

---

# Localisation automatique de champs de saisie sur des images de formulaires couleur par isomorphisme de sous-graphe <sup>1</sup>

Maroua Hammami\* — Pierre Héroux\* — Sébastien Adam\* — Vincent Poulain d'Andecy\*\*

\* Normandie Université – Université de Rouen – Normastic CNRS FR 3638 – LITIS EA 4108

\*\* Itesoft – Le Séquoia, 30470 Aimargues

---

*RÉSUMÉ.* Cet article présente une approche permettant la localisation de champs de saisie sur des images couleur de formulaires. Ces champs sont localisés grâce à une modélisation structurelle représentant leur contexte. Dans un premier temps, les zones informatives sont extraites. Les formulaires sont ensuite représentés par des graphes au sein desquels les nœuds représentent des rectangles de couleur uniforme tandis que les arcs modélisent les relations de voisinage. Finalement, le voisinage de la région d'intérêt à localiser est également représenté par un graphe. Une recherche d'isomorphisme de sous graphe vise à localiser le graphe modélisant le voisinage de la région d'intérêt au sein de la représentation structurelle du document cible. Une expérimentation est réalisée sur une base de 130 images de document. Les résultats montrent l'efficacité de la méthode même si la position de la région d'intérêt est variable.

*ABSTRACT.* This paper presents an approach for spotting textual fields in colored forms. We proceed by locating these fields thanks to their neighboring context which is modeled with a structural representation. First, informative zones are extracted. Second, forms are represented by graphs in which nodes represent colored rectangles while edges represent neighboring links. Finally, the context of the queried region of interest is modeled as a graph. Subgraph isomorphism is applied in order to locate this ROI in the structural representation of a whole document. Evaluated on a 130-document image dataset, experimental results show up that our approach is efficient and that the requested information is found even if its position is changed.

*MOTS-CLÉS :* Localisation de champs, Documents couleur, Isomorphisme de sous-graphe.

*KEYWORDS:* Field location, Color document, Subgraph isomorphism.

---

1. Traduction étendue d'un article publié en anglais à ICDAR 2015 (Hammami *et al.*, 2015)

## 1. Introduction

La lecture automatique des documents commerciaux et administratifs tels que les formulaires ou les factures permet aux organisations d'alimenter leur systèmes d'information en évitant les interventions humaines chronophages, fastidieuses et sources d'erreurs. Cette lecture automatique consiste à localiser, extraire et reconnaître une information cible souvent ajoutée en surimpression sur un fond (formulaire).

Pour ce faire, il est nécessaire de disposer d'un modèle de lecture indiquant la position des informations à lire, éventuellement associées à des méta-données renseignant le type d'information (patronyme, numéro d'abonné, adresse, code postal, numéro de téléphone, montant, date...) visant à guider la lecture ou permettant de la valider.

La partie du modèle de lecture relative à la localisation revêt une grande importance dans le sens où elle conditionne fortement la performance des traitements ultérieurs. En revanche, l'expression de la localisation peut être assez difficile à définir en raison de la variabilité avec laquelle les données à lire peuvent apparaître sur le document. En effet, même pour plusieurs instances du même formulaire, la position des informations peut varier, notamment en raison des conditions de numérisation. Ce phénomène est particulièrement notable dans le contexte de la reconnaissance "à la demande" dans lequel la numérisation est opérée par l'utilisateur mais où l'étape de reconnaissance est opérée par un système central. Par ailleurs, l'information à lire est fréquemment manuscrite. Il s'ensuit que la couleur de l'encre, l'épaisseur du tracé, la taille et la syntaxe de l'information à lire ne peuvent pas être parfaitement contrôlées. Par exemple, plusieurs syntaxes peuvent être utilisées pour indiquer la même date, différents montants peuvent contenir un nombre de chiffres très différent. L'ensemble de ces contraintes illustrent le fait qu'un modèle de lecture qui définirait la position et la taille des champs à lire de façon absolue et rigide serait inopérant.

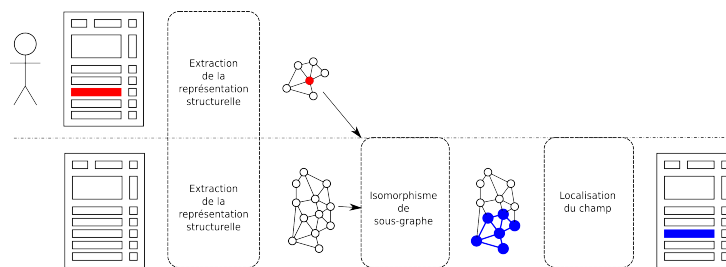
Dans cet article, nous proposons la définition d'un modèle de lecture permettant la définition de la position de façon souple et relative. Nous proposons également l'algorithme associé pour localiser de façon robuste les zones de saisie sur un document cible. L'approche répond aux contraintes des applications commerciales qui imposent que (i) la position et la taille des informations à extraire peuvent être amenées à varier entre plusieurs instances du même formulaire et que (ii) la saisie du modèle de lecture est opérée via une interface conviviale en considérant un exemple sur une seule instance.

Dans ce contexte, la représentation proposée modélise le voisinage des zones informatives sur la base des rectangles de couleurs environnants. Ce choix est justifié par le fait que les rectangles et les cadres de couleur imprimés sur le fond des formulaires sont des éléments essentiels de la structure de tels documents. En particulier, ils apparaissent sur le fond du document pour guider la saisie des champs informatifs par les utilisateurs. Du point de vue structurel, le voisinage est modélisé par un graphe dont les nœuds représentent les rectangles et cadres situés dans l'environnement de la zone ciblée. Dans ce graphe, les arcs représentent des relations de visibilité. Un modèle identique est utilisé pour représenter la structure du document cible complet sur le-

quel l'information sera cherchée. La procédure de localisation de la zone revient alors à rechercher l'occurrence du graphe le modélisant au sein de la représentation structurale du graphe complet. La recherche d'isomorphisme de sous-graphe correspondante se doit d'être robuste à la variabilité évoquée précédemment.

La suite de cet article est organisée de la façon suivante. La section 2 présente le contexte dans lequel l'approche proposée est appliquée. La section 3 décrit le processus d'extraction des rectangles de couleur sur les images de formulaire. La section 4 détaille la représentation structurelle utilisée pour modéliser aussi bien la zone informative recherchée et son contexte que la structure du document cible dans son ensemble. La section 5 présente les expériences menées pour l'évaluation de l'approche et les résultats obtenus. Enfin, la section 6 conclut l'article et propose quelques extensions possibles à ce travail.

## 2. Contexte



**Figure 1.** *Vue d'ensemble de l'approche*

La figure 1 présente les principales étapes de l'approche proposée. Dans un premier temps, une interface graphique permet à l'utilisateur de délimiter, sur une seule instance de document, l'information qu'il souhaite voir extraite automatiquement sur les autres instances du même type de documents. Le type des documents à traiter est en amont déterminé par un classifieur. À partir de cette information, nous construisons une représentation structurelle du contexte de voisinage de l'information ciblée. Cette représentation structurelle alimente le modèle de lecture du document. En production, la même représentation structurelle est extraite sur l'ensemble du document, et ce pour tous les documents à traiter. Un algorithme de recherche d'isomorphisme tolérant aux erreurs est ensuite appliqué, afin de retrouver le sous-graphe de la représentation structurelle du document qui correspond au mieux à celui modélisant l'information ciblée. Nous disposons alors en sortie de l'instance de l'information recherchée sur un document cible.

Dans un contexte applicatif similaire, Rusiñol *et al.* (Rusiñol *et al.*, 2013) proposent une approche pour s'abstraire de la variabilité en position et en taille de l'information ciblée en localisant cette information par rapport à son voisinage. Dans leur travaux, le contexte de voisinage est modélisé par un graphe en étoile dont le

nœud central correspond à l'information ciblée et dont les nœuds aux extrémités des branches correspondent à des mots reconnus automatiquement grâce à un logiciel de reconnaissance optique de caractères. La recherche de l'information sur un document cible consiste alors à (i) lui appliquer l'OCR, (ii) y localiser l'agencement des mots qui correspond au mieux à celui décrit dans le contexte de voisinage de l'information ciblée, (iii) en déduire la position estimée de l'information recherchée. L'importance accordée à un mot particulier dans le processus de mise en correspondance s'accroît en fonction de sa fréquence dans le contexte des documents d'apprentissage. Un des inconvénient principaux de cette approche vient du fait qu'elle dépend fortement des performances de la reconnaissance de caractères. Elle offre de bonnes performances lorsque l'OCR opère dans un cadre favorable (image binaire propre, caractères imprimés. . . ) mais peut échouer dans d'autres circonstances, notamment en présence de documents couleurs. D'autres travaux tels que ceux de Peanho *et al.* (Peanho *et al.*, 2012) et Cesarini *et al.* (Cesarini *et al.*, 1998) utilisent des graphes relationnels attribués. Cependant, ce type de méthodes est basé sur un modèle sémantique annoté par l'utilisateur, là où notre système travaille sur la seule base des informations ciblées par l'utilisateur.

De nos jours, les documents couleur constituent la très grande majorité des formulaires administratifs et commerciaux (cd. Fig. 2). En effet, sur ce type de documents les lignes, peignes, cadres ou rectangles en couleur sont présents sur le fond du document afin de guider la saisie (cf. Fig. 3).

Dans l'approche proposée, au lieu d'être une difficulté, l'information colorimétrique est exploitée pour déterminer la structure du document et le contexte de voisinage de la zone recherchée. En effet, les primitives colorées sont considérées comme faisant partie intégrante de la structure du document et sont utilisées comme référence à partir desquelles on peut définir la localisation des informations recherchées. Les sections 3 et 4 détaillent la façon dont la représentation structurelle basée sur les zones rectangulaires colorées est construite.

### **3. Recherche des champs de saisie**

Cette section décrit la première étape de la construction de la représentation structurelle. Cette étape vise à extraire les zones rectangulaires colorées. Ceci est réalisé en trois temps : les prétraitements, la quantification couleur et la sélection des composantes connexes rectangulaires.

#### **3.1. Prétraitements**

La première phase vise à prétraiter l'image initiale de façon à faciliter les étapes ultérieures.

Figure 2. Exemples de formulaires

Figure 3. Exemples de champs de saisie

### 3.1.1. EPSF

En raison des techniques d'impression des formulaires (techniques d'offset en quadrichromie CMYK) et à la numérisation, les valeurs des pixels des images issues d'une numérisation des formulaires ne correspondent pas aux couleurs qui sont perçues à une échelle macroscopique. Pour obtenir des données plus cohérentes et pour faciliter la phase ultérieure de quantification couleur, nous appliquons un filtrage à la valeur des

pixels. Pour ce faire, nous avons choisi le filtre EPSF (Edge Preserving Smoothing Filtering) proposé par Nikoulaou *et al.* dans (Nikoulaou et Papamarkos, 2009). Comme son nom l'indique, ce filtre médian adapte automatiquement son pouvoir de lissage en fonction du gradient de couleur. Les régions pour lesquelles le gradient est faible se voient appliquer un fort lissage alors que les zones à fort gradient ne sont que très faiblement lissées, permettant ainsi de préserver les contours des formes.

### 3.1.2. *Inpainting*

Nous proposons également d'appliquer une étape d'inpainting visant à remplacer la valeur des pixels appartenant aux éléments textuels par la valeur des pixels situés autour. Cette étape d'effacement du texte facilite également l'extraction des rectangles de couleur. En effet, la présence d'informations textuelles, qu'il s'agisse du texte ajouté sur le formulaire ou qu'il soit déjà présent sur le formulaire vierge, est susceptible de perturber l'extraction des rectangles qui peuvent se retrouver scindés en plusieurs composantes connexes.

Pour ce faire, nous détectons le texte en utilisant la binarisation intelligente proposée par Gaceb *et al.* (Gaceb *et al.*, 2013). La valeur des pixels classés comme texte est remplacée par la valeur du pixel classé non texte le plus proche.

### 3.1.3. *Espace colorimétrique*

Il existe plusieurs espaces colorimétriques, et l'espace de représentation *RVB* le plus communément utilisé n'est pas toujours le plus approprié. Fort de ce constat, avant d'appliquer l'étape ultérieure de quantification couleur, l'image peut être projetée dans d'autres espaces colorimétriques, notamment l'espace *CIELab* (Sgarbi *et al.*, 2014 ; Hase *et al.*, 2001) ou l'espace *YIQ* (Chen et Chen, 1998).

Une fois l'image prétraitée par filtrage, purgée de son information textuelle et/ou projetée dans un espace colorimétrique alternatif à *RVB*, nous appliquons une étape de quantification couleur afin d'identifier des composantes de couleur homogènes.

## 3.2. *Quantification couleur*

L'étape suivante nécessaire à l'extraction des rectangles de couleurs consiste à appliquer une quantification couleur qui peut être appliquée dans n'importe lequel des espaces colorimétriques. Cette étape vise à séparer l'image du document en plusieurs couches en fonction de la couleur que percevrait un humain sur l'image originale. Pour ce faire, nous appliquons l'algorithme des *k*-moyennes. Nous supposons que le nombre de couleurs dominantes sur le document est au moins de 3 : une pour le fond du document, au moins une pour les éléments de la structure du formulaire, dont les rectangles recherchés, et une pour le texte. Cependant, cette dernière couche a pu être éliminée par le traitement d'inpainting. Dans les expériences décrites en section 5, nous analysons l'influence du paramètre *k*.

### 3.3. Extraction des rectangles

Après que l'image a été traitée de sorte à séparer plusieurs couches correspondant à un nombre réduit de couleurs, la dernière étape consiste à isoler au sein de ces couches des formes rectangulaires. Pour ce faire, dans chaque couche couleur, nous extrayons dans un premier temps les composantes connexes au sein des différentes couches. L'ensemble des composantes connexes est filtré de sorte que seules celles dont la forme est rectangulaire sont conservées. Selon Rosin (Rosin, 1999), une région peut être considérée comme rectangulaire quand son aire rapportée à celle de sa boîte englobante est supérieure à un seuil  $\theta$ . Ce seuil  $\theta$  est considéré comme le critère de rectangularité. Dans les expériences décrites en section 5, la valeur de  $\theta$  a été fixée à 0,7. On peut remarquer que la quasi totalité des documents administratifs et commerciaux sont présentés avec une structure de type Manhattan. De ce fait, notre recherche se concentre sur la recherche de rectangles isothétiques (i.e. dont les côtés sont horizontaux et verticaux). La boîte englobante considérée dans le calcul de  $\theta$  est donc également isothétique, ce qui simplifie l'extraction puisqu'il suffit de déterminer les valeurs minimale et maximale de chacune des coordonnées des pixels de la composante connexe.

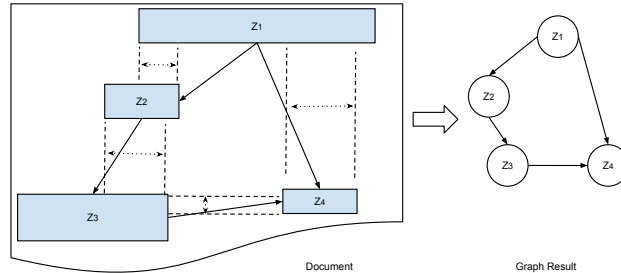
## 4. Représentation structurelle

Une fois les rectangles de couleur extraits selon la procédure définie précédemment, nous construisons une représentation structurelle modélisant leur agencement. Nous rappelons que l'objet de la représentation structurelle est d'être le support permettant de localiser la zone informative par le biais d'une recherche d'isomorphisme de sous-graphe permettant de retrouver au sein de la représentation globale du document le sous-graphe correspondant à une instance de la modélisation du voisinage. Dans ce contexte, nous proposons de modéliser le document par un graphe de visibilité entre les rectangles.

### 4.1. Description des nœuds du graphe

Comme indiqué précédemment, les nœuds du graphe représentent les rectangles extraits tel qu'indiqué dans la section précédente. Les nœuds du graphe sont étiquetés par des attributs de deux types :

**Attributs de couleur :** L'information de couleur est utilisée pour distinguer les documents ayant des structures similaires mais des couleurs différentes. Grâce à une étude expérimentale, nous avons déterminé que les caractéristiques correspondant à l'intensité des composantes  $I$  et  $Q$  de l'espace  $YIQ$  étaient celles qui permettaient d'obtenir les meilleures performances. À partir de l'espace  $RVB$ ,



**Figure 4.** Relations de visibilité

les composantes  $I$  et  $Q$  des pixels sont obtenues par les combinaisons linéaires suivantes :

$$I(p) = 0,596 \times R(p) - 0,275 \times G(p) - 0,321 \times B(p)$$

$$Q(p) = 0,212 \times R(p) - 0,528 \times G(p) - 0,311 \times B(p)$$

**Attributs géométriques :** Les étiquettes des nœuds comportent également la dimension (largeur  $W$  et hauteur  $H$ ) des rectangles afin de distinguer les rectangles de dimensions différentes lors du processus de mise en correspondance. Les dimensions sont normalisées par rapport à la taille globale du document afin de rendre ces caractéristiques indépendantes de la résolution de l'image.

Au final, chaque nœud du graphe est étiqueté par un vecteur numérique à 4 dimensions ( $I, Q, W, H$ )

#### 4.2. Description des arcs

Les seuls nœuds accompagnés de leurs attributs sont insuffisants pour décrire la structure du document. Une modélisation des agencements relatifs des rectangles les uns par rapport aux autres est nécessaire pour décrire plus précisément la structure topologique du document. Nous proposons de modéliser la notion de voisinage par le biais d'une relation de visibilité entre les rectangles. Ainsi, deux nœuds sont liés par un arc si les rectangles correspondant sont considérés visibles l'un de l'autre. Deux rectangles sont considérés visibles s'il partagent des intervalles suffisamment larges (resp. hauts) de points pouvant être mis en correspondance par un segment vertical (resp. horizontal) n'intersectant d'autres rectangles.

Ainsi, si  $\lambda$  est le seuil minimal de visibilité,  $I_V$  la taille de l'intersection visible verticale et  $I_H$  celle de l'intersection visible horizontale, un arc est établi entre les nœuds modélisant les rectangles  $z_i$  (de largeur  $w_i$  et de hauteur  $h_i$ ) et  $z_j$  (de largeur  $w_j$  et de hauteur  $h_j$ ) si  $I_V \geq \lambda \cdot \min(w_i, w_j)$  ou  $I_H \geq \lambda \cdot \min(h_i, h_j)$ .

La figure 4 illustre le concept de visibilité sur un exemple jouet sur lequel la valeur de  $\lambda$  est fixée à 0,5. Sur cette figure,  $z_1$  et  $z_2$  sont verticalement visibles l'un de l'autre car la largeur de  $I_V$  est plus grande que la moitié de la largeur du moins large des deux rectangles. La même configuration intervient pour  $(z_2, z_3)$  et  $(z_1, z_4)$ . À l'inverse,  $z_1$  et  $z_3$  ne sont pas visibles. Les arcs modélisant la relation de visibilité sont étiquetés par les distances horizontale  $H_D$  et verticale  $V_D$  entre les centres de gravité des rectangles, normalisées par rapport à la taille du document.

Au final, la structure du document est modélisée par un graphe  $G = (V, E, \mu, \nu)$  où

- $V$  est l'ensemble des nœuds de  $G$  représentant les rectangles ;
- $E \subseteq V \times V$  est l'ensemble des arcs représentant les relations de visibilité entre les rectangles ;
- $\mu : V \rightarrow \mathcal{L}_V$  est la fonction d'étiquetage des nœuds avec  $\mathcal{L}_V = (I, Q, W, H)$  ;
- $\nu : E \rightarrow \mathcal{L}_E$  est la fonction d'étiquetage des arcs avec  $\mathcal{L}_E = (H_D, V_D)$ .

Une fois les graphes construits, nous pouvons appliquer la recherche d'isomorphisme pour localiser l'information recherchée. La section suivante présente les expériences que nous avons menées afin d'évaluer notre approche et les résultats obtenus.

## 5. Évaluation expérimentale

Les expériences décrites dans cette section visent à valider la pertinence des graphes de visibilité définis précédemment pour la tâche de localisation d'information dans des images de formulaires. Le protocole expérimental proposé simule ce contexte applicatif. Il est présenté les paragraphes qui suivent, avant de présenter et commenter les résultats.

### 5.1. Protocole expérimental

La base sur laquelle nous opérons notre évaluation est composée de 130 images de formulaires commerciaux ou administratifs répartis en 8 classes dont certaines sont données sur la figure 2. L'effectif de chaque classe est donnée en en-tête du tableau 1. Comme indiquée précédemment, la classe de chaque document est supposée connue au moment de la recherche puisqu'elle a été déterminée par un classifieur. De ce fait les expériences sont menées indépendamment dans chaque classe.

Pour chaque classe de documents, 36 graphes de visibilité différents sont extraits. Ces 36 graphes correspondent à autant de paramétrages différents afin de déterminer l'impact de chacun des paramètres et de déterminer les meilleurs paramétrages. Les paramètres étudiés sont :

- l'impact du prétraitement par filtrage *EPSF* ;
- l'impact du prétraitement par inpainting ;

- le paramètre  $k$  de l’algorithme des  $k$ -moyennes ;
- l’impact de l’espace colorimétrique de travail :  $RVB$ ,  $YIQ$  ou  $CIELab$ .

Pour simuler la définition des régions d’intérêt par l’utilisateur, des graphes requêtes sont générés aléatoirement de la façon suivante. Dans un premier temps, le document d’apprentissage est sélectionné aléatoirement parmi les documents de la classe. Un nœud germe représentant la région d’intérêt<sup>1</sup> est alors sélectionné également aléatoirement au sein de la représentation structurelle du document d’apprentissage. L’ensemble des nœuds est alors complété avec les nœuds directement connectés au nœud germe. Enfin, le graphe requête est construit comme étant le sous-graphe de la représentation structurelle du document d’apprentissage induit par l’ensemble des nœuds sélectionnés. Les instances du graphe requête sont ensuite recherchées dans les représentations structurelles des documents autres que celui sélectionné pour l’apprentissage. Afin d’évaluer les capacités en généralisation de l’approche proposée, ce processus est reproduit 50 fois au sein de la classe de sorte que 50 graphes requêtes soient générés. Au total, cela nous a conduit à opérer 6100 recherches d’isomorphisme de sous-graphe pour chacune des 36 configurations d’extraction.

Pour chaque recherche, l’algorithme de recherche d’isomorphisme fournit l’ensemble des mises en correspondances entre les nœuds et entre les arcs et le coût associé à cette mise en correspondance. Pour la recherche d’isomorphisme, nous avons utilisé la méthode présentée dans (Lerouge *et al.*, 2016) qui retourne de façon exacte le sous-graphe du graphe cible dont le coût d’association au graphe requête est minimal. Cette méthode est une extension de la méthode (Le Bodic *et al.*, 2012) qui permet l’association des nœuds (resp. arcs) du graphe modèle à un nœud (resp. arc) dans le graphe cible. Un coût est associé à chaque association potentielle et l’association effective est celle qui minimise la somme des coûts d’association. En outre, la méthode décrite dans (Lerouge *et al.*, 2016) autorise moyennant des pénalités l’association des nœuds (resp. arcs) du graphe requête à des nœuds (resp. arcs) factices créés si besoin dans le graphe cible. Cette méthode de recherche d’isomorphisme entre un graphe requête est tolérante aux erreurs aussi bien sur l’étiquetage, via des coûts de substitution, qu’à des différences structurelles, via des coûts de création. Ces différences d’étiquetage sont classiques en reconnaissance de forme, les valeurs numériques des caractéristiques pouvant varier d’un document à l’autre. Les différences structurelles interviennent en cas de sur-détection, non-détection, sur-segmentation ou sous-segmentation de zones. Finalement, le problème d’isomorphisme de sous-graphe est modélisé comme un programme linéaire en nombres entiers et peut être résolu par un solveur mathématique.

---

1. Pour des questions liées à l’évaluation, la région d’intérêt est nécessairement un nœud du graphe correspondant à un rectangle coloré. Cependant, l’approche peut être généralisée à n’importe quelle zone de l’image

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} & \left( \sum_{i \in V_1} \sum_{k \in V_2} x_{i,k} \cdot (c(i \rightarrow k) - c(i \rightarrow \epsilon)) + \sum_{i \in V_1} c(i \rightarrow \epsilon) \right. \\ & \left. + \sum_{ij \in E_1} \sum_{kl \in E_2} y_{ij,kl} \cdot (c(ij \rightarrow kl) - c(ij \rightarrow \epsilon)) + \sum_{ij \in E_1} c(ij \rightarrow \epsilon) \right) \quad [1a] \end{aligned}$$

s. c.

$$\sum_{k \in V_2} x_{i,k} \leq 1 \quad \forall i \in V_1 \quad [1b]$$

$$\sum_{i \in V_1} x_{i,k} \leq 1 \quad \forall k \in V_2 \quad [1c]$$

$$\sum_{l \in V_2, kl \in E_2} y_{ij,kl} \leq x_{i,k} \quad \forall ij \in E_1, \forall k \in V_2 \quad [1d]$$

$$\sum_{k \in V_2, kl \in E_2} y_{ij,kl} \leq x_{j,l} \quad \forall ij \in E_1, \forall l \in V_2 \quad [1e]$$

$$x_{i,k} \in \{0, 1\} \quad \forall i \in V_1, \forall k \in V_2 \quad [1f]$$

$$y_{ij,kl} \in \{0, 1\} \quad \forall ij \in E_1, \forall kl \in E_2 \quad [1g]$$

Dans cette formulation le graphe  $G_1 = (V_1, E_1, \mu_1, \nu_1)$  est le graphe requête et le graphe  $G_2 = (V_2, E_2, \mu_2, \nu_2)$  est le graphe cible. Les variables d'association  $x_{i,k}$  sont à 1 lorsque le nœud  $i$  du graphe  $G_1$  est associé au nœud  $k$  du graphe  $G_2$  et à 0 sinon. De même, les variables d'association  $y_{ij,kl}$  sont à 1 lorsque l'arc  $ij$  du graphe  $G_1$  est associé à l'arc  $kl$  du graphe  $G_2$  et à 0 sinon. Les fonctions  $c(\cdot)$  sont les coûts associés aux opérations d'édition élémentaires (substitution et création).

L'évaluation de notre approche requiert de vérifier que les nœuds ont été correctement associés lors de la recherche d'isomorphisme de sous-graphe. Nous examinons donc la sortie proposée par l'algorithme d'isomorphisme et notamment la liste des nœuds effectivement associés  $S_{ij}$  et celle des nœuds créés  $C_i$ . Nous calculons alors la distance de Jaccard  $Jacc(Z_i, Z_j)$  entre les pairs de zones  $Z_i$  et  $Z_j$  des associations  $S_{ij}$ . Nous considérons qu'une association est correcte si la distance  $Jacc(Z_i, Z_j)$  est supérieure à un paramètre  $\alpha$  que nous avons fixé à la valeur 0,1. Les associations considérées correctes sont notées  $M_{ij}$ . Nous calculons ensuite le score de mise en correspondance de la requête visant à rechercher le sous-graphe  $q_i$  and le graphe  $t_j$  comme étant le taux de mises en correspondance correctes selon la formule donnée par l'équation (2).

$$match(q_i, t_i) = \frac{|M_{ij}|}{|S_{ij}| + |C_i|} \quad [2]$$

Enfin, pour chaque classe, nous calculons la performance de la recherche de zone selon l'équation (3). Dans cette formule,  $n$  correspond au nombre de documents dans la classe. Nous considérons que le sous-graphe a été correctement retrouvé si  $match$  est supérieur à un paramètre  $\beta$  dont nous avons fixé la valeur à 0,1 dans nos expériences.

$$Perf = \frac{\sum_{i=1}^{50} \sum_{j=1}^{n-1} \mathbf{1}_{match(q_i, t_j) > \beta}}{50 * (n - 1)} \quad [3]$$

## 5.2. Résultats

Les résultats obtenus sont donnés dans le tableau 1. Ce tableau présente les performances obtenues pour chacune des classes et pour chaque configuration de traitement, une configuration de traitement étant définie par le fait que le filtre EPSF ait été appliqué ou non, que l'inpainting ait été appliqué ou non, le choix de l'espace couleur ( $RVB$ ,  $YIQ$  ou  $CIELab$ ) et la valeur de  $k$  utilisée pour l'algorithme des  $k$ -moyennes ( $k \in \{2, 3, 4\}$ ). Ce paramétrage définit 36 configurations différentes. La dernière colonne du tableau 1 indique la valeur de  $Perf$  sur la globalité de la base.

Pour aider à la lecture de ce tableau, la meilleure performance sur l'ensemble des configurations est indiquée en gras. On peut observer que la meilleure performance est toujours obtenue avec une configuration qui inclut le traitement d'inpainting. Nous remarquons également que la qualification en deux couches couleurs ( $k = 2$ ) conduit à la meilleure performance dans la plupart des cas. Etant données ces observations, nous émettons l'hypothèse que, dès lors que la couche texte a été éliminée, la séparation en deux couches couleurs conduit aux meilleures performances. Il est en revanche difficile de tirer des conclusions quant à la nécessité d'appliquer ou non le filtrage EPSF ou sur l'espace colorimétrique le plus adapté. En effet, les meilleures performances sont obtenues avec des configurations différentes de ces deux derniers paramètres selon les classes. Nous émettons l'hypothèse que les configurations conduisant aux meilleurs performances diffèrent selon les classes car les classes diffèrent entre elles notamment du point de vue du nombre et de la nature des couleurs qui les composent et sur la densité du texte. Ce phénomène illustre le besoin de pouvoir déterminer automatiquement la configuration la plus adaptée pour chaque classe de documents.

Au global, la performance la plus adaptée sur la globalité de la base permet de retrouver la zone recherchée dans 86,9% des cas. Cependant, il existe de grandes disparités selon les classes. En effet, pour certaines classes la zone est retrouvée dans 100% des cas alors qu'elle n'est retrouvée que de 72% des cas pour les classes les plus difficiles.

Les figures 5.2, 5.2 et 5.2 donnent quelques exemples de localisation de zones. Ces figures présentent l'interface utilisateur sur laquelle le document de gauche correspond au document requête. Sur ce document requête, le cadre vert indique la zone recherchée, les zones bleues sont celles appartenant au contexte de voisinage. Le do-

**Tableau 1. Résultats expérimentaux**

Prétraitement		Quantification		Classes							Base		
EPSF	Inpaint	K	Espace couleur	1(11)	2(31)	3(29)	4(14)	5(12)	6(11)	7(9)		8(13)	
Oui	Oui	2	RGB	58,85%	59,57%	75,44%	98,00%	99,33%	60,90%	69,33%	50,62%	69,67%	
			YIQ	<b>83,45%</b>	89,35%	92,82%	<b>99,28%</b>	98,16%	44,72%	<b>92,66%</b>	58,29%	84,84%	
			Lab	76,00%	82,00%	<b>93,44%</b>	98,57%	<b>100,00%</b>	45,27%	90,00%	45,98%	81,25%	
		3	RGB	61,45%	65,54%	59,65%	90,29%	<b>100,00%</b>	78,54%	88,44%	42,92%	70,09%	
			YIQ	79,81%	47,74%	72,13%	90,28%	<b>100,00%</b>	44,18%	43,77%	58,39%	65,77%	
			Lab	68,54%	66,83%	84,96%	91,71%	97,16%	38,00%	56,22%	31,29%	69,70%	
		4	RGB	70,90%	58,70%	66,62%	90,42%	<b>100,00%</b>	39,63%	61,11%	61,07%	67,46%	
			YIQ	52,72%	55,87%	60,82%	84,85%	98,16%	37,63%	47,33%	49,53%	60,86%	
			Lab	49,63%	32,38%	69,51%	88,71%	98,16%	45,81%	48,44%	38%	57,07%	
		Non	2	RGB	37,09%	25,03%	53,86%	62,64%	70,16%	21,09%	34,88%	30,15%	41,34%
				YIQ	65,45%	59,87%	82,41%	92,00%	92,66%	49,09%	83,55%	37,07%	70,21%
				Lab	76,00%	47,41%	78,27%	86,28%	93,00%	27,09%	60,22%	36,30%	63,16%
	RGB			74,00%	51,54%	43,93%	90,14%	74,67%	45,09%	65,11%	50,30%	58,13%	
	YIQ			70,72%	62,96%	73,79%	87,71%	82,66%	40,36%	77,55%	51,07%	68,38%	
	Lab			62,36%	62,32%	72,48%	87,14%	85,00%	34,90%	59,77%	41,07%	64,64%	
	3		RGB	56,00%	61,93%	64,55%	87,85%	72,33%	38,72%	69,11%	24,92%	60,53%	
			YIQ	43,63%	52,00%	58,75%	92,14%	77,50%	36,90%	60,66%	36,46%	57,24%	
			Lab	56,36%	59,80%	64,55%	85,85%	75,16%	24,18%	65,33%	36,92%	59,84%	
			RGB	59,09%	48,95%	59,37%	87,71%	99,16%	61,09%	62,88%	39,28%	61,94%	
			YIQ	80,90%	<b>90,38%</b>	89,58%	99%	<b>100,00%</b>	<b>83,81%</b>	88,66%	55,21%	<b>86,90%</b>	
			Lab	77,63%	74,96%	83,37%	90,57%	<b>100,00%</b>	52,36%	85,77%	47,82%	77,08%	
	Non	Oui	RGB	55,45%	65,03%	59,65%	87,85%	98,33%	83,63%	86,44%	29,09%	68,00%	
			YIQ	74,70%	51,48%	85,37%	92,42%	<b>100,00%</b>	38,90%	36,44%	51,53%	67,78%	
			Lab	50,00%	55,03%	79,93%	89,14%	95,50%	40,36%	50,88%	32,92%	63,83%	
RGB			79,45%	59,41%	70,96%	86,42%	95,16%	41,81%	48,44%	<b>72,00%</b>	68,93%		
YIQ			51,27%	41,35%	67,31%	93,42%	99,66%	42,36%	38,66%	43,84%	59,10%		
Lab			56,18%	34,77%	65,44%	90,28%	96,50%	34,90%	44,88%	31,07%	55,43%		
Non		2	RGB	43,09%	22,83%	49,87%	55,31%	69,33%	31,81%	47,33%	23,84%	40,89%	
			YIQ	63,63%	60,83%	78,48%	82,42%	89,83%	35,09%	69,55%	30,46%	65,35%	
			Lab	58,00%	52,83%	64,82%	84,57%	84,66%	18,90%	64,66%	40,15%	58,95%	
		3	RGB	63,63%	48,12%	59,51%	90,14%	78,16%	70,18%	65,55%	41,36%	61,53%	
			YIQ	67,81%	72,45%	76,48%	85,85%	82,33%	40,18%	66,88%	46,46%	69,58%	
			Lab	53,09%	56,83%	67,79%	85,00%	80,50%	46,36%	42,66%	46,00%	61,16%	
4	RGB	49,27%	59,74%	66,82%	85,00%	67,66%	52,54%	53,77%	24,30%	59,20%			
	YIQ	56,00%	53,74%	62,27%	83,28%	79,83%	36,90%	59,11%	46,32%	59,49%			
	Lab	52,90%	57,87%	50,55%	81,28%	74,83%	40,90%	59,33%	40,00%	56,66%			

cument de droite est le document cible. Sur ce document, les cadres rouges indiquent les zones de couleur homogènes qui ont été détectées. Les zones bleues correspondent à celles qui ont été mise en correspondance par l’algorithme de recherche d’isomorphisme de sous-graphe avec celle modélisant le contexte de voisinage de la zone recherchée. Enfin, la zone verte est la zone correspondant à la zone recherchée.

## 6. Conclusion

Dans cet article, nous avons présenté une approche pour la localisation de champs de saisie sur des images de formulaires commerciaux ou administratifs. Cette approche repose sur la description des images de documents à base de graphes dont les nœuds représentent les rectangles de couleur et les arcs représentent des relations de visibilité. À partir d’un graphe modélisant la zone recherchée et son voisinage, nous effectuons une recherche d’isomorphisme de sous-graphe tolérante aux erreurs. Les résultats obtenus illustrent que la performance de la recherche dépend du paramétrage de la chaîne d’extraction de la représentation structurelle. Nos travaux futurs vont porter sur la mise en place d’un système permettant de déterminer automatiquement des configurations parmi les plus performantes pour chacune des classes. D’autres perspectives peuvent également porter sur l’extraction de primitives graphiques autres que

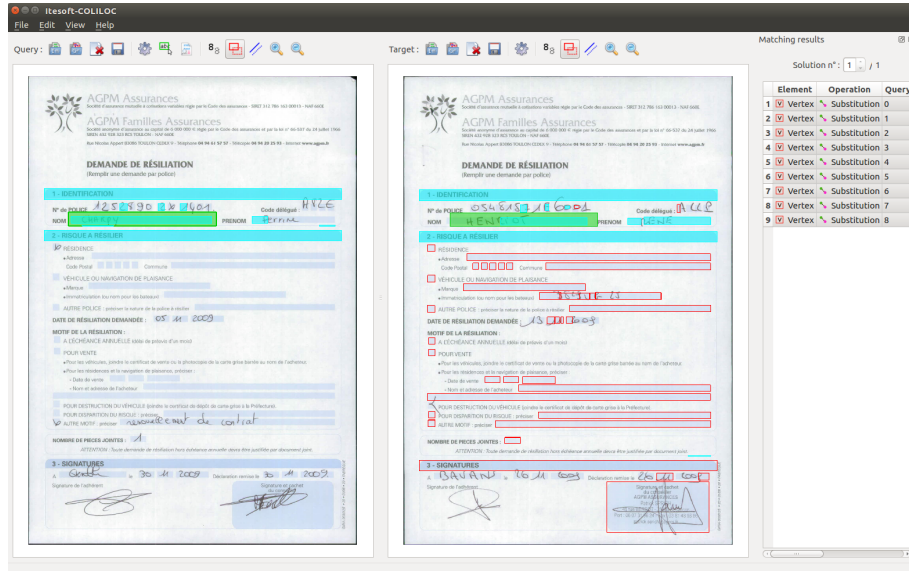


Figure 5. Exemple de localisation

les rectangles de couleur permettant d'enrichir la représentation dans le but d'améliorer la robustesse de la recherche des zones.

## 7. Bibliographie

- Cesarini F., Gori M., Marinai S., Soda G., « Informys : A flexible invoice-like form-reader system », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, n° 7, p. 730-745, 1998.
- Chen W.-Y., Chen S.-Y., « Adaptive page segmentation for color technical journals' cover images », *Image and vision computing*, 1998.
- Gaceb D., Lebourgeois F., Duong J., « Adaptative smart-binarization method for images of business documents », *Proceedings of the 12th International Conference on Document Analysis and Recognition*, p. 118-122, 2013.
- Hammami M., Héroux P., Adam S., Poulain d'Andecy V., « One-shot field spotting on colored forms using subgraph isomorphism », *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, p. 586-590, 2015.
- Hase H., Shinokawa T., Yoneda M., Suen C. Y., « Character string extraction from color documents », *Pattern Recognition*, vol. 34, n° 7, p. 1349-1365, 2001.
- Le Bodic P., Héroux P., Adam S., Lecourtier Y., « An integer linear program for substitution-tolerant subgraph isomorphism and its use for symbol spotting in technical drawings », *Pattern Recognition*, 2012.

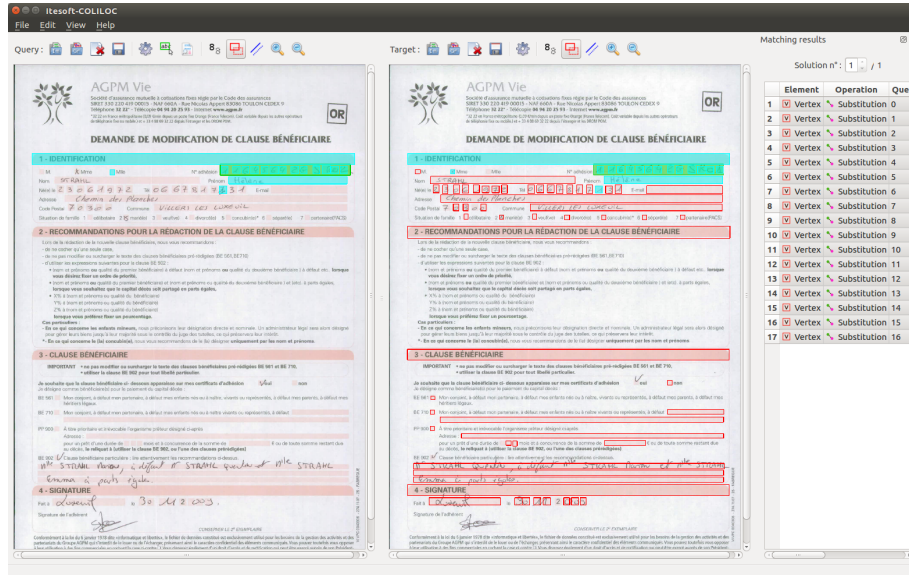


Figure 6. Exemple de localisation

- Lerouge J., Hammami M., Héroux P., Adam S., « Minimum cost subgraph matching using a binary linear program », *Pattern Recognition Letters*, vol. 71, p. 45 - 51, 2016.
- Nikoulaou N., Papamarkos N., « Color reduction for complex document images », *International Journal of Imaging Systems and Technology*, vol. 19, n<sup>o</sup> 1, p. 14-26, 2009.
- Peanho C. A., Stagni H., da Silva F. S. C., « Semantic Information », *Applied Intelligence*, vol. 37, n<sup>o</sup> 4, p. 543-557, 2012.
- Rosin P. L., « Measuring rectangularity », *Machine Vision and Applications*, vol. 11, n<sup>o</sup> 4, p. 191-196, 1999.
- Rusiñol M., Benkhelallah T., d'Andezy V. P., « Field extraction from administrative documents by incremental structural templates », *Proceedings of the 12th International Conference on Document Analysis and Recognition*, p. 1100-1104, 2013.
- Sgarbi E. M., Mura W. A. D., Nikolas M., Facon J., Ayala H. A. L., « Restoration of old document images using different color spaces », *Proceedings of the International Conference on Computer Vision Theory and Applications, VISAPP*, vol. 1, p. 82-88, 2014.

Matching results

Solution n°: 1 / 1

Element	Operation	Query Index	Target Index	Cost
1	V. Vertex	Substitution 0	75	0.00391572
2	V. Vertex	Substitution 1	72	0.00392416
3	V. Vertex	Substitution 2	74	0.00463826
4	V. Vertex	Substitution 3	79	0.00292446
5	V. Vertex	Substitution 4	140	0.0025687
6	V. Vertex	Substitution 5	73	0.00318549
7	V. Vertex	Substitution 6	65	0.00326564

**Figure 7.** Exemple de localisation