



**HAL**  
open science

## Un état de l'art des méthodes de localisation de symboles dans les documents graphiques

Thi Oanh Nguyen, Salvatore Tabbone

► **To cite this version:**

Thi Oanh Nguyen, Salvatore Tabbone. Un état de l'art des méthodes de localisation de symboles dans les documents graphiques. Colloque International Francophone sur l'Écrit et le Document - CIFED 2010, Mar 2010, Sousse, Tunisie. pp.217-232. inria-00472152

**HAL Id: inria-00472152**

**<https://inria.hal.science/inria-00472152>**

Submitted on 9 Apr 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.

---

# Un état de l’art des méthodes de localisation de symboles dans les documents graphiques

**Thi-Oanh Nguyen — Salvatore Tabbone**

LORIA - Université Nancy 2  
Campus Scientifique - BP 239  
54506 Vandoeuvre-lès-Nancy Cedex  
{thi-oanh.nguyen, tabbone}@loria.fr

---

*RÉSUMÉ.* Dans cet article, nous proposons un panorama de méthodes de localisation de symboles dans les documents graphiques. Nous les divisons suivant deux catégories : les approches structurelles et les pixelaires. Les approches structurelles sont basées souvent sur des représentations de types graphes et possèdent généralement une étape de segmentation préalable des documents en primitives. Le symbole est ensuite détecté via une étape de regroupements de primitives et sous certaines conditions. Dans les approches pixelaires, la localisation est effectuée directement sur les documents sans étape préalable de segmentation.

*ABSTRACT.* In this paper, we present a survey on symbol spotting methods for graphical documents. We classify these methods into two categories: structural and pixel-based approaches. Structural approaches are often based on graphs representations and frequently need a preliminary segmentation step in order to break documents into primitives. A symbol is then detected by regrouping neighbouring primitives under certain conditions. In pixel-based approaches, the symbol spotting is performed directly on the entire images without a preliminary segmentation step.

*MOTS-CLÉS :* localisation de symboles, symboles graphiques.

*KEYWORDS:* symbol spotting, graphical symbol.

---

## 1. Introduction

Nous abordons dans cet article le problème complexe de la *localisation de symboles*<sup>1</sup> dans les documents où les symboles ne sont pas isolés de leur contexte. Bien qu'il existe beaucoup de travaux visant à la définition de bons descripteurs pour la représentation d'un symbole, ces derniers ne peuvent généralement pas être utilisés directement pour le problème de localisation de symboles dans les documents. En effet, un descripteur peut fournir de bons résultats pour la recherche de symboles segmentés ou isolés mais être difficilement adaptable au cadre de la description des symboles d'un document. Une stratégie classique de description d'un document consiste à le décomposer en plusieurs composantes, puis à appliquer des descripteurs prédéfinis sur chacun de ces composantes. Souvent une étape supplémentaire de vectorisation est appliquée (Wenyin *et al.*, 2007, Fonseca *et al.*, 2005, Locteau *et al.*, 2007, Rusinol *et al.*, 2006). Certains travaux proposent de ne considérer que les symboles satisfaisants certaines conditions (par exemple : la convexité, la connectivité ou la fermeture du symbole) (Tabbone *et al.*, 2007, Rusinol *et al.*, 2006, Rusinol *et al.*, 2007).

Nous proposons un panorama récent de techniques de localisation de symboles dans les documents graphiques. Malgré de nombreux travaux dans l'analyse de documents, il y a peu de travaux qui attaquent le problème de la localisation de symboles. Nous les divisons suivant deux catégories : les approches structurelles et les pixelaires. Les approches se basant sur la représentation structurelle de l'image seront considérées comme des approches structurelles. Elles utilisent souvent une représentation sous forme de graphes et possèdent généralement une étape de segmentation préalable des documents en primitives. Le symbole est ensuite détecté via une étape de regroupements de primitives sous certaines conditions. Dans les approches pixelaires, la localisation est effectuée directement sur l'image entière sans étape préalable de segmentation.

## 2. Approches structurelles

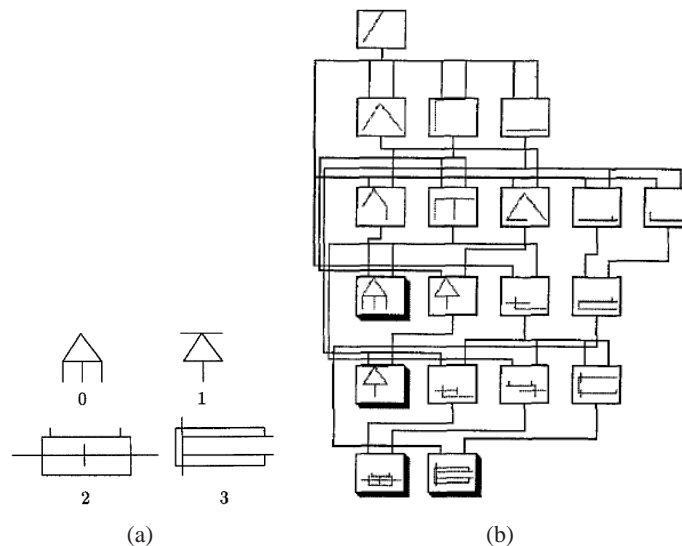
Une étape indispensable pour les approches structurelles consiste à décomposer l'image en primitives comme des lignes, des courbes, des régions ou bien des formes géométriques simples telles que des cercles, des rectangles, etc. Le critère d'adjacence ainsi que les caractéristiques de chaque primitive sont pris en compte pour agréger des parties qui répondent à des hypothèses prédéfinies ou qui correspondent à des (sous-)structures connues par le système. Les relations entre les primitives sont souvent représentées par un graphe et la mise en correspondance des objets revient à la recherche d'un isomorphisme de (sous-)graphes.

Le segment est l'une des primitives les plus simples à utiliser pour la description des symboles (Messmer *et al.*, 1995, Park *et al.*, 2000, Park *et al.*, 2003). Dans (Messmer *et al.*, 1995), les symboles connus par le système sont représentés dans un

---

1. Traduit de "symbol spotting"

réseau qui décrit le processus de construction au fur à mesure à partir des segments. Ce réseau se compose de plusieurs niveaux qui correspondent aux parties possibles des symboles. Ces parties s'élargissent de plus en plus en combinant des parties des niveaux précédents. Le niveau le plus bas contient les symboles complets (voir Fig. 1). Le module d'appariement à chaque partie permet de trouver des (sous-)isomorphismes des symboles dans le document. La représentation des symboles connus par un réseau rend la recherche des isomorphismes de (sous-)graphes plus efficace au niveau de la complexité des calculs. Ce système fournit aussi la capacité d'apprendre de nouveaux symboles à partir des documents. Cependant, ces nouveaux symboles doivent satisfaire trois heuristiques : i) le graphe du symbole doit être connecté ; ii) il doit contenir au moins un circuit fermé, et si possible, des circuits fermés adjacents ; iii) le segment qui croise le circuit fermé appartient aussi au symbole.



**Figure 1.** Réseau de représentation des symboles connus par le système. (a) exemple de quatre symboles. (b) la partie du réseau qui représente les symboles 0, 1, 2, 3 dans (a).

Afin de localiser rapidement les régions de l'image pouvant potentiellement contenir le symbole requête, (Dosch *et al.*, 2004) propose d'utiliser la signature vectorielle pour représenter le symbole requête ainsi que les différentes régions de l'image. La signature d'une entité (symbole) est définie par l'ensemble des relations existantes entre segments. Cinq types de relations sont considérés : le recouvrement, la colinéarité, le parallélisme, la jonction "T" et la jonction "V". Pour effectuer la localisation, l'image est divisée en plusieurs régions disjointes (appelées "bucket") dont la taille dépend de la taille du plus grand symbole de référence. Un test d'inclusion des signatures entre les symboles de référence et ces "bucket" permet d'étiqueter des symboles qui peuvent être présents dans la région. Cette étape de pré-traitement ne permet pas de déterminer

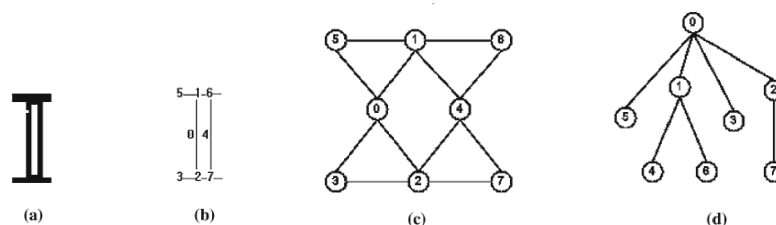
la localisation exacte de chacun des symboles mais permet de détecter rapidement le type de symbole potentiellement contenu dans chacune des régions.

Parmi les autres approches structurelles, (Tabbone *et al.*, 2007) propose de segmenter les symboles contenus dans les documents pour les localiser par rapport à un symbole requête. Ainsi, le document est décomposé en chaînes de points connectés qui sont ensuite regroupées pour redéfinir de nouveaux symboles. En effet, le document est représenté par un graphe de jonctions définissant les relations entre les primitives segmentées (chaînes de points). Des sous-ensembles de chaînes de points qui pourraient former un symbole sont ensuite déterminés grâce à la construction d'un dendrogramme. Cette dernière se fait de façon itérative en fusionnant à chaque étape deux chaînes de points. Le choix des fusions est guidé par deux hypothèses : (i) un symbole est constitué d'un ensemble de chaînes de points connectés ; (ii) les chaînes de points d'un symbole ont tendance à être convexes. La localisation d'un symbole dans ces documents revient ainsi à chercher les sous-ensembles de chaînes les plus proches parmi ceux créés par les dendrogrammes. Cependant, la méthode proposée ne permet pas de détecter les symboles dont les chaînes ne sont pas connectées, les symboles qui ne possèdent pas de points de jonction dans le document et ne permet pas de faire face au cas où deux symboles partagent une même chaîne de points.

L'approche proposée par (Qureshi *et al.*, 2008) suit le même principe : le document graphique est aussi décomposé en segments et représenté par un graphe dont les nœuds sont des *quadrilatères* (issus de l'étape de vectorisation, correspondant aux segments) et chaque arc relie deux quadrilatères adjacents. Les arcs sont étiquetés par le type de connexion existant entre deux *quadrilatères* et pondérés par l'angle relatif et le rapport de longueurs entre les deux segments adjacents. Afin de localiser des régions d'intérêt qui contiennent des symboles, les auteurs proposent de définir un poids pour chaque nœud du graphe. Celui-ci est calculé en fonction de 6 hypothèses de la composition d'un symbole. Ainsi, les régions qui intègrent des nœuds connectés possédant des poids élevés (supérieur à un seuil  $T_s$ ) sont extraites afin de générer un ensemble de régions pouvant potentiellement contenir des symboles. Lorsqu'une requête d'un symbole est effectuée, la mise en correspondance entre les (sous-)graphes de ces régions et celui de la requête permet directement de trouver les régions contenant des occurrences de la requête dans le document. Une remarque importante concernant cette approche est qu'il faut déterminer un seuil optimal  $T_s$  afin d'avoir des zones d'intérêt correctes. Une valeur trop grande ou trop petite entraîne une mauvaise segmentation conduisant à des erreurs pour la recherche de symboles.

Dans (Wenyin *et al.*, 2007), la vectorisation est utilisée pour extraire des primitives (lignes, arcs) d'un symbole ainsi que du document. Ces primitives sont utilisées pour construire un graphe attribué : les primitives représentent les nœuds du graphe et leurs relations (intersections, parallélisme, perpendicularité, ligne-arcs/cercles) les arcs du graphe. Pour chaque symbole requête, un *arbre-squelette* et un *chemin de traverse* sont extraits du graphe (Fig. 2). Ces derniers sont utilisés pour chercher des occurrences du symbole dans le document. En effet, lors de la localisation du symbole dans un document, chaque primitive contenue dans le *chemin de traverse* du sym-

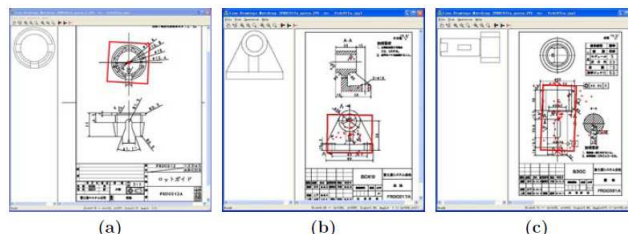
bole est successivement utilisée pour chercher des primitives correspondantes dans le document en vérifiant les contraintes existantes vis-à-vis des primitives précédentes. Ainsi, les hypothétiques occurrences du symbole requête sélectionnées sont celles qui répondent, lors du parcours du *chemin de traverse*, à toutes les contraintes imposées par le symbole. Pour augmenter l'efficacité de la recherche dans le cas de symboles déformés, les auteurs ont proposé d'intégrer un processus de retour de pertinence qui aide à la mise à jour de taux de tolérance utilisés dans l'étape de reconnaissance. Les résultats expérimentaux sont prometteurs. Cependant, la performance du système dépend fortement de l'étape de vectorisation. De plus, cette méthode ne permet pas d'effectuer des recherches sur tous les types de symboles tels que les courbes libres ("*free-curves*") ou les symboles dont le graphe est déconnecté.



**Figure 2.** Représentation du symbole. (a) symbole. (b) primitives du symbole obtenus après une étape de vectorisation. (c) graphe attribué. (d) arbre-squelette du graphe en désignant la primitive 0 comme la racine de l'arbre, son chemin de traverse est donc  $E(0, 5), E(0, 1), E(0, 3), E(0, 2), E(1, 4), E(1, 6), E(2, 7)$  où  $E(i, j)$  désigne l'arc liant deux nœuds  $i$  et  $j$  (extrait de (Wenyin et al., 2007)).

Similairement, Liu et al. (Liu et al., 2009) ont également utilisé des lignes et des courbes comme des primitives pour caractériser les images. Pour chaque primitive, la structure locale est définie par les voisins les plus proches. Elle est caractérisée par les distances et angles relatifs entre cette primitive et les primitives dans le voisinage. La détection de symboles dans l'image est effectuée par un processus de vérification des hypothèses. Tout d'abord, les  $k$  structures locales les plus proches de chaque primitive du modèle sont extraites et les paramètres de transformation correspondant sont estimés. En considérant que chaque estimation est un point dans l'espace de paramètres, une région dense dans cette espace signifie qu'il existe une occurrence du modèle dans l'image. De telles régions sont détectées et passées par une étape de vérification pour localiser les occurrences du modèle dans l'image. Les résultats obtenus sur quelques images de test montrent que la méthode est efficace (Fig. 3). Cependant, comme les autres méthodes structurales, la vectorisation a un impact important sur la performance de la méthode. Les résultats obtenus sont intéressants à condition que l'image d'entrée soit parfaite.

Dans (Locteau, 2008), les régions sont considérées comme des primitives pour décrire l'image par un graphe d'adjacence inexacte. Le squelette de l'image est utilisé pour construire un graphe d'adjacence dont les nœuds sont des régions (qui sont des composantes connexes, appelées *occlusions*) et dont les arcs relient les régions aux frontières limitrophes. Ces régions sont caractérisées par les pseudo-invariants de Zer-

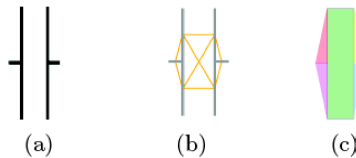


**Figure 3.** Illustration de résultats obtenus par la méthode proposée par (Liu et al., 2009).

nike. Les arcs quant à eux sont étiquetés par des informations liées aux deux régions qu'ils relient. Chaque nœud du graphe peut posséder une étiquette nominale, déterminée grâce à une étape d'apprentissage, pour faciliter l'emploi d'algorithme opérant sur le graphe. À cause des dégradations pouvant exister dans les images, la topologie des graphes construits n'est pas toujours parfaite. Pour faire face à ce problème, des règles supplémentaires définies à partir d'une base d'apprentissage sont utilisées pour corriger les erreurs. Ces dernières permettent de construire un graphe d'adjacence inexacte pour chaque image. La localisation de symboles dans les documents revient à chercher des isomorphismes de sous-graphes, un problème NP-complet. Pour le résoudre, l'auteur propose d'utiliser une approche d'extension d'isomorphisme partiel explorant un espace état-transition. Bien que cette méthode donne de bonnes performances pour la reconnaissance et la détection de symboles, elle possède néanmoins certaines limites dues à l'utilisation du graphe et à la squelettisation de l'image. Ainsi, le graphe défini ne permet pas de traiter les symboles qui ont des composantes déconnectées ou qui n'ont pas de composantes connexes. De plus, cette méthode ne permet pas de distinguer certaines classes de symboles ayant des graphes isomorphes. Enfin, la performance de cette méthode repose largement sur la squelettisation, elle est donc sensible à la connexité des objets même sur des images non-bruitées et il peut rester toujours des problèmes de régions sur-segmentées malgré l'application de pré-traitements intégrés. Utilisant également le graphe d'adjacence pour représenter un document, Bodic et al. (Bodic *et al.*, 2009) ont modélisé le problème de recherche des isomorphismes de sous-graphes en problème de programmation linéaire en nombres entiers. Le taux de détection obtenu est assez élevé. Cependant, comme les mêmes primitives que (Locteau, 2008) sont utilisées pour construire des graphes d'adjacence, cette méthode a des difficultés similaires avec celle présentée précédemment.

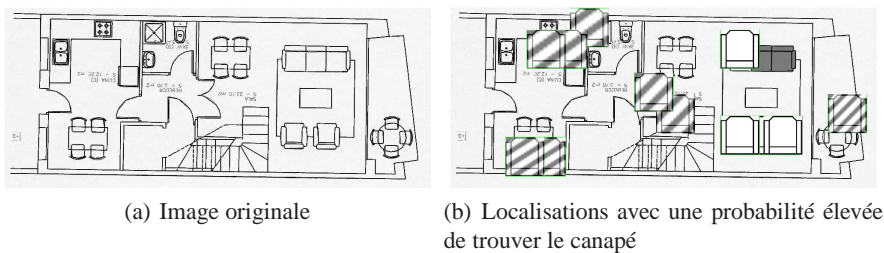
Avec la même idée de regroupement des primitives pour construire des symboles, (Locteau *et al.*, 2007) propose une approche perceptuelle basée sur la théorie de Gestalt et sur un graphe de visibilité. Il s'agit de redéfinir les régions caractérisant l'objet en regroupant des segments dans un graphe de visibilité pour obtenir des polygones convexes pouvant couvrir le symbole et caractérisant le plus fidèlement possible la structure de celui-ci. La représentation par un graphe de regroupements perceptuels résout les cas non-représentables ou non-différenciables du graphe d'adjacence inexacte proposé par le même auteur. La Fig. 4 présente quelques exemples de graphes

de visibilité et de régions segmentées correspondantes. Cette approche donne de bons résultats pour la reconnaissance de symboles, même pour les symboles n'ayant pas de contours fermés. Cependant, le problème de la localisation de symboles reste une tâche complexe en raison de la grande connexité du graphe. De plus, ce graphe est sensible à la qualité des données vectorielles.



**Figure 4.** (a) Un symbole non-traité avec le graphe d'adjacence inexacte. (b) Graphe de visibilité. (c) Régions (occlusions) segmentées à partir de graphe de visibilité (reprise de (Locteau, 2008)).

Dans la même veine, Rusinol et al. (Rusinol *et al.*, 2006) déterminent sur un document vectorisé des régions d'intérêt à partir des coordonnées de l'ensemble des segments connectés. Les régions d'intérêt sont définies comme des fenêtres dynamiques déterminées à partir des coordonnées maximales et minimales des segments adjacents. Une signature vectorielle est proposée pour la représentation de chaque région. Dans un premier temps, les régions non-pertinentes (tels que le mur et le câblage) sont facilement éliminées. Un vote est ensuite effectué pour sélectionner parmi les fenêtres détectées celles pouvant contenir le symbole requête. Afin de diminuer l'effet des erreurs de vectorisation, le document est repassé en basse résolution lors de la détection de régions d'intérêt. Les évaluations sur des documents réels montrent que les occurrences du symbole requête sont bien localisées malgré l'obtention de beaucoup de faux positifs (Fig. 5). Ces faux positifs sont causés par l'erreur de détection des régions d'intérêt et par le manque d'informations dans la base de signatures vectorielles. Un autre point négatif de cette approche, qui est tributaire de la qualité des résultats de la vectorisation, est que seuls les symboles connectés ont la possibilité d'être localisés.

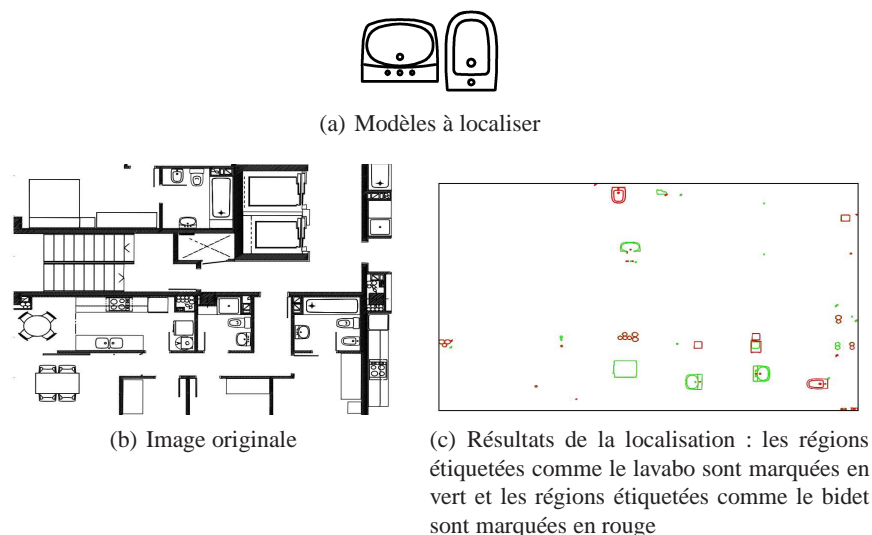


**Figure 5.** Exemple d'un résultat de la méthode proposée par (Rusinol et al., 2006).

Afin d'améliorer cette méthode, (Rusinol *et al.*, 2007) propose de localiser rapidement des régions qui contiennent probablement un symbole similaire à une requête sans étape préalable de segmentation ou d'apprentissage. Une analyse en composantes connexes est effectuée sur le document entier pour trouver des régions fermées qui sont ensuite représentées par des polygones d'approximation. L'auteur propose d'utiliser l'ovale de Cassini pour construire une fonction de hachage de façon à indexer



le document. Un polygone est représenté par un couple  $(a, b)$ , où  $a, b$  sont deux paramètres géométriques de l'ovale de Cassini couvrant le polygone. Les couples  $(a, b)$  (discretisés) sont utilisés comme entrées d'une table de hachage pour indexer tous les polygones ayant ces mêmes valeurs. Les régions dans le document contenant probablement des occurrences du symbole requête sont détectées par un système de vote qui se base sur l'identification de primitives (polygones) dans le document similaires à celles de la requête grâce à la table de hachage. L'approche n'a pas besoin d'une étape d'apprentissage ou bien de segmentation préalable mais elle se base sur une analyse en composantes connexes. Un défaut de cette approche est qu'elle favorise les faux positifs (Fig. 6). En effet, les primitives dans une entrée de la table de hachage peuvent ne pas être toujours similaires. De plus, cette approche ne prend pas en compte l'organisation structurelle du symbole et elle ne s'applique qu'aux symboles contenant des composantes connexes dont les contours sont fermés.



**Figure 6.** Exemple des résultats obtenus par la méthode de (Rusinol et al., 2007).

Basée sur ce même principe d'utiliser des régions fermées comme primitives pour indexer et localiser des symboles dans le document, (Rusinol *et al.*, 2009b) propose d'utiliser une chaîne de segments adjacents du contour de chaque région pour identifier et indexer les régions. La distance entre deux régions, représentées par la chaîne  $A$  et  $B$ , est mesurée par le coût de la transformation de la chaîne  $A$  en la chaîne  $B$ . La table de hachage est aussi utilisée pour augmenter la vitesse de recherche des primitives similaires. La chaîne médiane d'une entrée de la table est prise comme index d'accès. Ainsi, un nombre élevé de primitives similaires à celles du symbole requête indique l'existence d'une occurrence du symbole dans le document. Bien que cette approche offre une faible précision de localisation dans les documents, le rappel est lui élevé et le classement des réponses obtenues est généralement correct. Un avantage de cette méthode est que, grâce à la fonction de distance définie pour la mise en correspondance entre deux chaînes, elle s'avère tolérante au problème de fragmentation

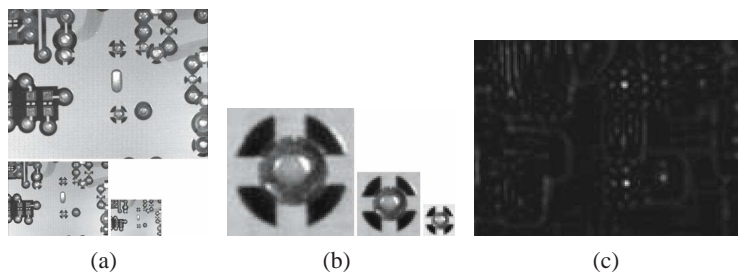
de segments lors de la construction des polygones d'approximations de chaque région. Néanmoins, en raison du fait que seules des régions fermées soient prises comme primitives, la méthode ne permet pas de traiter les cas de symboles dont le contour est ouvert, ni les symboles partiellement visibles.

Pour conclure sur les approches structurelles, nous pouvons constater que ces approches permettent de localiser pertinemment des symboles même dans des cas complexes (cf. Fig. 3). Cependant, elles nécessitent généralement des étapes intermédiaires, telles que la vectorisation ou la segmentation, pour décomposer les images en primitives. La performance de ces approches dépend ainsi fortement des résultats de ces étapes intermédiaires. Une erreur dans ces étapes peut entraîner des altérations dans la représentation des documents (par exemple, le fait de se retrouver avec trop de nœuds et de relations dans les graphes) et ainsi nuire aux résultats de la localisation. De plus, elles nécessitent souvent de poser des contraintes et des hypothèses sur les symboles considérés telles que la connectivité, la convexité ou la fermeture du symbole. Enfin, elles nécessitent de faire face à la complexité de la mise en correspondance entre (sous-)graphes qui est un problème NP-complet. En effet, dans ces approches le symbole et les documents sont souvent représentés par des graphes. Nous présentons dans la suite des approches ne nécessitant pas de traitements préalables, et qui imposent peu ou pas du tout d'hypothèses sur les symboles.

### 3. Approches pixelaires

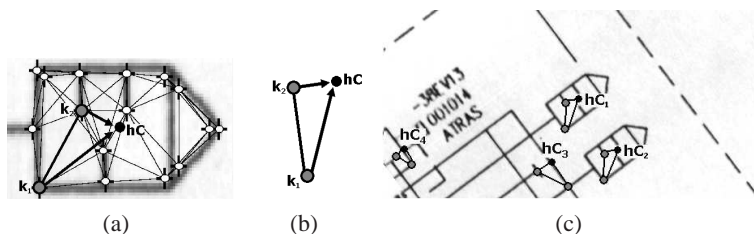
MacLean et Tsotsos (MacLean *et al.*, 2008) proposent une méthode de localisation et de reconnaissance rapide de symboles dans le document qui ne nécessite aucune étape de pré-traitement ou de décomposition. Cette méthode est basée sur la représentation pyramidale de l'image. Ainsi, une représentation pyramidale correspondant au modèle requête et une autre correspondant à l'image sont construites (Fig. 7(a), 7(b)). La première étape de la recherche des positions potentielles des occurrences du symbole requête consiste à chercher des maxima sur la surface de corrélation normalisée et calculée à partir des sommets des deux pyramides (Fig. 7(c)). Ces positions sont ensuite propagées, niveau par niveau, vers la base de la pyramide. Les positions exactes des occurrences sont trouvées au niveau le plus bas. Afin de déterminer le nombre maximal de niveaux de la pyramide correspondant à chaque modèle, les auteurs proposent d'utiliser une analyse du pire des cas. La méthode proposée localise rapidement et précisément les positions du symbole requête mais elle est sensible au changement de densité du fond de l'image et ne permet pas de vérifier les propriétés d'invariances à la rotation et au changement d'échelle.

Partant de la même idée de ne pas segmenter l'image, (Rusinol *et al.*, 2008) propose, pour localiser des symboles ou des mots dans des documents techniques, d'effectuer directement un système de vote sur l'image entière en validant l'organisation spatiale des points d'intérêt appariés. Chaque paire de points d'intérêt du document est mise en correspondance avec une paire de points du modèle. Ces mises en correspondance sont utilisées pour définir, au travers d'un système de vote, le centre hypo-



**Figure 7.** Représentation pyramidale (a) de l'image et (b) du modèle. (c) Surface de corrélation au niveau le plus haut de l'image, les deux extrema les plus forts (blancs) correspondent aux positions de deux occurrences du modèle (reprise de (MacLean et al., 2008)).

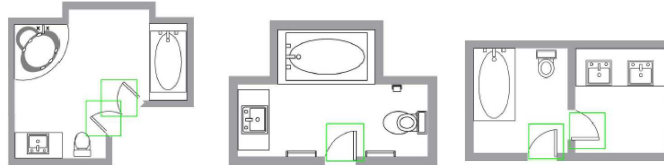
thétique du symbole dans le document. Ce centre hypothétique est déterminé de sorte que la relation spatiale existante entre ce centre et deux points considérés dans le document soit similaire à celle existante entre le centre et deux points d'intérêt appariés du modèle requête (Fig. 8). Afin d'éviter une explosion de traitements inutiles des paires de points, pour chaque point d'intérêt, seuls les cinq voisins les plus proches sont considérés pour former un couple (voir Fig. 8(a)). Bien que la méthode n'ait été testée que sur une petite base de documents et fournit des résultats non parfaits, la combinaison de descriptions locales et d'informations géométriques fournit une bonne discrimination pour la localisation et la reconnaissance de symboles.



**Figure 8.** (a) Graphe de proximité entre les points d'intérêt du modèle en considérant les 5 plus proches voisins. (b) Relation spatiale existante entre 2 points d'intérêt et le centre du modèle. (c) sous-configurations avec leurs centres hypothétiques ( $hC_1, hC_2, hC_3, hC_4$ ) (extraite de (Rusinol et al., 2008)).

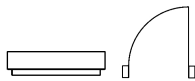
Pour représenter le symbole dans les documents, (Escalera *et al.*, 2009) propose le descripteur CBSM (*Circular Blurred Shape Model*). Celui-ci se base également sur la distribution des points de contour autour des centroïdes locaux. Une étape d'apprentissage est effectuée pour définir les paramètres optimaux du descripteur. Ce descripteur est ensuite utilisé pour vérifier si des régions locales du document contiennent des occurrences du symbole requête. Ces régions ne sont pas segmentées a priori mais dynamiquement par une fenêtre glissante qui parcourt le document (avec un pas de déplacement de 5 pixels). La taille de la fenêtre varie selon un intervalle prédéfini. Cette technique classique de localisation est simple et permet d'utiliser n'importe quel descripteur de formes pour la reconnaissance de symbole. Quelques résultats sont montrés

en Fig. 9. Néanmoins, un point crucial de cette approche concerne le choix des valeurs pour les paramètres liés à la taille de la fenêtre et au pas de déplacement. Le symbole peut ne pas être trouvé si sa taille est éloignée des tailles choisies pour la fenêtre. De plus, le nombre de régions de test peut devenir très important pour une base de documents, en particulier dans le contexte d'images de grandes tailles.



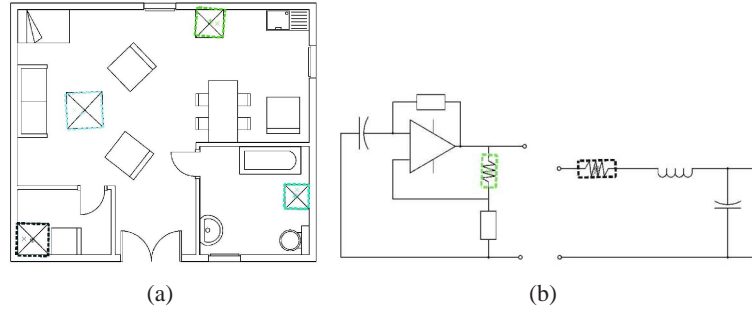
**Figure 9.** Quelques exemples de localisation de portes dans les documents de la méthode proposée par (Escalera et al., 2009).

Récemment, (Nguyen *et al.*, 2009) proposent également une approche pixelaire pour le problème de localisation de symboles dans les documents graphiques. Les documents sont représentés par les informations extraites aux points d'intérêt à différentes résolutions. Un ensemble de mots visuels est construit par regroupement de descriptions similaires extraites des documents. Chaque document graphique est donc représenté par des mots visuels grâce à un appariement multiple (une description locale mise en correspondance avec plusieurs mots). Des techniques de recherche d'informations textuelles sont ensuite utilisées pour indexer et localiser des symboles. Lorsqu'un symbole requête est recherché dans un document, les descriptions locales du symbole sont calculées et mises en correspondance avec les mots visuels. Des régions candidates pouvant contenir des occurrences du symbole requête sont identifiées par analyse de la relation spatiale entre la requête et deux points (l'un du document, l'autre de la requête) partageant le même mot visuel. Le modèle vectoriel est ensuite adapté pour déterminer la similarité entre la requête et les régions candidates afin de trouver les régions contenant les occurrences du symbole. L'approche proposée donne de bonnes performances sur des documents synthétiques (Fig. 11). Elle n'impose aucune hypothèse sur les symboles traités telle que la connexité, la convexité ou la fermeture. Elle permet donc de localiser des symboles ouverts ou déconnectés. Ni étapes préalable de squelettisation, ou de vectorisation sont nécessaires. Cependant, les symboles ayant peu de points d'intérêt discriminant (spécialement dans le cas de connexion avec d'autres parties dans le document) sont difficiles à localiser (Fig. 10).



**Figure 10.** Symboles manquants de points d'intérêt discriminants, et / ou la plupart se trouvent près de connexions externes.

Bien que peu de travaux se soient encore intéressés à cette catégorie d'approches, les approches pixelaires présentent néanmoins de nombreux avantages par rapport aux approches structurales. En effet, les approches pixelaires n'ont pas besoin d'une étape préalable de vectorisation ou de segmentation. De plus, elles n'imposent pas



**Figure 11.** Exemples de résultats de localisation de symboles par l'approche proposée dans (Nguyen et al., 2009)

de fortes contraintes sur les symboles considérés. Enfin, ces approches permettent d'éviter d'avoir à faire face au problème de l'appariement entre (sous-)graphes.

#### 4. Mesures d'évaluation

Considérées comme des systèmes de recherche d'information, les méthodes de localisation (de textes ou de symboles) sont souvent évaluées par des mesures classiques de ce domaine tels que la précision, le rappel et le *F-score* (Marcus, 1992, Rath et al., 2003, Tabbone et al., 2007, Valveny et al., 2007, Locteau, 2008). Ces valeurs sont calculées par analyse des réponses obtenues pour une requête. Une réponse est soit pertinente soit non-pertinente. Soit une base de données contenant un ensemble d'éléments  $X$  et une requête  $i$  consistant à chercher les occurrences d'un symbole dans  $X$ , nous notons  $X_p$  l'ensemble des éléments pertinents et  $X_n$  l'ensemble des éléments non-pertinents par rapport à  $i$ ,  $X_p, X_n \subset X$ . Soit  $Ret$  l'ensemble des éléments obtenus pour la requête  $i$ , les formules [1] définissent la précision et le rappel du résultat de la recherche. La précision  $P$  mesure la capacité du système à fournir des réponses pertinentes. Elle est définie par le rapport entre le nombre d'éléments pertinents trouvés et le nombre total d'éléments récupérés. Le rappel  $R$  est défini par le rapport entre le nombre d'éléments pertinents trouvés et nombre total d'éléments pertinents dans la base. Il mesure la capacité du système à trouver exhaustivement les éléments pertinents existants.

$$P = |Ret \cap X_p|/|Y|; \quad R = |Ret \cap X_p|/|X_p| \quad [1]$$

La plupart des systèmes de recherche d'informations ne fournissent pas simplement un ensemble d'éléments a priori pertinents, mais fournissent aussi un classement de ces éléments. Afin de prendre en compte ce classement, une approche consiste à ne considérer la précision que pour les  $n$  éléments ( $P_n$ ) définis comme les plus pertinents par le système. Pour fournir une vue d'ensemble sur la précision et le rappel, la courbe *précision/rappel*  $P(r)$  est souvent utilisée.  $P(r)$  désigne la précision du résultat lorsque le rappel atteint  $r$ . En pratique, la courbe est obtenue en calculant  $P_n$  pour plusieurs valeurs de  $n$  ( $n = 1, 2, 3, \dots$ ) et en calculant le rappel associé. Une autre

mesure pour donner une vue d'ensemble sur les performances d'un système consiste à combiner la précision et le rappel en une valeur unique, tels que le *F-score* (Shaw *et al.*, 1997), la *E-mesure* (van Rijbergen, 1979) (Eq. [2]), ou la *précision moyenne*.

$$F = \frac{2}{1/R + 1/P} ; \quad E = 1 - \frac{1 + b^2}{b^2/R + 1/P} \quad [2]$$

où  $b$  est un paramètre qui permet de définir l'importance relative de la précision par rapport au rappel. Si  $b = 1$ , la *E-mesure* est le complément du *F-score*. Une valeur élevée du *F-score* démontre que le système permet de fournir un bon compromis entre la précision et le rappel.

Il n'existait pas encore dans la littérature de mesures d'évaluation plus spécifiques au domaine de la localisation de symboles. Ainsi Rusiñol et Lladós (Rusiñol *et al.*, 2009a) ont proposé un ensemble de mesures pour évaluer les performances des systèmes de localisation de symboles en termes de capacité de reconnaissance, de précision de localisation et d'adaptation à de grandes bases de données. Ces mesures sont des adaptations de mesures classiques du domaine de la recherche d'informations. Elles se basent sur l'aire de recouvrement entre la région d'intérêt du résultat et celle de la vérité terrain. La région d'intérêt d'un symbole est déterminée par son enveloppe convexe.

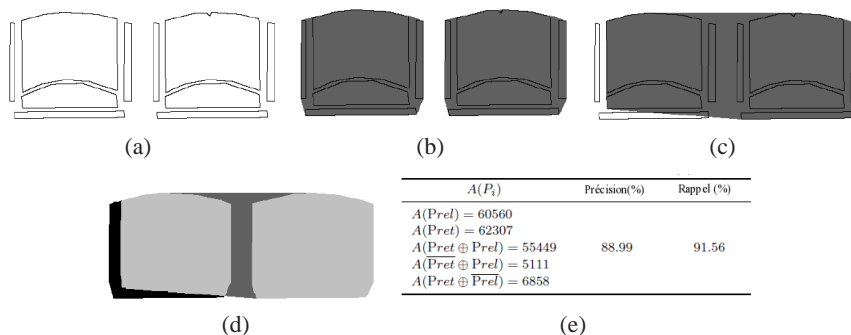
Étant donnée une collection de documents graphiques,  $P_{tot}$  désigne l'ensemble des polygones contenus dans l'ensemble des documents dans cette collection,  $P_{rel}$  l'ensemble des polygones de la vérité terrain où une occurrence du symbole  $S$  se trouve. Lors de la localisation du symbole  $S$  dans la collection,  $P_{ret}$  désigne l'ensemble des polygones récupérés. La précision et le rappel sont donc définis par :

$$P_A = A(P_{ret} \oplus P_{rel}) / A(P_{ret}) ; \quad R_A = A(P_{ret} \oplus P_{rel}) / A(P_{rel}) \quad [3]$$

avec  $A(P_i)$  la somme des aires de tous les polygones de l'ensemble  $P_i$ . L'opération  $P_i \oplus P_j$  fournit l'ensemble des polygones obtenus par l'intersection spatiale entre les polygones de l'ensemble  $P_i$  et ceux de l'ensemble  $P_j$  (Fig. 12). D'autres mesures d'évaluation sont également reformulées à partir de  $P_A$  et de  $R_A$ . Par exemple, le *F-score* est redéfini par :

$$F_A = \frac{2}{1/R_A + 1/P_A} \quad [4]$$

Ces mesures permettent d'évaluer les performances d'un système de localisation de symboles. Cependant, elles tendent à sur-évaluer la précision des systèmes dans le cas où un résultat fourni par ces derniers couvrent plusieurs zones de vérité terrain (comme dans l'exemple présenté Fig. 12(d)). En effet, l'objectif de la localisation de symboles est de fournir des régions où le symbole existe. Ainsi, si une réponse couvre plus d'une région de vérité terrain, une seule région parmi ces régions doit être considérée comme pertinente à la fois. Or, dans les cas comme celui montré en Fig. 12(c)



**Figure 12.** La précision et le rappel de la localisation. (a) document original. (b) vérité terrain ( $P_{rel}$ ). (c) résultat obtenu ( $P_{ret}$ ). (d) polygones de chevauchement du résultat et de la vérité terrain :  $P_{ret} \oplus P_{rel}$  correspond à deux zones gris clair. (e) Calcul de la précision et du rappel de la localisation (extraite de (Rusinol et al., 2009a)).

où un résultat couvre plusieurs régions en même temps, la précision calculée par la formule [3] est très proche de 1 lors que le symbole n'a pas été détecté correctement. Ceci soulève le problème de la pertinence de l'adéquation d'une requête aux résultats proposés par le système.

Concernant la base de vérité terrain, les seules bases disponibles à notre connaissance pour le problème de la localisation de symboles sont celles qui ont été définies au cours du projet SESYD<sup>2</sup> (Delalandre *et al.*, 2009). Malheureusement, ces bases de données ne peuvent être facilement utilisées pour calculer les mesures de performance proposées par (Rusinol *et al.*, 2009a). En effet, les mesures de performance proposées par (Rusinol *et al.*, 2009a) doivent être calculées au niveau du pixel mais cette précision est peu compatible avec les bases du projet SYSED où les régions pertinentes pour un symbole sont définies à partir de son rectangle englobant. Ainsi, le manque de bases de vérité terrain et de mesures d'évaluation communes expliquent que les méthodes de localisation n'aient pu être évaluées que sur des bases réduites d'images. Le problème reste donc ouvert et la communauté (cf. campagne d'évaluation ÉPEIRES<sup>3</sup>) se pose des questions dans ce sens sur la définition d'une vérité terrain. Dans quelle mesure peut-on dire qu'une réponse à une requête est bonne ?

## 5. Conclusion

Pour les images de documents graphiques, il semble que les approches structurales soient plus adaptées car les lignes et les formes de base sont les primitives principales pour construire le contenu de l'image en respectant des règles de combinaison, de connexion, etc. Intuitivement, ces approches permettent d'effectuer des localisa-

2. Systems Evaluation SYnthetic Documents

3. Évaluation de PErformances à appliquées à la REcognition de Symboles, site : <http://epeires.loria.fr>

tions précises et correctes si les primitives sont correctement extraites. Néanmoins, l'extraction des primitives d'un document exigent souvent des étapes préalables tels que : la vectorisation, l'approximation de polygones, la décomposition en segments, qui sont sensibles aux bruits. Les erreurs dans ces étapes peuvent causer des erreurs importantes dans les étapes postérieures. Donc des traitements supplémentaires sont nécessaires. De plus, utiliser des graphes pour représenter les documents convertit le problème de localisation en problème d'appariement de (sous-)graphes, qui est connu comme un problème NP-complet. En outre, les méthodes se basant sur les graphes nécessitent des hypothèses préalables sur les symboles pour les construire en regroupant des primitives. Ces hypothèses limitent les types de symboles traités. A l'opposé, les approches pixelaires ne sont pas confrontées au problème NP-complet d'appariement, ni au problème de vectorisation ou de segmentation en primitives. Généralement, elles n'ont pas de contraintes particulières sur les symboles traités. Cependant, les positions des symboles localisés par les approches pixelaires sont moins précises qu'avec les approches structurelles. Pour tirer au mieux partie des avantages des deux approches, l'idéal serait de les combiner au travers de nouvelles méthodes qui seraient sans doute plus robustes et plus précises.

## 6. Bibliographie

- Bodic P. L., Locteau H., Adam S., Heroux P., Lecourtier Y., Knippel A., « Symbol Detection Using Region Adjacency Graphs and Integer Linear Programming », *ICDAR*, IEEE Computer Society, Washington, DC, USA, p. 1320-1324, 2009.
- Delalandre M., Valveny E., Pridmore T., Karatzas D., « Generation of Synthetic Documents for Performance Evaluation of Symbol Recognition & Spotting Systems », *IJDAR*, 2009.
- Dosch P., Lladós J., « Vectorial Signatures for Symbol Discrimination », *Graphics Recognition*, Springer Berlin / Heidelberg, 2004.
- Escalera S., Fornés A., Pujol O., Escudero A., Radeva P., « Circular Blurred Shape Model for Symbol Spotting in Documents », *ICIP*, Novembre, 2009.
- Fonseca J., Ferreira A., Joaquim J., « Content-based retrieval of technical drawings », *IJCAT*, vol. 23, n° 2-3, p. 86-100, March, 2005.
- Liu R., Wang Y., Baba T., Masumoto D., « Shape Detection from Line Drawings by Hierarchical Matching », *CAIP*, vol. 5702/2009, Springer Berlin / Heidelberg, p. 922-929, 2009.
- Locteau H., Contribution à la localisation de symboles dans les documents graphiques, PhD thesis, Université de Rouen, 2008.
- Locteau H., Adam S., Trupin E., Labiche J., Heroux P., « Symbol Spotting Using Full Visibility Graph Representation », *International Workshop on Graphics Recognition*, Curitiba, Brazil, September, 2007.
- MacLean W. J., Tsotsos J. K., « Fast pattern recognition using normalized grey-scale correlation in a pyramid image representation », *Machine Vision and Application*, vol. 19, n° 3, p. 163-179, May, 2008.
- Marcus J. N., « A novel algorithm for HMM word spotting performance evaluation and error analysis », *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, IEEE Computer Society, Los Alamitos, CA, USA, p. 89-92, 1992.



Thi-Oanh NGUYEN and Salvatore TABBONE

- Messmer B. T., Bunke H., « Automatic Learning and Recognition of Graphical Symbols in Engineering Drawings », *Selected Papers from the First International Workshop on Graphics Recognition, Methods and Applications*, Springer-Verlag, London, UK, p. 123 - 134, 1995.
- Nguyen T.-O., Tabbone S., Boucher A., « A Symbol Spotting Approach Based on the Vector Model and a Visual Vocabulary », *ICDAR*, Barcelona, Spain, July, 2009.
- Park B. G., Lee K. M., Lee S. U., Lee J. H., « Recognition of partially occluded objects using probabilistic ARG (attributed relational graph)-based matching », *CVIU*, vol. 90, n° 3, p. 217-241, 6, 2003.
- Park S. H., Lee K. M., Lee S. U., « A Line Feature Matching Technique Based on an Eigenvector Approach », *CVIU*, vol. 77, n° 3, p. 263-283, March, 2000.
- Qureshi R. J., Ramel J.-Y., Barret D., Cardot H., « Spotting Symbols in Line Drawing Images Using Graph Representations », *Graphics Recognition. Recent Advances and New Opportunities*, Springer-Verlag, Berlin, Heidelberg, p. 91-103, 2008.
- Rath T. M., Manmatha R., « Features for Word Spotting in Historical Manuscripts », *ICDAR*, IEEE Computer Society, Washington, DC, USA, p. 218, 2003.
- Rusinol M., Lladós J., « Symbol Spotting in Technical Drawings Using Vectorial Signatures », *Graphics Recognition. Ten Years Review and Future Perspectives*, vol. 3926/2006, Springer Berlin / Heidelberg, p. 35-46, October, 2006.
- Rusinol M., Lladós J., « A Region-Based Hashing Approach for Symbol Spotting in Technical Documents », *International Workshop on Graphics Recognition*, Curitiba, Brazil, September, 2007.
- Rusinol M., Lladós J., « Word and Symbol Spotting Using Spatial Organization of Local Descriptors », *International Workshop on Document Analysis Systems (DAS'08)*, Nara, Japan, September, 2008.
- Rusinol M., Lladós J., « A Performance Evaluation Protocol for Symbol Spotting Systems in Terms of Recognition and Location Indices », *IJDAR*, vol. 12, n° 2, p. 83-96, July, 2009a.
- Rusinol M., Lladós J., Sánchez G., « Symbol Spotting in Vectorized Technical Drawings Through a Lookup Table of Region Strings », *Pattern Analysis and Applications*, 2009b.
- Shaw W. M. J., Burgin R., Howell P., « Performance standards and evaluations in IR test collections : Cluster-based retrieval models », *Information Processing & Management*, vol. 33, n° 1, p. 1-14, January, 1997.
- Tabbone S., Zuwala D., « An indexing method for graphical documents », *ICDAR*, vol. 2, Curitiba, Brazil, p. 789-793, 2007.
- Valveny E., Dosch P., Winstanley A., Zhou Y., Yang S., Yan L., Wenyin L., Elliman D., Delalandre M., Trupin E., Adam S., Ogier J.-M., « A general framework for the evaluation of symbol recognition methods », *IJDAR*, vol. 9, n° 1, p. 59-74, 2007.
- van Rijbergen C. J., *Information Retrieval*, Butterworths, 1979.
- Wenyin L., Zhang W., Yan L., « An interactive example-driven approach to graphics recognition in engineering drawings », *IJDAR*, vol. 9, n° 1, p. 13-29, March, 2007.