

Une approche de localisation de symboles non-segmentés dans des documents graphiques

A symbol spotting approach in graphic documents

Thi-Oanh Nguyen^{1 2}, Salvatore Tabbone¹, Alain Boucher^{2 3}

¹ LORIA - Université Nancy 2, Campus scientifique - BP 239 - 54506 Vandœuvre-les-Nancy, France.

² Institut de la Francophonie pour l'Informatique MSI, UMI 209 UMMISCO, Hanoi, Vietnam.

³ IRD, UMI 209 UMMISCO, IRD France Nord, Bondy, F-93143, France. {nguyenth,tabbone}@loria.fr,alain.boucher@auf.org

Manuscrit reçu le 15 décembre 2009

Résumé et mots clés

Dans cet article, nous proposons une méthode de localisation de symboles dans des documents graphiques. Les occurrences du symbole dans un document sont détectées grâce à un processus de vote sur des régions candidates. L'approche repose sur un vocabulaire visuel et afin de réduire la complexité d'appariement d'un symbole avec d'autres nous utilisons le modèle vectoriel et une indexation par un fichier inverse. Cette méthode s'appuie sur un descripteur défini à partir du concept de contexte de forme¹ adapté aux points d'intérêt. Ce descripteur est invariant à la rotation, à la translation et aux changements d'échelles. Les résultats expérimentaux sur la recherche de symboles isolés et sur la localisation de symboles non-segmentés dans le document sont très prometteurs.



Descripteur de symboles, symbole graphique, mot visuel, recherche par le contenu, localisation de symboles.

Abstract and key words

This paper addresses the problem of symbol spotting for graphic documents. We propose an approach where each graphic document is indexed as a text document by using the vector model and an inverted file structure. The method relies on a visual vocabulary built from a shape descriptor adapted to the document level and invariant under classical geometric transforms (rotation, scaling and translation). Regions of interest (ROI) selected with high degree of confidence using a voting strategy are considered as occurrences of a query symbol.

The symbol spotting problem consists in locating all instances of a symbol embedded in documents. The representation of these symbols is not straightforward by using a good shape (symbol) descriptor because they are not isolated from their context. Therefore, a common strategy for symbol spotting consists in decomposing documents into components and in applying a shape descriptor on each of them. A vectorization step is needed for most of the approaches and usually, only symbols which satisfy some conditions are retrieved (eg. convexity, connectivity, closure, ...). Our objective is to tackle the problem from a point of view where neither symbol hypothesis nor vectorization step is needed. First of all, we proposed a descriptor to represent graphic symbols and its extension to document level. Then, we exploit a

1. Traduit de "Shape Context".

technique based on the concept of visual words for indexing graphic documents and for spotting non-segmented symbols into documents. Finally, we introduce a voting process on the detected ROI in order to locate instances of a query symbol.

In order to represent graphic symbols, we propose an adaptive solution based on *shape contexts*. It consists in adapting *shape context* for points of interest. We use the DoG detector to extract the points of interest that are nearby the junctions of object model at different resolutions. The shape context at each point of interest (CFPI) is normalized by the dominant orientation of the point of interest and the mean distance between the point of interest and contour points to make the CFPI invariant under rotation and scaling. Therefore, a symbol s is described by a set of CFPI at the interest points of s . We use also this descriptor to extract the local information in a graphic document by computing the CFPI in the neighbour region of each point of interest. We define the neighbour region for each point of interest according to its resolution.

With the goal of reducing the complexity of on-line matchings (for searching and spotting), we use the concept of visual words for our system thanks to the information pre-computed in the off-line step. A clustering technique is executed on the set of descriptors CFPI, calculated from all documents in the database, to create visual words. As the CFPIs are matched with visual words, we can use indexing and retrieval techniques for text documents on graphic documents. While matching the CFPIs with visual words, we propose to associate one CFPI with several visual words according to its similarities with these visual words. The objective is to reduce the ambiguity for the CFPI which are nearby cluster boundaries.

For spotting the instances of a symbol in a document, first, we detect in the document the ROI corresponding to the query symbol and then we execute a voting process on these regions. The ROI detection is based on the relation between the considered keypoint and the *bounding box* of the query. The centre of each region of interest is voted from the similarity between this region and the query, based on a text retrieval technique (the vector model). This technique uses the appearance frequencies of visual words in each region to make the comparison between the query and ROI in the document. Regions having high values are considered as potential regions containing a symbol instance. We have tested the adaptation of CFPI for graphic symbols on the GREC'03 dataset composed of isolated symbols. This base contains symbols of 50 models with different sizes and orientations. The results obtained with the CFPI and the π -signature are shown. The two descriptors have similar performance while querying with complete symbols. However, the CFPI is more robust when the goal is to retrieve incomplete symbols. For evaluating the spotting system, our tests are executed on a collection of synthetic documents from the SESYD project. In spite of errors, the results are very promising and show the feasibility of our approach.

Symbol descriptor, graphic symbol, visual word, symbol spotting.

1. Introduction

La reconnaissance de formes est un domaine de recherche qui suscite encore beaucoup de travaux [1]. La reconnaissance des symboles y constitue un sous-domaine où beaucoup de progrès ont été faits pour reconnaître des symboles isolés [2]. Les descripteurs de formes proposés peuvent être divisés en deux grandes classes : les uns s'appuient sur la forme entière et les autres sur le contour du symbole. Les descripteurs basés sur le contour incluent les descripteurs de Fourier [3], [4], qui ont été largement utilisés. Les descripteurs basés sur la région prennent en compte tous les pixels de la forme et les méthodes les plus courantes sont basées sur la théorie des moments [5].

Ce papier adresse le problème difficile de la *localisation de symbole*² dans les documents où les symboles ne sont pas isolés de leur contexte. Même s'il y a beaucoup de travaux visant à la définition de bons descripteurs pour la représentation d'un symbole, ils ne peuvent pas être utilisés directement pour le problème de localisation de symboles dans le document. En effet, un descripteur peut fournir de bons résultats de recherche de symboles segmentés mais le passage direct d'une description de symbole au document entier est difficile. Une étape préalable commune à la plupart des approches est de décomposer le document en plusieurs composantes, puis d'appliquer des descrip-

2. Traduit de "symbol spotting"

teurs prédéfinis sur ces composantes. Une étape de vectorisation est souvent appliquée [6]–[9]. Parfois, seulement des symboles satisfaisants certaines conditions (par exemple : la convexité, la connectivité ou la fermeture du symbole) sont considérés [9]–[11].

Dans cette perspective, des auteurs [10], [12] proposent de segmenter les symboles dans des documents pour les localiser par rapport à un symbole requête. Tout d'abord, le document est décomposé en chaînes de points connectés. Puis, ces chaînes sont regroupées pour redéfinir des symboles grâce à la construction d'un dendrogramme. Cependant, les symboles doivent satisfaire deux hypothèses : (i) un symbole est constitué d'un ensemble de chaînes de points connectés; (ii) les chaînes de points d'un symbole doivent être convexes. Avec la même idée de regroupement des primitives pour construire des symboles, une approche perceptuelle basée sur la loi de Gestalt et sur un graphe de visibilité a été récemment proposée [8]. Ces symboles doivent aussi satisfaire des contraintes de fermeture et de convexité. Suivant une approche similaire, Rusinol *et al.* [9] détermine sur un document vectorisé des régions d'intérêt, après avoir filtré des symboles non-pertinents (le mur, câblage, ...) à partir des coordonnées des ensembles de segments connectés. Une signature vectorielle est proposée pour la représentation de chaque région. Avec cette approche qui est tributaire de la qualité des résultats de la vectorisation, seuls des symboles connectés peuvent être localisés. Afin d'améliorer leur méthode, les mêmes auteurs [11] proposent de localiser rapidement des régions qui contiennent probablement un symbole similaire à une requête et ceci sans étape préalable de segmentation ou d'apprentissage. Les régions d'intérêt sont détectées différemment grâce à une indexation par une fonction de hachage. Elles sont obtenues par un système de vote qui se base sur l'identification de primitives (polygones) similaires à la requête. L'approche est très intéressante mais ne s'applique qu'aux symboles contenant des composantes connexes avec des contours fermés. Dans [6], une requête de recherche est décrite par une séquence de contraintes à partir d'un graphe attribué représentant les relations entre les primitives (ligne, arc) d'un symbole. Ensuite, les occurrences de ce symbole sont détectées dans un document en vérifiant au fur et à mesure des hypothèses émises sur les contraintes pour des sous-configurations issues du document. Pour augmenter l'efficacité du système dans le cas de symboles déformés les auteurs ont intégré un processus de retour de pertinence qui aide à la mise à jour de seuils de tolérance utilisés dans l'étape de reconnaissance. Les résultats expérimentaux sont prometteurs mais, la performance du système dépend fortement de l'étape de vectorisation. De plus, si le graphe du symbole est déconnecté, le symbole n'est pas reconnu.

Cet article aborde également ce problème de localisation mais avec un autre point de vue où aucune hypothèse sur le symbole ni de pré-traitements telle la vectorisation sont pré-requis. Notre méthode se base sur des travaux précédents sur la reconnaissance de formes et la recherche d'images [13], [14].

Le schéma général de l'approche est présenté dans la figure 1. D'abord, nos travaux se sont portés sur le choix d'un descripteur. Ce choix est généralement conduit par des exigences liées à l'application telles que : la robustesse aux bruits, la stabilité par rapport aux distorsions, l'invariance aux transformations géométriques ou la tolérance aux occlusions. Dans cette perspective, nous introduisons le principe du *Contexte de Forme* pour des *Points d'Intérêt* (CFPI). Il s'agit d'une adaptation d'un descripteur basé sur les contours, défini dans l'article de Belongie *et al.* [13], aux symboles graphiques. Plus précisément, des points d'intérêt de chaque symbole sont extraits et des informations dans la région au voisinage de ces points sont représentées par des descripteurs de forme. Le choix du descripteur est motivé par la nature des documents graphiques qui sont, en général, en niveaux de gris ou binaires. Par ailleurs, le *contexte de forme* permet de représenter des objets partiellement visibles car il tient compte d'informations locales proche de points de référence et les symboles extraits d'un document graphique sont très souvent partiellement occultés à cause des erreurs de segmentation.

Pour chercher des formes similaires, une mesure de similarité entre les descripteurs doit être définie. Cependant, la comparaison entre deux formes n'est pas toujours simple et elle exige parfois des invariances à quelques transformations. De plus, la dimension du descripteur est souvent élevée, ce qui augmente la complexité lors de la recherche d'objets similaires dans une grande collection. Pour ce faire, les descripteurs sont regroupés en classes pour construire un vocabulaire visuel qui se traduit par un processus d'abstraction. Chaque classe est considérée comme un mot visuel et les *contextes de forme* appartenant à la même classe partagent une information de forme similaire quelque soit le symbole où le point d'intérêt est extrait. Finalement, chaque symbole est décrit par des mots visuels et apparié au symbole requête. L'approche est similaire à celle de [14] où le modèle vectoriel et le fichier inverse sont utilisés pour l'indexation et pour la recherche afin de diminuer les temps de réponse.

Cet article est organisé comme suit. Dans la section 2, nous proposons un descripteur de formes pour la représentation des symboles graphiques en se basant sur le *contexte de forme* avec une extension du descripteur proposé pour la description du document entier. L'approche par mots visuels pour indexer et chercher des documents graphiques est abordée dans la section 3. La section 4 est dédiée au problème de la localisation de symboles dans le document. Un système de vote est construit pour trouver des régions contenant le symbole en se basant sur des mots visuels et la définition de régions candidates. Ensuite, nous présentons une évaluation sur l'adaptabilité du descripteur proposé aux symboles graphiques ainsi que des résultats préliminaires pour le problème de localisation de symboles dans des documents graphiques (section 5). Enfin, dans la section 6, nous donnons des conclusions et des perspectives à notre travail.

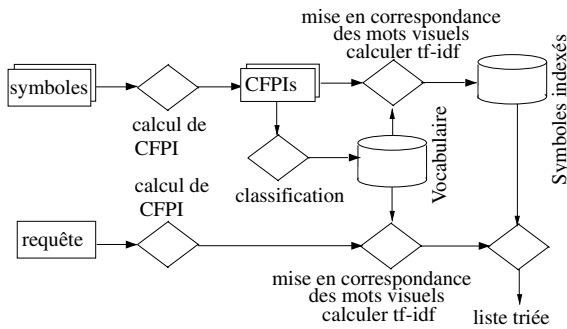


Figure 1. Système de recherche de symboles.

2. Notion de contexte de forme pour les points d'intérêt

Dans cette partie, nous présentons une adaptation du descripteur proposé par Belongie *et al.* [13] qui est invariant à la rotation et aux changements d'échelles. Notre motivation est de réduire la complexité et la redondance des informations définies par le *contexte de forme* en attachant le contexte uniquement aux points d'intérêt.

A. Rappel du contexte de forme

Le *contexte de forme* d'un point de contour p_i d'une forme est déterminé par la distribution des points de contour dans la région au voisinage de p_i [13]. Il s'agit d'un histogramme de fréquences d'apparition des points de contour dans des sous-régions entourant le point p_i , le *point de référence*. Ces sous-régions, appelées des *bins*, sont déterminées en divisant l'espace autour de p_i en plusieurs plages uniformes dans l'espace log-polaire. Cette espace utilise p_i comme origine pour déterminer les coordonnées relatives d'autres points de contour et calculer leurs distributions dans des sous-régions.

Soit $\mathcal{C} = \{p_1, p_2, \dots, p_n\}$ $p_i \in \mathbb{R}^2$, l'ensemble de points échantillonnés des contours externes et internes d'une forme et n le nombre de points de contour, les coordonnées relatives d'un point q par rapport au point p_i sont indiquées par (1).

$$q = (\log(r_{qpi}), \theta_{qpi}), \forall q \neq p_i \wedge q \in \mathcal{C} \quad (1)$$

où r_{qpi} est la distance entre q et p_i , θ_{qpi} est l'angle entre le vecteur $\vec{p_iq}$ et l'axe horizontal. Le *contexte de forme* h_i du point p_i est donc défini par :

$$h_i(l) = \#\{q \neq p_i : (r_{qpi}, \theta_{qpi}) \in bin(l)\}, l = \overline{1, L} \quad (2)$$

où $h_i(l)$ est le nombre de points de contour appartenant à la $l^{\text{ème}}$ classe de l'histogramme et $bin(l) = \{(r_{*pi}, \theta_{*pi}) : r_{*pi} \in [r_l, r_l + \Delta_{r_l}] \wedge \theta_{*pi} \in [\theta_l, \theta_l + \Delta_{\theta_l}]\}$, $\Delta_{r_l}, \Delta_{\theta_l}$ désignent la taille du $bin(l)$. Un objet \mathcal{O} est ainsi décrit comme l'ensemble des contextes de forme des points de contour :

$$\mathcal{O} \equiv \{h_i | p_i \in \mathcal{C}\} \quad (3)$$

Le *contexte de forme* décrit ci-dessus n'est pas invariant à la rotation et aux changements d'échelles. Pour obtenir l'invariance aux changements d'échelles, les distances radiales sont normalisées par la distance moyenne α des n^2 paires de points de la forme [13]. Les auteurs ont également proposé d'utiliser le vecteur tangent associé à chaque point au lieu de l'axe absolu horizontal pour que le *contexte de forme* soit aussi invariant à la rotation.

B. Contexte de forme pour les points d'intérêt (CFPI)

Comme le *contexte de forme* est un descripteur qui contient des informations sur la configuration locale associée à chaque point de contour, il semble bien adapté aux symboles graphiques auxquels nous nous intéressons. Cependant, une représentation d'un symbole en utilisant des *contextes de forme* de tous les points de contour engendre de la redondance, une grande complexité de calcul des descripteurs et aussi d'appariement au niveau des symboles lors de la recherche de symboles similaires.

Des travaux ont montré qu'un objet peut être efficacement localisé à partir de ses points d'intérêt [14]–[17]. Ainsi, dans notre contexte, nous proposons de retenir seulement des informations au niveau des points caractéristiques appelés points clés ou points d'intérêt pour la description d'un symbole.

Détection des points d'intérêt : Beaucoup de méthodes ont été proposées pour détecter les points d'intérêt d'une image [18]–[20]. Nous avons choisi le détecteur DoG (Difference of Gaussian) [15] où les points d'intérêt sont considérés comme des extrema existant dans une pyramide d'échelles construite par la convolution d'une image I avec des filtres de différences de Gaussienne (voir (4)) de tailles différentes. D'après l'évaluation de Mikolajczyk *et al.* [19], le descripteur SIFT (Scale Invariant Feature Transform [15]) défini aux points détectés par DoG donne de très bons résultats pour la recherche d'images par le contenu. De plus, dans [20], l'analyse du comportement du LoG (Laplacian of Gaussian) a montré qu'il fournit à proximité des jonctions un ou plusieurs extrema. Ces derniers jouent un rôle important pour distinguer un modèle d'un autre, surtout pour les symboles graphiques. Les positions des extrema, plus loin ou plus proche des jonctions, sont dépendantes de la résolution où le LoG est calculé (valeur de σ de LoG). Ici, nous étendons ce raisonnement aux documents graphiques, en utilisant le détecteur DoG qui est une approximation du LoG, aux différents niveaux de résolution pour détecter des extrema.

L'utilisation de ce détecteur permet d'éviter le choix arbitraire d'une résolution à traiter et d'avoir des extrema pour caractériser les éléments du document à plusieurs résolutions :

$$D(x, y, \delta) = (G(x, y, k\delta) - G(x, y, \delta)) * I(x, y) \quad (4)$$

Donc, un point d'intérêt ne caractérise l'élément du document à qu'une seule résolution. Des points à différentes localisations caractérisent l'information à plusieurs résolutions, les résolutions les plus grandes étant caractérisées par les points les plus éloignés.

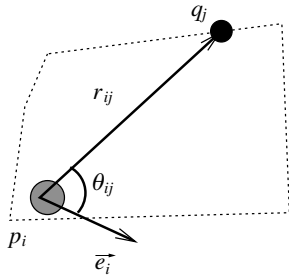


Figure 2. Coordonnées relatives de q_j par rapport à p_i .

Descripteur CFPI : Supposons $\mathcal{IP} = \{p_1, p_2, \dots, p_N\}$ l'ensemble des points d'intérêt et $\mathcal{C} = \{q_1, q_2, \dots, q_n\}$ l'ensemble des points de contour de l'objet. Chaque point dans \mathcal{IP} est considéré comme un point de référence pour calculer le *contexte de forme* correspondant. Pour que l'objet soit bien représenté quelles que soient son orientation et sa taille, le descripteur doit être invariant à la rotation et aux changements d'échelles, les coordonnées relatives des points de contour doivent ainsi être normalisées. De par l'utilisation d'un point d'intérêt comme point de référence pour calculer le descripteur, plutôt qu'un point de contour, il devient nécessaire d'adapter l'étape de normalisation liée à l'orientation car les points d'intérêt ne correspondent pas aux points de contour [20], c'est-à-dire $\mathcal{IP} \not\subseteq \mathcal{C}$. L'utilisation du vecteur tangent [13] n'est plus applicable ici. Dans notre cas, nous proposons d'utiliser l'orientation dominante du point d'intérêt comme axe des abscisses lors du calcul des coordonnées relatives. Chaque point d'intérêt p_i est localisé par ses coordonnées, la résolution où il est détecté (δ_i) et son orientation dominante (\vec{e}_i) qui forme avec l'axe horizontal un angle θ_i :

$$p_i = \{x_i, y_i, \delta_i, \theta_i\} \quad (5)$$

Les coordonnées log-polaires relatives d'un point de contour $q_j \in \mathcal{C}$ dans l'équation (1) sont réécrites comme suit :

$$q_j^{p_i} = (\log(r_{ij}), \theta_{ij}) \quad (6)$$

où p_i est le point de référence, r_{ij} est la distance normalisée de q_j à p_i et $\theta_{ij} = \langle \vec{p}_i \vec{q}_j, \vec{e}_i \rangle$ (voir Fig. 2).

Le *contexte de forme* associé au point p_i est défini suivant l'équation (2). C'est un histogramme de L classes (L *bins*). Si

par exemple, comme dans [13], 5 intervalles sont choisis pour la décomposition radiales $\log(r)$ avec r entre 0.125α et 2α et 12 intervalles pour la décomposition angulaire θ , alors L vaut 60. Un objet \mathcal{O} est maintenant décrit par l'ensemble des *contextes de forme* des points d'intérêt p_i .

$$\mathcal{O} \equiv \{h_i | p_i \in \mathcal{IP}\} \quad (7)$$

La construction des *bins* dans l'espace log-polaire permet premièrement de représenter de façon relative (distance, angle) les points de contour formant l'objet. Deuxièmement, elle donne une grande importance aux points qui sont plus proches par rapport à ceux qui sont plus éloignés (grâce à la transformation \log). Troisièmement, elle permet de diminuer la sensibilité au bruit grâce à la discrétisation de l'espace en *bins*.

C. CFPI au niveau de document

Dans la partie 2-B, nous avons défini le CFPI comme la distribution des points de contour du symbole par rapport aux points d'intérêt. Comme dans les documents, les symboles ne sont pas segmentés, il faut définir une région de voisinage pour chaque point d'intérêt pour calculer le CFPI. Cependant, il n'est pas évident de fixer la taille de la région à l'avance car la résolution des symboles contenus dans le document peut-être différentes d'un symbole à un autre. Pour bien capturer des informations aux différentes résolutions, nous proposons de définir une région de voisinage d'un point d'intérêt en fonction de la résolution où le point est détecté. La région de voisinage \mathcal{N}_i d'un point $p_i = (x_i, y_i, \delta_i, \theta_i)$ pour calculer le CFPI est déterminée par un cercle dont le centre est p_i et le rayon $R_i = \beta\delta_i$. Le descripteur CFPI, h_i , calculé dans cette région est défini par :

$$h_i(l) = \#\{q_j \neq p_i, q_j \in \mathcal{C} \cap \mathcal{N}_i : (q_j - p_i) \in \text{bin}(l)\}, l = \overline{1, L} \quad (8)$$

et un document \mathcal{D} est représenté par l'ensemble des h_i tel que :

$$\mathcal{D} \equiv \{h_i | p_i \in \mathcal{IP}\} \quad (9)$$

Comme dans un document les symboles sont souvent proches ou connectés, il est possible que d'autres points de contour présents dans le voisinage et n'appartenant pas au symbole perturbent le calcul du CFPI. Cependant, comme chaque symbole est représenté par plusieurs CFPI basés sur des points d'intérêt à des résolutions différentes, il sera affecté de façon moindre par ces points de contour extérieurs.

3. Vocabulaire visuel pour la représentation des documents

Dans la recherche de symboles isolés ou pour la localisation de symboles dans le document, une partie importante est d'apparier des points (des descripteurs) similaires pour trouver la similarité entre deux objets. Selon [13], la distance entre deux formes est mesurée comme la somme symétrique des coûts des meilleurs appariements des *contextes de forme*. Cela pose un problème de complexité quand on cherche des formes (objets) similaires parmi de nombreux candidats. Ainsi, dans cette partie, nous présentons une adaptation d'une technique de recherche textuelle pour l'indexation et la recherche de symboles. L'objectif est de réduire la complexité temporelle de l'étape d'appariement, calculée *en ligne*, grâce aux informations prédéterminées pendant une étape *hors ligne*. Cette approche est semblable à celle de Sivic *et al.* [14] qui consiste à rechercher des images à niveaux de gris en adaptant le modèle vectoriel à une recherche textuelle efficace et basée sur des mots visuels.



A. Construction d'un vocabulaire visuel

Tout d'abord, des descripteurs CFPI de l'ensemble des documents de la base considérée sont calculés comme indiqué dans la section 2. Chaque descripteur est un vecteur de dimension L . Ensuite, des classes contenant des descripteurs similaires sont créées en utilisant une technique de classification. Chaque classe est considérée comme un mot visuel identifié par le centre de la classe. Les descripteurs appartenant à cette classe sont considérés comme des occurrences d'un mot visuel dans la base. Un document graphique est ainsi représenté par un ensemble de mots visuels et peut être traité comme un document textuel. La figure 3 présente un exemple des structures appartenant à trois

classes définies à partir d'une base de symboles pour caractériser trois mots visuels. En principe, n'importe quelle méthode de classification peut être utilisée, tout dépend de la distribution des vecteurs. Ici nous avons utilisé la méthode des k-moyennes.

B. Appariement

Le vocabulaire visuel est construit pour que le CFPI associé à chaque point soit représenté par un mot visuel. L'objectif est de diminuer le nombre d'appariements redondants dans les étapes postérieures. Cependant, lors de la recherche d'un mot visuel correspondant à un CFPI, un problème peut se poser lorsque ce point se trouve proche des frontières entre classes. Un choix arbitraire de mot apparié avec ce point peut entraîner des erreurs. Afin de réduire ce problème, nous proposons d'effectuer des appariements multiples, ce qui signifie que le CFPI d'un point peut être apparié avec plus d'un mot. Des mots, dont les similarités avec le CFPI au point p_i sont très proches, sont choisis comme des mots correspondants du CFPI à p_i . Par soucis de clarté, nous parlerons de *mots correspondants de p_i* pour les désigner. Nous définissons un degré de confiance pour chaque appariement. Ce dernier dépend du nombre de mots appariés avec p_i et de leur similarité correspondante. Il est défini comme suit :

$$dConf_{p_i, w_j} = \frac{sim_{p_i, w_j}}{\sum_{w_k \in V_i} sim_{p_i, w_k}} \tag{10}$$

$$\sum_{w_j \in V_i} dConf_{p_i, w_j} = 1 \tag{11}$$

où V_i est l'ensemble des mots appariés avec p_i et sim_{p_i, w_j} la similarité entre le CFPI au point p_i et le mot w_j :

$$V_i = \{w_j \in \mathcal{V} \mid \frac{sim_{p_i, w_j}}{sim_{p_i, w_0}} > \epsilon\} \tag{12}$$

\mathcal{V} est le vocabulaire visuel, w_0 le mot le plus proche de p_i et ϵ est un seuil prédéfini (fixé à 0.96 dans nos résultats expérimentaux).

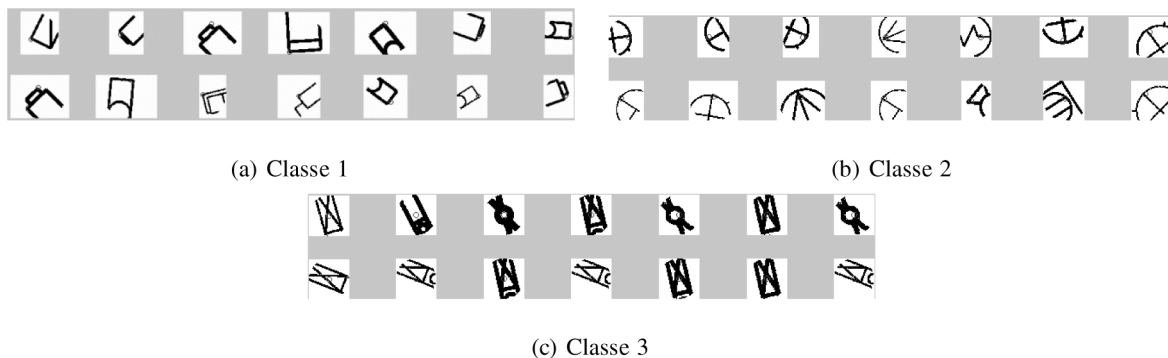


Figure 3. Exemple de quelques structures classées selon trois classes caractérisant trois mots visuels.

C. Modèle vectoriel

Lors de la recherche de symboles similaires, nous utilisons le modèle vectoriel. Il s'agit d'une technique couramment utilisée en recherche d'information car elle est plus performante que d'autres techniques classiques dans le cas général [21]. Avec ce modèle, un document est représenté par un vecteur de fréquences d'apparition des mots. Il est souvent décrit par un vecteur de fréquences pondérées dont chaque composante apporte un compromis entre deux facteurs : la fréquence d'apparition des termes (mots) dans le document (facteur tf) et l'estimation du degré d'importance de ce mot pour distinguer un document pertinent d'un non pertinent (facteur idf).

Les points d'intérêt pour un document j sont appariés avec des mots visuels et un document j est représenté par un vecteur de $tf-idf$ \vec{s}_j :

$$\vec{s}_j = \{f_{1,j}, f_{2,j}, \dots, f_{K,j}\} \quad (13)$$

où K est la taille du vocabulaire et $f_{i,j}$ est la fréquence pondérée du mot i dans le document j :

$$f_{i,j} = tf_{i,j} * idf_{i,j} = \overline{1, K}$$

$$tf_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}}; idf_{i,j} = \log \frac{N}{n_i}$$

avec $freq_{i,j}$ la fréquence d'apparition du mot i dans le document j , N le nombre total des documents dans la base et n_i le nombre de documents dans lesquels le mot i apparaît. Ainsi, $tf_{i,j}$ représente la fréquence normalisée du terme.

Dans le cas où un point est apparié avec plus d'un mot visuel, la fréquence $freq_{i,j}$ d'apparition du mot i dans le document j est redéfinie par (14) où $\mathcal{W}_j^D = \{w_1, w_2, \dots, w_M\}$ est l'ensemble des mots associés aux points d'intérêt dans le document j avec des degrés de confiance respectifs $\{dConf_1, dConf_1, \dots, dConf_M\}$:

$$freq_{i,j}^F = \frac{\sum_{w_m \in \mathcal{W}_j^D, w_m = i} dConf_m}{\sum_{m=1}^M dConf_m} \quad (14)$$

Pour déterminer des documents similaires à la requête, le vecteur $tf-idf$ de la requête \vec{s}_q est calculé de la même façon : 1) déterminer les descripteurs CFPI, puis, 2) mettre en correspondance ces descripteurs avec des mots visuels et enfin, 3) calculer les fréquences pondérées des mots existants dans la requête. Le degré de similarité entre la requête et un document j de la base de données est quantifié par leur corrélation mesurant l'angle de projection entre les deux vecteurs \vec{s}_j et \vec{s}_q .

$$sim(s_q, s_j) = \frac{\vec{s}_j \bullet \vec{s}_q}{|\vec{s}_j| \times |\vec{s}_q|} \quad (15)$$

Les degrés de similarité entre la requête et les documents de la base sont utilisés afin d'obtenir une liste de documents ordonnée par ordre de pertinence.

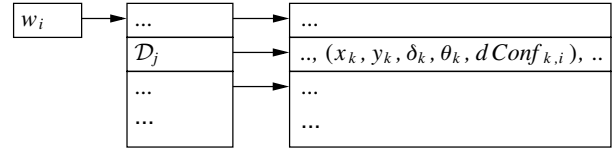


Figure 4. Structure de l'entrée du mot w_i dans le fichier inverse

D. Fichier inverse pour l'indexation des grands documents

Pour un document graphique pouvant contenir plusieurs symboles non-segmentés, le vecteur $tf-idf$ n'est pas suffisamment représentatif de celui-ci pour permettre de localiser précisément les symboles (il contient uniquement une information globale). Ainsi, nous nous appuyons sur une décomposition des documents par la structure de fichier composée de deux éléments : le vocabulaire et les occurrences des symboles. Pour chaque mot, la liste des positions de ses occurrences dans les documents est sauvegardée. Chaque élément dans la liste indique le document où ce mot est apparu, les positions des occurrences dans le document ainsi que les degrés de confiance correspondants. La figure 4 illustre une entrée du fichier pour un mot w_i .

4. Localisation de symboles dans les documents graphiques

Dans cette partie, nous montrons comment localiser des occurrences du symbole requête dans un document. Des régions candidates sont tout d'abord déterminées et couplées avec un processus de vote pour mettre en évidence les régions susceptibles de contenir la requête.

A. Régions candidates

Pour un symbole requête, les points d'intérêt et leurs mots correspondants sont déterminés comme dans les sections précédentes. Le rectangle englobant du symbole $rect = (x_c, y_c, w, h)$ (voir Fig. 5(a)) est défini à partir de son centre $C(x_c, y_c)$ et w, h dénotent respectivement sa hauteur et sa largeur.

Nous essayons de localiser dans le document des régions qui contiennent probablement des occurrences du symbole requête. Ces régions sont déterminées en se basant sur les relations entre les points d'intérêt et le rectangle englobant $rect$ de la requête. Supposons que le point $p_i = (x_i, y_i, \delta_i, \theta_i)$ (de la requête) et le point $p_j^d = (x_j^d, y_j^d, \delta_j^d, \theta_j^d)$ (dans le document) sont associés au même mot visuel (c'est-à-dire que ces deux points sont simi-

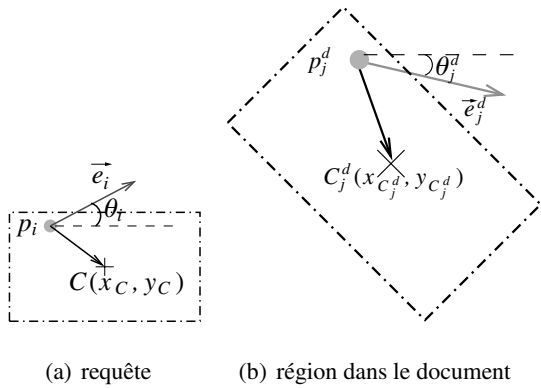


Figure 5. Localisation d'un rectangle englobant dans le document correspondant à la requête.

lares), la région candidate $rect_j^d = (x_{C_j^d}, y_{C_j^d}, w_j^d, h_j^d, \varphi_j^d)$ dans le document est déterminée en fonction de $\{rect, p_i, p_j^d\}$ (cf. Fig. 5.b) via des transformations de p_i à p_j^d . Il s'agit d'une translation du repère cartésien de la requête vers le point p_i (l'équation (22) donne les nouvelles coordonnées du centre C après la translation), suivie par une rotation vectorielle de la requête par un angle φ_j^d (cf. eq (20)) pour que l'orientation dominante de p_i ait la même direction de celle de p_j^d , puis un changement d'échelle par un facteur ξ (cf. eq (21)) et d'une translation du repère cartésien de p_j^d à l'origine $(0, 0)$ du document. Les coordonnées du centre de la région candidate obtenues après ces transformations sont définies par les équations (16) et (17). Les équations (18) et (19) définissent la largeur et la longueur de la région. p_i est considéré comme le point de contrôle de la région requête et p_j^d celui de la région à apparier dans le document.

$$x_{C_j^d} = x_j^d + \xi * (x_{piC} * \cos(\varphi_j^d) - y_{piC} * \sin(\varphi_j^d)) \quad (16)$$

$$y_{C_j^d} = y_j^d + \xi * (x_{piC} * \sin(\varphi_j^d) + y_{piC} * \cos(\varphi_j^d)) \quad (17)$$

$$w_j^d = w * \xi \quad (18)$$

$$h_j^d = h * \xi \quad (19)$$

où :

$$\varphi_j^d = \theta_j^d - \theta_i \quad (20)$$

$$\xi = \delta_j^d / \delta_i \quad (21)$$

$$(x_{piC}, y_{piC}) = (x_C, y_C) - (x_i, y_i) \quad (22)$$

B. Processus de vote

Lorsque les régions candidates sont définies, leurs centres sont votés par similarité entre ces régions et la requête en utilisant le modèle vectoriel.

Supposons que $W_r = \{w_{r1}, w_{r2}, \dots, w_{rM}\}$ est l'ensemble des mots associés aux points d'intérêt pour une région candidate r avec les degrés de confiance correspondants $\{dConf_{r1}, dConf_{r2}, \dots, dConf_{rM}\}$, la fréquence d'apparition du mot i dans cette région est définie par tf_i^r :

$$tf_i^r = \frac{\sum_{w_{rk} \in W_r, w_{rk} = i} dConf_{rk}}{\sum_{k=1}^M dConf_{rk}} \quad (23)$$

et sa fréquence pondérée pour cette région :

$$f_i^r = tf_i^r * idf_i \quad (24)$$

La région est donc représentée par un vecteur de fréquences des mots pondérées s^r :

$$s^r = (f_1^r, f_2^r, \dots, f_K^r) \quad (25)$$

La distance cosinus entre s^r et s^q (le vecteur de fréquences des mots pondérées de la requête) représente la valeur de vote pour la région r et celle qui obtient une valeur de vote élevée est considérée comme une occurrence du symbole requête dans le document.

5. Résultats expérimentaux

La première partie de cette section est dédiée à l'évaluation du descripteur proposé (CFPI) pour les symboles graphiques. Nous montrons expérimentalement que notre descripteur est adapté à la reconnaissance de symboles graphiques. La deuxième partie est focalisée à l'évaluation de notre méthode pour la localisation de symboles non-segmentés dans des documents.

A. Recherche de symboles isolés

Puisque notre objectif s'inscrit dans un système de recherche de symboles similaires à une requête, classés selon leur degré de similarité, nous utilisons une mesure classique pour évaluer la performance du descripteur : la courbe de *précision/rappel* [19], [21]–[23].

Dans notre contexte, le **rappel** est défini comme le nombre de symboles pertinents retrouvés (Ra) au regard du nombre de symboles pertinents dans toute la base (R).

$$Rappel = |Ra|/|R| \quad (26)$$

Le **précision** est le rapport entre le nombre de symboles pertinents et le nombre de symboles trouvés (X) similaires à une requête.

$$Precision = |Ra|/|X| \quad (27)$$

Pour les données de test, nous avons choisi la collection de symboles GREC³. Cette collection se compose de deux ensembles. L'un contient 50 symboles différents correspondant à 50 classes (l'ensemble A), l'autre (l'ensemble B) 250 occurrences de 50 classes de symboles, obtenues par des transformations linéaires (changement d'échelle et rotation) de chaque élément de A. Le nombre d'occurrences de chaque classe n'est pas égal, il varie de 1 à 10.

Nous effectuons une validation croisée où les symboles de A et de B sont partitionnés en deux : un ensemble de requêtes et une base de symboles à indexer. Nous avons créé 6 partitions en prenant 20 %, 30 %, 50 % de symboles comme requêtes grâce à des échantillons aléatoires sur chaque modèle de symbole. Pour chaque partition, une base de symboles à indexer sert à créer un vocabulaire visuel (dont la taille $K = 200$). Comme le nombre maximum des symboles pour une classe est égal à 10, nous nous intéressons qu'aux 10 premiers symboles trouvés. Les valeurs de la précision et du rappel sont calculées pour chaque valeur r ($r = 1, 10$) qui correspond aux r premiers symboles récupérés. Les valeurs de précision/rappel sont déterminées par la moyenne de toutes les requêtes pour chaque valeur de r .

Le premier objectif est de vérifier le comportement de notre descripteur par rapport aux changements de taille du vecteur de description et de montrer, expérimentalement, si des informations locales aux points d'intérêt sont suffisantes pour décrire un symbole. De ce fait, nous avons défini un vecteur de descripteur de dimension $L = 36$ et un de dimension $L = 60$. Dans ce contexte, au lieu de diviser l'intervalle $[0.125\alpha, 2\alpha]$ en 5 intervalles pour classer les distances radiales r_{ij} , nous ne prenons en compte que les points dont la distance radiale est comprise dans l'intervalle $[0.125\alpha, \alpha]$ soit une division en 3 intervalles. Nous utilisons toujours 12 intervalles pour les coordonnées angulaires θ_{ij} . La figure 6 présente les résultats moyen de précision/rappel pour 50 requêtes. Comme nous pouvons le remarquer sur cette figure, il n'y a pas de différence significative entre $L = 36$ et $L = 60$. Cela montre que même si l'information représentée par un descripteur est réduite, l'ensemble des descripteurs aux points d'intérêt assure une bonne représentation du symbole et ainsi de bonnes performances de recherche.

Par ailleurs, d'après la figure 6, nous pouvons aussi constater que la courbe atteint une précision élevée (80 %) avec un rappel de 70 %. Le degré de précision le moins bon (44 %) est obtenu avec $r = 10$. En effet, les valeurs de précision et de rappel sont calculées à partir des dix premiers symboles récupérés mais le nombre de symboles pertinents dans la base pour chaque requête n'est pas toujours égal à 10. Ainsi, les requêtes dont le nombre total de symboles pertinents dans la base est inférieur au nombre de documents récupérés (r) biaisent notre calcul de moyenne des précisions.

Nous avons également effectué des expérimentations avec SIFT et la \mathcal{R} -signature. Le descripteur SIFT évalué dans [19] comme un descripteur donnant de très bonnes performance en termes de

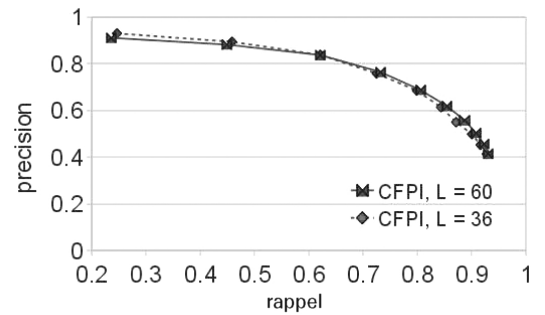


Figure 6. Précision/rappel avec descripteurs CFPI (dimension $L = 36, 60$).

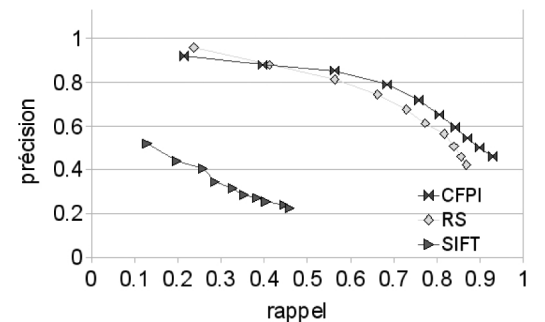


Figure 7. Précision/rappel avec CFPI, SIFT (vecteur de dimension 128) et la \mathcal{R} -signature

reconnaissance d'objets quelconques et la \mathcal{R} -signature [24] a prouvé également ses performances pour la reconnaissance de symboles. Notre but n'est pas de faire une évaluation de ces descripteurs, mais de montrer que ces derniers sont moins bien adaptés aux symboles graphiques pour SIFT et aux symboles graphiques incomplets pour la \mathcal{R} -signature. Lors de la recherche de symboles complets, les deux premières réponses plus proches obtenues par la \mathcal{R} -signature sont quasi-semblables à notre approche mais globalement le résultat obtenu par le CFPI est meilleur que celui obtenu par \mathcal{R} -signature (cf Fig. 7). Afin de vérifier l'adaptabilité du descripteur et du modèle vectoriel pour la recherche de symboles incomplets, nous avons comparé notre méthode avec la \mathcal{R} -signature en prenant des symboles incomplets comme requêtes (cf Fig. 8). Nous avons créé un ensemble de 77 symboles incomplets à partir des symboles complets de l'ensemble A et de ceux obtenus après rotation et zoom. Une validation croisée est effectuée sur l'en-

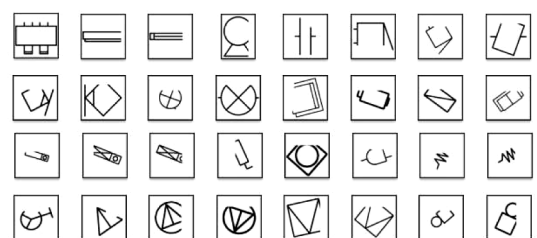


Figure 8. Exemples des symboles incomplets.

3. <http://www.cvc.uab.es/grec2003/SymRecContest/>

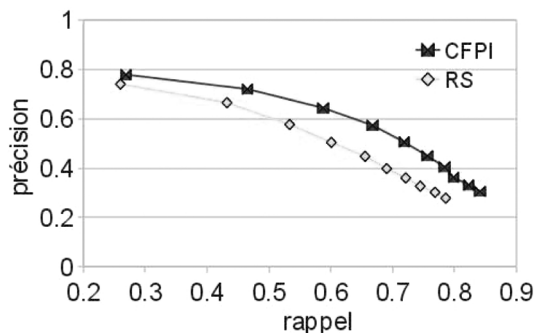


Figure 9. Résultats obtenus avec le CFPI et la \mathcal{R} -signature pour des requêtes avec des symboles incomplets.

semble de ces symboles et de ceux dans B. Les résultats obtenus (Fig. 9) montrent que notre approche donne des résultats meilleurs que ceux obtenus par la \mathcal{R} -signature. Comme le CFPI se base sur les points d'intérêt près des jonctions, il est moins robuste aux occlusions de jonctions mais plus robuste aux occlusions d'arêtes que la \mathcal{R} -signature. En effet, contrairement à la \mathcal{R} -signature, le descripteur CFPI fournit une description locale en chaque point d'intérêt. Cette capacité à prendre en compte des symboles partiellement occultés est un avantage dans le cadre de la localisation de symboles non-segmentés dans les documents graphiques.

B. Localisation de symboles

Pour vérifier la performance de notre approche pour localiser des symboles requêtes dans des documents, nous avons effectué des tests sur des documents synthétiques du projet SESYD [25]. Les CFPIs sont calculés pour les points d'intérêt sur 100 documents graphiques. La technique de classification des k-moyennes est appliquée sur ces descripteurs pour créer des mots visuels. Les CFPIs sont ensuite mis en correspondance avec ces mots en effectuant des appariements multiples et le fichier inverse est construit. Lors d'une requête, les CFPIs de la requête sont extraits et mis en correspondance avec l'ensemble des mots visuels construits. Grâce au fichier inverse, nous détermi-

nons des régions d'intérêt dans les documents. Les similarités entre ces régions et la requête sont calculées grâce au modèle vectoriel. Les régions dont la similarité est élevée sont considérées comme des occurrences potentielles du symbole requête. Le tableau I présente les résultats obtenus de la localisation des requêtes de la figure 10 dans quinze documents du projet SESYD. La première colonne du tableau indique les requêtes. La deuxième ($N^{\circ}VT$) indique le nombre d'occurrences du symbole existant dans le document (vérité terrain). La troisième ($N^{\circ}DC$) et la quatrième ($N^{\circ}errP$) contiennent respectivement le nombre de détections correctes et le nombre de fausses détections (faux positifs). La cinquième colonne ($N^{\circ}errN = N^{\circ}VT - N^{\circ}DC$) indique le nombre d'occurrences de la requête dans le document qui ne sont pas trouvées (faux négatifs). Les deux dernières colonnes montrent la précision ($= N^{\circ}DC / (N^{\circ}DC + N^{\circ}errP)$) et le rappel ($= N^{\circ}DC / N^{\circ}VT$) correspondant à chaque requête. Les faux positifs ont une influence sur la précision du résultat et les faux négatifs sur la capacité à fournir toutes les réponses possibles. Donc, moins il y a de faux positifs, plus la précision est élevée et plus les faux négatifs sont faibles, plus le rappel est élevé.

Nous pouvons voir que notre méthode a bien répondu pour les requêtes de la figure 10(a), 10(e) et 10(f) (cf. Tab. I et Fig. 11). Pour les autres, la précision n'est pas toujours élevée mais le rappel est très bon. En effet, la méthode a localisé la plupart des bonnes réponses mais elle contient également des réponses incorrectes (voir Fig. 11(b)).

Nous constatons que notre système n'exige pas de fortes contraintes sur le symbole (comme la convexité ou que les composantes soient connectées). Cependant, si le symbole se compose particulièrement de courbes, la réponse actuelle de notre méthode n'est pas très stable car le détecteur des points d'intérêt n'est pas adapté pour extraire des points de salience sur des courbes. Il manque dans ce cas des points pour avoir une description pertinente. Par ailleurs, certaines fausses réponses telles que celles identifiées dans la figure 11(b) sont dues à la représentation trop locale de notre méthode qui ne tient pas compte de la relation spatiale qui peut exister entre les différents points d'intérêt composant un symbole. Dans ce contexte il sera nécessaire dans de futurs travaux d'ajouter également une étape de filtrage a posteriori sur les réponses obtenues.

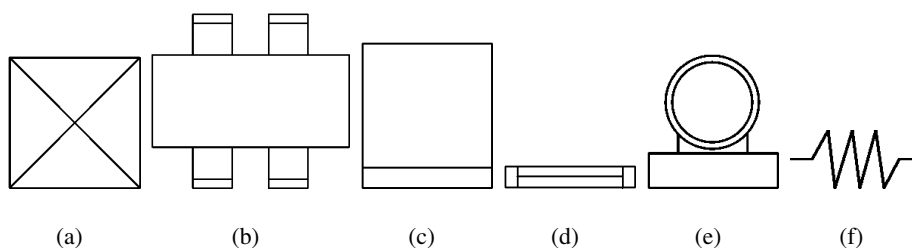


Figure 10. Requêtes

Tableau 1. Résultats des requêtes en Fig 10. N°VT : Nombre d'occurrences de la requête dans le document. N°DC : Nombre de détections correctes. N°errP : Nombre de faux positifs. N°errN : Nombres de faux négatifs, P : précision, R : rappel.

Requêtes.	N°VT	N°DC	N°errP	N°errN	P	R
Fig. 10(a)	58	58	0	0	58/58	58/58
Fig. 10(b)	15	15	30	0	15/45	15/15
Fig. 10(c)	56	32	32	24	32/64	24/56
Fig. 10(d)	15	15	30	0	15/45	15/15
Fig. 10(e)	15	15	0	0	15/15	15/15
Fig. 10(f)	32	28	10	10	28/38	28/38
Average	Precision = 0.7, Recall = 0.88					

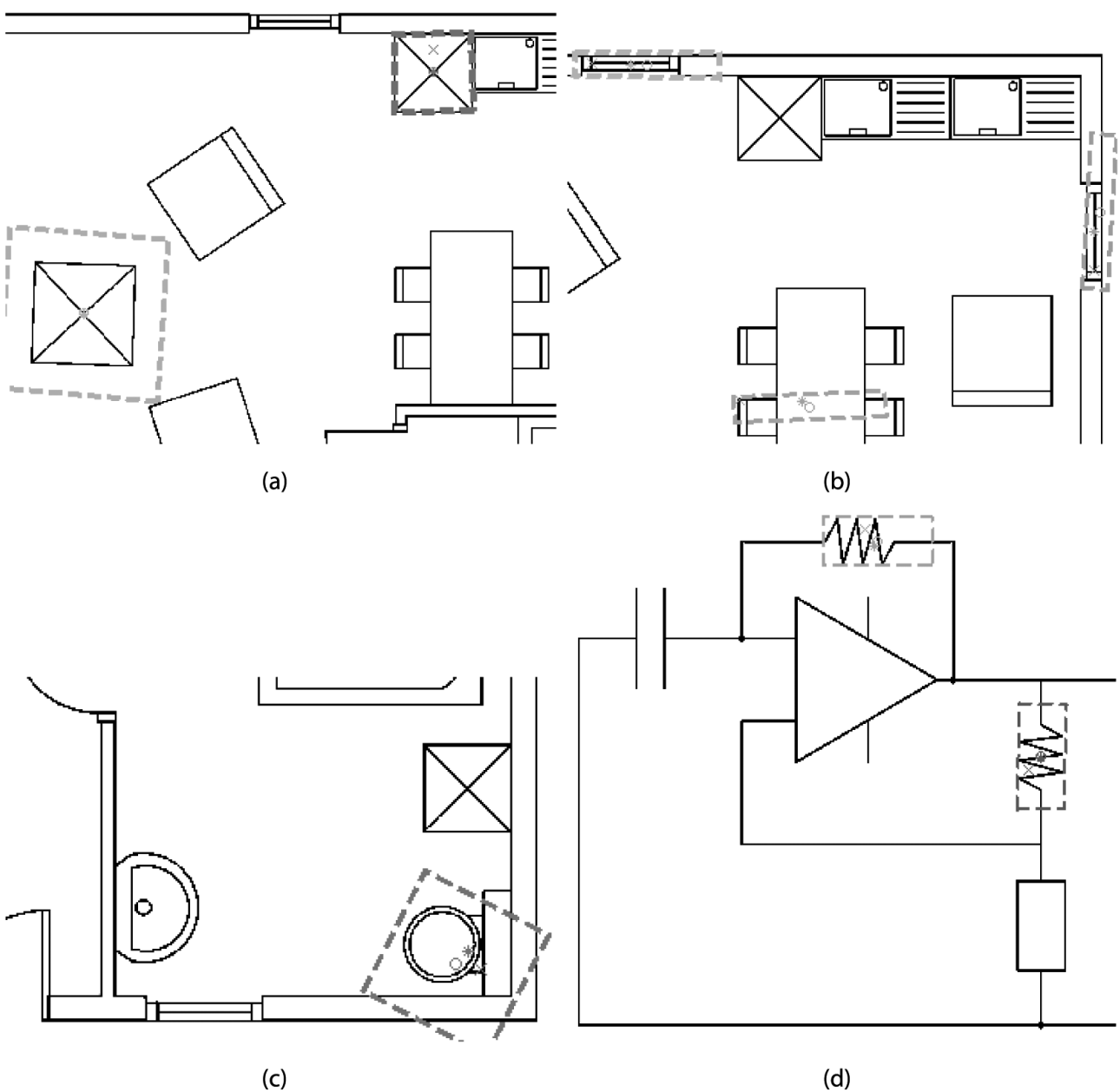


Figure 11. Réponses de l'approche pour localiser les requêtes des figures 10(a), 10(d), 10(e) et 10(f).

6. Conclusions et perspectives

Nous avons proposé une méthode de localisation de symboles dans le document. La première étape a été de définir un descripteur adapté aux symboles. Pour ce faire, nous avons proposé une adaptation du *contexte de forme* pour décrire des symboles graphiques et des documents. Le descripteur CFPI (*Contexte de forme pour les Points d'Intérêt*) est simple et invariant à la rotation et aux changements d'échelles. Il fournit une bonne représentation de la configuration locale correspondant à chaque point d'intérêt et nous permet de réduire la complexité de description d'un symbole. Ensuite, les descripteurs sont regroupés en classes pour construire un vocabulaire visuel et chaque symbole est décrit par un ensemble de mots visuels et apparié à un symbole requête. La complexité d'appariement est aussi réduite grâce à l'utilisation d'un fichier inverse et du modèle vectoriel. Les résultats expérimentaux ont montré la capacité de notre approche à localiser des occurrences d'un symbole requête dans des documents.

Cependant ces résultats considérés comme prometteur méritent des améliorations. En l'occurrence, quand un symbole est représenté par des CFPI, si le nombre de points d'intérêt du symbole est trop petit (cas de symboles représentés par des courbes), le vecteur représentant le symbole n'assure pas une description efficace du « contenu » du symbole en raison du faible nombre de mots visuels. Par ailleurs, lors de la localisation de symboles dans un document, nous n'avons pas pris en compte la structure spatiale entre les différents mots visuels existant dans un symbole. Une perspective est donc d'intégrer un filtrage tenant compte des relations spatiales entre les mots visuels du symbole afin de réduire les faux positifs.

Références

[1] A. JAIN, R. DUIN, and J. MAO, "Statistical pattern recognition: a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.

[2] S. TABBONE and L. LLADOS, "A propos de la reconnaissance de documents graphiques: synthèse et perspectives," in *TAIMA*, Hammamet, Tunisia, 2007.

[3] Y. RUI, A. SHE, and T. HUANG, "A modified fourier descriptor for shape matching in mars," in *Workshop on Image Databases and Multi Media Search*, vol. 8, Amsterdam, Netherlands, 1998, pp. 165–180.

[4] D. ZHANG and G. LU, "Study and evaluation of different fourier methods for image retrieval," *Image and Vision Computing*, vol. 23, pp. 33–49, 2005.

[5] R. PROKOP and A. REEVES, "A survey of moment-based techniques for unoccluded object representation and recognition," *CVGIP: Graphical Models and Image Processing*, vol. 54, no. 5, pp. 438–460, 1992.

[6] L. WENYIN, W. ZHANG, and L. YAN, "An interactive example-driven approach to graphics recognition in engineering drawings," *International Journal of Document Analysis and Recognition*, vol. 9, no. 1, pp. 13–29, March 2007.

[7] J. FONSECA, A. FERREIRA, and J. JOAQUIM, "Content-based retrieval of technical drawings," *International Journal of Computer Applications in Technology*, vol. 23, no. 2-3, pp. 86–100, March 2005.

[8] H. LOCTEAU, S. ADAM, E. TRUPIN, J. LABICHE, and P. HEROUX, "Symbol spotting using full visibility graph representation," in *Seventh IAPR International Workshop on Graphics Recognition*, Curitiba, Brazil, September 2007.

[9] M. RUSINOL and J. LLADOS, "Symbol spotting in technical drawings using vectorial signatures," in *Graphics Recognition. Ten Years Review and Future Perspectives*. Springer Berlin / Heidelberg, October 2006, vol. 3926/2006, pp. 35–46.

[10] S. TABBONE and D. ZUWALA, "An indexing method for graphical documents," in *International Conference on Document Analysis and Recognition*, vol. 2, Curitiba, Brazil, 2007, pp. 789–793.

[11] M. RUSINOL and J. LLADOS, "A region-based hashing approach for symbol spotting in technical documents," in *Seventh IAPR International Workshop on Graphics Recognition*, Curitiba, Brazil, September 2007.

[12] D. ZUWALA and S. TABBONE, "Une méthode de localisation et de reconnaissance de symboles sans connaissance a priori," in *Colloque International Francophone sur l'Ecrit et le Document – CIFED'06*, Fribourg, Suisse, 2006, pp. 127–131.

[13] S. BELONGIE, J. MALIK, and J. PUZICHA, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, Avril 2002.

[14] J. SIVIC and A. ZISSERMAN, "Video google: Efficient visual search of videos," in *Toward Category-Level Object Recognition*. Springer Berlin / Heidelberg, 2006, vol. 4170/2006, pp. 127–144.

[15] D. G. LOWE, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.

[16] S. AGARWAL, A. AWAN, and D. ROTH, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, November 2004.

[17] A. BOSCH, A. ZISSERMAN, and X. MUNOZ, "Scene classification via pls," in *Computer Vision – ECCV 2006*. Springer Berlin / Heidelberg, May 2006, vol. 3954/2006, pp. 517–530.

[18] C. SCHMID, R. MOHR, and C. BAUCKHAGE, "Comparing and evaluating interest points," in *In Proceedings of the ICCV*, Bombay, India, 1998, pp. 230–235.

[19] K. MIKOLAJCZYK and C. SCHMID, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, October 2005.

[20] S. TABBONE, L. ALONSO, and D. ZIOU, "Behavior of the laplacian of gaussian extrema," *Journal of Mathematical Imaging and Vision*, vol. 23, no. 1, pp. 107–128, July 2005.

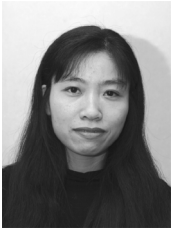
[21] R. BAEZA-YATES and B. RIBEIRO-NETO, *Modern Information Retrieval*. New York: ACM Press / Addison-Wesley, 1999.

[22] J. R. SMITH, "Image retrieval evaluation," in *IEEE Workshop on Content-based Access to Image and Video Databases, Bombay, India*. IEEE Computer Society Washington, DC, USA, June 1998, p. 112.

[23] TH.GEVERS and A. SMEULDERS, *Emerging Topics in Computer Vision*. Addison-Wesley / Prentice Hall, 2004, ch. 8. Content-Based Image Retrieval: An Overview.

[24] S. TABBONE and L. WENDLING, "Recognition of symbols in grey level line drawings from an adaptation of the radon transform," in *In Proceedings of 17th International Conference on Pattern Recognition*, vol. 2, Cambridge (UK), 2004, pp. 570–573.

[25] M. DELALANDRE, T. PRIDMORE, E. VALVENY, H. LOCTEAU, and E. TRUPIN, "Building synthetic graphical documents for performance evaluation revised," in *Selected Papers of Workshop on Graphics Recognition (GREC), Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2008, vol. 5046, pp. 288–298.



Thi-Oanh **Nguyen**

THI-OANH NGUYEN est née à Bac Ninh, Vietnam. Elle a obtenu le titre de docteur en informatique de l'Université de Nancy 2 en Décembre 2009. Sa thèse est effectuée au Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA) dans le cadre de co-direction entre le LORIA (France) et l'IFI (Institut de la Francophonie pour l'Informatique, Vietnam) depuis 2006. La problématique de recherche concerne la localisation de symboles dans les documents graphiques qui l'a amenée à focaliser sur le problème de reconnaissance de symboles et sur l'intégration des techniques de recherche d'informations à la recherche d'images par le contenu, particulièrement au problème de localisation de symboles.



Salvatore **Tabbone**

SALVATORE-ANTOINE TABBONE est professeur en informatique à l'université de Nancy 2. Il a obtenu, en 1994, son doctorat en informatique de l'INPL sur la détection multi-échelle de contours subpixel et de jonctions. Il a été en délégation à l'INRIA Lorraine de 2000 à 2002. Il a soutenu, en 2005, son habilitation à diriger des recherches de l'université de Nancy 2 sur la reconnaissance de formes dans les documents graphiques. Depuis 2007, il est responsable (responsable adjoint de 2005 à 2007) de l'équipe QGAR du LORIA et ses thèmes de recherche concernent la reconnaissance de formes, la recherche d'images par le contenu, le filtrage d'images, la segmentation d'images et de documents graphiques, l'extraction d'indices visuels, l'indexation de documents graphiques et l'apprentissage numérique. Il est auteur/co-auteur de plus de 100 publications dans des revues et conférences nationales et internationales avec comité de lecture et membre de plusieurs comités de programme de conférences nationales et internationales.



Alain **Boucher**

ALAIN BOUCHER a obtenu son diplôme de génie informatique de l'École Polytechnique de Montréal en 1994, puis son DEA, option robotique, vision et images, de l'Institut National Polytechnique de Grenoble en 1995. Il a obtenu son doctorat en informatique de l'Université Joseph Fourier en 1999 pour ses travaux sur la segmentation et l'interprétation d'images de cellules en mouvement par un système multi-agent. Il a ensuite travaillé pendant 3 ans à l'INRIA Sophia Antipolis (projet Orion) pour le projet européen ASTHMA sur la reconnaissance de grains de pollen. Depuis septembre 2002, il est professeur d'informatique à l'Institut de la Francophonie pour l'Informatique (IFI-AUF). Depuis 2008, il est également le responsable des études à l'IFI. Ses travaux de recherche se font dans le cadre de MSI (Modélisation et Simulation Informatique), équipe membre de l'UMI UMMISCO soutenue par l'IRD et l'Université Pierre et Marie Curie (Paris 6). Ses principaux thèmes de recherche sont la vision par ordinateur, l'indexation et recherche d'images par le contenu et l'intelligence artificielle.



