995, pp. 1119–1122.

[HAM05] Hamza H., Smigiel E., Belaid E., Neural based binarization techniques, Document Analysis and Recognition, ICDAR, proceedings. Eighth International Conference on 29

Aug.-1 Sept. 2005, vol.1, pp.317-321.

[HAN07] Eunjung Han, Kirak Kim, HwangKyu Yang, and Keechul Jung Frame Segmentation Used MLP-Based X-Y Recursive for Mobile Cartoon Content, Book:Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, vol.4552, pp. 872-881.

[HAN05] Han Z., Liu C.PP., Yin X.C., A two-stage handwritten character segmentation approach in mail address recognition, Document Analysis and Recognition, Proceedings. Eighth International Conference on 29 Aug.-1 Sept. 2005, vol.1, pp. 111-115.

[HAR70] Harary F., Hedetniemi S., The achromatic number of a graph. Jonrnal of Combinatorial Theory, 1970, vol.8, pp. 154-161.

[HAR81] Haralick R. M., Some neighborhood operations, Plenum Press, Addison-Wesley, Reading, MA, 1981, pp.11–35.

[HAS85] Hase M., Hoshino Y., Segmentation Method of Document Images by Two-Dimensional Fourier Transformation, System and Computers in Japan, 1985, vol.16, n°3, pp. 38-47.

[HEJ05] He J., Do Q.D.M., Downton A.C., Kim J.H., A comparison of binarization methods for historical archive documents, Document Analysis and Recognition, ICDAR. 2005. Proceedings. Eighth International Conference on 29 Aug.-1 Sept. 2005, vol.1, pp.538-542.

[HER98] HÉROUX PP. and al, Classification method study for automatic form class identification, the 14th International Conference on Pattern Recognition, Brisbane, Australia, 1998, pp. 926-929.

[HEU03] Heuberger C., On planarity and colorability of circu-lant graphs. Discrete Mathematics, 2003, vol.268, pp.153-169.

[HIL01] Hilton A., Slivnik T., Stirling D., Aspects of edge list-colourings. Discrete Mathematics, 2001, vol.231, pp.253-264.

[HOC93] Hoch R., Kieninger T., On virtual partitioning of large dictionaries for contextual post-processing to improve character recognition, Document Analysis and Recognition Proceedings of the Second International Conference on 20-22 Oct. 1993, pp. 226 -231.

[HUI00] Huiping Li, Doermann D., Kia O., Automatic text detection and tracking in digital video Image Processing, IEEE Transactions, Jan 2000, vol.9, Issue1, pp.147-156.

[IMA93] Imade S., Tatsuta S., Wada T., Segmentation and Classification for Mixed Text/Image Document Using Neural Network, In Proc. 2nd ICDAR, 1993, pp. 930-934.

[ING99] INGLIS S. J., Lossless Document Image Compression, thèse de doctorat, Université de Waikato, New Zealand, Mars 1999.

[IRV99] Irving R., Manlove D., The b-chromatic number of a graph. Discrete Applied Mathernatics, 1999, vol.91, pp.127-141.

[ITT93] Ittner D. J. and H. S. Baird, 'Language Free layout analysis', in Proceedings of the Third International conference on Document Analysis and Recognition, 1993, pp. 336–340.

[IWA95] Makoto Iwayama, Takenobu Tokunaga, Cluster-based text categorization: a comparison

of category search strategies. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval table of contents Seattle, Washington, United States, 1995. pp. 273-280. ISBN:0-89791-714-6.

[JAI92a] Jain A., Bhattacharjee S., Address block location on envelopes using Gabor filters, Pattern Recognition, 1992, vol. 25, no.12, pp. 1459-1477.

[JAI92b] Jain A. K., Bhattacharjee S., Text segmentation using Gabor filters for automatic document processing, Mach. Vis. Appl. , 1992, vol.5, pp. 169-184.

[JAI96] Anil K. Jain, Bin Yu, Fast address Block Location and Orientation Detection MSU-CPS, , 1996, pp. 97-12.

[JAI98] Jain A., Yu B., Document representation and its application to page decomposition," IEEE TPAMI March 1998, vol. 20, pp. 294–307.

[JEO04] Seon Hwa Jeong, Seung Ick Jang, Yun-Seok Nam, Locating destination address block in Korean mail images, ICPR 2004, IEEE, vol.2, pp. 387-390.

[JEO04] Seon Hwa Jeong, Seung Ick Jang, Yun-Seok Nam, Locating destination address block in Korean mail images, Pattern Recognition, ICPR 2004, IEEE, 17th International Conference , vol.2, pp.387-390.

[JIA05] Xiao-Gang Jiang; Jian-Yang Zhou; Jiang-Hong Shi; Hui-Huang Chen; FPGA implementation of image rotation using modified compensated CORDIC, ASIC, 2005. ASICON 2005. 6th International Conference, 24-27 Oct. 2005, vol.2, pp.752-756.

[JON05] Jonathan Milgram, Mohamed Cheriet, and Robert Sabourin. Estimating accurate multiclass probabilities with support vector machines. In Int. Joint Conf. on Neural Networks, 2005, pp.1906-1911.

[JOU05] Journet N., Eglin V., Ramel J.Y., Mullot R., Text/graphic labelling of ancient printed documents. Document Analysis and Recognition, ICDAR.2005. Proceedings. Eighth International Conference on 29 Aug.-1 Sept. 2005, vol.2, pp.1010 -1014.

[KAM99] Kamada H., Fujimoto K., High-speed, high-accuracy binarization method for recognizing text in images of low spatial resolutions, Document Analysis and Recognition, ICDAR '99. Proceedings of the Fifth International Conference on 20-22 Sept. 1999, pp.139-142.

[KAN95] Kanai J., Rice S. V., Nartker T. A., Nagy G., Automatic evaluation of OCR zoning, IEEE Trans. Pattern Anal. Machine Inteli, 1995, vol. 17, n°.1, pp. 86-90.

[KAP85] Kapur J. N., Sahoo PP. K., Wong A. K. C., A new method for gray-level picture thresholding using the entropy of the histogram, Graph. Models Image Process, 1985, 29, pp. 273-285.

[KAV03] KAVALLIERATOU E. and al. EGRATED SYSTEM FOR HANDWRITTEN DOCUMENT IMAGE PROCESSING, 2003, International Journal of Pattern Recognition and Artificial Intelligence, vol. 17, n°4, pp. 617-636.

[KAV04] Kavallieratou E., Stamatatos S., Discrimination of machine-printed from handwritten text using simple structural characteristics Pattern Recognition, 2004. ICPR 2004. Pro-

ceedings of the 17th International Conference, 23-26 Aug. 2004, vol.1, pp.437-440.

[KAV99] Kavallieratou E., Fakotakis N., Kokkinakis G., New algorithms for skewing correction and slant removal on word-level, in Proc. IEEE 6th International Conference on Electronics, Circuits and Systems, Pafos, Cyprus, Sept. 1999, pp. 1159-1162.

[KEM98]A. Kemnitz and H. Kolberg. Coloring of integer dis-tance graphs. Discrete Mathematics, 1998, vol.191, pp.113-123.

[KIS 96] Kise K., Yanagida O., Takamatsu S.: Page Segmentation Based on Thinning of Background. In: Proceedings of 13th Int'l Conf. Pattern Recognition, 1996, pp.788–792.

[KLA02] Klaus Schulz and Stoyan Mihov, Fast String Correction with Levenshtein-Automata,International Journal of Document Analysis and Recognition, 2002, vol.5, pp. 67-85.

[KOC99] Kochi T., Saitoh T., User-defined template for identifying document type and extracting information from documents. In: Proceedings of the 5th International Conference on Document Analysis and Recognition, Bangalore, India, 20–22 September 1999, pp. 127-130.

[KOH81] Köhler R.. A segmentation system based on thresholding. Graphical Models and Image Processing, 1981, vol.15, pp.319-338.

[KOH88] Kohonen T., Learning Vector Quantization, Neural Networks, 1(suppl 1), 303, 1988.

[KOH95] Kohonen T., Self-Organizing Maps, Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001. Third Extended Edition, 501 p.

[KOU02] Kouider M. and Maheo M. Some bounds for the b-chroma tic number of a graph, Discrete Mathematics, 2002, 256, pp. 267-277.

[KRI93] Krishnamoorthy, M., Nagy, G., Seth, S., Viswanathan, M.: Syntactic Segmentation and Labeling of Digitized. From Technical Journals. IEEE Trans. Pattern Analysis and Machine Intelligence , 1993, vol.15, pp.743-747.

[KRU01] Kruatrachue B., Suthaphan PP., A fast and efficient method for document segmentation for OCR. Electrical and Electronic Technology, 2001. TENCON. Proceedings of IEEE Region 10 International Conference, 19-22 Aug. 2001, vol.1, pp.381-383.

[KUH95] Kuhnke K., Simoncini L., Kovacs-V Zs.M., A system for machine-written and hand-written character distinction, Document Analysis and Recognition, 1995., Proceedings of the Third International Conference, 14-16 Aug. 1995, vol.2, pp.811-814.

[KUM86] Kumar V. K. P, Eshaghian M. M., Parallel geometric algorithms for digitized pictures on mesh of trees, Proc. 1986 Intl. Con5 on Parallel PGcessing (ICPP), pp.270-273.

[LAM94] Lam S., An adaptive approach to document classification and understanding. In: Proceedings of International Association for Pattern Recognition Workshop on Document Analysis Systems, Kaiserslautern, Germany, October 1994, pp. 231–251.

[LEB92] Lebourgeois F., Bublinski Z., Emptoz H., A fast and efficient method for extracting text

paragraphs and graphics from unconstrained documents, ICPR92, vol.2, pp.272-276.

[LEB97] LeBourgeois F., Robust multifont OCR system from gray level images, fourth ICDAR, International Conference on Document Analysis and Recognition, Ulm, 1997, pp. 1-5.

[LEB99] LeBourgeois F., Emptoz H., Document Analysis in Gray Level and Typography Extraction Using Character Pattern Redundancies, Int. Conf. On Doc. Analysis and Recognition ICDAR'99, 1999, India, pp.177-180.

[LEC01] LeCun Y., Bottou L., Bengio Y., Haffner PP., Gradient-based learning applied to document recognition. In Intelligent Signal Processing, pp. 306-351. IEEE Press, 2001.

[LED94] Le, D.X., Thoma, G., and Weschler, H.: 'Automated page orientation and skew angle detection for binary document images', Pattern Recognit., 1994, 27, 10, pp.1325-1344.

[LED94] Le D.X., Thoma G., Weschler H., Automated page orientation and skew angle detection for binary document images, Pattern Recognit, 1994, 27, (10), pp. 1325-1344.

[LEE94] Seong-Whan Lee, Ki-Cheol Kim, Locating destination address block on handwritten Korean envelopes, Pattern Recognition, Computer Vision & Image, 12th IAPR International, IEEE, 1994, Vol.2, pp. 619 - 621.

[LEE94] S-W. Lee, K-C. Kim, Locating destination address block on handwritten Korean envelopes, Pattern Recognition, Computer Vision & Image, 12th IAPR International, IEEE, 1994, Vol. 2, pp. 619-621.

[LEE98] référence 2001 pas 98. Seong-Whan Lee et Dae-Seok Ryu, Parameter-Free Geometric Document Layout Analysis. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, NOVEMBER 2001, vol.23, n°.11, pp. 1240-1256.

[LEV07] Benjamin Lévêque. Coloration de graphes : structures et algorithmes, thèse de Doctorat, Université Joseph Fourier, 2007, 123p.

[LEV85] Levine M. D., Ahmed M. Nazif. Dynamic Measurement of Computer Generated Image Segmentations. IEEE Transaction on Pattern Analysis and Machine intelligence, PAMI-, March 1985, 7(2), pp. 155-164.

,[LEW94] Lewis D.D., Ringuetee M., A Comparison of Two Learning Algorithms for Text Categorization. Proc. of 3rd An. Sympp.on Document Analysis and Information Retrieval, 1994, pp.81-93.

[LEY04] Leydier Y., Le Bourgeois F. et Emptoz H., Sérialisation du k-means pour la segmentation des images en couleur. Dans Actes du 8ième colloque international francophone sur l'écrit et le document, CIFED, 21-25 juin 2004, La Rochelle. pp. 191-196.

[LI 95] Li S.Z., Markov Random Field Modeling in Computer Vision, Springer, Tokyo, 1995.

[LIA02a] Liang, J., Doermann, D., Ma, M., Guo, J.K., Page classification through logical labelling. In: Proceedings of the 16th International Conference on Pattern Recognition, Quebec, Canada, 11–15 August 2002, pp. 477-480.

[LIA02b] Liang J., Doermann D.S, Logical Labeling of Document Images Using Layout Graph Matching with Adaptive Learning Source Lecture Notes In Computer Science; archive Proceedings of the 5th International Workshop on Document Analysis Systems V

(DAS), 2002, vol. 2423, pp. 224-235. ISBN:3-540-44068-2.

[LIF07] He Lifeng, ChaoYuyan; Kenji Suzuki, A Linear-Time Two-Scan Labeling Algorithm, Image Processing, 2007. ICIP 2007. IEEE International Conference, Sept. 16 2007-Oct. 19 2007, vol.5, pp.V-241-V-244.

[LIF08] He Lifeng; Chao Yuyan; K. Suzuki, A Run-Based Two-Scan Labeling Algorithm, Image Processing, IEEE Transactions on, May 2008, vol.17, Issue 5, pp.749-756. DOI 10.1109/TIPP.2008.919369.

[LII93] Lii J., Palumbo PP.W., Srihari S.N., Address block location using character recognition and address syntax Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on 20-2 2 Oct. 1993, pp.330-334.

[LIJ98] Li J., Gray R.M., Text and picture segmentation by distribution analysis of wavelet coefficients. In: Proceedings of the 5th International conference on Image Processing Chicago, Illinois, October 1998, pp. 790-794.

[LIK94] Likforman-Sulem L., Faure C., Une méthode de résolution de conflits d'alignements pour la segmentataion des documents manuscrits, CNED 94, 3ème Colloque National Sur l'Ecrit et le Document, 1994, pp .265-273.

[LIO01]. Liolios. N, Fakotakis. N, Kokkinakis. G, Improved document skew detection based on text line connected-component clustering, Image Processing, Proceedings, International Conference, 2001, vol.1, pp. 1098-1101.

[LU99] Lu Z., Chi Z., Siu W.C., Shi PP. A background-thinning based approach for separating and recognizing connected handwritten digit strings. Pattern Recognition, 1999, vol. 32, pp. 921–933.

[LUC04] Corinne Lucet, Florence Mendes, Aziz Moukrim: Pre-processing and Linear-Decomposition Algorithm to Solve the k-Colorability Problem. WEA 2004, pp. 315-325

[LUM83] Lumia R., Shapiro L., Zuniga 0., A new connected components algorithm for virtual memory computers, Computer Vision, Graphics, and Image Processing, 1983,vol. 22, pp. 287-300.

[MAJ06] Majumdar A., Chaudhuri B.B., A MLP Classifier for Both Printed and Handwritten Bangla Numeral Recognition, Computer Vision, Graphics and Image Processing, Lecture Notes in Computer Science,Springer Berlin / Heidelberg,Volume 4338/2006, Pp. 796-804, ISBN: 978-3-540-68301-8.

[MAR01] D. Martin, C. Fowlkes, D. Tal, et J. Malik. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. 8th Int'l Conf. Computer Vision, July 2001, pp. 416–423.

[MEH96] Mehrotra, A. et Trick, M. A. (1996). A column generation approach for graph coloring. INFORMS Journal on Computing, 8(4), pp.344–354.

[MIL96] Miletzki U., Documents on the Move: DA&IRDriven Mail Piece Processing Today and Tomorrow, in Proc. of DAS' 96, pp. 547-563.

[MOH07] MOHAMED H.K., Automatic documents classification, IEEE International Conference,

Computer Engineering & Systems, 2007, pp. 33-37.

[MOR63] Morgan J.N., Sonquist J.A., Problems in the analysis of survey data, and a proposal. JASA, 1963, vol.58, N°302, Zbl 0114.10103.

[MOR92] MORI S., SUEN C.Y., YAMAMOTO K., Historical rcview of OCR research and development , Proceedings of the IEEE, juillet 1992, vol. 80, Issue 7, pp. 1029-1058.

[MUL06] Rémy Mullot, Livre: Les documents écrits de la numérisation à l'indexation par le contenu, Editeur : Hermes science Publication, Nov 2006, 365 pp., ISBN-10 : 2746211432.

[MWO97] Wolf M., Niemann H., Schmidt W., Fast address block location on handwritten and machine printed mail-piece images, Document Analysis and Recognition, Fourth International Conference, IEEE, 1997, Vol. 2, pp.753-757.

[NAG84] Nagy, G., Seth, S.: Hierarchical Representation of Optically Scanned Documents. In: Proceedings of Seventh Int'l Conf. Pattern Recognition, 1984, pp.347-349.

[NAG92] NAGY G., At the frontiers of OCR, Proceedings of the IEEE, 1992, 80 (7), pp.1093-1100, 1992.

[NAS80] Nassimi D., Sahni S., Finding connected components and connected ones on a mesh-connected parallel computer, SIAM J. of Computing, Nov. 1980, vol.9, n°4. pp.744-757.

[NAT01] Nattee, C., Numao, M.: Geometric method for document understanding and classification using on-line machine learning. In: Proceedings of the 6th International Conference on Document Analysis and Recognition, Seattle, USA, 10–13 September 2001, pp. 602–606.

[NIB86] Niblack W., An Introduction to Digital Image Processing, Prentice Hall, 1986, pp.115-116.

[NIC06] Nicolas S., Heutte, L., Paquet T., Extraction de la structure de documents manuscrits complexes à l'aide de champs Markoviens, in DIAL, pp. , 2006.

[OGA03] Ogata H., Watanabe S., Imaizumi A., Yasue T., Furukawa N., Sako H., Fujisawa H., Form type identification for banking applications and its implementation issues. In: Proceedings, of Document Recognition and Retrieval X (IS&T/SPIE electronic imaging), Santa Clara, California, 20-24 Jan2003, Series 5010, pp.208–218.

[ORI94] Oriot JC., Barba D., Gilloux M., Localisation du bloc adresse sur les objets postaux par une méthode de segmentation ascendante : évaluation et optimisation, Actes du 9ème Congrès AFCET Reconnaissance des formes et intelligence artificielle, 1994, pp. 473-484,

[OTS78] OTSU N., A threshold selection method from gray-level histograms, IEEE Trans. on SMC, 1979, 9 (1), pp.62-66.

[PAL92] Palumbo PP.W., Srihari S.N., Soh J., Sridhar R., Demjanenko V., Postal address block location in real time Computer, IEEE, 1992, vol.25, pp. 34-42.

[PAL99] Pal U., Chaudhuri B.B., Automatic separation of machine-printed and hand-written text lines Document Analysis and Recognition, ICDAR '99. Proceedings of the Fifth Interna-

tional Conference on 20-22 Sept. 1999, pp.645-648.

[PAP96] Papamarkos N., Tzortzakis J., Gatos B., Determination of run-length smoothing values for document segmentation, Electronics, Circuits, and Systems, ICECS '96., Proceedings of the Third IEEE International Conference, 1996, vol. 2, pp. 684-687.

[PAR05] Park J.H., Jang I.H., Kim N.C., Skew correction of business card images acquired in PDA Vision, Image and Signal Processing, IEE Proceedings, 9 Dec.2005, vol.152, Issue 6, pp.668-676.

[PAR97]. Parker J., Algorithms for image processing and computer vision, Pub. Wiley, John & Sons, Incorporated, 1997, n°1, 417 pp. ISBN-13: 9780471140566.

[PAS07] PASCHOS V., livre, Optimisation combinatoire5: problèmes paradigmatiques et nouvelles problématiques, Lavoisier, France, 2007, pp. 270.

[PAV77] Pavlidis T., Structural Pattern Recognition. Berlin: Springer-Verlag, 1977, vol.1, 302 p.

[PAV91] Pavlidis T., Zhou J., Page segmentation by white streams. In: Proceedings of the First International Conference on Document Analysis and Recognition, St, Malo, France, sep 1991, pp.945-953.

[PAV92] Pavlidis, T. (1992). Why progress in machine vision is so slow". Pattern Recognition Letters, vol.13, pp.221-225

[PEN01] Liangrui Peng, Ming Chen, Changsong Liu, Xiaoqing Ding, Jirong Zheng, An Automatic Performance Evaluation Method for Document Page Segmentation,Document Analysis and Recognition, ICDAR. 2001. Proceedings. Sixth International Conference, 2001, pp.134 -137.

[PHI01] Philipp-Foliguet S., Evaluation de la segmentation. Rapport Technique, 2001.

[POL01] Polikar R., Udpa L., Udpa S., Honavar V., Learn++, An Incremental Learning Algorithm for Supervised Neural Networks, IEEE Transactions on Systems,Man, and Cybernetics, 2001, vol. 31, pp. 497-508.

[POS86] Postl W., Detection of linear oblique structures and skew scan in digitized documents. Proc. Int. Conf. on Pattern Recognition, 1986, pp. 687-689.

[PRA03] Prasanna PP.S.R., Balaji S., Khezhie T.H., Vasanthanayaki C., Annadurai S., Destination address interpretation for automating the sorting process of Indian Postal System, TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region, 15-17 Oct. 2003, vol.2 pp.858-862.

[RAB89] Rabiner L.R., A tutorial on hidden Markov models and selected application in speech recognition, Proceedings of IEEE, 1989, vol.77, pp. 257-286.

[RAH03] RAHMAN A.F.R., FAIRHURST M.C., «Multiple classifier decision combination strategies for character recognition; A review », International Journal on Document Analysis and Recognition, 2003, vol. 5, pp. 166-194.

[RAM93] Ramponi G., Fontanot PP., Enhancing document images with a quadratic filter, Signal Process, 1993, vol. 33, pp. 23-34.

[RAM04] Ramel J.Y., Leriche S., Segmentation et analyse interactives de documents anciens imprimés, CIFED Colloque International Francophone sur l'Écrit et le document N07, la Rochelle, France, 2004, VOL.22, N°3 PP: 209-222.

[RAN91] Ranganathan N., Mehrotra R., Subramaniam S., A high speed systolic architecture for labeling connected components in an image, Proc. IEEE International Symposium on Parallel and Distributed Processing, Dallas, 1991, pp. 818-825.

[RAN95] Ranganathan N., Mehrotra R., Subramanian S., A high speed systolic architecture for labeling connected components in an image, Systems, Man and Cybernetics, IEEE Transactions, March 1995, vol.25, Issue 3, pp. 415-423.

[RAS97] Rasquinha A., Ranganathan N., C3L: a chip for con-nected component labeling VLSI Design, ICVD. Proceedings, Tenth International Conference on 4-7 Jan. 1997, pp. 446 - 450.

[RIN05] Ringlstetter, C.; Schulz, K.U.; Mihov, S.; Louka, K, The same is not the same - postcorrection of alphabet confusion errors in mixed-alphabet OCR recognition, Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference, 29 Aug.-1 Sept. 2005, vol.1, pp. 406-410.

[ROS66] Rosenfeld A., Pfaltz PP., Sequential Operations in Digital Picture Processing, Journal of the Association for Computing Machinery, 1966, pp.471-494.

[ROS82] ROSENFELD A., KAK A.C., Digital Picture Processing, Academic Press, Edition 2 , New York, Publisher : Academic Press, Inc. Orlando, FL, USA, 1982, pp. 349. ISBN:0125973020.

[ROS76] ROSENFELD A., HUMMEL R.A., ZUCKER S.W., « Scene labeling by relaxation operations », IEEE trans SMC 6, June 1976. pp.420-433.

[ROY05] K. Roy, S. Vajda, U. Pal, B.B. Chaudhuri, A. Belaid, A system for Indian postal automation Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on 29 Aug.-1 Sept. 2005 Pp.1060 - 1064 Vol. 2.

[RUM86] Rumelhart D. E., Hinton G. E., Williams R. J., Learning internal representations by error backpropagation. In J. L. McClelland In D. E. Rumelhart and the PDP research group, editors, Parallel distributed processing: explorations in the microstructure of cognition, 1986, pp. 318–362.

[RWO94] Ralph Wolf, C. Platt John, Postal Address Block Location Using A Convolutional Locator, Syllaptics, JIIc, 2698 Orchard Parkway, Sail Jose, CA 95134, Jalluary 7, 1994 Network.

[SAF00] Safabakhsh.R, Khadivi.S, Document skew detection using minimum-area bounding rectangle, Information Technology: Coding and Computing, 2000, Proceedings, International Conference on 27-29, pp.253-258.

[SAK03] Sako, H., Seki, M., Furukawa, N., Ikeda, H., Imaizumi, A.: Form reading based on form-type identification and form-data recognition. In: Proceedings of the 7th International Conference on Document Analysis and Recognition, Edinburgh, Scotland, 3–6 August 2003, pp. 926-930.

[SAM86] Samet H., Tamminen M., An Improved Approach to connected component labeling of images. Proceedings of CVPR'86, 1986, pp. 312-318.

[SAR07] M. Sarfraz, S. A. Mahmoud, Z. Rasheed, On Skew Estimation and Correction of Text, Computer Graphics, Imaging and Visualisation, CGIV '07 14-17 Aug. 2007, pp.308 - 313.

[SAU97] Sauvola J. and al., Adaptive document binarisation, Document Analysis and Recognition, ICDAR, Proceedings of the Fourth International Conference, 18-20 Aug. 1997, vol.1, pp.147-152.

[SAV98] Savakis A.E., Adaptive document image thresholding using foreground and background clustering Image Processing, ICIPP. 1998, Proceedings. ICIPP.1998 International Conference on 4-7 Oct. 1998, vol.3, pp.785-789.

[SAY73] Sayre K.M., Machine Recognition of Handwritten Words: A Project Report, Pattern Recognition, 1973, vol. 5, pp. 213-228.

[SEU04] Seung Ick Jang, Seon Hwa Jeong, Yun-Seok Nam, Classification of machine-printed and handwritten addresses on Korean mail piece images using geometric features Pattern Recognition, ICPR'04. Proceedings of the 17th International Conference on Volume 2, 23-26 Aug. 2004, vol.2, pp.383-386.

[SEW96] Sewell E., An improved algorithm for exact graph coloring. vol. 26 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science,American Mathematical Society, Providence, RI, USA, 1996, pp. 359-373.

[SEZ01] Sezgin M., Sankur B., Selection of thresholding methods for nondestructive testing applications, Image Processing, ICIPP. Proceedings. 2001 International Conference, 7-10 Oct. 2001, vol.3, pp.764-767.

[SHA98] Shan Mo J., Mathews, Adaptive, quadratic preprocessing of document images for binarisation Image Processing, IEEE Transactions, July 1998, vol.7, Issue 7, pp.992-999.

[SHA03] Sharat Chandran, Ananth K. Potty, and Milind Sohoni, Fast Image Transforms Using Diophantine Methods, IEEE TRANSACTIONS ON IMAGE PROCESSING, 2003, vol.12, n°6, pp. 678-684.

[SHI01] Shin, C., Doermann, D., Rosenfeld, A.: Classification of document pp. using structure-based features. Int. J. Doc. Anal. Recognit. 2001, 3(4), pp.232–247.

[SHI03] Zhixin Shi, Govindaraju.V, Skew detection for complex document images using fuzzy run length Document Analysis and Recognition, Proceedings, Seventh International Conference, 2003, pp.715 – 719.

[SHI04] Shi Z., Govindaraju Venu, Line separation for complex document images using fuzzy runlength, Document Image Analysis for Libraries, DIAL 2004, Proceedings, First International Workshop on 2004, pp.306-312.

[SHI90] Shizawa.M, Discrete invertible affine transformations, Pattern Recognition, 1990, Proceedings. 10th International Conference, 1990, vol.2, 134 -139.

[SIM97] Simon A., Pret J., Johnson A., A Fast Algorithm for Bottom-Up Document Layout Anal-

ysis,º IEEE Trans. Pattern Analysis and Machine Intelligence, 1997, vol. 19, Issue 3, pp. 273-277.

[SIV95] Sivaramakrishnan R., Phillips I.T., Ha J., Subramanium S., Haralick R.M., Zone classification in a document using the method of feature vector generation Document Analysis and Recognition, ICDAR., Proceedings of the Third International Conference on Volume 2, 14-16 Aug. 1995, vol.2, pp.541-544.

[SMI95] Smith.R, A simple and efficient skew detection algorithm via text row accumulation, Document Analysis and Recognition, 1995, Proceedings of the Third International Conference on, vol.2, pp. 1145-1148.

[SOU02] Souafi-Bensafi S., Contribution à la reconnaissance des structures des documents écrits : approche probabiliste, Thèse, Université Laval, Québec et École doctorale Informatique et information pour la société, Institut national des sciences appliquées de Lyon, France. 2002, pp.182-189.

[SRI96] Srihari S.N., Shin Y.C., Ramanaprasad V., Lee D.S., A System to Read Names and Addresses on Tax Forms, July 1996, PIEEE(84), n°7, pp. 1038-1049.

[SRI97] SRIHARI S.N., KUEBERT EJ., "Integration of Hand-Written Address Interpretation Technology into the United States Postal Service Remote Computer Reader System ", 41h International Conference on Document Analysis and Recognition (ICDAR '97), 1997.

[SRI89] Srihari S.N, Govindaraju V, Analysis of textual images using the Hough transform, Machine Vision and Applications, 1989, vol. 2, n°3, pp. 141-153.

[SRI95] SRIHARI S.N., LAM S.W., Character Recognition, Center of Excellence for Document Analysis and Recognition, State University of New York at Buffalo, Techical report CEDAR-TR-95-1, janvier 1995.

[STE99] Di Stefano L., Bulgarelli A., A simple and efficient connected components labeling algorithm, Image Analysis and Processing, ICIAPP.1999. Proceedings. International Conference on 27-29 Sept. 1999, pp.322-327.

[STR03] Strohmaier, C.M. Ringlstetter, C. Schulz, K.U. Mihov, S., Lexical postcorrection of OCR-results:the web as a dynamic secondary dictionary? Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on 3-6 Aug. 2003 pp. 1133- 1137; ISBN: 0-7695-1960-1.

[SUC04] Suchitra S., Lam S.K., Srikanthan T., Novel schemes for high-throughput image rotation Signals, Systems and Computers, Conference Record of the Thirty-Eighth Asilomar Conference, 7-10 Nov. 2004, vol.2, pp.1884 -1888.

[SUN05] Hung-Ming Sun, Page Segmentation for Manhattan and Non-Manhattan Layout Documents via Selective CRLA Full text Publisher Site Source ICDAR, Proceedings of the Eighth International Conference on Document Analysis and Recognition table of contents, 2005, pp. 116 – 120.

[SUN08] Junxi Sun,; Dongbing Gu,; Hua Cai,; Guangwen Liu,; Guangqiu Chen,;Bayesian document segmentation based on complex wavelet domain hidden Markov tree models , Information and Automation, 2008. ICIA 2008. International Conference on 20-23 June

2008, pp.493-498

[SUN97] Changming Sun, Deyi Si, Skew and Slant Correction for Document Images Using Gradient, 4th International Conf, on Document Analysis and Recognition, Ulm, Germany, 1997, pp.142-146.

[SUR99] Sural S., Das PP.K., two-step algorithm and its parallelization for the generation of minimum containing rectangles for document image segmentation, Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999. ICDAR '99, pp. 173-176. ISBN: 0-7695-0318-7.

[SUZ03] Suzuki K., Horiba I., Sugie N., Linear-time connected-component labeling based on sequential local operations, Computer Vision and Image Understanding, 2003, vol. 89, pp.1-23.

[TAG04] Taghva, Kazem; Coombs, Jeffrey S.; Pereda, Ray; Nartker, Thomas A, Address extraction using hidden Markov models, Document Recognition and Retrieval XII. Edited by Barney Smith, Elisa H.; Taghva, Kazem. Proceedings of the SPIE, 2004, vol. 5676, pp. 119-126.

[TAN98] Tang Y.Y., Cheriet M., Liu J., Said J.N., Suen C.Y., Document analysis and recognition by computers. In: Handbook of Pattern Recognition and Computer Vision, 2nd edn. World Scientific, Singapore, 1998, pp. 579–612.

[TAY95] Taylor S., Lipshutz M., R. Nilson, Classification and functional decomposition of business documents. In: Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, Canada, 14-15 August 1995, pp. 563–566 .

[THI04] Thillou C., Gosselin B., Combination of binarization and character segmentation using color information Signal Processing and Information Technology, ISSPIT. Proceedings of the Fourth IEEE International Symposium on 18-21 Dec. 2004, pp.107-110.

[THU99] Thulke M., Margner V., Dengel A., Quality Evaluation of Document Segmentation Results,ICDAR'99, Bangalore, India, 1999, pp. 450.

[TOM02] Tombre K., Tabbone S., Pélissier L., Lamiroy B., Dosch PP., Text/Graphics Separation Revisited, IARP workshop on document analysis systems, N°5, DAS 2002.Princeton, Etats-Unis. Vol.2423, pp. 200-211.

[TRE04] Trémeau A., Fernandez-Maloigne C., Bonton PP., Image numérique couleur, de l'acquisition au traitement, DUNOD, pp. 300.

[TRI95a] Trier O.D., Taxt T., Improvement of 'integrated function algorithm' for binarization of document images, Pattern Recognition Letters, 1995, vol. 16, n°3, pp. 277-283.

[TRI95b] Trier O.D., Jain A.K., Goal-directed evaluation of binarization methods, Pattern Analysis and Machine Intelligence, IEEE Transactions, Dec. 1995, vol.17, Issue 12, pp.1191-1201.

[TRI96] TRIER O., JAIN A., T AXT T., Feature extraction methods for character recognition A survey , Pattern Recognition, 1996, vo.29, pp. 641-662.

[TSA 07] Yao-Hong Tsai, A New Approach for Image Thresholding under Uneven Lighting Con-

ditions, Computer and Information Science, ICIS 2007. 6th IEEE/ACIS International Conference on, 11-13, July 2007. pp.123-127.

[UNS95] Michael Unser, Philippe ThCvenaz, and Leonid Yaroslavsky, Convolution-Based Interpolation for Fast, High-Quality Rotation of Images, 1995, IEEE TRANSACTIONS ON IMAGE PROCESSING, 1995, vol.4, n°10, pp.1371-1381.

[VAL00] Valverde J.S., Grigat R.R., Optimum binarization of technical document images, Image Processing, ICIPP. 2000. Proceedings. 2000 International Conference, 10-13 Sept. 2000, vol.3, pp.985-988.

[VAP79] Vapnik V.N., Estimation of Dependences Based on Empirical Data in Russian.Nauka, Moscow, Russia, 1979.

[VIA91] Viard-Gaudin C., Barba D., A multi-resolution approach to extract the address block on flat mail pieces, ICASSP-91, International Conference, 1991, vol.4, pp. 2701- 2704.

[VIN00] Alessandro Vinciarelli, Juergen Luettin, A new normalization technique for cursive handwritten words, Pattern recognition letters 22, 2000, pp.1043-1050.

[VIN02] Vinciarelli A., A surveyon o!-line Cursive Word Recognition, 2002, The journal of the Pattern Recognition, society, 2002, pp.1433-1446.

[VOL59] Volder J.E., The CORDIC Trigonometric Computing, Technique, IRE Trans. Elecron. Comput, 1959, vol. EC-8, pp 330-334.

[VUU03] VUURPIJL L., SCHOMAKER L., V AN ERP M., «Architectures for detecting and solving conflicts : Two-stage classification and support vector classifiers », International Journal on Document Analysis and Recognition, vol. 5, pp. 213-223, 2003.

[WAH82] Wahl F. M., Wong K. Y., Casey R. G., Block Segmentation and Text Extraction in Mixed Text/Image Documents, Computer Graphics Image Processing, 1982, vol.20, pp. 375-390.

[WAN88] Wang C.H., Palumbo PP.W. and Srihari S.N. , Object Recognition in Visually Complex Environments: An Architecture for Locating Address Blocks on Mail Pieces.Proc. Ninth Intl. Conf. on Pattern Recognition, Rome, 1988, pp.365-367.

[WAN89] Wang D., Shihari S.N., Classification of newspaper image blocks using texture analysis, Computer Vision Graphics Image Process, 1989, vol.47, pp.327-352.

[WAN95] Shin-Ywan Wang; Yagasaki, T; Block selection: a method for segmenting a page image of various editing styles, ICDAR.1995, vol.1, pp.128-133.

[WAT95] Watanabe T., Luo Q., Sugie N., Layout recognition of multikinds of table-form documents. IEEE Trans. Pattern Anal. Mach. Intell, 1995, vol.17, n°4, 432–445.

[WER03] Werra D., Hansen PP., Using stable sets to bound the chromatic number. Information Processing Letters 87, 2003, pp.127-131.

[WER74] Werbos PP. J., Beyond: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis, Masters Thesis, Harvard University, 1974.

[WIL01]. William K. Pratt, Digital Image Processing, PIKS Inside, Third Edition, 2001.

[WOL02] Wolf C., Jolion J.M., Chassaing F., Text Localization, Enhancement and binarization in Multimedia Documents, In Proceedings of the International Conference on Pattern Recognition ICPR, Quebec City, Canada, August 11th-15th, 2002, vol.4, pp.1037-1040.

[WOL90] Wolberg George, Digital image warping, IEEE Computer Society Press, Los Alamitos, CA, 1990.

[WON82] Wong K.Y., Casey R.G., Wahl F.M., document Analysis System, IBM journal Res. Dev. 1982, vol. 26. n°. 6, pp. 647-656.

[WU05] Kesheng Wu, Ekow Otoo, and Arie Shoshani. Optimizing connected component labeling algorithms. Proceedings of SPIE Medical Imaging Conference, 2005.

[WUS03] Sue Wu; Adnan Amin, Automatic thresholding of gray-level using multistage approach, Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on Volume , Issue , 3-6 Aug. 2003, vol.1, pp. 493-497.

[WUV99] Victor Wu, Raghavan Manmatha, Edward M. Riseman, TextFinder: An Automatic System to Detect and Recognize Text In Images, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, nov 99, vol.21, n°11., pp. 1224-1229.

[XIA99] Xiaoyi Jiang, H.Bunke, D.Widmer-Kljajo, Skew detection of document images by focused nearest-neighbor clustering, 1999, Document Analysis and Recognition, ICDAR 99, Proceedings of the Fifth International Conference , pp. 629 - 632.

[XUE99] Junliang Xue, Xiaoqing Ding, Changsong Liu, Shiyan Pan, Hongwei Kong, Destination address block location on handwritten Chinese envelope, Document Analysis and Recognition, ICDAR '99. Fifth International Conference, IEEE, 1999, pp. 737 - 740.

[YAM96] Yamashita A., Amano T., Hirayama Y., Itoh N., Katoh S., Mano T., Tokokawa K., A Document Recognition System and Its Applications. IBM Journal of Research and Development, 1996, vol. 40, pp. 341-352.

[YAN06] Feng Yang, Zheng Ma, Mei Xie, A Novel Binarization Approach for License Plate, Industrial Electronics and Applications, ICIEA 1ST IEEE Conference on May 2006 pp.1 - 4.

[YAN89] Yanowitz S.D., Bruickstein A.M., A new method for image segmentation, Computer Vision, Graphics and Image Processing, Apr.1989, vol. 46, n°.1, pp. 82-95.

[YAN93] Yan H., Skew correction of document images using interline crosscorrelation, Graph. Models Image Process, 1993, vol.55, n°6, pp. 538-543.

[YAN98] Yanikoglu B. A., Vincent L., Pink Panther, a complete environment for groundthruthing and benchmarking document page segmentation, Pattern Recognirion, vol. 31, n° 9, 1998, pp. 1191-1204.

[YAN99] Yang Y., Liu X.(1999)A Re-examination of Text Categorization Methods Proc. of the 22nd ACM SIGIR Conference, Pp 42-49.

[YAN88] Yang, X.D.; Design of fast connected components hardware, Computer Vision and Pattern Recognition, Proceedings CVPR '88., Computer Society Conference on 5-9 June 1988, pp.937-944.

[YAO07] Jin-Liang Yao, Yan-Qing Wang, Lu-Bin Weng, Yi-Ping Yang, Locating text based on connected component and SVM, Wavelet Analysis and Pattern Recognition, 2007. ICWAPR 07. International Conference, 2-4 Nov. 2007, vol.3, pp.1418 – 1423.

[YEH87] A. L. PP.S. Yeh, S. Antoy and A. Rosenfeld. Address location on envelopes. Pattern Recognition, 1987, vol.20, n°2, pp.213-227.

[YON03] Yonekura E.A., Facon J., 2D histogram-based segmentation of postal envelopes , Computer Graphics and Image Processing. SIBGRAPI. XVI Brazilian Symposium on 12-15 Oct. 2003, pp.247-253.

[YUB96] Yu B., Jain A. K., A robust and fast skew detection algorithm for generic documents, Pattern Recognition, 1996, vol. 29 , n°10, pp. 1599 -1629.

[YUB97] Yu B., Jain A.K., Mohiuddin M., Address block location on complex mail pieces, Document Analysis and Recognition, Fourth International Conference, IEEE, 1997, vol.2, pp. 897-901.

[ZHA96] Zhang Y. J., A Survey on Evaluation Methods for Image Segmentation, Pattern Recognition, 1996, vol. 29, n°.8, pp: 1335-1346.

[ZHE03] Zheng Yefeng, Li Huiping, D.Doermann, Text identification in noisy document images using Markov random model, Document Analysis and Recognition. Proceedings. Seventh International Conference on 3-6 Aug. 2003, vol.1, pp.599 - 603.

[ZHO02] Zhou J., Krzyzak A., Suen C.Y., Verification - A Method of Enhancing the Recognizers of Isolated and Touching Handwritten Numerals Pattern Recognition. Pattern Recognition, 2002, vol.35, n°5, pp.1179-1189.

[ZHU02a] Hui Zhu; Zhizhong Fu; Zaiming Li, A new image thresholding method based on relative entropy, Communications, Circuits and Systems and West Sino Expositions, IEEE 2002 International Conference, 29 June-1 July 2002, vol.1, pp.634- 638.

[ZHU02b] Xiaoyan Zhu; Xiaoxin Yin, A new textual/non-textual classifier for document skew correction Pattern Recognition, ICPR.. Proceedings. 16th International Conference, 11-15 Aug. 2002, vol.1, pp.480 -482.

FOLIO ADMINISTRATIF

THESE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

NOM : GACEB
(avec précision du nom de jeune fille, le cas échéant)

DATE de SOUTENANCE : 19/09/2009

Prénoms : Djamel

TITRE : Contributions au tri automatique de documents et de courrier d'entreprises

NATURE : Doctorat

Numéro d'ordre : 2009-ISAL-0077

Ecole doctorale : Informatique et Mathématiques

Spécialité : Documents numériques, Images et Systèmes d'Information Communicants

Cote B.I.U. - Lyon : T 50/210/19 / et bis

CLASSE :

RESUME :

Ce travail de thèse s'inscrit dans le cadre du développement de systèmes de vision industrielle pour le tri automatique de documents et de courriers d'entreprises. Ces systèmes sont par nature très exigeants en temps de traitement mais aussi en justesse et précision des résultats. Les systèmes actuels sont composés, pour la plupart, de modules séquentiels exigeant des algorithmes efficaces et rapides tout au long de la chaîne des traitements, depuis les étapes de bas niveau jusqu'aux étapes de niveau supérieur d'analyse fine et de reconnaissance des contenus. Les architectures existantes, dont nous avons balayé les spécificités dans les trois premiers chapitres de la thèse, présentent des faiblesses qui se traduisent par des erreurs de lecture et des rejets que l'on impute encore trop souvent aux OCR. Or, les étapes responsables de ces rejets et de ces erreurs de lecture sont les premières à intervenir dans le processus, à savoir celles de segmentation et de localisation de zones d'intérêts ; ces deux étapes qui s'impliquent mutuellement conditionnent les performances des systèmes et le rendement des chaînes de tri automatique.
Nous avons ainsi choisi porter notre contribution sur les aspects inhérents à la segmentation des images de courriers et la localisation de leurs régions d'intérêt (comme la zone d'adresse) en investissant une nouvelle approche pyramidale de modélisation par coloration hiérarchique de graphes ; à ce jour, la coloration de graphes n'a jamais été exploitée dans un tel contexte. Elle intervient dans notre contribution à toutes les étapes d'analyse de la structure des documents ainsi que dans la prise de décision pour la reconnaissance (reconnaissance de la nature du document à traiter et reconnaissance du bloc adresse). La partie de reconnaissance a été conçue autour d'un apprentissage traité à l'aide d'un modèle unique portant sur la b-coloration de graphe.
Notre architecture a été conçue pour réaliser essentiellement les étapes d'analyse de structures et de reconnaissance en garantissant une réelle coopération entres les différents modules d'analyse et de décision. Elle s'articule autour de trois grandes parties : une partie de segmentation bas niveau (binarisation et recherche de connexités), une partie d'extraction de la structure physique par coloration hiérarchique de graphe et une partie de localisation de blocs adresse et de classification de documents. Les algorithmes impliqués dans le système ont été conçus pour leur rapidité d'exécution (en adéquation avec les contraintes de temps réels), leur robustesse, et leur compatibilité. Les expérimentations réalisées dans ce contexte sont très encourageantes et offrent également de nouvelles perspectives à une plus grande diversité d'images de documents.

MOTS-CLES : Extraction de la structure physique, catégorisation de documents, localisation de bloc adresse, coloration et b-coloration de graphes, tri de courriers en temps réel.

Laboratoire (s) de recherche : LIRIS – INSA de Lyon

Directeurs de thèse : M. Hubert EMPTOZ et Mme. Véronique EGLIN

Président de jury : M. Jean-Marc OGIER

Composition du jury : M. Mohamed CHERIET, VIARD-GAUDIN, Jean-Marc OGIER, Hubert EMPTOZ, Véronique EGLIN, Bruno MAISONNEUVE.