Détection de logos pour l'annotation d'images de presse

Pierre Tirilly[1] Vincent Claveau[1] Patrick Gros[2]

[1] CNRS-IRISA
[2] INRIA Rennes - Bretagne Atlantique

Campus de Beaulieu
35042 Rennes, France
pierre.tirilly@irisa.fr

Résumé

Dans cet article, nous proposons une méthode d'annotation d'images tirées d'un corpus d'articles de presse. Pour contourner le problème du fossé sémantique posé par l'utilisation de descripteurs de bas-niveau, nous mettons en relation des indices visuels de haut-niveau extraits des images (présence de logos ou de panneaux) et des indices textuels issus de l'analyse du texte des articles (un sous-ensemble des entités nommées – noms d'entreprises…–). Dans un premier temps, nous proposons un détecteur de logos et de panneaux reposant sur le modèle des mots visuels proposé par Sivic et Zisserman [1], dont les descripteurs sont particulièrement adaptés à la détection de ce type d'objets. Notre détecteur ne nécessite qu'une phase d'apprentissage légère (données faciles à obtenir, peu de temps de calcul), et permet de détecter les logos de manière rapide et avec une bonne précision. Nous l'évaluons sur 413 images extraites du corpus initial d'articles de presse. Dans un second temps, nous présentons la méthode d'annotation d'images. Nous associons aux images où sont détectés des logos les entités nommées extraites de l'article accompagnant l'image et susceptibles de décrire ces logos. Nous montrons que cette méthode d'annotation est rapide et ne pose pas de problèmes de passage à l'échelle. Nous l'évaluons sur le corpus d'articles de presse complet, qui contient plus de 40000 images.

Mots Clef

Détection de logos, détection de panneaux, mots visuels, annotation d'images, entités nommées, indexation texte-image

Abstract

In this paper, we propose a new method to annotate news images. To avoid the semantic gap problem due to the use of low-level visual features, we associate high-level visual features (presence of logos and panels) and high-level textual features (a subset of named entities). In one part, we propose a logo and panel detector based on the bag of visual words model as it was proposed by Sivic and Zisserman [1]. The descriptors provided by this model are well suited to the detection of this kind of objects. Our detector requires only a simple learning stage (training data obtained easily, little computation time) and detects logos quickly with a good precision. We evaluate the detector on 413 images from a news corpus. In a second part, we present our annotation method. We associate images that contain logos or panels and some named entities extracted from the news text coming with the images. This annotation method is very fast and is fitted to large-scale applications. We evaluate it on a news corpus that contains more than 40,000 images.

Keywords

Logo detection, board detection, visual words, image annotation, named entities, text-image indexing

1 Introduction

La démocratisation des moyens d'acquérir, échanger et stocker des documents numériques a provoqué une explosion de la quantité d'images disponibles, pour les professionels comme pour les particuliers. Mais comment classer et consulter efficacement de telles quantités de données ? Les techniques d'indexation d'images tentent de résoudre ce problème. Parmi celles-ci, les techniques d'annotation d'images par des mots-clefs sont particulièrement prisées, car elles permettent une formulation naturelle des requêtes par les utilisateurs, mais ces techniques se heurtent au problème du fossé sémantique, qui désigne l'écart existant entre les caractéristiques de bas-niveau que l'on sait extraire des images (couleur, texture…) et le contenu sémantique réel des images, représenté par les mots-clefs. De nombreuses techniques basées sur l'apprentissage artificiel ont montré une certaine réussite pour l'indexation de bases artificielles d'images, mais passent mal à l'échelle des bases d'images réelles, en raison de la taille et de la complexité de ces dernières.

D'autres méthodes proposent d'exploiter les données textuelles accompagnant les images pour en extraire des mots-clefs intéressants [2, 3, 4]. Dans cet article, nous proposons une méthode d'annotation d'images de presse exploitant les articles que ces images illustrent. Nous contournons le problème du fossé sémantique en mettant en relation des indices visuels de haut-niveau, les logos et panneaux, avec des indices textuels de haut-niveau, les entités nommées (noms d'organisation, d'entreprises...). En effet, une image contenant le logo d'une entité donnée (organisation, entreprise...) illustre généralement un article au sujet de cette entité. Dans un premier temps, nous proposons une technique de détection des logos et de panneaux, dont le contenu visuel est proche des logos, rapide, indépendante de la forme des logos et de leur prise de vue, et nécessitant une phase d'apprentissage très légère. Ensuite, nous utilisons ce détecteur conjointement avec un détecteur d'entités nommées dans les textes pour réaliser l'annotation des images par l'entité nommée correspondant au logo détecté. Nous validons notre approche sur un corpus réel d'articles de presse contenant plus de 40000 images.

Dans une première partie, nous donnons un aperçu des travaux proches de ceux que nous présentons ici. Puis nous décrivons le fonctionnement de notre détecteur. Nous présentons ensuite notre méthode d'annotation, puis les expériences que nous avons menées pour tester notre approche. Enfin, nous concluons et donnons quelques perspectives dans la dernière partie.

2 Travaux antérieurs

2.1 Détection de logos

La détection de logos dans les images naturelles a été peu abordée dans la littérature jusqu'à présent. Il existe néanmoins des travaux connexes. En matière de détection de logos, certains travaux visent à détecter les logos dans des documents scannés [5, 6]. Ce problème est plus simple que celui que nous traitons ici car il nécessite uniquement de différentier les logos des zones de texte. Certains travaux s'intéressent également à la détection de logos dans des vidéos [7, 8]. Dans ce cas, le problème est là aussi plus simple car les logos détectés sont en général statiques, au sein d'images en mouvement. Un article récent présente des travaux plus proches des nôtres [9]. Les auteurs utilisent des descripteurs SIFT pour reconnaître des images de logos contenus dans une base. Leur problème est cependant différent de celui que nous traitons car il ne nécessite pas de pouvoir découvrir de nouveaux logos, mais uniquement des logos connus. Un dernier ensemble de travaux proches de ceux que nous présentons ici sont les méthodes de détection de texte dans les images naturelles. Ces travaux sont plus nombreux [10, 11, 12] et proches des nôtres car les logos contiennent souvent une ou plusieurs lettres. Cependant, leur tâche est simplifiée par le fait que les lettres à identifier ont chacune un aspect et des proportions constants, alors que notre tâche nécessite de détecter des logos de taille, proportion et aspect variables.

2.2 Annotation d'images à partir de textes

Les techniques d'annotation d'images sont très souvent évaluées sur des collections catégorisées de type Corel où les images sont annotées par des mots-clefs, une partie de ces images servant de données d'apprentissage. Il existe néanmoins quelques travaux exploitant le texte accompagnant les images, dans des collections d'images issues d'applications réelles, pour annoter les images avec des mots-clefs. Parmi ceux-ci, on distingue des méthodes utilisant des techniques de traitement automatique du langage pour extraire précisément les mots pertinents (patrons, entités nommées, Wordnet...) [13, 2, 14] et des techniques utilisant plutôt des techniques statistiques utilisées en recherche d'information (pondérations, LSA...) [4, 3, 15, 16]. Parmi ces travaux, la plupart reposent sur l'utilisation des légendes accompagnant les images [2, 16, 4, 14], d'autres disposent de textes décrivant explicitement et précisément les images [13], ou utilisent les tags HTML de pages web pour extraire les parties de textes dédiées aux images, mais ils n'opèrent pas de recherche de termes d'annotation au sein de texte plus généraux illustrés par les images. À notre connaissance, seul [3] extrait par analyse du texte les parties de celui-ci décrivant l'image, mais leur méthode n'est évaluée que sur un faible nombre de documents (300) cantonnés à un thème précis (le terrorisme). Comme ce dernier, nous cherchons à exploiter l'intégralité du texte accompagnant chaque image pour en extraire des termes d'annotation, mais notre corpus est beaucoup plus grand et varié.

3 Détection de logos

3.1 Modèle en sac de mots visuels

Nous utilisons le modèle en sac de mots visuels tel qu'il a été défini par Sivic et Zisserman [1]. Ce modèle permet de décrire les images comme des ensembles de régions élémentaires appelées mots visuels. Le processus de description d'un image est le suivant (voir figure 1) :

1. Construction d'un vocabulaire visuel :
   (a) Détection de régions d'intérêt (ellipses rouges sur la figure 1) et description de chaque région par un vecteur numérique.
   (b) Quantification des descripteurs de régions d'intérêt à l'aide d'un algorithme de clustering. Chaque cluster ainsi formé correspond à un mot visuel.
2. Description de l'image :
   (a) Extraction de descripteurs locaux de l'image.
   (b) Association de chaque descripteur local à son centroïde le plus proche dans le vocabulaire. Chaque descripteur local est ainsi associé à un mot visuel.
   (c) Représentation de l'image comme un ensemble de mots visuels ou un vecteur d'occurrences de mots visuels.

Cette représentation des images est utilisée dans un grand nombre de travaux d'indexation, d'annotation et de classification d'images car elle utilise des éléments locaux du contenu des images et peut être manipulée de manière très efficace en termes de temps de calcul. Elle est adaptée à la détection de logos et de panneaux pour plusieurs raisons. D'une part, l'aspect local des mots visuels dans l'image permet de délimiter les zones contenant les objets recherchés. D'autre part, les descripteurs locaux utilisés dans ce genre d'approche permettent de différencier les zones d'images complexes des zones plus uniformes, ou des zones se situant à la frontière de régions uniformes. Les logos, qui sont généralement simples et composés d'aplats pour en faciliter l'identification, correspondent à ces dernières. Enfin, la phase de quantification des descripteurs (clustering générant les mots visuels) peut faciliter la généralisation, et donc la découverte de logos inconnus.

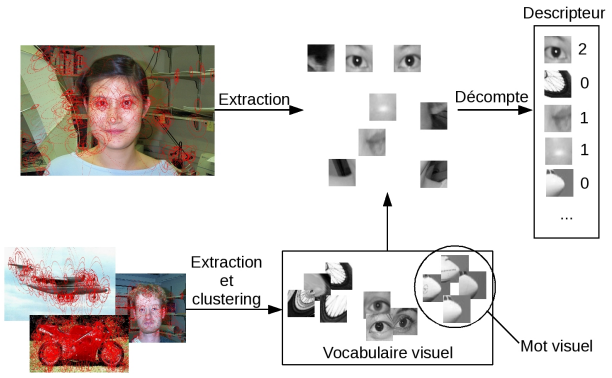

FIG. 1 – Construction d'un sac de mots visuels

3.2 Algorithme de détection

L'algorithme 1 décrit notre technique de détection de base. Nous nous appuyons sur les mots visuels de l'image pour construire les régions susceptibles de contenir un logo. Par rapport aux stratégies classiques utilisant des régions candidates de taille fixée à l'avance [17], notre méthode ne nécessite pas d'*a priori* sur la forme à détecter. Ce point est essentiel car les logos peuvent avoir des proportions variables en fonction de leur aspect et de la prise de vue des images. Un score est calculé pour chaque région candidate en fonction des mots visuels qu'elle contient. À chaque étape, l'algorithme retient la région dont le score est le plus élevé. Si ce score est inférieur au seuil de détection fixé, il n'existe plus de zone intéressantes, la détection s'arrête. Sinon, la région est conservée, puis les mots visuels qu'elle contient sont retirés de l'ensemble des mots visuels de l'image. Cette élimination permet d'éviter de détecter de nombreuses régions se chevauchant.

3.3 Calcul du score de détection

Pour chaque candidat logo, un score est calculé (fonction `score` de l'algorithme 1) puis comparé à un seuil pour savoir si ce candidat est effectivement un logo ou non. Notre

M_I : **mots visuels de l'image I**
L_I : **logos détectés dans l'image I**
logo_trouvé : **booléen**
$L_I \leftarrow \emptyset$
logo_trouvé $\leftarrow$ vrai
Tant que (logo_trouvé) **faire**
  L : **logos candidats**
  $L \leftarrow \emptyset$
  Pour $(m_i, m_j) \in M_I \times M_I$ **faire**
    $L \leftarrow L \cup \{\text{nouveau\_logo}(M_I, m_i, m_j)\}$
  Fin Pour
  l_{max} : **logo candidat**
  $l_{max} \leftarrow \text{argmax}_{l \in L}(\text{score}(l))$
  Si (score$(l_{max}) >$ MIN_SCORE) **Alors**
    $L_I \leftarrow L_I \cup \{l_{max}\}$
    $M_I \leftarrow M_I - \text{mots}(l_{max})$
  Sinon
    logo_trouvé $\leftarrow$ faux
  Fin Si
Fait
Retourner L_I

Algorithme 1: Algorithme de détection

calcul de score s'inspire de la théorie bayésienne. Soit w un mot visuel et L l'événement "est un logo", alors la quantité

$$q(w) = \frac{\Pr(w|L)}{\Pr(w)} \quad (1)$$

indique si la présence du mot w est en faveur de la présence d'un logo ($q(w) > 1$) ou non ($q(w) < 1$). Un cadre bayésien strict nécessiterait d'y inclure une probabilité a priori $\Pr(L)$. Nous choisissons de l'ignorer car l'estimation de cette probabilité nécessiterait de disposer de données adaptées et ne ferait pas réellement sens en dehors de ces données.

La phase de quantification des mots visuels pouvant induire du bruit dû aux erreurs de quantification, $q(w)$ peut être corrigé pour prendre en compte la fiabilité de l'occurrence du mot visuel. Nous nous basons pour cela sur la distance du descripteur à son centroïde le plus proche lors de l'assignement aux clusters, durant l'étape de construction des mots visuels de l'image. Si le descripteur est proche du centroïde, cette occurrence du mot visuel est fiable et il faut donc augmenter son score s'il est supérieur à 1, le diminuer sinon, et inversement si le descripteur est éloigné de son centroïde. Le score $q(w)$ peut alors être corrigé en $q_{dist}(w^k)$ de la manière suivante :

$$q_{dist}(w^k) = 1 + (q(w) - 1) * \frac{d_{min}(w)}{d(w^k)} \quad (2)$$

où w^k représente l'occurrence k du mot visuel w, $d(w^k)$ la distance du descripteur au centroïde pour w^k et $d_{min}(w)$

la distance minimale observée entre un descripteur et le centroïde pour le mot visuel k. Nous disposons ainsi d'une méthode améliorée de calcul des contributions des mots visuels. Les deux méthodes sont testées en Section 5.
En faisant l'hypothèse que les occurrences de mots visuels sont indépendantes les unes des autres, la probabilité qu'une région R d'une image contienne un logo peut être calculée ainsi :

$$S(R) = \prod_{w^k \in R} q'(w^k) \quad (3)$$

où $q'(w^k)$ peut être $q(w)$ ou $q_{dist}(w^k)$.
Nous proposons d'améliorer ce score par la prise en compte de la densité des mots visuels dans la région d'image R. En effet, les détecteurs ont tendance à détecter beaucoup de régions d'intérêt au niveau des logos et des écrits. Nous normalisons donc le score précédent en tenant compte de la densité de la région en points d'intérêt. Soient N_R le nombre de régions d'intérêt dans R et A_R sa surface (en pixels), nous normalisons le score de R ainsi :

$$S(R) = n. \prod_{w^k \in R} q'(w^k) \quad (4)$$

où n désigne l'un de ces scores de normalisation :

$$n_1 = \frac{N_R}{A_R} \quad (5) \quad \text{ou} \quad n_2 = \frac{N_R^2}{A_R} \quad (6)$$

La seconde normalisation n_2, utilisant le nombre de mots élevé au carré, a pour but de favoriser les fortes densités par rapport à n_1. Ces normalisations peuvent de plus être vues comme une compensation à la suppression des termes $\Pr(L)$ dans le calcul du score.
Le calcul du score nécessite de connaître les probabilités $\Pr(w)$ et $\Pr(w|L)$. Ces scores sont appris respectivement sur un corpus d'images quelconques et sur un corpus d'images de logos (les corpus que nous utilisons sont décrits en Section 5), en utilisant un lissage Laplacien pour éviter les probabilités nulles :

$$\Pr(w) = \frac{1 + \mathrm{Occ}(w)}{|V| + T_{occ}} \quad (7)$$

où $Occ(w)$ représente le nombre d'occurrences du mot visuel w dans le corpus considéré, $|V|$ la taille du vocabulaire et T_{Occ} le nombre total d'occurrences de mots visuels dans le corpus. Cette estimation peut être corrigée de la même manière que nous corrigeons le score d'un mot visuel, en prenant en compte à chaque occurrence d'un mot visuel sa distance au centroide :

$$\Pr_{dist}(w) = \frac{1 + \sum_{k=1}^{Occ(w)} \frac{d_{min}(w)}{d(w^k)}}{|V| + \sum_{v \in V} \sum_{j=1}^{Occ(v)} \frac{d_{min}(v)}{d(v^j)}} \quad (8)$$

3.4 Complexité

Phase d'apprentissage. Contrairement à de nombreux détecteurs basés sur de l'apprentissage supervisé (celui d'openCV [17] par exemple), notre détecteur nécessite une phase d'apprentissage très légère puisqu'il suffit d'estimer les probabilités $\Pr(w)$ et $\Pr(w|L)$ sur des ensembles d'apprentissage adaptés. Ces ensembles d'apprentissage peuvent eux-mêmes êtres obtenus à moindre coût (voir Section 5).

Phase de détection. La phase de détection nécessite de tester un grand nombre de régions candidates, nombre qui dépend directement de la quantité de mots visuels initialement détectés dans les images. Chaque région étant déterminée par un rectangle dont deux sommets sont des mots visuels, le nombre total de régions à tester dans une image contenant n mots visuels est de $\frac{n(n-1)}{2}$, soit une complexité en $O(n^2)$. Cette complexité peut handicaper lourdement l'efficacité de notre détecteur lorsque l'image contient un grand nombre de mots visuels. En pratique, deux méthodes permettent néanmoins de limiter l'impact de cette complexité quadratique : réduire le nombre de mots visuels dans l'image et imposer une limite de taille pour les régions candidates.

Réduction du nombre de mots. Une méthode efficace pour réduire le nombre de régions candidates lors de la détection est de limiter le nombre de mots visuels utilisés pour définir ces régions. Nous proposons trois façons de procéder :

- éliminer des mots en fonction de leur position : plusieurs mots visuels peuvent être détectés à des coordonnées identiques. Il est donc possible de réduire le nombre de régions à tester en ne considérant qu'une fois chaque coordonnée où apparaissent plusieurs mots visuels.
- éliminer les mots en fonction de $q(w^k)$: les occurrences de mots visuels telles que $q(w^k) < 1$ ont peu de chances de délimiter des logos. Nous ne les utilisons donc pas comme supports pour délimiter les régions candidates. Cette approximation peut faire rater la région optimale mais reste acceptable pour notre application où la détection prime sur la localisation.
- éliminer des mots en fonction de leur surface : les mots visuels sont détectés à plusieurs échelles, ils peuvent donc couvrir des surfaces variables de l'image. Les mots visuels qui couvrent une grande surface de l'image représentent généralement des zones complexes, et sont donc proches des mots visuels d'aire réduite détectés dans des zones complexes de l'image. Ils ne correspondent pas au type de mots visuels que nous recherchons pour détecter les logos. Les mots visuels dont la surface dépasse un seuil donné peuvent donc être éliminés, non seulement lorsqu'il s'agit d'établir les régions candidates, mais également pour la phase de calcul des scores des régions, à laquelle ils ajoutent du bruit.

Taille minimale des logos. Une autre manière de limiter le nombre de régions candidates est d'imposer un seuil de taille pour les régions détectables. L'intérêt est double. D'une part, nous évitons ainsi de tester de nombreuses zones de taille insignifiante (entre deux mots vi-

suels très proches). D'autre part, cela évite certains artefacts de détection, comme des zones très larges mais n'occupant qu'un ou deux pixels de hauteur. Ces régions sont détectées uniquement quand les deux mots visuels qui leur servent de support ont tous les deux un score d'apparition très élevé.

4 Annotation d'images contenant des logos

Nous utilisons une technique simple, similaire à celle que nous avions utilisée pour l'annotation d'images de personnes [18]. Le processus d'annotation est le suivant :

1. Détection de logos dans l'image.
2. Si l'image contient au moins un logo, détection des entités nommées dans le texte.
3. Élimination des entités nommées dont la fréquence (nombre d'occurrences) est inférieure à un seuil fixé à l'avance.
4. S'il reste des entités nommées, annotation de l'image par l'entité nommée la plus fréquente. En cas d'ambiguïté entre plusieurs entités nommées (fréquences identiques), l'annotation n'est pas réalisée.

Nous annotons une image par une seule entité nommée à la fois, même si plusieurs logos sont détectés dans l'image. En effet, si, dans le cas des personnes, la détection de deux visages sur une même photo indique nécessairement la présence de deux personnes différentes, rien n'indique ici que deux logos détectés sur une même image soient nécessairement distincts. Comme nous n'utilisons pas de reconnaissance des logos, deux images correspondant au même texte et contenant un logo recevront la même annotation. Nous utilisons le logiciel Némésis [19] pour détecter et catégoriser les entités nommées. Il détecte à la fois les entités nommées présentes dans un dictionnaire et à partir de patrons qu'il infère automatiquement. Nous n'utilisons pas toutes les entités nommées pour réaliser les annotations, mais seulement un sous-ensemble, car il n'est pas pertinent d'associer certaines d'entre-elles avec des logos (noms de personnes ou de lieux par exemple). Parmi les catégories d'entités nommées proposées par Némésis, nous nous restreignons aux suivantes : organisations, entreprises, marques ou produits, établissements, ensembles artistiques, événements. Comme cette méthode d'annotation est rapide et purement locale aux couples texte-image, elle ne pose aucun problème de passage à l'échelle pour de très grandes collections. L'évaluation de notre détecteur sur des images correspondant à un problème réel montre de plus que cette méthode fonctionne dans des contextes variés.

5 Évaluation du détecteur de logos

5.1 Conditions expérimentales

Données d'apprentissage. Notre détecteur nécessite de déterminer par apprentissage les probabilités d'occurrence des mots visuels. Les probabilités à priori $\Pr(w)$ d'occurrence des mots visuels sont calculées sur l'ensemble des 40947 images de presse dont nous disposons. Les probabilités conditionnelles $\Pr(w|L)$ d'occurrences des termes dans les logos nécessitent des données adaptées. Plutôt que d'annoter manuellement un sous-ensemble de notre corpus, nous avons téléchargé les 489 premières images retournées par le moteur de recherche `google images` en réponse à la requête "logo". Nous avons ainsi obtenu des données d'apprentissage à moindre coût, bien qu'elles soient légèrement bruités (logos répétés, présence d'images qui ne sont pas des logos).

Conditions d'évaluation. Nous utilisons 413 images extraites d'articles de presses téléchargés sur le site TV5.org entre mars et novembre 2006. 209 d'entre elles contiennent un ou plusieurs logos que nous avons délimités à la main. Nous comptons une détection réussie quand il y a intersection entre le zone détectée et la zone de la vérité-terrain. Cette métrique est adaptée à notre application, pour laquelle la détection est plus importante que la localisation. En faisant varier le seuil du score de détection, on peut obtenir différents points de rappel et précision et obtenir des courbes rappel-précision.

Construction des mots visuels. Nous utilisons le détecteur de points d'intérêt Hessian-affine. C'est l'un des plus utilisés pour construire des mots visuels et il offre de bonnes performances dans de nombreux cas [20]. Nous utilisons le descripteur local SIFT qui est le plus utilisé et offre d'excellentes propriétés [21]. Le vocabulaire est constitué grâce à une implémentation de l'algorithme de clustering *k-means* pour GPU.

5.2 Résultats

Taille du vocabulaire. La figure 2 montre l'évolution des performances lorsque la taille du vocabulaire varie. Nous remarquons que les performances diminuent fortement si la taille du vocabulaire augmente trop. Deux phénomènes peuvent expliquer cela. D'une part, augmenter la taille du vocabulaire favorise les erreurs d'assignation des descripteurs locaux aux mots visuels. D'autre part, la quantification des descripteurs locaux en mots visuels favorise la robustesse du système en le rendant insensible aux petites variations dans les descripteurs. Augmenter la taille du vocabulaire limite cette effet généralisateur.

Taille minimale des logos. La figure 3 montre les performances du détecteur lorsque l'on fait varier la taille minimale des régions détectées. Globalement, augmenter la taille minimale permet d'augmenter la précision du détecteur, puisque les plus petites régions, qui sont souvent litigieuses, sont ignorées. En contrepartie, augmenter la taille minimale fait diminuer le rappel : le rappel maximal diminue quand la taille minimale augmente. Ceci est logique puisqu'on s'interdit dès lors de détecter les plus petits logos des images. Dans le cadre de notre application d'annotation, ceci n'est pas nécessairement un problème car les logos les plus intéressants seront ceux qui occupent une

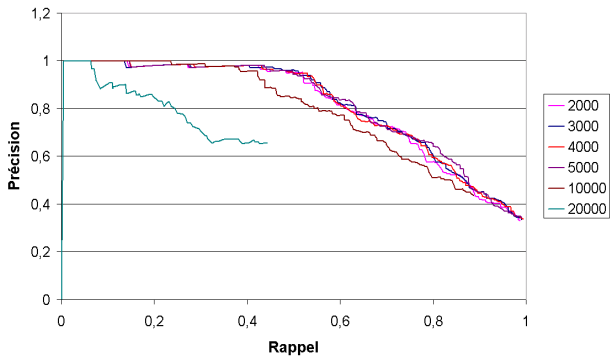

FIG. 2 – Performances du détecteur pour différentes tailles de vocabulaire

taille conséquente de l'image, et qu'il peut en exister des petits (publicités par exemple) qui ne sont pas pertinents pour indexer l'image.

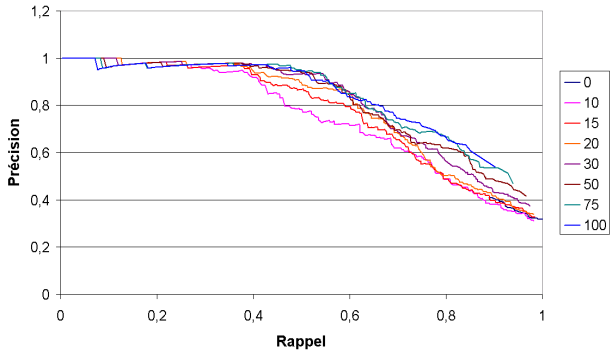

FIG. 3 – Performances du détecteur en fonction de la taille minimale des régions détectées (la taille indiquée est celle d'un coté du rectangle)

Élimination des mots superflus. La figure 4 montre l'évolution des performances du détecteur en fonction de la taille maximale des mots visuels pris en compte. Bien que les différences soient limitées, nous pouvons faire deux observations. D'une part, l'utilisation de tous les mots visuels n'apporte pas les meilleurs résultats, ce qui confirme que les mots visuels qui recouvrent une grande partie de l'image ne sont pas pertinents pour décrire les logos et génèrent du bruit. D'autre part, les plus mauvais résultats sont obtenus en éliminant un maximum de mots (aire de la région > 100). Il faut donc trouver un compromis sur la taille optimale des mots à conserver, qui se situe ici entre 300 et 500. En plus d'apporter un léger gain en termes de précision, ce filtrage permet de limiter fortement la nombre de mots visuels décrivant chaque image, et d'accélérer d'autant la vitesse de détection.

Scores de détection. La figure 5 montre les performances du détecteur selon que nous utilisions un score calculé sur les occurrences seules des mots visuels (`q-Pr-n2`), ou calculé en prenant en compte les

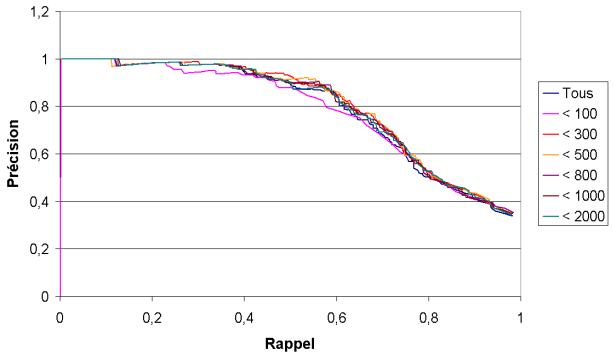

FIG. 4 – Performances du détecteur en fonction de la taille des mots visuels retenus

distances aux centroïdes dans le calcul du score seul (`qdist-Pr-n2`) ou du score et de l'estimation des probabilités (`qdist-Prdist-n2`), à chaque fois avec une normalisation $n2$. Nous voyons que l'introduction des distances aux centroïdes, qui permet de limiter l'influence des mots visuels dont l'assignation est moins fiable, est bénéfique aussi bien pour la phase de détection que pour celle d'apprentissage.

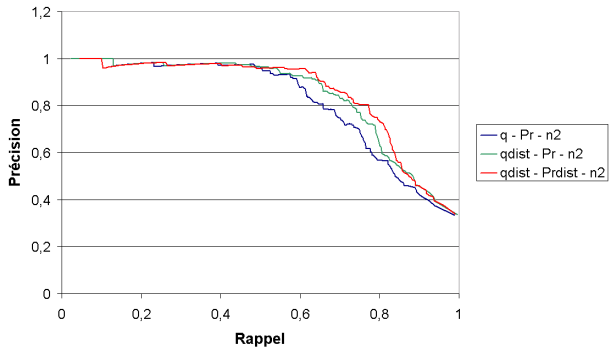

FIG. 5 – Performances du détecteur en fonction du type de score utilisé

La figure 6 montre l'influence de la normalisation du score utilisée. $n0$ signifie qu'aucune normalisation n'est utilisée. L'efficacité de la normalisation n_2 confirme qu'il est cohérent de favoriser les hautes densités. En revanche, la normalisation n_1 fonctionne moins bien. En effet, les normalisations ont tendance à favoriser les petites régions candidates, et n_1 ne favorisant pas suffisamment les hautes densités, elle détecte plus de petites régions non significatives.

Temps d'exécution. Le tableau 1 donne les temps d'exécution de la phase d'extraction et description des régions d'intérêt, de quantification des descripteurs, de détection des logos en utilisant tous les mots visuels (logos - base), et de détection des logos en limitant le nombre de mots visuels (logos - rapide) selon les techniques décrites en Section 3.4 (taille minimale des régions candidates : 50 pixels de côté, aire maximale des mots visuels : 500). On voit

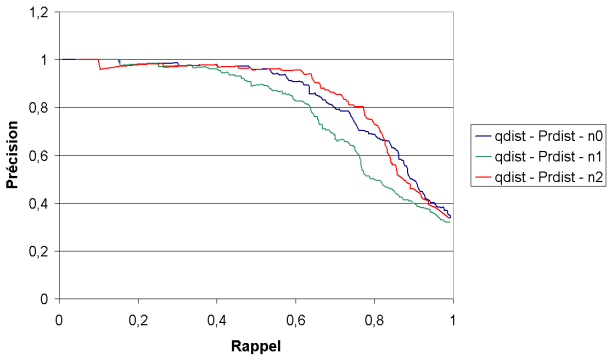

FIG. 6 – Performances du détecteur en fonction de la normalisation de score utilisée

que si la durée moyenne de la détection de base est prohibitive pour tout usage à grande échelle, la version accélérée offre des temps d'exécution raisonnables, en plus d'une meilleure précision. La phase d'apprentissage nécessite, hors prétraitements, (extraction et quantification) environ 1 minute.

extraction	quantification	logos - base	logos - rapide
0.54	0.05	257	1.63

TAB. 1 – Temps d'exécution moyen par image (en secondes) des prétraitements et du détecteur

6 Évaluation de l'annotation d'images

6.1 Conditions expérimentales

Données. Nous évaluons notre tâche d'annotation sur un corpus constitué de 26988 articles de presse, illustrés par 40947 images. Ce corpus a été constitué en téléchargeant des articles du site `www.tv5.org` entre avril et novembre 2006. Chaque image est de plus accompagnée d'une légende la décrivant. Nous utilisons cette légende comme vérité-terrain pour évaluer notre système.

Évaluation. Nous évaluons notre méthode d'annotation en nous basant sur les légendes des images. Nous considèrons qu'une annotation est juste dès que le terme d'annotation apparaît dans la légende.

Détection des logos. Nous utilisons notre détecteur de logos. Les paramètres du score sont q_{dist}, Pr_{dist} et n_2. Nous éliminons les mots visuels dont l'aire est inférieure à 500 et nous éliminons les régions candidates de côté inférieur à 50 pixels. Nous utilisons deux seuils de détection : 50 (qui correspond à une précision d'environ 98% dans nos tests) et 200 (précision de 80%).

Détection des entités nommées. Nous utilisons le logiciel Némésis [19] pour la détection des entités nommées.

6.2 Résultats

La figure 7 montre les performances en annotation pour deux seuils de détection de logos données. Chaque courbe est obtenue en faisant varier le seuil minimal d'acceptation des entités nommées. Les proportions d'images annotées sont calculées en fonction du nombre d'images retenues à l'issue de la détection de logos, respectivement 11144 pour le seuil de 50 et 6022 pour le seuil de 200. Nous observons qu'utiliser un seuil de détection plus élevé permet d'améliorer la précision du système pour un rappel donné, car cela évite un certain nombre de fausses détections, et donc d'annotations injustifiées. Néanmoins, dans l'absolu, le seuil le plus bas permet d'annoter un plus grand nombre d'images, au prix d'une légère perte de précision. Plus le seuil des entités nommées est élevé, plus les entités retenues sont fiables, nous obtenons donc une meilleure précision mais nous annotons une proportion moindre d'images. Pour les valeurs de ce seuil les plus élevées, la précision baisse légèrement. Ceci est lié au fait que, la proportion d'images annotées étant très faible, le retrait d'une annotation correcte modifie de manière notable la précision. Un examen qualitatif des résultats montre que Némésis, bien qu'offrant d'excellents résultats dans la détection de certaines catégories d'entités nommées (les noms de personnes par exemple), effectue de nombreuses erreurs de détection et de catégorisation sur les catégories dont nous nous servons. Ceci a un impact négatif sur notre système, tant en termes de précision que quantité d'images annotées.

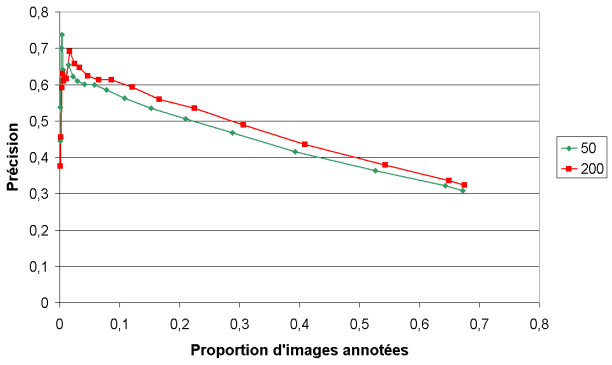

FIG. 7 – Performance de l'annotation d'images pour deux seuils de détection de logos donnés et un seuil de détection des entités nommées variable.

7 Conclusion et perspectives

Nous avons présenté une méthode efficace d'annotation d'images basée sur l'association d'un détecteur de logos et d'un détecteur d'entités nommées pour mettre en relation des images contenant des logos et les entités nommées décrivant ces logos. Cette méthode d'annotation est à la fois rapide et peu coûteuse en termes d'apprentissage, et ne pose pas de problèmes de passage à l'échelle.
Il existe plusieurs perspectives possibles à ce travail. Nous pourrions regrouper les occurrences d'un même logo sur des critères visuels pour affiner et propager les annotations,

Libération 8
CE 2
SCPL 2
Rotschild 2
Société Civile des
Personnels de Libération 1
Le Monde 1
Comité d'Entreprise 1

Arcelor 17
Mittal 5
ADAM 2
Goldman-Sachs 1
Institutional Shareholders
Services 1
ISS 1
Association française 1

FIG. 8 – Exemples d'annotations : les termes candidats sont indiqués avec leur fréquence, l'annotation retenue est en gras.

à la manière de ce qui se fait pour les visages [14]. L'analyse du texte peut également être affinée, en utilisant la reconnaissance d'acronymes : de nombreuses entités nommées utilisées ici sont des acronymes mais nous comptons séparément leurs occurrences et celles de leur définition. Nous envisageons également de traiter les ambiguités entre entités nommées de même fréquence, au lieu de les ignorer. Enfin, notre détecteur pourrait être utile en indexation vidéo, par exemple pour les journaux télévisés où des logos sont utilisés comme marqueurs d'un thème ou d'un sujet.

Références

[1] J. Sivic and A. Zisserman. Video Google : A text retrieval approach to object matching in videos. In *Proceedings of ICCV*, volume 2, pages 1470–1477, Nice, France, 2003.

[2] K. Deschacht and M.-F. Moens. Text analysis for automatic image annotation. In *Proceedings of the annual meeting of the Association of Computational Linguistics (ACL)*, 2007.

[3] T. Jiang and A.-H. Tan. Learning image-text associations. *IEEE Transactions on Knowledge and Data Engineering*, 21(2), 2009.

[4] T. Westerveld. Image retrieval : Content versus context. In *Proceedings of the Conference on Computer Assisted Information Retrieval (RIAO)*, 2006.

[5] S. Seiden, M. Dillencourt, S. Irani, R. Borrey, and T. Murphy. Logo detection in document images. In *Proceedings of the International Conference on Imaging Science, Systems, and Technology*, 1997.

[6] T. D. Pham. Unconstrained logo detection in document images. *Pattern Recognition*, 36 :3023–3025, 2003.

[7] H. Pan, B. Li, and M. I. Sezan. Automatic detection of replay segments in broadcast sports programs by detection of logos in scene transitions. In *Proceedings of ICASSP*, 2002.

[8] K. Meisinger, T. Troeger, M. Zeller, and A. Kaup. Automatic tv logo removal using statistical based logo detection and frequency selective inpainting. In *Proceedings of EUSIPCO*, 2005.

[9] S. Sanyal and S. H. Srinivasan. Logoseeker : a system for detecting and matching logos in natural images. In *Proceedings of ACM Multimedia*, 2007.

[10] J. Gao and J. Yang. An adaptative algorithm for text detection from natural scenes. In *Proceedings of CVPR*, 2001.

[11] J. Gllavata, R. Ewerth, and B. Freisleben. Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. In *Proceedings of ICPR*, 2004.

[12] C.-T. Wu. Embedded-text detection and its application to anti-spam filtering. Master's thesis, University of California, Santa Barbara, 2005.

[13] K. Pastra, H. Saggion, and Y. Wilks. NLP for indexing and retrieval of captionned photographs. In *Proceedings of the European Conference on Computational Linguistics*, 2003.

[14] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y.-W. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *Proceedings of the CVPR*, 2005.

[15] Z. Hua, X.-J. Wang, Q. Liu, and H. Lu. Semantic knowledge extraction and annotation for web images. In *Proceedings of ACM Multimedia*, 2005.

[16] H. Feng, R. Shi, and T.-S. Chuan. A bootstrapping framework for annotating and retrieving www images. In *Proceedings of ACM Multimedia*, 2004.

[17] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *Proceedings of the annual Symposium of the German Association for Pattern Recognition (DAGM)*, 2003.

[18] P. Tirilly, E. Martienne, V. Claveau, and P. Gros. Association d'un détecteur de visages et d'un détecteur d'entités nommées pour l'annotation automatique d'images. In *Rencontres Jeunes Chercheurs en Recherche d'Information (RJCRI)*, Saint-Etienne, France, 2007.

[19] N. Fourour. Nemesis, un système de reconnaissance incrémentielle des entités nommées pour le français. In *Actes de la conférence Traitement Automatique des Langues Naturelles (TALN)*, 2002.

[20] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 2005.

[21] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. on PAMI*, 27(10), 2005.