

Extraction automatique de champs numériques dans des documents manuscrits

Clément Chatelain, Laurent Heutte, Thierry Paquet

Laboratoire PSI, CNRS FRE 2645,
Université de Rouen, 76800 Saint Etienne du Rouvray, FRANCE
clement.chatelain@univ-rouen.fr

Résumé. Nous décrivons dans cet article une chaîne de traitement complète et générique permettant d’extraire automatiquement les champs numériques (numéros de téléphone, codes clients, codes postaux) dans des documents manuscrits libres. Notre chaîne de traitement est constituée des trois étapes suivantes: localisation des champs numériques potentiels selon une approche markovienne sans reconnaissance chiffre ni segmentation, reconnaissance des séquences extraites, et vérification des hypothèses de localisation / reconnaissance en vue de limiter la fausse alarme générée lors de l’étape de localisation. L’évaluation de notre système sur une base de 300 courriers manuscrits montre des performances en rappel-précision intéressantes.

1 Introduction

Aujourd’hui, la lecture automatique des documents manuscrits se limite à quelques cas applicatifs particuliers : lecture automatique de chèques ou d’adresses postales, reconnaissance des champs d’un formulaire. Cette lecture est possible car le contenu de ces documents est très largement contraint : structure du document stable, position des informations connue, redondance de l’information, lexique limité, etc. Lors de la lecture, le système bénéficie ainsi d’informations *a priori* importantes permettant de limiter ou de vérifier les hypothèses de reconnaissance, autorisant une lecture fiable des documents.

Peu de travaux abordent des problèmes de reconnaissance moins contraints car il est alors plus difficile de bénéficier de moyens automatiques de vérification des hypothèses de reconnaissance. C’est le contexte de nos travaux portant sur la lecture automatique des courriers entrants manuscrits. Il s’agit de courriers manuscrits tels que des lettres de réclamation, de changement d’adresse, de modification de contrat, etc., reçus en très grand nombre quotidiennement par des grandes organisations. Contrairement aux applications précédemment citées, aucune information *a priori* n’est disponible : le contenu, la structure, l’expéditeur ou encore l’objet du document sont totalement inconnus du système de lecture, ce qui rend la lecture intégrale du document extrêmement délicate. Il est cependant possible de considérer des problèmes de lecture partielle du document, visant à en extraire l’information pertinente. C’est ce que nous envisageons dans cet article en proposant une méthode de localisation et de reconnaissance de champs numériques (numéros de téléphones, codes clients, etc.) dans des courriers entrants manuscrits (voir figure 1). La reconnaissance de ces champs permettra par

Extraction automatique de champs numériques dans des documents manuscrits

exemple d'identifier l'expéditeur par le biais du numéro de téléphone, ou de déterminer le type de contrat à l'aide du code client, ce qui autorise un acheminement du courrier vers le service concerné au sein de l'organisation.

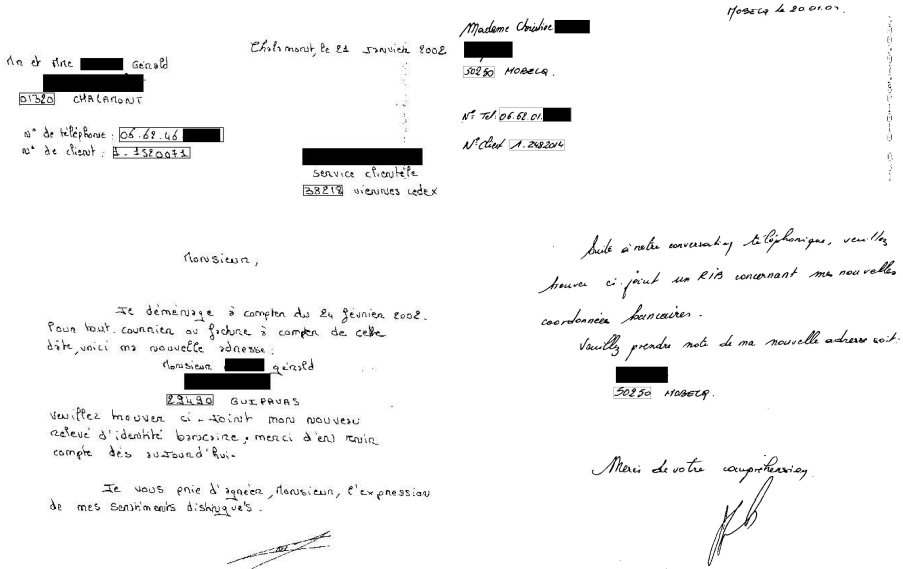


FIG. 1 – Exemple de courriers manuscrits où les champs numériques à extraire sont encadrés.

La méthode présentée comporte trois grandes étapes :

- Une première étape de localisation rapide sans reconnaissance chiffre ni segmentation permet d'extraire des séquences de composantes susceptibles de constituer des champs numériques. Cette étape basée sur l'exploitation de la syntaxe connue des champs a déjà été présentée dans Koch et al. (2004) puis améliorée dans Chatelain et al. (2004). Nous la décrivons donc sommairement dans cet article et rappelons ses performances pour justifier les deux étapes de traitement suivantes.
- La deuxième étape consiste à soumettre les hypothèses de localisation à un module de reconnaissance de champs fournissant leur valeur numérique. Cette étape repose sur l'utilisation d'un classifieur chiffre et d'un module de segmentation de chiffres liés.
- La troisième étape consiste à traiter le problème des fausses alarmes générées par l'étape de localisation. Nous présentons un module de vérification qui accepte ou rejette les hypothèses de champ numérique en exploitant une combinaison d'informations provenant des différentes étapes de traitement.

L'article est organisé de la manière suivante : la section 2 décrit sommairement la méthode d'extraction des champs dans les documents manuscrits libres. La section 3 présente la méthode de reconnaissance des champs basée sur un classifieur chiffre et une méthode de reconnaissance de chiffres liés. Nous présentons dans la partie 4 les performances de notre

système, ainsi qu'une étape de vérification des hypothèses de reconnaissance des champs afin de rejeter les fausses alarmes.

2 Localisation des champs

2.1 Une approche "dirigée par la syntaxe"

L'approche proposée ici pour la localisation des champs est basée sur une modélisation markovienne d'une ligne de texte. Nous avons déjà eu l'occasion de présenter cette approche dans Koch et al. (2004); Chatelain et al. (2004). Rappelons seulement que ce modèle exploite la syntaxe spécifique des champs numériques que l'on souhaite extraire (nombre de chiffres, présence et position de séparateurs...) pour parvenir à localiser les séquences numériques, sans toutefois procéder à la segmentation des composantes connexes ni à la reconnaissance des chiffres. Nous interprétons globalement la séquence des composantes connexes de chaque ligne pour associer à chaque composante son étiquette : textuelle ou numérique. Toutefois, puisque l'approche ne procède pas à la segmentation des composantes connexes, une composante numérique peut correspondre à un ou plusieurs chiffres, ou même un séparateur (point, tiret...). De ce fait, on doit introduire dans le modèle de ligne des étiquettes correspondant à ces situations : D (Digit ou chiffre), DD (Double Digits ou chiffres liés), S (Séparateur). En ce qui concerne les composantes textuelles, le modèle ne comprend qu'une seule classe, appelée classe Rejet, pour décrire l'ensemble des situations possibles : caractère isolé, fragment de mot, mot, diacritique, signe de ponctuation. La figure 2 représente une ligne de texte avec les étiquettes associées à chacune des composantes qui la constitue.

de client 1.3989360. D'ailleurs je vous remercie
 R R R D S DD DDDDD D R R RR RRR R RRR R RRRR

FIG. 2 – Exemple d'étiquetage des composantes d'une ligne comprenant un code client.

Ces quatre classes constituent les états du modèle markovien. La construction des modèles se fait de la manière suivante : nous fixons le nombre d'état Digit, Double Digit et Séparateur pour chaque type de champ, ainsi qu'un état Rejet. La matrice des probabilités de transitions est déterminée par une estimation statistique sur une base annotée. La figure 3 montre les modèles de Markov ainsi construits, où les flèches entre les états représentent les probabilités de transition non nulles.

L'alignement des séquences de composantes reconnues sur ces modèles garantit de ne conserver que les séquences syntaxiquement correctes. L'extraction des champs numériques dans les lignes de textes consiste alors à rechercher le meilleur alignement dans le treillis des hypothèses de classification. Ceci est réalisé par l'algorithme de Viterbi Forney (1973).

Le processus d'extraction des champs repose donc sur les étapes suivantes :

Segmentation en lignes : les lignes de texte sont extraites grâce à une approche de regroupement des composantes connexes inspirée de Likforman-Sulem et Faure (1995).

Classification des composantes connexes : il s'agit de classer les composantes connexes de chaque ligne selon qu'elles appartiennent à un champ numérique (Digit, DoubleDigit, Séparateur) ou non (Rejet). La caractérisation des composantes est réalisée à l'aide de deux jeux

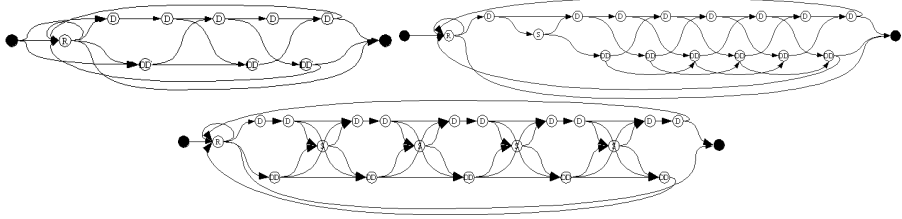


FIG. 3 – Modèles de Markov pour une ligne de texte contenant un code postal, un code client ou un numéro de téléphone.

de caractéristiques, présentés à deux classificateurs de type MLP entraînés grâce à l’algorithme de rétropropagation du gradient. Nous combinons ensuite les résultats des deux MLP.

Analyse syntaxique : cette dernière étape permet d’extraire les champs recherchés grâce à l’analyse syntaxique des lignes de texte. L’analyseur syntaxique corrige les éventuelles erreurs de classification de l’étape précédente en alignant les hypothèses de reconnaissance sur un modèle markovien d’une ligne de texte pouvant contenir un champ numérique.

Cette méthode d’extraction des champs est une alternative intéressante à l’utilisation d’une stratégie de segmentation-reconnaissance sur l’intégralité du document, puisque seuls les champs extraits seront soumis à un reconnaiseur.

2.2 Résultats

Les expérimentations ont été réalisées sur deux bases distinctes d’images de courriers entrants manuscrits : la première (292 images) a été utilisée comme base d’apprentissage pour la classification des composantes connexes ainsi que pour déterminer les probabilités de transition des modèles de Markov et pour paramétrer le système ; la seconde (293 documents) a servi à tester notre approche.

La détection des champs numériques est réalisée en effectuant l’analyse de chaque ligne d’un document. L’analyseur syntaxique se prononce pour la présence (détection) ou l’absence (rejet) d’un champ sur la ligne en cours d’analyse. Un champ est considéré comme convenablement détecté si et seulement si “aucune composante du champ étiqueté n’est rejetée et si toutes les composantes connexes dans le champ détecté appartiennent au champ étiqueté”.

Le tableau 1 donne les taux de détection des champs en rang 1, 2 et 5.

détection	codes postaux	téléphones	codes client
RANG1	69	75	81
RANG2	81	81	89
RANG5	89	91	94

TAB. 1 – Taux de détection en rang 1/ rang 2/ rang 5

On constate que suivant le type de champ, 70 à 80 % des champs sont détectés en première proposition. Ces résultats augmentent significativement lorsque l'on considère les 2 ou 5 premières propositions de l'analyseur. Les résultats sont meilleurs pour les champs qui possèdent une syntaxe plus contraignante tels que le numéro de téléphone et le code client (nombre de chiffres plus important, présence de séparateurs) que sur les champs faiblement contraints (codes postaux).

La majorité des champs ont ainsi été localisés, sans reconnaissance chiffre. L'étape suivante consiste à soumettre les hypothèses de localisation des champs à un module de reconnaissance afin d'obtenir leur valeur numérique.

3 Reconnaissance des champs

Contrairement à la majorité des systèmes de reconnaissance de documents manuscrits où la localisation et la reconnaissance des informations sont intimement liées, l'exploitation de la connaissance *a priori* sur la syntaxe des champs nous a permis de localiser les champs numériques sans les reconnaître. La reconnaissance intervient donc en fin de traitement et permet la vérification des hypothèses de localisation.

L'étape de reconnaissance des champs numériques s'appuie sur l'exploitation des hypothèses de classification fournies lors de l'étape de détection. En effet, nous bénéficions pour chaque champ extrait de l'hypothèse de classification "Digit", "Séparateur" ou "Double digit" des composantes. Il s'agit donc de déterminer l'hypothèse de classification *chiffre* pour chacune de ces composantes (voir figure 4).

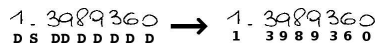


FIG. 4 – Détermination des hypothèses de classification chiffre à partir des hypothèses de classification Digit, Séparateur, Double Digit.

Pour les composantes dont l'hypothèse de classification est "Digit", il suffit de soumettre l'imagette à un classifieur chiffre qui déterminera la meilleure hypothèse de classification "chiffre". La description du classifieur chiffre est présentée dans la section 3.1. Les composantes "Séparateur" sont ignorées lors de cette étape, puisqu'elles n'interviennent pas dans la valeur numérique du champ à reconnaître. La reconnaissance des composantes classifiées comme "Double digit" est effectuée de la manière suivante : comme nous savons que la composante contient deux chiffres liés, il nous faut trouver la meilleure segmentation des deux chiffres, et les reconnaître. Cette étape est présentée dans la section 3.2. Dans la section 3.3, nous présentons les résultats obtenus sur les champs numériques isolés.

3.1 Classifieur chiffre

La reconnaissance de chiffres isolés a bénéficié de très nombreux travaux ces dernières années, notamment dans le cadre de la reconnaissance de montants numériques de chèques, de champs numériques dans les formulaires, ou encore de reconnaissance de codes postaux dans les adresses postales Plamondon et Srihari (2000). Ces systèmes reposent sur l'extraction

de nombreuses caractéristiques Trier et al. (1996) et l'utilisation de classifieurs performants Jain et al. (2000). Néanmoins, aucun extracteur ni classifieur n'a pu montrer de supériorité incontestable par rapport aux autres. Partant de ce constat, il est intéressant d'exploiter la complémentarité entre plusieurs classifieurs par une combinaison de type parallèle ou séquentielle. Nous avons ainsi choisi d'effectuer une combinaison parallèle de deux classifieurs de type perceptron multicouche (ou "MultiLayer Perceptron : MLP") auxquels sont soumis deux vecteurs de caractéristiques :

- Le vecteur de caractéristiques du *chaincode* extrait du contour des composantes a montré son efficacité dans de nombreux problèmes de reconnaissance Kimura et al. (1994). Après avoir effectué un pavage de l'imagette, l'histogramme des directions de Freeman des pixels est extrait dans chaque zone de l'image. Les histogrammes constituent les caractéristiques du vecteur. Nous considérons un voisinage 8-connexe et un pavage $4 * 4$, ce qui fournit un vecteur à 128 caractéristiques.
- Nous utilisons également le vecteur statistique/structurel développé dans nos travaux antérieurs Heutte et al. (1998), constitué de 117 caractéristiques réparties en 6 familles (projections, profils, intersections, fin de traits et jonctions, concavités, et extrema) et qui a prouvé son efficacité dans la discrimination robuste de caractères manuscrits tels que les chiffres, lettres majuscules et même graphèmes Heutte et al. (1998).

Ces vecteurs sont soumis à deux MLP construits sur le même schéma : une couche d'entrée contenant autant de neurones que de caractéristiques : 128 et 117 ; une couche cachée contenant $(\text{nombre d'entrees} + \text{nombre de sorties})/2$ neurones ; et une couche de sortie contenant autant de neurones que de classes : 10. Ces deux MLP ont été entraînés sur une base de 115000 chiffres manuscrits étiquetés provenant de formulaires. La base de test permettant de contrôler les apprentissages des réseaux et la base de validation sont constituées chacune de 39000 éléments. Nous appellerons par la suite "MLP117" le MLP entraîné sur le vecteur statistique / structurel, et "MLP128" le classifieur entraîné sur le vecteur du chaincode.

MLP117 et MLP128 sont combinés avec une règle de type produit. Le tableau 2 donne les taux de reconnaissance de MLP117, MLP128, et la combinaison des deux.

taux de reconnaissance	TOP1	TOP2	TOP3
MLP128	95,76	98,39	99,18
MLP117	96,87	98,80	99,38
combinaison	98,00	99,24	99,65

TAB. 2 – Taux de reconnaissance en TOP 1, 2, 3 pour les classifieurs MLP117, MLP128 et la combinaison des deux avec un opérateur produit.

3.2 Reconnaissance des chiffres liés

Nous avons présenté le classifieur chiffre permettant de reconnaître les chiffres isolés, nous nous intéressons dans cette partie à la reconnaissance des composantes dont l'hypothèse de classification lors de la première étape est "chiffres liés" (DD). Une stratégie pour cette opération pourrait être la reconnaissance globale de la composante à l'aide d'un classifieur "chiffres liés" comportant autant de classes que de combinaison possibles, soit 100 (classifieur

100 classes [00..99]). Cette stratégie nécessite toutefois une base d'apprentissage conséquente comportant un nombre suffisamment élevé d'éléments dans chaque classe, afin de couvrir la variabilité inhérente à un problème réel : différents type d'écriture, nature des liaisons entre les chiffres (liaisons hautes, basses, multiples), etc. La conception d'un tel classifieur est donc a priori difficile à envisager. Nous avons donc orienté notre approche vers une segmentation de la composante pour identifier les deux chiffres qui la constituent. Dans la mesure où il est très difficile de déterminer sans reconnaissance le meilleur chemin de coupure pour séparer deux chiffres liés, nous avons mis en œuvre une stratégie de segmentation-reconnaissance à l'échelle de la composante. Plusieurs hypothèses de segmentation sont générées et soumises au classifieur chiffre qui se prononce sur les deux chiffres. Le choix de la meilleure hypothèse est déterminé à partir des confiances fournies par le classifieur. Cette stratégie repose donc sur la mise en œuvre d'un module de segmentation permettant la génération de plusieurs chemins de coupures, et sur un module de décision qui se prononce sur le choix du meilleur chemin de segmentation. Nous décrivons maintenant ces deux étapes.

3.2.1 Segmentation des composantes

Il existe de très nombreuses méthodes de segmentation explicite, généralement basées sur l'analyse des contours Casey et Lecolinet (1996). Nous avons utilisé une méthode de segmentation inspirée de l'algorithme "drop fall" Congedo et al. (1995), qui consiste à segmenter la composante selon le chemin emprunté par une goutte d'eau qui coulerait selon les contours de la composante. Lorsque la goutte est boquée au fond d'une vallée, celle-ci coupe la composante et continue sa chute. Cet algorithme permet de générer quatre chemins de coupures, suivant que la goutte descende ou qu'elle monte, et suivant la direction prioritaire (gauche ou droite) qu'on lui impose lorsqu'elle rencontre un extrema (mont ou vallée). Ces quatre variantes fournissent généralement des chemins différents contenant au moins une bonne segmentation (voir figure 5).

3.2.2 Sélection du meilleur chemin de segmentation

Nous pouvons donc générer quatre chemins de coupures selon les variantes du drop fall présentées précédemment. Il s'agit dans ce module de sélectionner le "meilleur" chemin parmi les hypothèses générées, décision pour laquelle il nous faut définir un critère fiable traduisant la qualité de la segmentation. Nous proposons de soumettre chaque paire de composantes segmentées à notre classifieur chiffre, et d'utiliser comme critère le produit des scores de confiance associés aux propositions du classifieur pour les deux chiffres. En effet, si les chiffres liés sont bien segmentés, les confiances associées aux deux premières propositions seront élevées ; dans le cas contraire, les hypothèses de classification chiffre devraient voir leur score chuter. La figure 5 présente la segmentation et la reconnaissance d'une composante "double digit" selon les quatre variantes du "drop fall" ; ici le drop fall ascendant gauche maximise le produit des confiances, cette hypothèse est donc conservée.

Le taux de reconnaissance des chiffres liés est de 90% en première proposition sur une base étiquetée d'environ 150 "Double Digit".

La reconnaissance de chiffres liés est évaluée sur une base étiquetée d'environ 150 "double digit" extraits de séquences numériques. Une composante est comptabilisée comme bien re-





Drop fall	ascendant, gauche	ascendant, droit	descendant, gauche	descendant, droit
Chemin de coupure				
	0[98] 8[82]	2[27] 8[35]	0[73] 8[36]	0[92] 8[34]
produit des confiances	81	09	26	32

FIG. 5 – Exemple de segmentation d’un chiffre lié selon les quatre variantes du drop fall, et reconnaissance par le classifieur chiffre. Le chemin de coupure généré par le drop fall ascendant gauche produit des confiances maximum ; nous conservons donc cette hypothèse de segmentation.

connue si les deux chiffres qui la constituent sont bien classifiés. Le taux de reconnaissance obtenu sur cette base est de 90%.

3.3 Résultats de la reconnaissance des champs isolés

Pour évaluer la reconnaissance des champs numériques, nous avons constitué une base d’environ 500 champs isolés disposant de l’étiquetage “syntaxique” (Digit, Séparateur, Double Digit), et annoté au niveau chiffre. La base provient de courriers entrants manuscrits réels, et les trois types de champ recherchés sont représentés (codes postaux, numéros de téléphone et codes clients). Nous ne comptabilisons comme bien reconnus que les champs dont toutes les composantes ont été bien reconnues au niveau chiffre. Le taux de reconnaissance au niveau champ est de 80%.

4 Performances du système

4.1 Rappel-précision du système

L’évaluation d’un système d’extraction d’information se fait classiquement par la mesure du rappel et de la précision du système. Ces deux critères sont définis de la manière suivante :

$$\text{rappel} = \text{nombre de champs bien reconnus} / \text{nombre de champs à reconnaître}$$

Le rappel traduit donc la capacité du système à localiser et reconnaître correctement tous les champs numériques d’un document.

$$\text{précision} = \text{nombre de champs bien reconnus} / \text{nombre de champs proposés par le système}$$

La précision indique la pertinence des résultats, c’est-à-dire la capacité du système à ne proposer que des champs d’intérêt et à limiter les fausses alarmes. En effet, notre système a tendance à proposer à l’issue de la première étape (localisation sans reconnaissance) des séquences de composantes qui ne sont pas des champs. Ces “fausses alarmes” ont plusieurs origines : il peut s’agir de séquences textuelles (détection d’un champ dans une zone de texte en présence notamment de caractères bâtons) ; numériques et textuelles (défaut d’alignement) ; ou même strictement numérique (détection d’un champ dans un autre, défaut d’alignement, ou erreur lors de l’étape de reconnaissance chiffre sur un champ bien localisé).

Nous présentons sur la table 3 le compromis rappel-précision de notre système à l’issue de la reconnaissance des hypothèses de localisation, en fonction du rang considéré.

TAB. 3 – *compromis rappel-précision du système.*

	TOP1	TOP2	TOP3	TOP4	TOP5
rappel	0,541	0,603	0,624	0,626	0,63
precision	0,054	0,031	0,022	0,017	0,014

On constate que le système est capable de reconnaître de 54 à 63% des codes postaux, codes clients et numéros de téléphone des documents. La précision du système est en revanche relativement faible. Nous proposons donc un module de vérification permettant d’accepter ou de rejeter les séquences de composantes reconnues.

4.2 Vérification des hypothèses de reconnaissance

Le but de cette étape de vérification est d’analyser les hypothèses de champs de manière à rejeter les fausses alarmes et à accepter les séquences numériques qui étaient effectivement à détecter. Ce module est basé sur l’interprétation d’un certain nombre d’informations obtenues tout au long de la chaîne de traitement, permettant d’accepter ou de rejeter ces hypothèses. L’étape de localisation fournit des scores d’alignement des séquences de composantes sur les modèles markoviens traduisant la qualité de l’alignement, l’étape de reconnaissance fournit des scores de confiance permettant de déceler les éventuelles composantes non numériques. Ces scores, auxquels nous avons rajouté des informations sur la régularité des boîtes englobantes des composantes, constituent les caractéristiques d’un vecteur soumis à un classifieur de type MLP, entraîné sur une base de champs numériques et de fausses alarmes. L’unique sortie du classifieur se prononce sur l’acceptation (sortie du MLP $> 0,5$) ou le rejet (sortie $< 0,5$) de l’hypothèse de champ. Le MLP a été entraîné sur une base de 17000 séquences de composantes (16800 fausses alarmes et 200 véritables champs).

Nous décrivons maintenant le vecteur de 14 caractéristiques provenant des trois familles : caractéristiques issues de la localisation, de la reconnaissance, et des boîtes englobantes des composantes.

4.2.1 Caractéristiques provenant de la localisation

Lors de l’étape de localisation, l’analyseur syntaxique fournit pour chaque ligne un score d’alignement des composantes sur les modèles (voir figure 6). Ce score est une indication précieuse sur la fiabilité de la localisation du champ et doit donc être retenu comme caractéristique dans notre vecteur. Lorsque le champ n’est pas proposé en première solution par l’analyseur syntaxique, nous remarquons que l’écart entre les scores est généralement faible avec les premiers alignements. Nous avons donc retenu comme caractéristiques les écarts entre le score de l’alignement du champ et les scores des autres alignements de la même ligne. L’expérience montre que la bonne proposition n’est jamais au delà de la cinquième proposition de l’analyseur. Nous avons ainsi retenu 6 caractéristiques issues de l’étape de localisation.

		20	200	Saint	Renon				
RANG1	D-D	D-D-D	-	R	R-R	-	R		[-0.49]
RANG2	R-D	D-D-D	-	D	R-R	-	R		[-0.56]
RANG3	R-D	D-DD	-	R	R-R	-	R		[-0.59]
RANG4	D-D	D-DD-R	-	R	R-R	-	R		[-0.59]
RANG5	R-DD	D-D-D	-	R	R-R	-	R		[-0.62]

FIG. 6 – Les cinq premiers alignements proposés par l’analyseur syntaxique pour une ligne de texte contenant un code postal, avec les scores des alignements (scores logarithmiques).

4.2.2 Caractéristiques provenant de la reconnaissance

Une autre famille de caractéristiques pour la discrimination des fausses alarmes provient de l’étape de reconnaissance. Partant de l’hypothèse selon laquelle une séquence de composantes non numériques produit des confiances basses lors de l’étape de reconnaissance (voir figure 7), nous avons choisi d’intégrer dans le vecteur les trois caractéristiques suivantes :

- Les moyennes arithmétiques et géométriques des scores de la reconnaissance chiffre
- Parmi tous les chiffres du champ, le score minimum de la reconnaissance chiffre

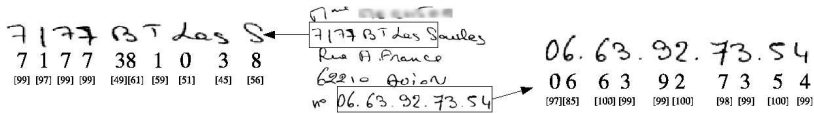


FIG. 7 – En-tête d’un document dans lequel l’analyseur a détecté deux numéros de téléphones. La reconnaissance de ces champs fournit des scores de confiance en première proposition : pour le premier champ détecté (fausse alarme) certaines confiances sont faibles, alors que pour le deuxième champ (champ bien localisé et reconnu), les confiances sont très proches de 1.

4.2.3 Caractéristiques morphologiques

L’observation d’un certain nombre de champs numériques et de fausses alarmes a montré que les boîtes englobantes des chiffres constituant un champ numérique présentent généralement des régularités que ne possèdent pas les fausses alarmes (voir figure 8).



FIG. 8 – Boîtes englobantes d’une fausse alarme et d’un champ numérique. Les boîtes englobantes du champ numérique présentent généralement davantage de régularité (hauteur, largeur, position relative) que celles des fausses alarmes.

Nous avons donc ajouté dans le vecteur 5 caractéristiques traduisant la régularité dans la succession des boîtes englobantes :

- L’écart type des ordonnées minimum et maximum des chiffres

- L'écart type des hauteur et largeur des chiffres
- L'écart type entre les abscisses des centres de gravité des chiffres

Résultats à l'issue de la vérification

La figure 9 montre l'évolution du rappel et de la précision du système avant et après l'étape de vérification des hypothèses de champs numériques.

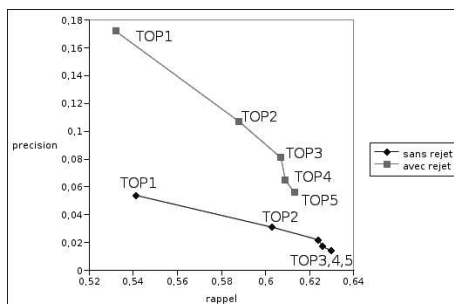


FIG. 9 – Courbe rappel/précision du système avant et après vérification des hypothèses de reconnaissance.

Nous constatons que le rejet permet d'améliorer considérablement la précision du système, pour tous les rangs considérés. Le rappel du système est peu affecté par ce rejet pour le rang 1, mais diminue de quelques points pour les rangs plus élevés.

5 Conclusion et perspectives

Dans le cadre du traitement automatique de courriers manuscrits, nous avons présenté une méthode d'extraction des champs numériques dirigée par la syntaxe, et la méthode de reconnaissance associée. L'intérêt de la méthode réside dans le fait qu'elle utilise la syntaxe d'un champ numérique comme information *a priori* pour le localiser. La reconnaissance des champs est largement contrainte par la méthode de localisation utilisée et permet de reconnaître plus de 60% des champs.

Notons que l'intégration d'un tel système en milieu industriel pourra bénéficier d'un certain nombre de connaissances *a priori* spécifiques aux types de champs recherchés (connaissances que nous n'avons pas intégrées ici dans le processus de localisation) afin d'en améliorer les performances et en particulier la précision. Par exemple, un numéro de téléphone commence toujours par un '0' ; les codes postaux se trouvent généralement dans la partie supérieure du document ; etc. Un module de mise en concurrence des champs pourra également être développé pour éviter les fausses alarmes dues à l'inclusion d'un champ dans un autre.

Afin de fiabiliser notre système, la mise en œuvre de stratégies alternatives pour la localisation des champs pourra être effectuée. Il serait intéressant d'appliquer des modèles de lignes intégrant les valeurs numériques des chiffres, afin de pouvoir prendre en compte les contraintes mentionnées précédemment (numéro de téléphone commençant par "06", etc.) dès la phase de localisation. Cette méthode impose la localisation de tous les chiffres dans le document : chiffres isolés et chiffres liés. Contrairement à la méthode présentée dans cet article, une

phase de segmentation des composantes est donc nécessaire. La clé du problème réside dans le contournement des stratégies classiques de sur-segmentation systématique des composantes qui entraînent une combinatoire très importante et donc des temps de traitement conséquents. Nous travaillons actuellement sur ce sujet.

Les deux stratégies pourront ainsi être mise en concurrence afin de fiabiliser les hypothèses de localisation et de reconnaissance des champs.

Références

- Casey, R. et E. Lecolinet (1996). A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(7), 690–706.
- Chatelain, C., L. Heutte, et T. Paquet (2004). A syntax-directed method for numerical field extraction using classifier combination. *9th International Workshop on Frontiers in Handwriting Recognition, Tokyo, Japan*, 93–98.
- Congedo, G., G. Dimauro, S. Impedovo, et G. Pirlo (1995). Segmentation of numeric strings. *ICDAR'95 2*, 1038–1041.
- Forney, G. (1973). The Viterbi algorithm. *Proc. IEEE* 61, 268–278.
- Heutte, L., T. Paquet, J. Moreau, Y. Lecourtier, et C. Olivier (1998). A structural/statistical feature based vector for handwritten character recognition. *Pattern Recognition Letters* 19, 629–641.
- Jain, A., R. Duin, et J. Mao (2000). Statistical pattern recognition : A review. *IEEE Trans. on PAMI* 22(1), 4–37.
- Kimura, F., S. Tsuruoka, Y. Miyake, et M. Shridhar (1994). A lexicon directed algorithm for recognition of unconstrained handwritten words. *IEICE Trans. on Information & Syst. E77-D(7)*, 785–793.
- Koch, G., L. Heutte, et T. Paquet (2004). Extraction de séquences numériques dans des courriers manuscrits. *RFIA'2004 3*, 1503–1511.
- Likforman-Sulem, L. et C. Faure (1995). Une méthode de résolution des conflits d'alignements pour la segmentation des documents manuscrits. *Traitement du signal* 12, 541–549.
- Plamondon, R. et S. Srihari (2000). On-line and off-line handwriting recognition : A comprehensive survey. *IEEE Trans. on PAMI* 22(1), 63–84.
- Trier, O., A. Jain, et T. Taxt (1996). Feature extraction methods for character recognition : A survey. *Pattern Recognition* 29(4), 641–662.

Summary

In this paper, we describe a complete and generic processing flow for the automatic extraction of numerical fields from unconstrained handwritten documents. The processing flow is made of three stages : 1) numerical field localization by means of a markovian-based method without segmentation nor numeral recognition, 2) numeral recognition of the extracted sequences, 3) verification of the hypotheses in order to limit the false alarm. The evaluation of our system on 300 handwritten mail documents gives encouraging recall-precision results.