



Vers une approche générique pour la reconnaissance de formes manuscrites structurées : Application aux équations mathématiques et aux caractères chinois

Adrien Delaye, Harold Mouchère

► To cite this version:

Adrien Delaye, Harold Mouchère. Vers une approche générique pour la reconnaissance de formes manuscrites structurées : Application aux équations mathématiques et aux caractères chinois. Colloque International Francophone sur l'Écrit et le Document (CIFED2010), Mar 2010, Sousse, Tunisie. 2010. <hal-00491000>

HAL Id: hal-00491000

<https://hal.archives-ouvertes.fr/hal-00491000>

Submitted on 10 Jun 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Vers une approche générique pour la reconnaissance de formes manuscrites structurées

Application aux équations mathématiques et aux caractères chinois

Adrien Delaye¹, Harold Mouchère²

¹IRISA, INSA de Rennes - Université Européenne de Bretagne
Campus Universitaire de Beaulieu, 35000 Rennes

²IRCCyN, équipe IVC, Université de Nantes
Rue Christian Pauc, 44306 Nantes

adrien.delaye@irisa.fr, harold.mouchere@univ-nantes.fr

RÉSUMÉ. Nous présentons ici une approche générique pour la reconnaissance des formes manuscrites structurées. Leurs caractéristiques : structure 2D complexe, topologie récursive, grand nombre de primitives et de relations spatiales, imposent la mise en œuvre de stratégies d'analyse adaptées pour limiter l'explosion combinatoire. GASPR est une approche originale consistant à guider la construction et l'exploration d'un graphe de segmentation par des connaissances a priori (grammaire du langage, relations spatiales...). La maîtrise de la complexité est assurée par un algorithme inspiré de A qui limite la construction du graphe aux meilleures hypothèses. Les expérimentations sur différents types de données en-ligne (équations mathématiques et caractères chinois) confirment l'intérêt de la méthode et sa généricité.*

ABSTRACT. GASPR is a generic method for interpreting structured handwritten patterns, having a complex recursive 2D structures, many symbols and spatial relations, and thus require efficient segmentation strategies. A graph-based approach is proposed where the construction of the segmentation graph is directed by domain-dependant knowledge. The complexity of analysis is handled by a A-inspired algorithm. Experiments on mathematical expressions and Chinese characters confirm the interest and the genericity of the method.*

MOTS-CLÉS : écriture en-ligne, formes structurées, graphe de segmentation, relations spatiales

KEYWORDS: handwriting recognition, structured patterns, segmentation graph, spatial relations

1. Introduction

La reconnaissance de formes structurées manuscrites complexes, telles que des expressions mathématiques, soulève plusieurs problèmes spécifiques qui rendent difficile leur analyse. D'abord cette analyse est complexe à cause du grand nombre de symboles qui constituent ces formes, et le problème de segmentation est particulièrement difficile à résoudre. Ensuite, outre le nombre de constituants, la nature bidimensionnelle des formes accentue cette complexité d'analyse, puisqu'il existe un grand nombre de combinaisons et de relations spatiales envisageables entre les symboles.

De nombreux travaux ont été menés pour la reconnaissance d'une grande variété de formes manuscrites structurées : équations mathématiques (Awal *et al.*, 2009), (Fitzgerald *et al.*, 2007), (Rhee *et al.*, 2009), (Zanibbi *et al.*, 2002), caractères Chinois (Ota *et al.*, 2007), notations chimiques (Ramel *et al.*, 2000), (Ouyang *et al.*, 2007), diagrammes (Blostein, 1995), (Blostein *et al.*, 2002)... Paradoxalement, il n'y a pas eu à notre connaissance de recherche visant à mettre au point un moteur d'analyse générique pour traiter différents types de formes, malgré leurs nombreux points communs : organisation bidimensionnelle, structure hiérarchique et récursive, grand nombre de symboles constituants, rôle des relations spatiales dans leur interprétation.

Notre objectif dans ce papier est de proposer une méthode générique d'analyse pour les formes manuscrites structurées, nommée GASPR (Generic Approach for Structured Pattern Recognition). L'originalité de ces travaux est de chercher à utiliser au mieux les connaissances formalisées sur le domaine de formes à reconnaître (grammaire décrivant un langage, vocabulaire des symboles, jeu de relations de positionnement) pour guider un algorithme efficace d'analyse de ces formes dans un graphe multi-hypothèse. L'idée centrale est de limiter l'exploration des hypothèses de segmentation en injectant le plus tôt possible dans le système de segmentation et de reconnaissance les connaissances sur le domaine considéré, afin de réduire la complexité de l'analyse.

Pour illustrer la méthode sur l'écriture en-ligne et attester de sa généralité, nous la déploierons sur deux types de formes structurées très différentes : les expressions mathématiques et les caractères Chinois. Les premières constituent un exemple caractéristique des formes structurées, car elles peuvent être très complexes à analyser, avec de nombreux symboles et beaucoup de relations de positionnement possibles. Elles sont composées de symboles tels que des chiffres, des lettres, des opérateurs, des parenthèses, ..., et la position relative des symboles détermine l'interprétation globale de l'expression (ainsi " a^2 " est différent de " a_2 "). Les secondes correspondent aussi à un type de formes structurées, mais dont la complexité reste plus mesurée puisque les caractères sont constitués de 1 à 10 symboles appelés *radicaux*, agencés entre eux selon des relations de positionnement caractéristiques. Ces relations sont également déterminantes puisque deux caractères distincts peuvent être composés des mêmes radicaux organisés différemment. Si le domaine des expressions mathématiques est infini, et peut être décrit à l'aide d'une grammaire, les caractères Chinois appartiennent quant à eux à un lexique large (de l'ordre de 10 000 caractères sont utilisés) mais fini.

La deuxième section de cet article présente un bref état de l'art des méthodes de reconnaissance de formes structurées développées pour différentes applications. Ensuite, les principes de notre approche pour un système générique de reconnaissance de formes structurées sont exposés dans la troisième partie, puis l'architecture et le fonctionnement du système proposé sont présentés. La cinquième partie décrit les deux domaines d'applications étudiés et les spécificités de chaque implémentation, ainsi que les premiers résultats expérimentaux, avant la sixième section qui conclut l'article et offre quelques perspectives d'améliorations futures.

2. État de l'art

Les méthodes structurelles pour la reconnaissance de formes ont été très étudiées par le passé, et ont été largement appliquées à de nombreux domaines (Miclet, 1984) : vision par ordinateur, analyse de la parole, fouille de données. . . Le principe commun à ces méthodes est de modéliser explicitement la structure des objets traités, en les décrivant au moyen de primitives et de relations entre ces primitives.

Pour l'analyse de formes manuscrites, il existe de nombreuses méthodes dédiées à différentes natures de formes, le contexte applicatif le plus répandu étant sans doute celui de l'interprétation des équations mathématiques. D'une façon générale, on peut schématiser la plupart des méthodes existantes par une approche séquentielle en trois grandes étapes. La première est l'étape dite de segmentation, qui consiste à trouver les meilleurs regroupements des tracés individuels en symboles. Des hypothèses de reconnaissance de ces segments sont ensuite formulées dans la seconde étape, dite de reconnaissance. Enfin, la troisième étape a pour but de donner une interprétation globale à l'ensemble en analysant les relations spatiales intervenant entre les symboles (Ouyang *et al.*, 2007), (Blostein *et al.*, 2002). Or une caractéristique des formes structurées est qu'elles comportent souvent des ambiguïtés locales qui ne peuvent être levées qu'au niveau de l'analyse complète de la forme, c'est-à-dire en tenant compte d'une interprétation globale. Par conséquent, le schéma évoqué ci-dessus est insuffisant : il doit être accompagné de mécanismes de retours en arrière (remise en cause de la segmentation par exemple), ou alors il doit assurer une exploration exhaustive de l'espace de recherche, ce qui est souvent impossible à cause de la grande complexité engendrée.

Différentes techniques sont mises en œuvre dans les travaux de reconnaissance de formes structurées, afin de résoudre ce problème. Par exemple, dans le cadre de reconnaissance de notations chimiques (Ouyang *et al.*, 2007), les auteurs remettent en cause les relations spatiales qui conduisent à des interprétations syntaxiquement incohérentes. En reconnaissance d'expression mathématiques, dans (Rhee *et al.*, 2009), toutes les hypothèses de reconnaissance sont considérées et conservées et la décision est différée au moment de l'interprétation finale de l'expression globale, ce qui induit une complexité importante.

Adrien Delaye, Harold Mouchère

Pour l'analyse d'expressions mathématiques, plusieurs solutions ont été proposées permettant de gérer la difficulté de leur structure récurrente en deux dimensions. Dans (Awal *et al.*, 2009), des règles en une seule dimension, soit verticales soit horizontales, sont combinées par programmation dynamique. Dans (Zanibbi *et al.*, 2002), (Rhee *et al.*, 2009), (Tapia *et al.*, 2004), c'est le principe de symboles dominants qui est utilisé, chaque relation spatiale est portée par un symbole structurant la relation : un symbole est *au dessus* d'une barre de fraction, la barre de fraction est alors le symbole dominant. L'interprétation est représentée par un arbre avec une branche principale (représentant la ligne de base de l'expression) et des branches secondaires issues des symboles dominants (pour les symboles *nichés* en numérateur, dénominateur, exposant, indice...).

Pour la reconnaissance de caractères chinois, il existe des méthodes statistiques efficaces (Liu *et al.*, 2004) rendues possibles puisque le vocabulaire est fermé, cependant certains travaux mettent à profit la nature hiérarchique des caractères pour développer des approches structurelles (Zheng *et al.*, 1997), (Ota *et al.*, 2007). Par exemple, dans (Ota *et al.*, 2007) les auteurs décrivent les caractères par une grammaire stochastique couplant des terminaux symboliques avec des relations de positionnement. L'analyse d'une forme est alors faite en considérant toutes les règles applicables, et en élaguant les règles trop coûteuses au regard de la meilleure solution trouvée.

L'inconvénient des techniques d'interprétation de formes structurées évoquées ci-dessus est que l'analyse structurelle y est effectuée après le processus de segmentation et de reconnaissance. Celui-ci multiplie les hypothèses de segments, dont la majorité sont structurellement et syntaxiquement incongrues au regard du langage à reconnaître. De plus, ces méthodes sont fortement dépendantes de la réussite de l'étape de reconnaissance et ne permettent généralement pas d'en détecter les erreurs.

Par son approche originale pour la reconnaissance d'expressions mathématiques, le travail de (Rhee *et al.*, 2009) incorpore les hypothèses de relations spatiales au cœur même de la tâche de segmentation, les relations spatiales guidant les hypothèses de segmentation. Ainsi, si un segment est identifié comme un symbole de racine carrée, la recherche du segment suivant est concentrée spatialement vers la zone située sous la racine. À la fin de la segmentation, l'interprétation structurelle est déjà faite, puisque les segments sont déjà interprétés en contexte et reliés les uns aux autres par des relations de positionnement. Cette approche se retrouve aussi dans des travaux d'analyse de documents structurés (Macé *et al.*, 2009), où l'analyse des tracés est guidée par le contexte, en suivant des règles formalisées par une grammaire CD-CMG (Context-Driven Constraint Multiset Grammar).

Nos travaux rejoignent ceux de (Rhee *et al.*, 2009) dans l'approche générale : construire un arbre de relations de symboles (SRT *Symbol Relation Tree*) alternant des hypothèses de symboles avec des hypothèses de relations spatiales. La différence majeure réside dans la recherche du SRT solution. Dans (Rhee *et al.*, 2009), un arbre de recherche multi-hypothèse est parcouru en construisant progressivement plusieurs SRT en concurrence (chaque hypothèse est un SRT). Dans le système GASPR, c'est

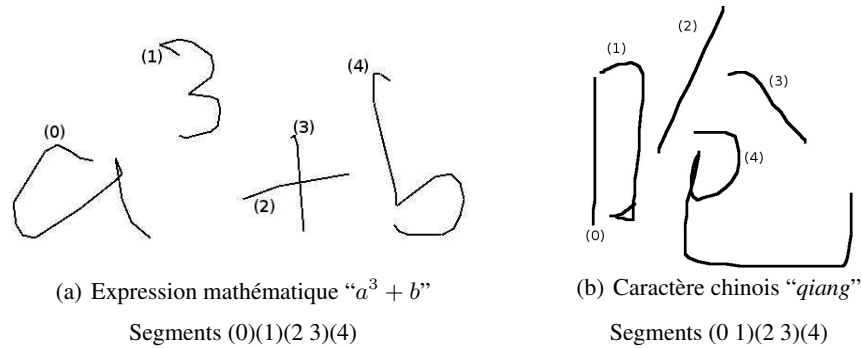


Figure 1. Exemple de formes structurées comportant chacune 5 tracés, avec l’interprétation et la segmentation associée

l’arbre de recherche lui-même qui constitue le SRT solution, ce qui évite répéter des hypothèses de symboles/rerelations/interprétations dans chaque SRT.

Dans la section qui suit, nous présentons les principes de notre approche qui reprennent cette idée d’introduction du contexte spatial ainsi que des contraintes syntaxiques pour guider au mieux la tâche de segmentation en réduisant sa complexité.

3. Principe

Pour mesurer la complexité du problème prenons l’exemple de la figure 1 qui montre une expression mathématique et un caractère chinois chacun composé de 5 traits, en considérant la complexité de leur reconnaissance dans le pire des cas qui correspondrait à une reconnaissance structurale naïve.

Supposons que chaque trait n’appartient qu’à un seul symbole. L’étape de segmentation consiste à tester toutes les combinaisons de traits pour tenter de former des symboles. Le nombre de segmentations possibles est défini par le nombre de Bell ¹. Sur notre exemple, on obtient $B_5 = 52$ segmentations distinctes. Ces 52 segmentations possibles contiennent $\sum_{i=0}^5 \binom{5}{i} = 31$ symboles distincts. Puis l’étape d’interprétation consiste à proposer une interprétation à chaque hypothèse de symbole. Il est courant de maintenir deux (ou plus) interprétations en parallèle pour chaque symbole. Dans notre exemple nous aurions $31 \times 2 = 62$ hypothèses. Ensuite s’ajoute l’étape d’évaluation des relations possibles entre chaque paire de symboles. Dans l’exemple, nous n’avons que 2 relations possibles à droite et en exposant. Il faut évaluer chacune des relations avec tous les couples de symboles, soit $\binom{31}{2} \times 2 = 930$ évaluations. Enfin, à partir de toutes ces informations, le graphe de segmentation est construit : il contiendra 62

1. Les nombres de Bell définis par : $B_{n+1} = \sum_{k=0}^n \binom{n}{k} B_k$ représentent le nombre de partitions possibles d’un ensemble de taille n .

nœuds d'hypothèses et 930 arcs de relations spatiales. Il s'agit alors de trouver dans ce graphe l'interprétation (le SRT) qui minimise le coût global de reconnaissance. Nous avons vu dans la section précédente différentes stratégies proposées dans l'état de l'art pour réduire cette complexité.

L'objectif de notre approche est de dépasser ce schéma classique de la reconnaissance de formes structurées : segmentation, reconnaissance puis identification des relations entre symboles, qui impose d'explorer un très vaste champs d'hypothèses dont la grande majorité sont incohérentes au regard du langage que l'on cherche à reconnaître. Notre approche permet de faire coopérer les trois processus en même temps, chacun utilisant les informations disponibles issues des deux autres. La tâche de segmentation, délicate a priori, est ainsi guidée par la connaissance introduite relative au positionnement spatial et aux structures de langage autorisées. Par exemple dans l'exemple de la figure 1, nous ne chercherons pas à commencer l'interprétation de l'expression par le '*b*' qui est trop à droite, ou à mettre le '*b*' en indice ou exposant du '+' puisque le '+' ne peut pas supporter ce type de relation. De cette façon, seules les interprétations qui respectent le langage du domaine d'expressions considéré sont explorées, ce qui permet de réduire considérablement l'espace de recherche.

GASPR s'appuie sur un graphe de segmentation/interprétation dont l'architecture est décrite en détails dans la section 4. Des ressources dépendantes du domaine sont exploitées par le système pour générer ou évaluer des hypothèses de segmentation et de relations spatiales. Des exemples de définition de ces éléments (reconnaisseurs de symboles, opérateurs de positionnement relatif, et modélisation du langage) sont donnés à la section 5 pour les deux domaines d'application envisagés : équations mathématiques et caractères chinois.

4. GASPR : Approche générique pour la reconnaissance de formes structurées

L'approche GASPR que nous proposons pour aborder l'analyse de formes manuscrites structurées s'appuie sur une représentation par un graphe de segmentation et d'interprétation. Ce graphe est construit dynamiquement au fur et à mesure de l'analyse, au moyen d'un algorithme de parcours efficace basé sur A* : le principe est de trouver la solution optimale valide avant d'avoir construit tout le graphe, en n'explorant pas les hypothèses de segmentation ou d'interprétation incohérentes.

4.1. Structure du graphe de segmentation/interprétation

Le graphe utilisé par GASPR ressemble au SRT utilisé pour la reconnaissance d'expressions mathématiques mais supportant plusieurs hypothèses de symboles et hypothèses de relation. Le graphe construit peut être vu comme un graphe multi-hypothèses où sont explorées plusieurs hypothèses (mais pas toutes) de symboles, de relations et d'interprétations.

Soit $G = \langle N, A \rangle$ le graphe de segmentation/interprétation d'un ensemble de tracés manuscrits T constituant une forme à reconnaître, avec N l'ensemble des nœuds et A l'ensemble des arcs du graphe. Nous définissons deux types de nœuds, de sorte que $N = N_S \cup N_R$:

- un nœud $s \in N_S$ constitue une *hypothèse de symbole*, qui associe une hypothèse de reconnaissance σ_s (une classe de symbole) à un ensemble de tracés $T_s \subset T$;
- un nœud $r \in N_R$ constitue une *hypothèse de relation spatiale* qui met en relation les hypothèses de symboles qui lui sont liés par des relations de positionnement : par exemple “ s_{arg} est à droite de s_{ref} ” ou “ s_{arg} est en exposant de s_{ref} ”. Ces relations sont asymétriques, l'un des symboles joue le rôle de référence (s_{ref}) tandis que l'autre joue le rôle d'argument (s_{arg}).

Les arcs du graphe ne portent quant à eux pas d'information et servent uniquement à lier entre eux des sommets, modélisant ainsi la structure de la forme à reconnaître. Le graphe est orienté, et les arcs relient toujours un nœud de symbole avec un nœud de relation spatiale : $A \subset (N_S \times N_R) \cup (N_R \times N_S)$:

- un arc $a_{s,r} = (s, r) \in (N_S \times N_R)$ représente un lien entre une hypothèse de symbole s et une hypothèse de relation spatiale r , s jouant le rôle de référence pour r ;
- un arc $a_{r,s} = (r, s) \in (N_R \times N_S)$ représente un lien entre une hypothèse de relation spatiale r et une hypothèse de symbole s , s jouant le rôle d'argument pour r .

Une interprétation pour une forme à reconnaître constituée d'un ensemble de tracés T est un sous-graphe connexe $G' = \langle N', A' \rangle$ du graphe de segmentation G dont toutes les hypothèses de symboles sont compatibles deux à deux (aucun tracé n'est utilisé par plusieurs hypothèses de symboles) :

$$\forall m, n \in N'_S, T_m \cap T_n = \emptyset$$

De plus, l'interprétation G' est dite *complète* si ses hypothèses de symboles couvrent l'ensemble des tracés de la forme à reconnaître :

$$\bigcup_{n \in N'_S} T_n = T$$

Une interprétation complète est donc telle que les hypothèses de symboles qui la composent constituent une partition des tracés d'entrée. Elle peut être réduite à un simple chemin du graphe total si l'expression reconnue est linéaire (par exemple “ $a+b=3$ ”), mais se présente en général sous forme d'un arbre comportant plusieurs branches.

4.2. Formalisation du problème

Le problème de reconnaissance de formes structurées peut être formulé comme suit : étant donné un ensemble de tracés $T = t_0, \dots, t_N$, on cherche la meilleure interprétation, c'est-à-dire l'interprétation G^* , parmi l'ensemble des possibles, qui minimise une fonction de coût $C(G)$. L'interprétation G^* est cherchée en construisant et

explorant progressivement le graphe de segmentation dont la structure a été présentée plus haut. Les fonctions de coût sont de deux types, selon qu'elles sont associées aux hypothèses de symboles ou aux hypothèses de positionnement relatif.

- les coûts associés aux nœuds $s \in N_S$ sont liés au score de reconnaissance du segment porté par le nœud (l'ensemble de tracés T_s) par rapport à une hypothèse de reconnaissance σ_s , par $C_S(s) = -\log(\text{scoreReco}(T_s, \sigma_s))$;

- les coûts associés au positionnement relatif des symboles sont portés par un arc de type $a = (r, s) \in (N_R \times N_S)$. Ils sont liés à l'adéquation du positionnement du symbole T_s porté par s par rapport au modèle de positionnement P_r porté par le nœud de relation r , $C_R(a_{r,s}) = -\log(\text{scorePos}(T_s, R_r))$;

- les coûts des arcs $a_{s,r} \in (N_S \times N_R)$ sont nuls.

Finalement, le score total d'une interprétation $G' = \langle N'_S \cup N'_R, A' \rangle$ se calcule par

$$C(G') = \sum_{s \in N'_S} C_S(s) + \sum_{a \in (A')} C_R(a)$$

4.3. Construction et parcours intelligent du graphe

Les interprétations sont construites dynamiquement avec le graphe. Un algorithme inspiré de A* guide cette construction et cette exploration, à l'aide de deux piles : une pile d'hypothèses de symboles P_S et une pile d'hypothèses de relations P_R . Ces piles sont triées selon une fonction de coût f , calculé pour un nœud n par :

$$f(n) = \frac{g(n)}{N} + h(n), \quad [1]$$

où $g(n)$ est le coût réel cumulé des nœuds et des arcs du meilleur chemin de n à "START" (le point d'entrée du graphe) et N est le nombre de traits expliqués par ce meilleur chemin. La normalisation de $g(n)$ par N permet de ne pas favoriser les chemins courts (expliquant peu de traits) dans l'exploration. Si n^* est le nœud parent de n avec le plus faible coût, on a :

$$g(n) = \begin{cases} g(n^*) + C_S(n) + C_R(a_{n^*,n}) & \text{si } n \in N_S \\ g(n^*) & \text{si } n \in N_R \end{cases} \quad [2]$$

L'heuristique $h(n)$ donne une estimation minimale du coût nécessaire pour atteindre un nœud "STOP" à partir du nœud n . Elle permet donc d'éviter d'explorer des hypothèses en élaguant dans le graphe les chemins dont le coût estimé ne peut pas être meilleur qu'une solution déjà trouvée. Afin de garantir que l'algorithme trouve toujours la meilleure solution, cette heuristique est définie telle que $h(n)$ soit toujours inférieur au coût réel nécessaire pour terminer le chemin (heuristique *admissible*).

Les nœuds les plus prometteurs sont donc dépilés en premier. À chaque étape, l'algorithme évolue avec un nœud courant, et alterne grâce à l'utilisation des deux piles la

création d'hypothèses de symboles avec la création d'hypothèses de positionnements. L'algorithme se compose de 6 étapes principales :

-
- 1) empiler le noeud relation START dans P_R
 - 2) dépiler un noeud relation r de P_R si elle est n'est pas vide
 - 3) si r est un noeud STOP
 - a) calculer la partition partielle correspondante et la mémoriser
 - b) combiner avec les partitions existantes pour chercher une solution
 - c) si une partition complète existe terminer l'algorithme
 - d) sinon retour à l'étape 2
 - 4) sinon
 - a) générer des hypothèses de symboles $\{s_i\}$ cohérentes avec le noeud r
 - b) évaluer $\forall i a_{r,s_i}, g(s_i)$ et $h(s_i)$
 - c) empiler les s_i dans P_S si $f(s_i) < \infty$
 - 5) dépiler un symbole s de P_S si elle est n'est pas vide
 - a) générer des hypothèses de relations $\{r_i\}$ cohérentes avec le noeud s
 - b) évaluer les $\forall i g(r_i)$ et $h(r_i)$
 - c) empiler les r_i dans P_R si $f(r_i) < \infty$
 - 6) retour à l'étape 2 si P_R ou P_S n'est pas vide
-

Remarquons que puisque plusieurs relations peuvent partager des hypothèses de symboles certains noeuds peuvent être visités plusieurs fois. Dans ce cas, ils ne sont ré-empilés que si la nouvelle évaluation $f(n)$ est meilleure que la précédente. Cette ré-évaluation peut se propager de noeud en noeud jusqu'à un noeud *STOP*, la partition associée sera alors remise en cause. L'algorithme ne garantit pas de trouver une segmentation/interprétation complète de la forme, dans ce cas la combinaison de partitions incomplètes expliquant le plus de tracés est donnée comme solution.

Quatre points particuliers de ce déroulement exploitent des connaissances extérieures pour guider la segmentation en construisant ou évaluant des hypothèses de symboles et de relations spatiales, rendant cet algorithme apte à construire efficacement une interprétation correcte de la forme considérée :

- au point 4a, la relation spatiale r est utilisée pour restreindre la génération des hypothèses de symboles s_i aux tracés situés dans la zone respectant cette relation ;
- au point 4b, un reconnaiseur de symboles dépendant du domaine est utilisé pour évaluer la qualité d'une hypothèses de symbole. En particulier un rejet peut intervenir si un segment ne correspond à aucun symbole du vocabulaire ;
- au point 4b, des opérateurs de positionnement spécifiques sont également utilisés pour attribuer un score d'adéquation entre la relation r et le symbole argument s_i ;
- au point 5a, la connaissance extérieure sur le langage est utilisée pour générer des hypothèses de relations cohérentes avec le symbole s : par exemple, dans une expression mathématique, les relations *indice*, *exposant* et *à droite* peuvent être trouvées après un symbole "a", tandis que la seule relation autorisée après un "+" est *à droite*.

4.4. Exemple

Nous décrivons ici un exemple simple de déroulement de l'algorithme pour la reconnaissance de l'expression mathématique de la figure 1(a) avec des connaissances du domaine simplifiées. Nous supposons qu'une expression n'est composée que de symboles comportant deux tracés au maximum et sans sauts temporels et de trois types de relations : à droite (noté R->), en indice (Rv) ou en exposant (R^). Le classifieur utilisé peut proposer deux hypothèses de reconnaissance dont une spéciale (notée #) correspondant au rejet de l'hypothèse de symbole.

Le tableau 1 décrit toutes les étapes de déroulement de l'algorithme permettant d'arriver à la solution. Les deux premières étapes de la construction du graphe sont illustrées par les figure 2(c) et 2(d). La figure 3 représente quant à elle l'ensemble du graphe tel qu'il est à la fin de la recherche, à l'étape 14. L'arbre de solution est représenté par les arcs rouges.

| Etape | Nœud dépilé | Nœuds empilés |
|---|-----------------------|---------------------------------|
| E1 | START | S(0)[a], S(0 1)[#] |
| E2 | S(0)[a] | Rv, R^, R->, STOP |
| E3 | R-> | S(2 3)[#], S(2 3)[+], S(1)[3] |
| E4 | S(2 3)[+] | R-> |
| E5 | R-> | S(4)[b] |
| E6 | S(4)[b] | Rv, R^, R->, STOP |
| E7 | STOP | |
| partition (0)(2 3)(4) : "a + b" (incomplète) | | |
| E8 | R^ | S(1)[3] (mise à jour de g et f) |
| E9 | S(1)[3] | Rv, R^, R->, STOP |
| E10 | Rv | aucun symbole possible |
| E11 | pile de symboles vide | |
| E12 | R-> | aucun symbole possible |
| E13 | pile de symboles vide | |
| E14 | STOP | |
| partition (0)(1) : "a ³ " (incomplète) combinaison-> partition (0)(1)(2 3)(4) : "a ³ + b" (solution) | | |

Tableau 1. Etapes de reconnaissance de l'expression $a^3 + b$ de la figure 1(a). Les hypothèses de symboles sont notées $S(i_k, k = 1..n)[\sigma]$ où i_k sont les indices des tracés constituant l'hypothèse et σ est la classe attribuée.

Nous pouvons noter quelques points particuliers dans cet exemple qui illustre la simplification de l'exploration permise par notre algorithme. Tout d'abord, l'utilisation d'un classifieur avec une option de rejet permet d'évaluer certaines hypothèses avec un coût infini. À l'étape E4, seule la relation R-> est générée grâce à la connaissance de la grammaire autorisée. Le nœud S(1)[3] est activé deux fois, à l'étape E4

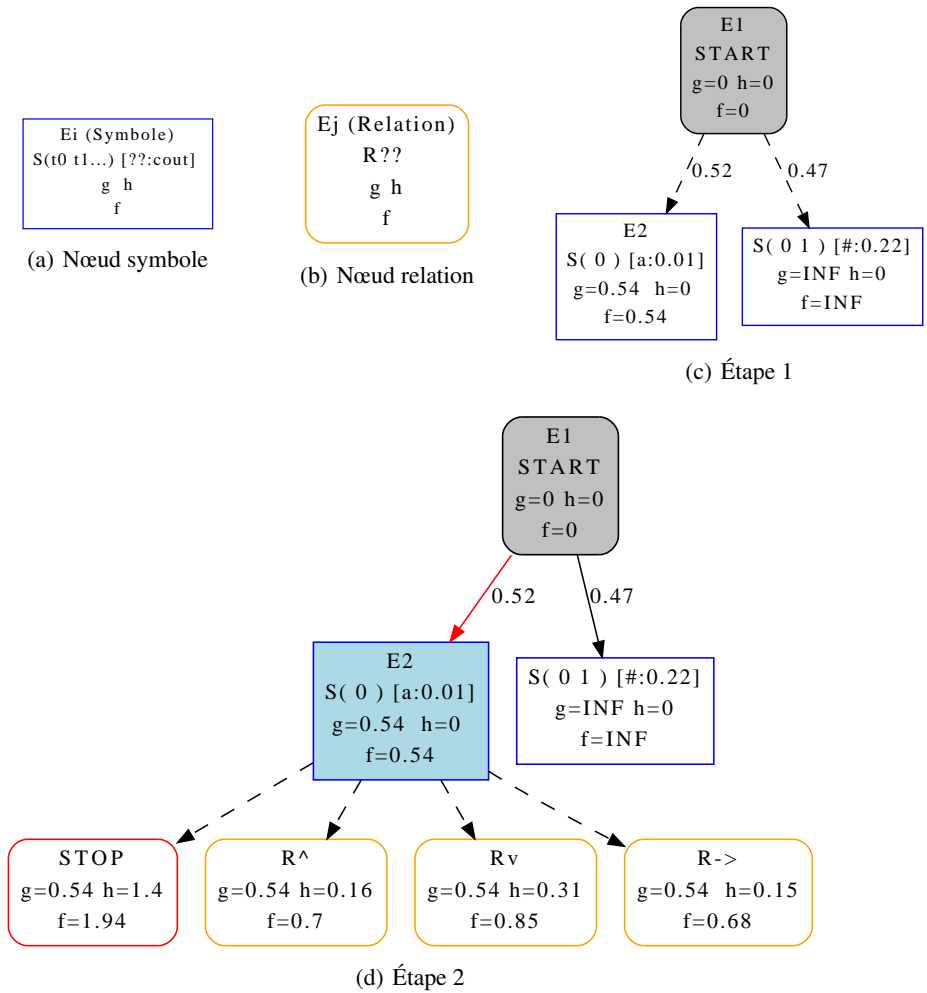


Figure 2. Les deux types de nœuds sont illustrés par (a) et (b). Les figures (c) et (d) présentent l'état du graphe aux étapes E1 et E2 du tableau 1. Les nœuds pleins sont les nœuds visités au cours de l'algorithme. Les arcs rouges symbolisent le meilleur chemin courant, et les arcs pointillés relient le nœud courant avec les nœuds créés à l'étape considérée.

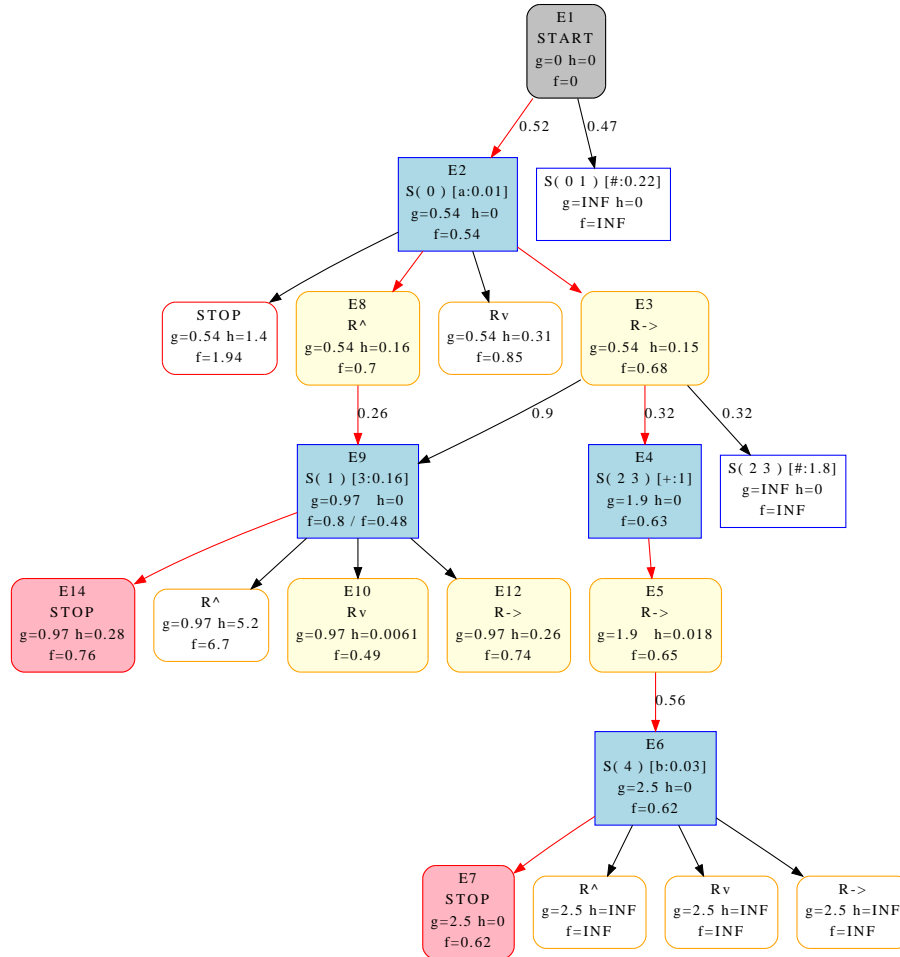


Figure 3. État du graphe obtenu à l'étape E14 du tableau 1. Le sous-graphe solution est décrit par les arcs rouges. L'analyse complète menant à cette solution a généré 6 hypothèses de segments et 13 hypothèses de relations spatiales.

avec un coût $g(n)$ important puis E8 avec coût $g(n)$ plus faible. À la fin de l'algorithme, la solution est bien un arbre correspondant exactement à l'interprétation de l'expression mathématique.

Cet exemple est le même que celui détaillé dans la section 3. Nous pouvons constater que notre approche réduit considérablement le nombre d'hypothèses explorées : 6 symboles au lieu de 62 et 14 relations évaluées au lieu de 930.

5. Exploitation pour les expressions mathématiques et les caractères Chinois

Nous présentons dans cette partie les deux applications de l'approche générique présentée à la section 4, consacrées à la reconnaissance d'expressions mathématiques et de caractères chinois. Dans un premier temps, nous présentons les spécificités de chacune, en ce qui concerne la connaissance qui est injectée dans l'algorithme de segmentation-reconnaissance. Ensuite, nous présentons les résultats obtenus sur chacun des deux types de données, et discutons de ces résultats.

5.1. Spécificités d'implémentation

Ainsi que présenté dans la partie 3, la mise en œuvre de la stratégie efficace de segmentation et interprétation nécessite une incorporation de connaissances dépendantes du domaine d'application considéré, tel que des contraintes de tracé, un lexique de symboles, un jeu de relations spatiales admises et une formalisation du langage permettant d'exprimer des contraintes de structure. Nous présentons ici ces connaissances spécifiques pour les expressions mathématiques et les caractères chinois. Le calcul de la fonction heuristique $h(n)$ est commun au deux applicatifs. Pour les nœuds symboles, $h(n)$ est nul ; pour les nœuds relations $h(n)$ représente le cout minimal d'application de cette relation aux tracés non interprétés.

5.1.1. Équations mathématiques

Pour la reconnaissance d'expressions mathématiques nous nous sommes intéressés à un corpus réduit de 36 expressions déjà utilisées dans (Awal *et al.*, 2009). Il est composé de 34 symboles différents (1, 2, 3, 4, 5, a, b, c, d, e, i, k, n, x, y, +, -, =, \pm , \neq , γ , π , ϕ , \sum , \int , $\sqrt{\quad}$, (,), \dots , cos, sin, sinh, cosh, log). Une 35^{ème} classe rejet est ajoutée au classifieur pour détecter les mauvaises segmentations. La classe '-' est utilisée pour le signe moins mais aussi pour la barre de fraction.

Pour les relations spatiales, des modèles géométriques basés sur les rectangles englobants des symboles sont développés, qui calculent un score flou de positionnement pour un couple de symboles donnés. Nous utilisons 7 relations spatiales possibles :

- à droite : possible après tous les symboles ;
- en exposant (resp. en indice) : impossible après {+, -, =, \pm , \neq , $\sqrt{\quad}$, (},
- au dessus (resp. au dessous) : possible seulement après la barre de fraction,
- à l'intérieur : possible seulement après $\sqrt{\quad}$
- STOP : impossible après les symboles {+, -, =, \pm , \neq , $\sqrt{\quad}$, (}.

5.1.2. Caractères chinois

Les caractères chinois sont des formes fortement structurées, constituées de sous-éléments appelés radicaux, qui constituent un lexique d'environ 200 symboles. Tous les caractères chinois (au nombre de plusieurs milliers) sont créés à partir de la com-

Adrien Delaye, Harold Mouchère

position de ces radicaux selon des schémas de positionnement prédéfinis qui sont appliqués récursivement (Wang *et al.*, 2001). L'exemple de la figure 1(b) est un caractère chinois constitué de 3 radicaux.

Le vocabulaire de base est constitué des radicaux. Pour les premières expérimentations menées, le lexique a été réduit à un petit nombre de ces radicaux (seulement 13 classes) reconnaissables par un classifieur simple de type RBFN. Les relations spatiales possibles sont au nombre de 9 et correspondent aux modèles de structuration des caractères chinois : *gauche-droite*, *haut-bas*, *gauche-haut*, *haut-droite*, *gauche-bas*, *haut-gauche-bas*, *gauche-haut-droite*, *gauche-bas-droite* et *autour*. Neuf opérateurs de positionnement relatif correspondants ont été définis, à partir de fonctions floues basées sur les rectangles englobants des segments. La description du langage est un lexique des caractères, qui sont décrits par un arbre de leurs radicaux et relations spatiales, dont on extrait des contraintes. Par exemple, on constate que certaines relations spatiales ne peuvent se présenter qu'après certains radicaux, ou que certains radicaux ne peuvent se trouver que dans des configurations bien spécifiques... Pour notre expérimentation, le lexique est constitué de 80 caractères composés à l'aide des 13 radicaux et des 9 opérateurs de positionnement.

5.2. Expérimentations

Il s'agit de premières expérimentations dont le but est de montrer la généralité et la faisabilité de notre approche. Dans les deux cas étudiés, nous traitons des données en-ligne, c'est à dire que le signal est une séquence de points décrivant la trajectoire manuscrite.

Pour les expressions mathématiques nous utilisons la base d'expression de (Awal *et al.*, 2009). 280 scripteurs ont saisi l'ensemble des symboles isolés puis un générateur a synthétisé 36 expressions différentes pour chaque scripteur. Les expressions contiennent en moyenne 11,4 symboles. Les données de 180 scripteurs sont utilisées comme base d'apprentissage du classifieur (soit 6480 expressions) et celles des 100 autres pour la base de test (soit 3600 expressions). La base de caractères Chinois contient les 80 classes de caractères écrits chacun une fois par 60 scripteurs différents. Les données de 40 scripteurs constituent la base d'apprentissage du classifieur de radicaux, et les données des 20 autres scripteurs constituent la base de test pour notre méthode. Les 80 caractères comportent de 1 à 5 radicaux, avec en moyenne 2,2 radicaux par caractère.

La table 2 présente les résultats pour les deux domaines. Le taux de reconnaissance global représente le nombre de formes manuscrites (expressions ou caractères) correctement interprétées (symboles et relations). Pour le domaine des expressions mathématiques, un taux de bonne segmentation et aussi indiqué, qui représente la proportion de symboles correctement segmentés.

Ces premiers résultats, obtenus sur des bases de données modestes mais réalistes, montrent le bien fondé d'une approche générique pour la reconnaissance de formes

| Base | taux de segmentation | taux de reconnaissance |
|-----------------------|----------------------|------------------------|
| Exp. math., B_{APP} | 68,3% | 24% |
| Exp. math., B_{TST} | 66,2% | 16,9% |
| Chinois B_{APP} | - | 95,78% |
| Chinois B_{TST} | - | 92,97% |

Tableau 2. Résultats sur les expressions mathématiques et les caractères Chinois.

structurées. En effet, malgré l'hétérogénéité des formes analysées, comportant de 1 radical pour le plus simple caractère chinois à 21 symboles pour l'expression mathématique reconnue la plus complexe, la méthode présentée est capable de reconnaître correctement des exemples de tous types.

La nette différence de performance entre les deux applications s'explique en partie par le fait que la grammaire des expressions mathématiques est ouverte et permet un ensemble de solutions bien plus large que le vocabulaire fermé des caractères chinois. De plus, étant donné le nombre moyen de symboles par expressions nettement supérieur, le taux global de bonne reconnaissance est naturellement inférieur au taux obtenus pour des formes plus simples. Le taux de bonne segmentation de près de 66% permet de mieux apprécier la qualité de la segmentation, et suggère que l'amélioration du reconaisseur de symboles ou des critères de génération des hypothèses de symboles permettraient un sensible gain de performance du système global. De la même façon pour les caractères chinois, la différence importante entre les taux recueillis sur la base d'apprentissage par rapport à la base de test montre que le reconaisseur de symboles est responsable de la majorité des erreurs de reconnaissance.

Les solutions à envisager pour l'amélioration de la segmentation concernent la modélisation des modèles de relations spatiales. Il peut être envisagé d'adopter des techniques issues de la morphologie mathématique floue comme dans les travaux de (Delays *et al.*, 2010), où des modèles de positionnement relatif sont définis, qui peuvent être appris automatiquement à partir de données. Un travail sur l'heuristique h permettrait aussi d'améliorer les résultats, par exemple en liant des coût grammaticaux aux hypothèses de reconnaissance de symboles.

6. Conclusion

Dans ce papier nous proposons GASPR, un algorithme générique permettant la reconnaissance de formes structurées. Dans un graphe de segmentation/interprétation nous explorons un grand nombre d'hypothèses de symboles et de relations spatiales. Un algorithme inspiré de A* et une utilisation précoce de connaissances spécifiques au domaine permettent de maîtriser la complexité de la recherche. Ainsi seules des solutions cohérentes avec le domaines sont explorées. Nous montrons la faisabilité de notre approche ainsi que sa généralité grâce à de premières expérimentations sur deux domaines très différents : la reconnaissance d'expressions mathématiques et la reconnaissance de caractères Chinois.

Adrien Delaye, Harold Mouchère

7. Bibliographie

- Awal A., Mouchère H., Viard-Gaudin C., « Towards handwritten mathematical expression recognition », *Proceedings of the 10th ICDAR*, Barcelona, Spain, p. 1046-1050, 2009.
- Blostein D., « General Diagram-Recognition Methodologies », *Proc. Int. Workshop on Graphics Recognition*, Springer, p. 106-122, 1995.
- Blostein D., Cordy J. R., Zanibbi R., « Applying Compiler Techniques to Diagram Recognition », *Proceedings of the 6th ICPR*, vol. 3, Quebec City, Canada, p. 123-136, 2002.
- Delaye A., Anquetil E., « Modélisation du positionnement relatif de tracés manuscrits par morphologie mathématique floue », *Actes du 17ème colloque RFIA*, 2010.
- Fitzgerald J., Geiselbrechtinger F., Kechadi T., « Mathpad : A Fuzzy Logic-Based Recognition System for Handwritten Mathematics », *Proceedings of the 9th ICDAR*, vol. 2, 2007.
- Liu C., Jaeger S., Nakagawa M., « Online Recognition of Chinese Characters : the State-of-the-Art », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, n° 2, p. 198-213, 2004.
- Macé S., Anquetil E., « Eager interpretation of on-line hand-drawn structured documents : The DALI methodology », *Pattern Recognition*, vol. 42, p. 3202-3214, 2009.
- Miclet L., *Méthodes structurelles pour la reconnaissance des formes*, Eyrolles, 1984.
- Ota I., Yamamoto R., Sako S., Sagayama S., « On-line Handwritten Kanji Recognition Based on Inter-stroke Grammar », *Proceedings of the 9th ICDAR*, vol. 2, p. 1188-1192, 2007.
- Ouyang T., Davis R., « Recognition of hand drawn chemical diagrams », *Proceedings of the national conference on artificial intelligence*, vol. 22, Menlo Park, CA ; Cambridge, MA ; London ; AAAI Press ; MIT Press ; 1999, p. 846, 2007.
- Ramel J., Boissier G., Emptoz H., « A structural representation adapted to handwritten symbol recognition », *Lecture Notes in Computer Science*, Springer, p. 228-237, 2000.
- Rhee T., Kim J., « Efficient search strategy in structural analysis for handwritten mathematical expression recognition », *Pattern Recognition*, vol. 42, p. 3192-3201, 2009.
- Tapia E., Rojas R., « Recognition of On-line Handwritten Mathematical Expressions Using a Minimum Spanning Tree Construction and Symbol Dominance », *Lecture Notes in Computer Science, GREC 2003*, vol. 3088, p. 329-340, 2004.
- Wang A.-B., Fan K.-C., « Optical recognition of handwritten Chinese characters by hierarchical radical matching method », *Pattern Recognition*, vol. 34, n° 1, p. 15-35, 2001.
- Zanibbi R., Blostein D., Cordy J., « Recognizing mathematical expressions using tree transformation », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, p. 1455-1467, 2002.
- Zheng J., Ding X., Wu Y., « Recognizing On-line Handwritten Chinese Character via FARG Matching », *Proceedings of the 4th ICDAR*, p. 621-624, 1997.