Classification de formulaires manuscrits en-ligne à l'aide de réseaux bayésiens

Emilie Philippot, Yolande Belaïd, Abdel Belaïd

Classification de formulaires manuscrits en-ligne à l'aide de réseaux bayésiens

Emilie Philippot — Yolande Belaïd — Abdel Belaïd

Université Nancy 2
LORIA - Campus scientifique - BP 239 - 54506 Vandoeuvre-lès-Nancy Cedex - France
{emilie.philippot, yolande.belaid,abdel.belaid}@loria.fr

RÉSUMÉ. *Dans cet article, nous proposons une méthode de reconnaissance de formulaires manuscrits en-ligne à l'aide d'un stylo électronique de type clip. Ce mode d'écriture non contraint entraîne un remplissage incertain du formulaire et la perte du type du formulaire rempli. Nous proposons une méthode permettant d'identifier la classe du formulaire en se fondant sur le positionnement des champs remplis. Notre démarche repose sur l'emploi de réseaux bayésiens permettant d'exploiter, à l'aide des probabilités conditionnelles, les occurrences de remplissage des champs les uns en fonction des autres. Les tests ont été effectués sur une base de 3200 formulaires fournis par la société Actimage. Les premières expériences montrent un taux de reconnaissance de 90%.*

ABSTRACT. *In this paper, we propose a method of recognizing handwritten forms online using an electronic pen-type clip. This mode of writing leads unconstrained filling uncertain form and the loss of the type of the completed form. We propose a method to identify the class of the form based on the positioning of fields filled. Our approach relies on the use of Bayesian networks to exploit, using conditional probabilities, occurrences of filling some fields depending on others. The tests were conducted on the basis of 3200 forms provided by the company Actimage. The first experiments show an average recognition of 90%.*

MOTS-CLÉS : *Formulaires, écriture manuscrite en-ligne, classification, réseaux bayésiens*

KEYWORDS: *forms classification, on-line handwritting, classification, bayesian networks*

1. Introduction

Le travail présenté dans cet article concerne la classification de formulaires remplis manuellement à l'aide d'un stylo numérique et est réalisé en collaboration avec la société Actimage. La figure 1 montre un exemple de la problématique. Dans un premier temps, le formulaire vierge est rempli à l'aide d'un système de prise de notes manuscrites en-ligne. Seules ces notes sont transmises au système, on perd donc la trame du formulaire. A partir des informations transmises, l'objectif est de déterminer la classe du formulaire rempli.

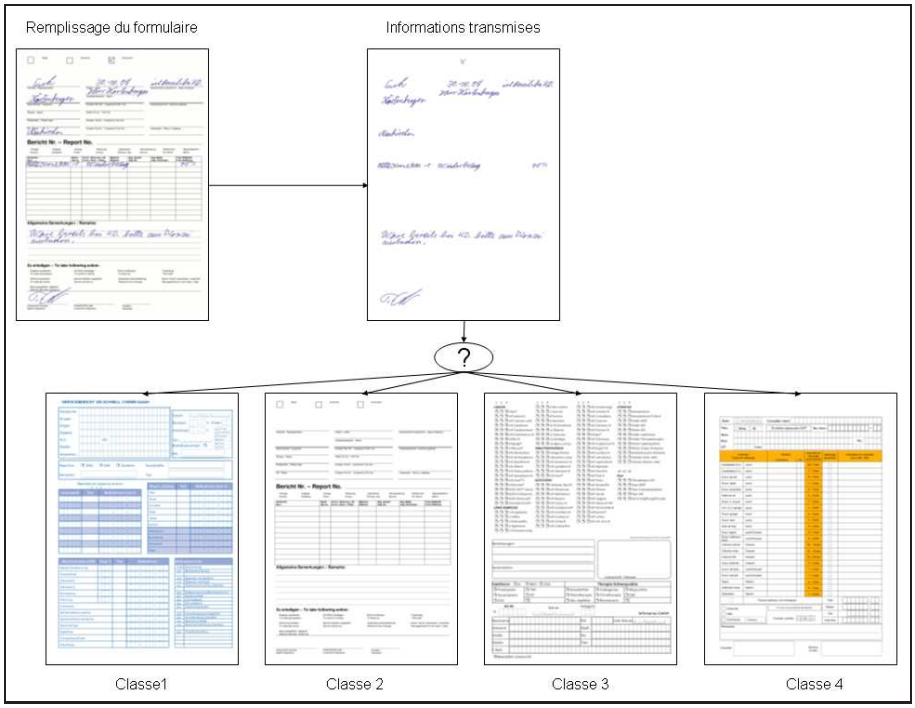

Figure 1. *Problématique*

La littérature montre, pour la classification des formulaires, une majorité de travaux portant sur les formulaires hors-ligne. Le contexte est donc présent dans l'image du document. Nous avons limité nos recherches aux formulaires pré-imprimés remplis manuellement.

Dans (Ramdane *et al.*, 1997), les auteurs segmentent les formulaires en blocs afin de définir leur structure physique, puis apprennent une distribution de probabilités d'occurrences de ces blocs. Ensuite, la reconnaissance s'appuie sur cette distribution couplée à un calcul de distance afin d'obtenir la classe d'un formulaire inconnu. Dans (Ramdane *et al.*, 2001), les auteurs utilisent également la structure physique du formulaire par segmentation en bloc mais l'étape de classification se fait en utilisant des

modèles de Markov cachés planaires. Ces approches sont intéressantes dans la mesure où les formulaires sont séparés en blocs ce qui permet une reconnaissance hiérarchique des formulaires.

(Watanabe *et al.*, 1995) proposent une autre méthode de classification de formulaires basée également sur l'extraction de la structure mais en utilisant des arbres de décision pour la classification. Comme dans (Ramdane *et al.*, 1997) et (Ramdane *et al.*, 2001), cet article propose une approche hiérarchique de la reconnaissance en créant des arbres de structure locale puis des arbres de structure globale des formulaires.

Dans (Neschen, 1996), Neschen propose un système de lecture automatique de formulaires bancaires allemands. Son approche repose sur la segmentation du formulaire puis sur l'utilisation d'un classifieur (l'algorithme des plus proches voisins) et enfin sur une unité de correction. L'intérêt repose sur l'unité de correction qui permet d'améliorer la reconnaissance.

Dans (Doermann *et al.*, 1993), les auteurs utilisent des modèles de formulaires et en extraient automatiquement les principales caractéristiques (orientation, position et taille). Chaque formulaire rempli est ensuite apparié au modèle par un sondage des tracés sur les champs de ce dernier. Cette méthode a pour objectif d'extraire le texte manuscrit de la trame du formulaire et d'en corriger les anomalies d'extraction.

Aucune de ces approches ne peut être appliquée à notre problématique dans la mesure où nous ne disposons pas de la trame du formulaire. Nous pouvons cependant envisager d'utiliser la segmentation par bloc et le sondage des champs. Ces deux techniques permettent en effet d'obtenir les champs remplis d'un formulaire et de simplifier le formulaire en le divisant en plusieurs zones, des zones d'intérêts dans notre cas.

La suite de l'article sera organisée de la maniére suivante : dans la section 2, nous présentons les outils de prise de notes en-ligne. La section 3 décrit l'approche proposée. Dans la section 4, nous effectuons quelques rappels concernant les réseaux bayésiens, puis, dans la section 5, nous détaillons la méthode de classification de formulaires manuscrits en-ligne. Enfin, dans la section 6, nous présentons les premiers résultats obtenus avant de conclure et de proposer quelques perspectives.

2. Outils de prise de notes en-ligne

Les outils de prise de notes en-ligne sont de plus en plus nombreux et de plus en plus hétérogènes. Afin de permettre une grande mobilité, les dispositifs portables sont en constante évolution. Il existe deux types de dispositifs : les systèmes tout électronique comme les tablets PC ou les netbooks à écran tactile et les systèmes basés sur un support papier.

Les systèmes tout électronique offrent une grande fiabilité de traitement dans la mesure où le support permet l'ajout de nombreuses informations ainsi que la localisation précise des tracés. De plus, le traitement de l'information est instantané et

permet une interaction directe avec l'utilisateur. Bien que sûres, ces solutions restent onéreuses et parfois fragiles. Leur portée reste donc limitée.

Les systèmes basés sur un support papier comme les stylos numériques ou le papier interactif permettent une bonne alternative en offrant un support d'écriture en-ligne solide tout en limitant l'investissement et les coûts d'utilisation. Le tramage du papier interactif permet l'ajout d'information parfois nécessaire au traitement de la prise de notes et une localisation précise des tracés. Les stylos numériques utilisent du papier quelconque. Les contraintes et les coûts d'utilisation sont donc plus faibles mais il est nécessaire de mettre en place une série d'applications afin de palier les pertes d'informations liées au support.

La société Actimage souhaite mettre en place un système de classification de formulaires manuscrits en-ligne saisis avec un stylo numérique. Ce système devra permettre à un client de remplir un formulaire imprimé sur une feuille quelconque en lui assurant la reconnaissance de la classe du formulaire rempli afin qu'il puisse appliquer le traitement adéquat.

2.1. *Technologie utilisée : le stylo numérique*

L'outil d'écriture en-ligne utilisé dans cette application est un clip muni d'un stylo. Le stylo contient un émetteur d'ultrason et un rayon infrarouge. Le clip est équipé de deux récepteurs audio et d'un récepteur infrarouge. Lorsque le crayon entre en contact avec le papier, il émet un signal sonore inaudible et transmet l'instant de départ et d'arrivée des signaux audio par infrarouge. Le clip réceptionne les informations et détermine les déplacements du stylo sur la feuille grâce à une méthode de triangulation (figure 2).

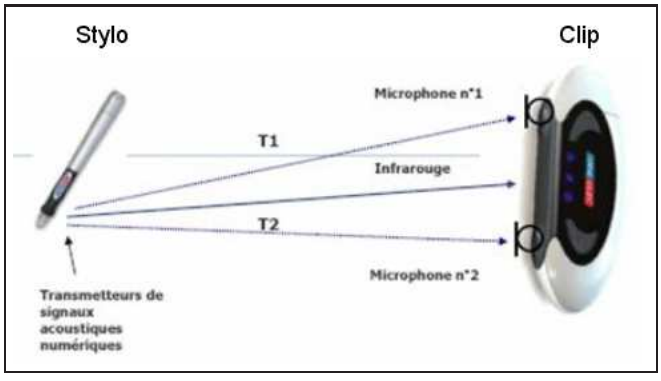

Figure 2. *Exemple de stylo numérique avec son capteur.(Image Zpen)*

Tous les tracés sont enregistrés dans le clip qui contient une mémoire d'un giga octet. Une fois la prise de notes terminée, il suffit de brancher le clip sur un port USB

pour en extraire les fichiers qui contiennent les notes de l'utilisateur. Ces fichiers sont appelés fichiers d'encre électronique.

L'avantage de cette technologie est qu'elle utilise un support papier classique. Le coût d'utilisation est donc limité. Par contre, il est impossible de transmettre de l'information concernant la classe du formulaire rempli grâce au papier ou au crayon. Il est donc nécessaire de s'appuyer uniquement sur les fichiers d'encre électronique récupérés pour analyser et traiter la prise de notes. Ceci explique la difficulté et l'originalité du problème à résoudre.

La figure 3 donne un exemple de formulaire rempli. A gauche, il s'agit du formulaire rempli vu par l'utilisateur, la trame est présente et le remplissage se fait de manière intuitive. A droite, nous avons le fichier d'encre électronique retourné par le système. Nous avons donc perdu toutes les informations concernant la classe du formulaire rempli.

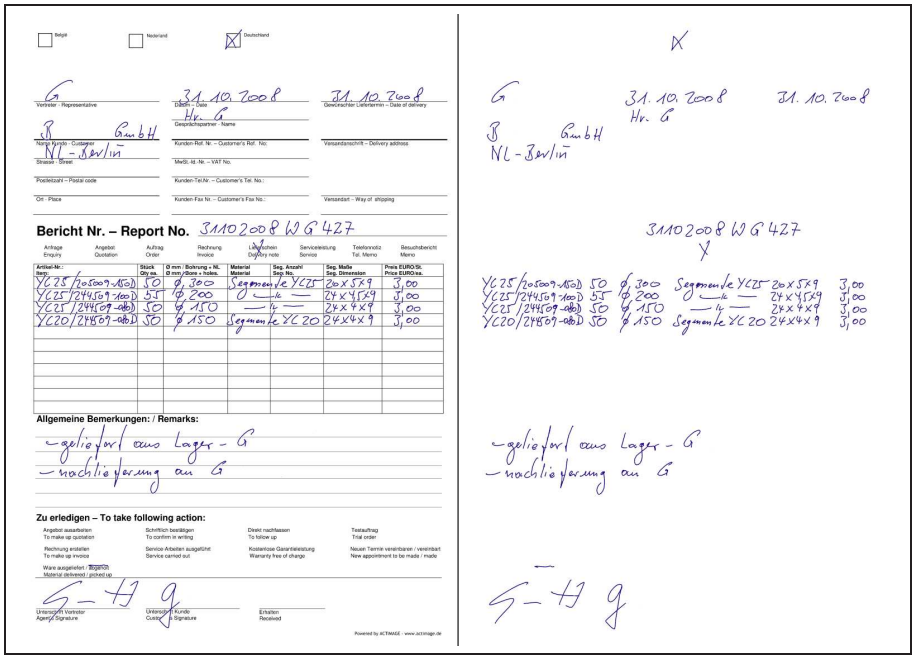

Figure 3. *Exemple de formulaire à traiter : à gauche le formulaire rempli et à droite le fichier d'encre électronique retourné*

La figure 4 montre un exemple de codage de tracés dans un fichier d'encre électronique. Au cours de l'écriture, les coordonnées x et y sont échantillonnées à intervalles de temps réguliers puis regroupées en tracés en fonction des posés et levés de crayon. On obtient ainsi une liste de vecteurs représentant les mouvements effectués. L'exemple donné correspond à la lettre "i". Le premier tracé correspond à la partie inférieure du "i" et le deuxième au point diacritique.

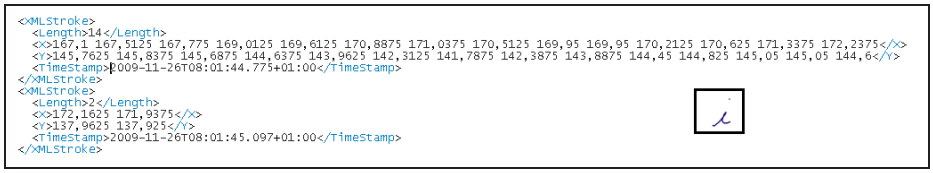

Figure 4. *Exemple de fichier d'encre électronique obtenu pour un ``i ´´*

3. Approche proposée

Il s'agit d'analyser le fichier d'encre électronique et de retrouver le modèle de formulaire correspondant. Notre approche repose en partie sur l'observation suivante : il existe une dépendance d'une part entre les champs d'un formulaire et plus précisement entre les champs d'une même zone d'un formulaire et d'autre part entre un formulaire et ses zones. Par exemple : les cases Mme, M. et Melle d'un formulaire ne sont jamais cochées simultanément ; la présence d'un numéro d'identification client implique l'absence de remplissage des champs concernant les coordonnées de ce dernier.

La figure 5 montre un exemple des dépendances pouvant exister pour la zone d'adresse d'un formulaire. Les liens et les probabilités permettent de localiser et de quantifier les dépendances qui existent entre les champs eux-mêmes et entre les champs et la classe. Par exemple, le champ *ville* est dépendant du champ *code postal*. Ainsi si *code postal* est rempli, on aura une probabilité de 0,7 pour que *ville* soit également rempli. Inversement si *code postal* est vide alors *ville* le sera aussi avec une probabilité de 0,8.

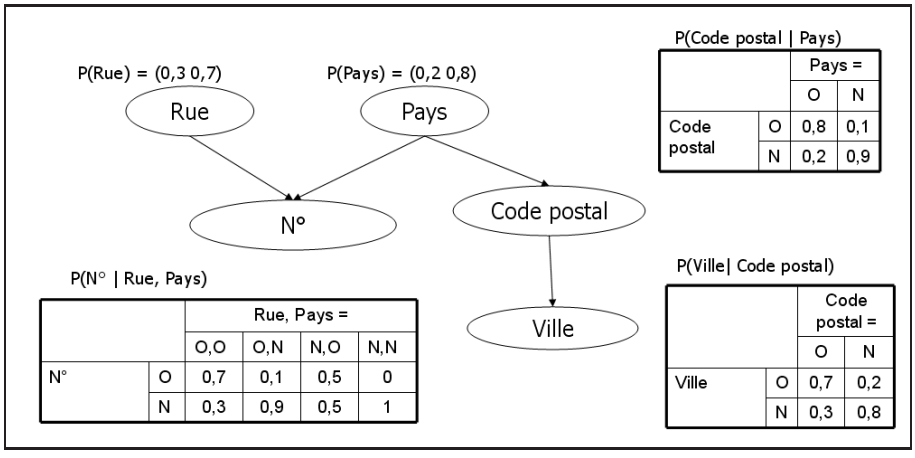

Figure 5. *Exemple de réseau bayésien pour la zone d'adresse d'un formulaire*

Par ailleurs, compte tenu de la technologie utilisée, les données récupérées peuvent être incertaines car leur interprétation est fonction du contexte qui n'est plus disponible et des notes peuvent avoir été ajoutées en dehors des champs prédéfinis. Les données peuvent aussi être incomplètes car certains champs peuvent être omis.

À partir de ces constatations, nous avons opté pour l'utilisation des réseaux bayésiens car ils permettent une description qualitative et quantitative des dépendances et la gestion de l'incertain.

Afin de permettre une classification par étape du formulaire (complexe dans sa globalité), nous développons des réseaux bayésiens par zones d'intérêt. La figure 6 montre un exemple de découpage d'un formulaire en 3 zones d'intérêts : la zone supérieure correspond à l'identification du client, la zone du milieu à la commande et la zone du bas à la validation de la commande. Cette division en plusieurs réseaux offre des avantages :

– l'apprentissage de la structure des réseaux est facilité par le nombre réduit de variables,

– un même réseau bayésien va pouvoir représenter des zones de différents formulaires,

– lors de la modification d'un formulaire, il ne sera pas nécessaire de réapprendre tout le réseau mais juste celui de la zone concernée par les modifications.

Tous les réseaux bayésiens des zones de toutes les classes sont ensuite regroupés dans un seul réseau bayésien global qui permet la classification des formulaires.

Cette approche, originale pour le type d'application traité, est détaillée dans la suite de l'article et validée par les résultats expérimentaux obtenus.

4. Réseaux bayésiens

Un réseau bayésien est un graphe acyclique dirigé (Naïm *et al.*, 2007). Les noeuds représentent un ensemble de variables aléatoires et les arcs les dépendances conditionnelles entre les noeuds.

La construction d'un réseau bayésien se fait en deux étapes : la définition de la structure du graphe et la définition des tables de probabilités des variables aléatoires.

4.1. *Apprentissage de la structure*

Les algorithmes d'apprentissage de la structure d'un réseau bayésien peuvent être regroupés en trois grandes familles : par recherche de causalité tel que l'algorithme PC, par recherche du meilleur score et par parcours des structures.

Nous avons choisi d'utiliser l'algorithme PC (Spirtes *et al.*, 2000) qui repose sur l'indépendance conditionnelle des variables. Le point de départ de cet algorithme est

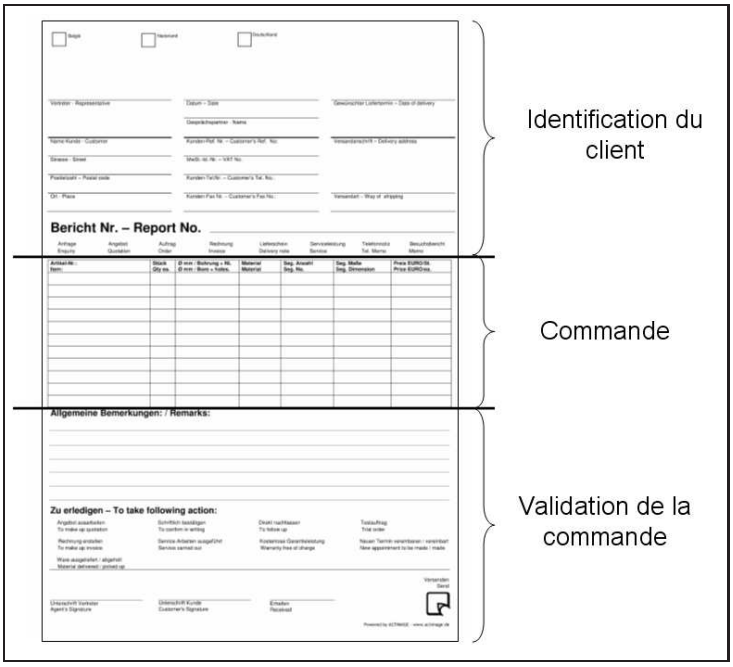

Figure 6. *Exemple de découpage d'un formulaire en zones.*

un graphe complètement connecté. Ensuite, pour chaque couple de variables aléatoires connectées par un arc, on teste s'il y a une indépendance conditionelle et si oui, on supprime l'arc correspondant.

Dans notre cas, cet algorithme a l'avantage de mettre en avant les dépendances entre les différents champs des formulaires et aussi entre les différentes zones des formulaires. Ce qui pourra par la suite nous permettre d'analyser les formulaires et les résultats afin d'en modifier l'aspect.

4.2. *Apprentissage des paramètres*

Dans notre application, toutes les variables du réseau, excepté la classe, sont observées . En effet, la variable associée a un champ a toujours une valeur que ce champ soit rempli ou non. Nous avons donc pu opter pour l'utilisation du maximum de vraisemblance pour l'apprentissage des paramètres. Il s'agit simplement de calculer la fréquence d'apparition d'un événement pour déterminer sa probabilité, ce qui donne le calcul suivant :

$$\hat{p}(X_i = x_k | pa(X_i) = x_j) = \frac{N_{i,j,k}}{\sum_k N_{i,j,k}} \quad [1]$$

où $N_{i,j,k}$ est le nombre d'événements dans la base de données pour lesquels la variable aléatoire X_i est dans l'état x_k et ses parents sont dans la configuration x_j.

4.3. *Reconnaissance*

La reconnaissance repose sur l'inférence qui consiste à propager les informations connues au reste du réseau bayésien afin de modifier les probabilités des variables aléatoires qui n'ont pas été observées. Dans notre cas, il s'agit de déterminer la probabilité d'appartenance à une classe d'un formulaire en se basant sur le remplissage des champs du dit formulaire. Dans un premier temps, on transforme la structure du réseau bayésien en un arbre en utilisant l'algorithme de l'arbre de jonction. Puis on utilise le "message passing" pour propager l'information au sein de l'arbre.

5. Classification de formulaires à l'aide de réseaux bayésiens

La figure 7 donne un aperçu de l'approche proposée. L'idée est de définir pour chaque classe de formulaire, un formulaire modèle qui servira de base à tout le système aussi bien pour l'apprentissage que pour la reconnaissance. Ce modèle servira à extraire les champs des formulaires afin de les utiliser pour la création des réseaux bayésiens au moment de l'apprentissage et pour la sélection des noeuds dans ces réseaux pour la reconnaissance. Nous allons détailler dans la suite les parties importantes de ce schéma.

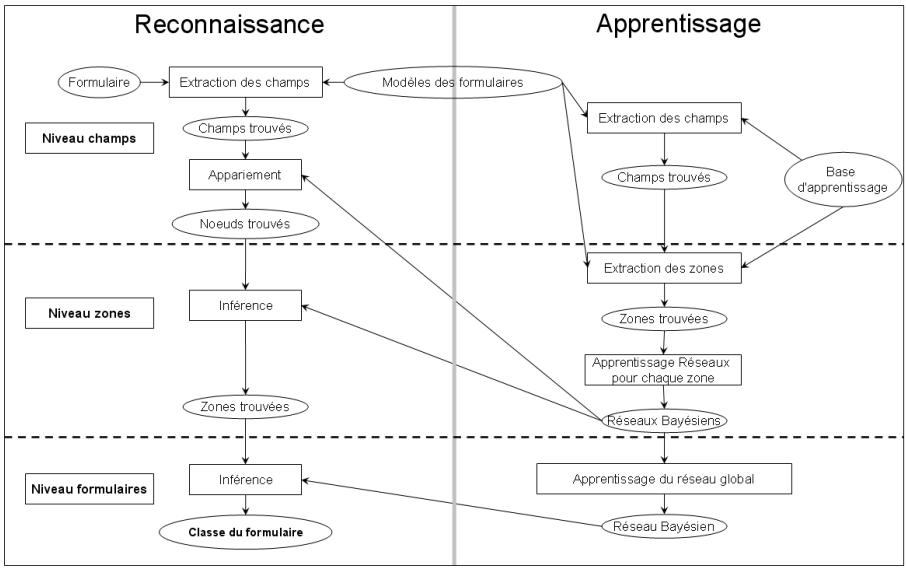

Figure 7. *Schéma global de l'approche*

5.1. *Formulaire modèle*

Chaque classe de formulaires est représentée par un formulaire modèle. Ce modèle créé manuellement est le squelette du formulaire de la classe. Il contient des informations sur les champs et sur la division en zones. Ces zones ont été définies manuellement.

Un formulaire modèle est un fichier XML contenant la liste des champs du formulaire en suivant l'ordre classique de lecture, de gauche à droite puis de haut en bas. Chaque champ est défini par les coordonnées de sa boîte englobante, son type (case à cocher, chaîne de caractères, nombre, ...) et la zone à laquelle il appartient. Le type est utile pour permettre la validation de la classification dans le cas où le résultat ne serait pas concluant. La figure 8 montre un extrait de formulaire modèle et des champs correspondants dans la trame.

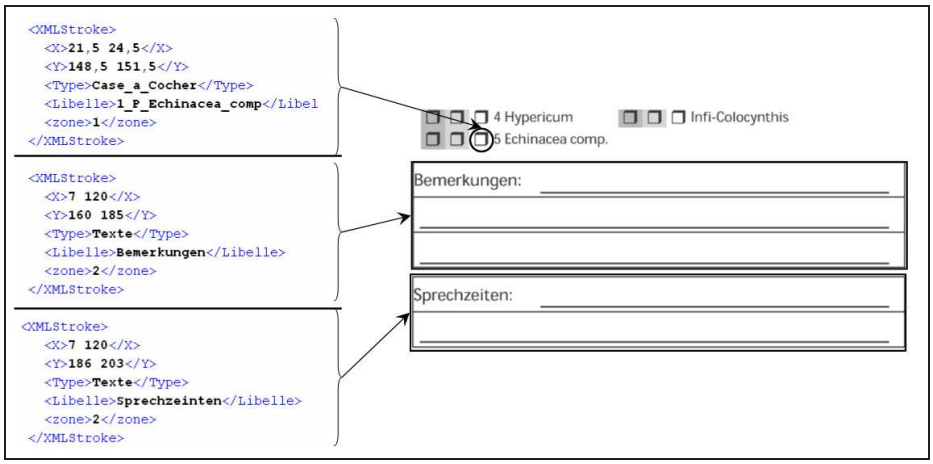

Figure 8. *Extrait du contenu d'un formulaire modèle et lien avec la trame correspondante*

Ce modèle permet d'extraire les champs des formulaires remplis à partir des fichiers d'encre électronique, de définir les différentes zones d'un formulaire et de valider la classe obtenue en s'assurant de la cohérence des types en fonction des tracés attribués à chaque champ.

5.2. *Extraction des champs*

L'extraction des champs est la première étape à la fois de l'apprentissage et de la reconnaissance. Elle a pour résultat la liste des champs trouvés. Actuellement, l'extraction des champs se fait en superposant les formulaires modèles au fichier d'encre électronique. Il s'agit d'apparier les tracés aux champs des formulaires modèles.

Chaque tracé est décomposé en autant de vecteurs qu'il a de coordonnées. On associe ensuite chaque vecteur du tracé au champ auquel il appartient par comparaison des coordonnées de ce vecteur avec ceux des boîtes englobantes des champs du modèle. Si un champ regroupe plus de 85% des vecteurs d'un tracé, on considère alors que ce tracé appartient à ce champ et donc que ce champ est présent dans le formulaire rempli. Ce pourcentage a été défini et validé expérimentalement pour autoriser un tracé à dépasser légèrement d'un champ.

Une fois tous les tracés traités, on dispose de la liste des tracés par champ et de la liste des tracés qui n'ont pas pu être appariés, soit parce qu'ils ne correspondent à aucun champ du formulaire, soit parce que leur taux d'appartenance à un champ est trop faible. Si le taux de tracés non appariés est élevé alors soit le formulaire rempli ne correspond pas au formulaire modèle, soit le formulaire rempli a été mal orienté par rapport au clip. Dans ce cas, il sera nécessaire de détecter l'orientation pour la corriger. Actuellement, nous travaillons sur des formulaires correctement orientés.

La figure 9 montre un exemple de différentes possibilités d'interprétation des tracés en fonction du contexte. Dans le cas (1), le texte est associé à deux cases à cocher et à un champ texte. Dans le cas (2), les tracés sont associés à deux champs texte différents. Le cas (3), qui correspond à la réalité, montre que le tracé dépasse du champ trop petit. La difficulté va donc résider dans le fait de trouver le formulaire initialement rempli même si les tracés ne correspondent pas entièrement aux champs de ce dernier.

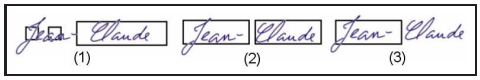

Figure 9. *Exemple d'interprétations de tracés en fonction du contexte*

La figure 10 est un exemple d'appariement entre un fichier d'encre électronique et le formulaire modèle correspondant. Les champs remplis sont ceux dans lesquels apparait un tracé. Le tracé entouré d'un cercle est un tracé pour lequel nous n'avons pas trouvé de champ correspondant. L'appariement reste néanmoins valide dans la mesure où il y a des tracés dans tous les champs remplis par l'utilisateur.

5.3. *Apprentissage*

Nous travaillons sur une base d'exemples de fichiers d'encre électronique étiquetés et normalisés. Les formulaires sont correctement orientés. Nous sommes donc sûrs des coordonnées des tracés que nous utilisons. L'apprentissage se fait en deux temps. Dans un premier temps, on apprend les réseaux bayésiens des zones (section 5.3.1) identifiées dans les formulaires puis on apprend le réseau bayésien global (section 5.3.2) permettant la classification des formulaires.

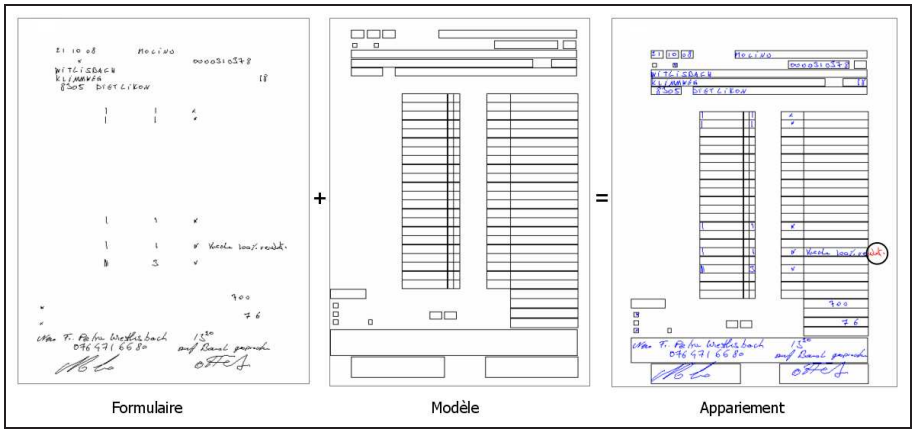

Figure 10. *Exemple d'appariement entre un formulaire et son modèle*

5.3.1. *Apprentissage des zones*

Dans un premier temps, on définit manuellement les différentes zones du formulaire à apprendre à partir des formulaires modèles. Puis, on associe à chaque zone la liste des champs extraits lui appartenant. Chaque champ correspond à un noeud du réseau bayésien de sa zone de formulaire. A partir de la liste des noeuds d'une zone, on construit un graphe complètement connecté qui servira de base pour l'apprentissage de la structure du réseau bayésien de cette zone.

Cet apprentissage se fait en utilisant l'algorithme PC (voir 4.1). L'apprentissage des probabilités utilise l'algorithme du maximum de vraisemblance (voir 4.2).

La figure 11 montre un exemple de la structure d'un réseau bayésien et de l'entête du formulaire correspondant. Il met en avant les dépendances existantes entre les champs de cette zone. On peut par exemple observer une corrélation entre les champs d'une date. La présence ou l'absence des champs *jour*, *mois* et *année* sont liées. L'apprentissage permet également de mettre en avant des dépendances moins intuitives entre certains champs. C'est notamment le cas entre les champs *Code Postal* et *Conseiller client*.

5.3.2. *Apprentissage du réseau global*

Le réseau global regroupe toutes les zones de toutes les classes et permet de mettre en relation les différentes zones des formulaires et leurs classes pour tous les modèles.

Dans un premier temps, on utilise les réseaux bayésiens obtenus précédemment afin de définir pour chaque zone une distribution de probabilités. Chaque distribution sert de noeud au réseau global. On applique ensuite l'algorithme PC afin de déterminer la structure du réseau global. La figure 12 montre un exemple de réseau global obtenu

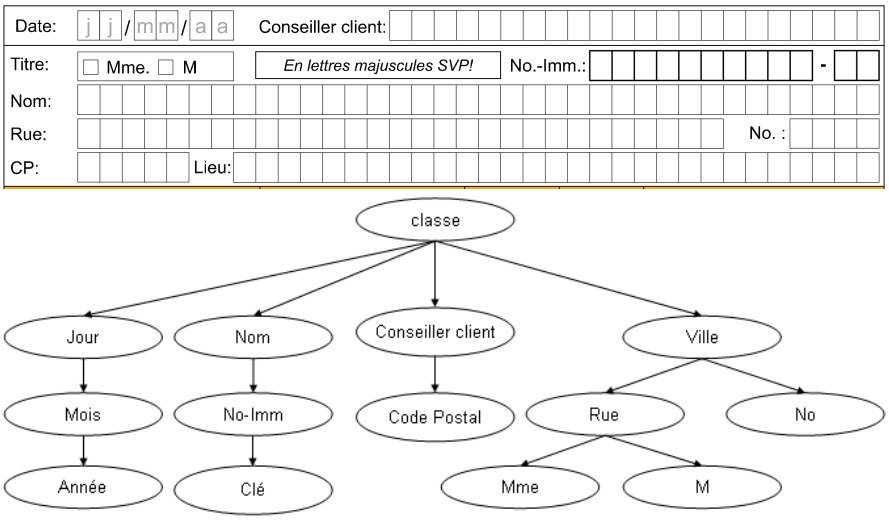

Figure 11. *Exemple de zone d'en-tête de formulaire et de la structure du réseau bayésien correspondant*

pour la classification de deux classes de formulaires. On observe que certaines zones peuvent faire référence aux deux formulaires.

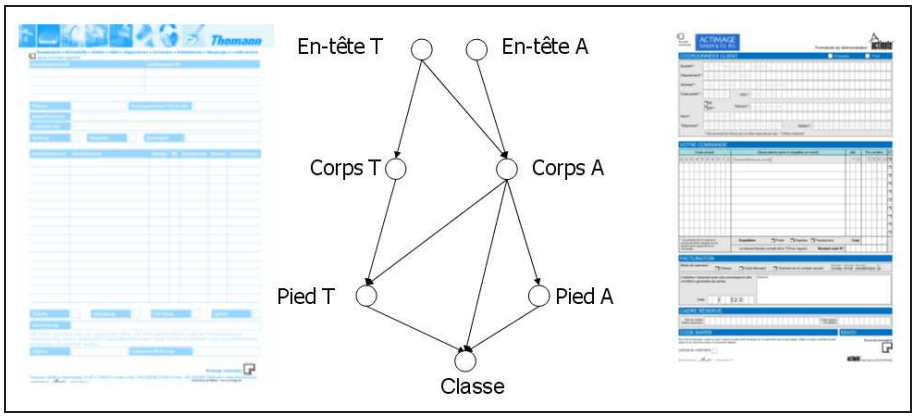

Figure 12. *Exemple de structure de réseau bayésien global permettant la classification de formulaires de classe A ou de classe T.*

5.4. *Reconnaissance*

La reconnaissance se déroule en plusieurs étapes. Tout d'abord comme lors de l'apprentissage, on extrait les champs en appariant les tracés avec les champs des formulaires modèles. Ensuite, pour chaque zone, on calcule une probabilité d'appartenance. C'est à dire la probabilité que les champs extraits appartiennent à une zone donnée en utilisant les réseaux définis pour les zones. Puis, on utilise les différentes probabilités obtenues afin de définir la classe du formulaire à l'aide du réseau global.

6. Résultats

Nous avons expérimenté notre approche sur une base fournie par la société Actimage. Elle comprend 4 classes de formulaires présentées dans la figure 1 et 800 échantillons par classe. Nous avons réalisé les tests en utilisant une méthode de validation croisée. La base de tests a été divisée en 4 parties contenant chacune 200 formulaires de chaque classe soit 800 échantillons. A partir de ces subdivisions, nous avons créé 4 bases d'apprentissages différentes composées chacune de 3 parties choisies parmi les 4 possibles, la partie non retenue a été utilisée pour la reconnaissance. Nous avons donc obtenu 4 bases d'apprentissage et de reconnaissance différentes sur lesquelles nous avons appliqué notre méthode de classification.

Les tests ont été effectués sous Matlab en utilisant la toolbox BNT (Murphy, 2001). Les résultats présentés dans les tableaux 1 et 2 sont encourageants.

Classe	Zone en-tête	Zone corps de texte	Zone pied de page	Formulaire
1	99,38	98,5	99,25	96,88
2	99,7	66,88	98,5	91,8
3	0,13	89,13	2,25	98,75
4	99,02	99,21	0.13	75,63
Moyenne	74,56	96,63	50,03	90,7

Tableau 1. *Rappel en %*

Classe	Zone en-tête	Zone corps de texte	Zone pied de page	Formulaire
1	56,68	98,5	37,62	80,2
2	83,86	66,88	74,7	99,6
3	25	89,23	0,5	95,98
4	97,66	98,88	0,13	98,77
Moyenne	65,8	88,37	28,23	93,9

Tableau 2. *Précision en %*

Le rappel et la précision sont calculés sur les 3 zones puis sur les formulaires dans leur globalité pour chaque classe puis sur l'ensemble des classes. Il semble nécessaire

d'améliorer le choix de la structure des réseaux. En effet, au cours des expérimentations, nous avons observé que la structure avait un fort impact sur la reconnaissance finale. Par exemple, pour la reconnaissance du formulaire global, la classe 4 n'a un taux de rappel moyen que de 75,63%. En fonction de la base d'apprentissage utilisée (et donc de la structure apprise), ce taux varie de 60,31% à 99,12%. Cette constation se fait également sur la reconnaissance des zones puisque pour la zone *corps de texte* de la classe 2, le taux de rappel varie entre 30,2% et 91,45%.

Le taux de précision de la classe 1 qui n'est que de 80,2% pour la reconnaissance globale s'explique par la complexité de sa structure. En effet, ses champs sont courts, nombreux et rapprochés. L'étape d'extraction des champs est donc fortement biaisée par cette particularité puisque un champ texte d'une autre classe pourra recouvrir plusieurs champs de la classe 1. Par exemple, on remarque que le taux de rappel du pied de page de la classe 4 est seulement de 0,13%. De même on observe que la précision du pied de page de la classe 1 n'est que de 37,62%. Ceci s'explique par le chevauchement des champs dans ces deux classes comme le montre la figure 13. Les champs du *pied de page* de la classe 4 sont complètement englobés par les champs du *pied de page* de la classe 1. L'appariement est donc biaisé et la reconnaissance de la partie du formulaire faussée. Malgré tout, les taux de reconnaissance des classes 1 et 4 sont bons, dans la mesure où le réseau global accepte la possibilité qu'une classe soit définie par une zone n'appartenant pas à son modèle d'origine.

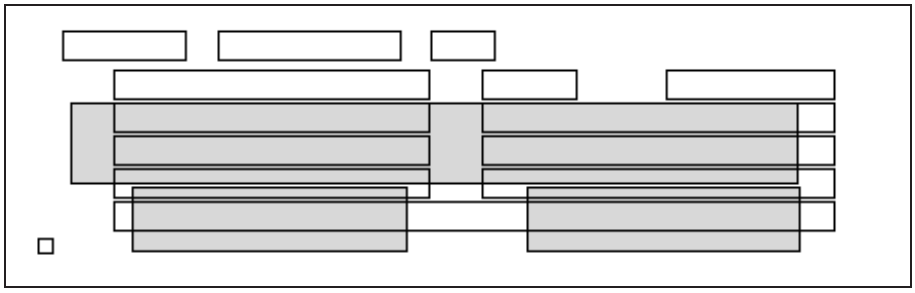

Figure 13. *Exemple de chevauchement de champs. Les champs grisés appartiennent à la classe 4 et les champs à fond blanc à la classe 1. Au moment de l'appariement, les tracés des champs seront appariés aux deux classes et donc les réseaux bayésiens des deux classes pourront renvoyer une probabilité d'appartenance acceptable.*

7. Conclusion

Nous avons mis en place et testé une première approche de la classification de formulaires manuscrits en-ligne non contraints en utilisant deux niveaux de réseaux bayésiens. Les premiers résultats sont encourageants et ouvrent la voie à de nombreuses perspectives. Dans un premier temps, il serait intéressant de valider la robustesse de notre système avec un plus grand nombre de classes, d'étudier l'apport d'autres algorithmes d'apprentissage de structure de réseaux bayésiens et de comparer notre ap-

proche avec une méthode basique d'appariement. Ensuite, nous envisageons de tester les limites du système quant à l'orientation de la feuille et en fonction des résultats modifier le système d'appariement des champs et des tracés en procédant au préalable à une étape de segmentation pour limiter les erreurs d'appariement.

Enfin, l'utilisation des réseaux bayésiens sur les formulaires pourrait être un moyen d'analyser les stratégies de remplissage de ces derniers et ainsi permettre la modification de la mise en page et la révision des contenus de formulaires afin de les adapter aux scripteurs.

Remerciements

Ce travail est réalisé dans le cadre d'une convention CIFRE. Nous tenons à remercier la société Actimage d'avoir collaboré à ces travaux et de nous avoir fourni la base d'apprentissage nécessaire.

8. Bibliographie

Doermann D., Rosenfeld A., « The processing of form documents », *Proceedings of the Second International Conference on Document Analysis and Recognition*, vol. 2, p. 497-501, 1993.

Murphy K., « The BayesNet Toolbox for Matlab », *Computing Science and Statistics : Proceedings of Interface*, vol. 33, 2001.

Naïm P., Wuillemin P., Leray P., Pourret O., Becker A., *Réseaux bayésiens*, Eyrolles, 2007.

Neschen M., « Reconnaissance de Formulaires Manuscrits Basée sur la Quantification Vectorielle Hiérarchique », *Colloque National sur l'Ecrit et le Document*, vol. 4, p. 245-250, 1996.

Ramdane S., Taconet B., Zahour A., Faure A., « Identification de formulaires par modèles de Markov cachés planaires », *Dix-huitième colloque du Groupe de Recherche et d'Etudes de Traitement du Signal et des Images*, vol. 18, p. 805-808, 2001.

Ramdane S., Taconet B., Zahour A., Kebairi S., « Apprentissage et Reconnaissance Automatique de types de Formulaires par une Mèthode Statistique », *Seizième colloque du Groupe de Recherche et d'Etudes de Traitement du Signal et des Images*, vol. 16, p. 111-114, 1997.

Spirtes P., Glymour C., Scheines R., *Causation, Prediction, and Search*, The MIT Press, 2000.

Watanabe T., Luo Q., Sugie N., « Layout Recognition of Multi-Kinds of Table-Form Documents », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, n° 4, p. 432-445, 1995.