Rehaussement de la classification textuelle d'une base de données photographiques par son contenu visuel

Sabrina Tollari
Sous la direction de Hervé Glotin et Jacques Le Maitre
Laboratoire SIS - Equipe Informatique

9 juin 2003

Résumé

Ce travail consiste en une expérience ayant pour objectif de tester l'existence d'une cohérence entre l'indexation textuelle (un ensemble de mots-clés) d'une image et son indexation visuelle (attributs de couleurs et de formes). Cette expérience est menée sur un corpus de photos de presse indexées manuellement par un ensemble de mots-clés extraits d'un thésaurus structuré hiérarchiquement. Elle consiste à établir une classification de référence de ces photos à partir de leur indexation textuelle, considérée comme pertinente, puis à construire des indices textuels et visuels caractérisant ces classes et enfin à utiliser ces indices pour évaluer les performances obtenues par une recherche d'images combinant description textuelle et description visuelle. Nous obtenons par cette fusion 54% de gain de classification par rapport à l'information textuelle seule.

Introduction

La recherche d'images sur de grandes masses de données (Google : 425 millions d'images, Lycos Image Gallery : 18 millions, Altavista, ...) nécessite des outils adaptés pour, d'une part, extraire efficacement des descripteurs significatifs, et d'autre part, retrouver les images pertinentes. Les systèmes actuels permettent des recherches par mot-clés comme pour les systèmes de recherche d'informations textuelles, ou bien par une image requête ressemblant aux images que l'on recherche. Mais peu de systèmes tiennent compte à la fois du texte et du contenu visuel.

L'originalité de ce travail est d'étudier la cohérence entre l'indexation textuelle et l'indexation visuelle. En effet, l'indexation textuelle d'une image se fait manuellement ou bien automatiquement à partir de la légende de l'image, ou du texte qui l'entoure si cette image est insérée dans un document. Cela a pour conséquence que les images obtenues n'ont parfois aucun rapport avec la requête textuelle effectuée. Par exemple, la requête 'femme' et 'ouvrière' pourra donner des images de femmes travaillant, mais aussi des logos d'usines, des graphiques sur la population ouvrière... Nous nous sommes donc particulièrement intéressés à la façon d'améliorer des recherches d'images posées avec peu de mot-clé, mais rehaussées grâce à leur contenu visuel.

La première et la deuxième partie de ce rapport sont consacrées à quelques rappels sur la recherche d'information dans les images et sur les classifications en général. La troisième partie présente le corpus et les indices avec lesquels nous avons travaillé, ainsi que le protocole que nous avons mis en place pour réaliser cette expérience. Ensuite, la quatrième partie explique les méthodes utilisées pour construire une base de référence. Enfin, la cinquième partie présente les résultats que nous avons obtenus pour les classifications textuelles, visuelles et par la fusion des deux.

1 Recherche d'information sur des bases d'images

Un système de recherche d'information est constitué principalement de deux outils : l'outil d'indexation et l'outil de recherche.

La performance du système de recherche d'images dépend notamment de l'indexation des images qui doit permettre de retrouver la sémantique associée à l'image, du modèle de représentation qui doit être efficace et de la mesure de similarité qui doit permettre de retrouver les documents pertinents.

1.1 Indexation des images

Les premiers systèmes de recherche d'images proposaient un processus de recherche basé uniquement sur des descriptions textuelles, les images étant indexées manuellement. Mais l'indexation manuelle des images est une tâche fastidieuse et nécessite un temps non-négligeable. De plus, les résultats des interrogations dépendent de l'ensemble des mot-clés disponibles et de la subjectivité humaine. Contrairement aux documents textuels, l'image ne porte pas de sémantique directement accessible à la machine. C'est pourquoi depuis les années 90, de nombreux travaux de recherche ont été menés pour développer l'indexation automatique d'images par le contenu. La difficulté principale est d'extraire des descripteurs visuels suffisamment significatifs pour permettre de retrouver la sémantique associée à l'image.

Pour qu'un système de recherche d'images soit performant, il faut que l'indexation logique soit pertinente et que l'indexation physique permette un accès rapide aux documents recherchés [BK02].

Indexation logique L'indexation logique consiste à extraire et à modéliser les caractéristiques de l'image qui sont principalement la forme, la couleur et la texture. Chacune de ces caractéristiques pouvant être considérée pour l'image entière ou pour une région de l'image (localisation spatiale et segmentation en régions d'intérêt [Smi02]).

Forme Les techniques de modélisation sont classées en deux catégories. L'approche « contour » décrit une région au moyen des pixels situés sur son contour (figure 1 page 9 image 'edges'). L'approche « région » considère une région par rapport aux caractéristiques des pixels que cette région contient.

Couleur La couleur est en général définie au moyen de triplets numériques permettant de coder l'intensité de ces composantes. On distingue les espaces de couleurs définis selon des propriétés optiques comme RGB(Red, Green, Blue), et ceux basés sur la perception humaine des couleurs comme HSV(Hue, Saturation, Value). Pour modéliser la distribution des couleurs, on utilise généralement un histogramme indiquant l'intensité d'une couleur en abscisse, et le nombre de pixels en ordonnée (figure 1 page 9). Pour des questions de performance, on réduit souvent le nombre de couleurs (l'histogramme d'une image codée sur 24 bits possèdera 16 millions de couleurs).

Texture Une texture peut être caractérisée par les attributs de contraste, de directionalité, de régularité et de périodicité du motif. Dans le cadre de la recherche par le contenu, elle permet de distinguer des zones de couleurs similaires, mais de sémantique différente (par exemple, le bleu du ciel et le bleu de la mer).

Indexation physique L'indexation physique consiste à déterminer une structure efficace d'accès aux données pour trouver rapidement une information. De nombreuses techniques basées sur des arbres (arbres-B, arbres-R, arbres quaternaires, ...) ont été proposées, mais ces techniques souffrent de faiblesses dues notamment à la multi-dimentionnalité de l'indexation logique (recherche sur plusieurs caractéristiques à la fois : forme, couleur, texture,...) et au grand volume de données. Une technique visant à réduire l'espace de recherche pour améliorer

la rapidité est le clustering. Elle consiste à regrouper les données qui sont en relation en fonction de certains critères.

Les principaux systèmes actuels de recherche d'images par le contenu sont QBIC[Nib93], MARS[Rui97], VisualSeek[SC96], SurfImage[NMMB98] et NeTra[MM99]. Ils se basent principalement sur les caractéristiques visuelles, et n'utilisent que peu ou pas les indices textuels.

1.2 Modèle classique de représentation des documents

Pour les documents textuels, il existe des représentations efficaces. Soit $D = \{d_1, d_2, \ldots, d_i, \ldots, d_m\}$ un ensemble de documents (le corpus) et $T = \{t_1, t_2, \ldots, t_j, \ldots, t_n\}$ un ensemble de mot-clés décrivant (indexant) ces documents, les principaux modèles de recherche d'information pour des documents textuels sont le modèle booléen, le modèle vectoriel et le modèle probabiliste [BYRN99].

Exemple 1 *Nous présentons ici un exemple de corpus qui nous servira aussi dans la suite de ce rapport.* $T = \{Communication, Téléphonie, Multimédia, Média, Radio, Télévision\}$, $D = \{d_1, d_2, d_3\}$, $T_{d_1} = \{Communication\}$, $T_{d_2} = \{Radio\}$ *et* $T_{d_3} = \{Téléphonie, Radio\}$.

1.2.1 Modèle booléen

Chaque document d est représenté par un ensemble de termes non-pondérés sous la forme d'une expression logique :

$$d = t_{j_1} \wedge t_{j_2} \wedge \ldots \wedge t_{j_p}, \tag{1}$$

ce qui signifie que les termes $t_{j_1},\ t_{j_2},\ \ldots\ , t_{j_p}$ sont présents dans le document et que les autres termes t_{j_a} sont absents du document(noté $\neg t_{j_a}$).

La requête q est une expression booléenne dont les termes sont reliés par les opérateurs de conjonction($\wedge$), disjonction($\vee$) ou de négation($\neg$).

Un document d correspond à une requête q s'il vérifie l'implication logique :

$$d \ \rightarrow \ q. \tag{2}$$

Exemple 2 *Si l'on cherche les documents qui contiennent les mots Téléphonie ou Radio sans Multimédia, alors on a* $d_1 = Communication$, $d_2 = Radio$, $d_3 = Téléphonie \wedge Radio$, $q = Téléphonie \vee (Radio \wedge \neg Multimédia)$ *et* $d_1 \not\rightarrow q$, $d_2 \rightarrow q$, $d_3 \rightarrow q$. *La réponse à la requête q est* $\{d_2, d_3\}$.

Ce modèle ne permet pas de pondérer les mots dans le document. Un document est soit pertinent, soit non pertinent, par conséquent les réponses ne sont pas ordonnées. L'expression de la requête est dépendante de l'utilisateur, car le 'et' et le 'ou' ne corresponde pas tout à fait au '$\wedge$' et '$\vee$'.

Le modèle booléen est actuellement très peu utilisé, et souvent c'est sous la forme d'une extension du modèle[SFW83].

1.2.2 Modèle vectoriel

Le modèle vectoriel ([Sal71], [SL68]) représente un document d_i et une requête q par un vecteur dans un espace à n dimensions :

$$\vec{d_i} = (\omega_{1,i},\ \omega_{2,i},\ \ldots,\ \omega_{j,i},\ \ldots,\ \omega_{n,i}) \tag{3}$$

$$\vec{q} = (\omega_{1,q},\ \omega_{2,q},\ \ldots,\ \omega_{j,q},\ \ldots,\ \omega_{n,q}) \tag{4}$$

où $\omega_{j,i} \in [0,1]$ est le poids du terme t_j dans le document d_i et $\omega_{j,q} \in [0,1]$ est le poids du terme t_j dans la requête q.

La formule la plus classique pour calculer le poids est la suivante :

$$\omega_{j,i} = tf_{j,i} \times \log \frac{m}{m_j} \tag{5}$$

où $tf_{j,i}$ est la fréquence du mot clé t_j dans le document d_i et m_j le nombre de documents du corpus indexés par le mot clé t_j. Elle est appelée TF-IDF[SB88].

Exemple 3 $d_1 = (\omega_{1,1}, 0, 0, 0, 0, 0)$, $d_2 = (0, 0, 0, 0, \omega_{5,2}, 0)$ *et* $d_3 = (0, \omega_{2,3}, 0, 0, \omega_{5,3}, 0)$.

1.2.3 Modèle probabiliste

Le modèle probabiliste [RJ76] essaye d'estimer la probabilité qu'un utilisateur a de trouver un document d pertinent. Ce modèle suppose qu'il y a un sous-ensemble R de documents que l'utilisateur veut retrouver parmi ceux disponibles, les autres documents $\overline{R}$ étant considérés non-pertinents. Un document d et une requête q sont représentés par un vecteur comme dans le modèle vectoriel, mais les poids sont binaires. Si $P(R|\vec{d})$ est la probabilité que le document d soit pertinent pour la requête q et si $P(\overline{R}|\vec{d})$ est la probabilité que le document d ne soit pas pertinent pour la requête q, alors la similarité entre le document d et la requête q est :

$$sim(d, q) = \frac{P(R|\vec{d})}{P(\overline{R}|\vec{d})}. \tag{6}$$

Ce modèle est difficile à mettre en oeuvre en raison du calcul de probabilité initiale.

1.3 Mesure de similarité

Pour mesurer la similarité entre deux documents x et y (ou bien entre un document x et une requête y) représentés par des vecteurs multi-dimensionels $\vec{x} = (x_1, x_2, \cdots, x_n)$ et $\vec{y} = (y_1, y_2, \cdots, y_n)$, on a coutume de prendre l'inverse ou l'opposé d'une distance comme les distances L_p ou Kullback-Leibler, ou bien directement un cosinus.

1.3.1 Distance L_p

$$L_p(x, y) = \left(\sum_{j=1}^{n} (x_j - y_j)^p \right)^{\frac{1}{p}} \qquad p \in [1, \infty[\tag{7}$$

Pour $p = 2$, elle correspond à la distance euclidienne qui est la plus utilisée.

1.3.2 Distance de Kullback-Leibler

La divergence de Kullback-Leibler(KL) entre deux distributions de probabilité x et y est aussi connue sous le nom d'entropie relative :

$$KL(x, y) = \sum_{j=1}^{n} x_j \log \frac{x_j}{y_j}. \tag{8}$$

De plus, comme $KL(x, y) \neq KL(y, x)$, on définit la distance de Kullback-Leibler(DKL) (1951) comme :

$$DKL(x, y) = DKL(y, x) = KL(x, y) + KL(y, x). \tag{9}$$

C'est la meilleure mesure pour la recherche d'information dans les grandes bases de données[MM02].

1.3.3 Similarité par cosinus

Cette mesure de similarité généralement associée au modèle vectoriel (1.2.2) correspond au cosinus de l'angle formé par les vecteurs $\vec{x}$ et $\vec{y}$ dans l'espace multi-dimentionnel.

$$sim(x,y) = cos(\vec{x},\vec{y}) = \frac{\sum_{j=1}^{n} x_j \times y_j}{\sqrt{\sum_{j=1}^{n} x_j^2} \times \sqrt{\sum_{j=1}^{n} y_j^2}} \tag{10}$$

2 Classification

La classification automatique consiste à regrouper divers objets (les individus) en sous-ensembles d'objets (les classes). Elle peut être :

- supervisée : les classes sont connues à priori, elles ont en général une sémantique associée
- non-supervisée (en anglais clustering) : les classes sont fondées sur la structure des objets, la sémantique associée aux classes est plus difficile à déterminer

Dans les deux cas, on a besoin de définir la notion de distance entre deux classes : le critère d'agrégation.

2.1 Classification supervisée

Soit $D = \{d_1, d_2, \ldots, d_i, \ldots, d_m\}$ un ensemble de documents représentés chacun par une description $\vec{d_1}, \vec{d_2}, \ldots, \vec{d_m}$, et $C = \{C_1, C_2, \ldots, C_k, \ldots, C_c\}$ un ensemble de classes, la classification supervisée suppose connues deux fonctions. La première fait correspondre à tout individu d_i une classe C_k. Elle est définit au moyen de couples (d_i, C_k) donnés comme exemples au système. La deuxième fait correspondre à tout individu d_i sa description $\vec{d_i}$. La classification supervisée consiste alors à déterminer une procédure de classification :

$$C^f : \vec{d_i} \rightarrow C_k \tag{11}$$

qui à partir de la description de l'élément détermine sa classe avec le plus faible taux d'erreurs. La performance de la classification dépend notamment de l'efficacité de la description. De plus, si l'on veut obtenir un système d'apprentissage, la procédure de classification doit permettre de classer efficacement tout nouvel exemple (pouvoir prédictif).

2.2 Classification non-supervisée

La classification non-supervisée est utilisée lorsque que l'on possède des documents qui ne sont pas classés et dont on ne connaît pas de classification. A la fin du processus de classification non-supervisée, les documents doivent appartenir à l'une des classes générées par la classification. On distingue deux catégories de classifications non-supervisées : hiérarchiques et non-hiérarchiques.

Dans la **classification hiérarchique(CH)**, les sous-ensembles créés sont emboîtés de manière hiérarchique les uns dans les autres. On distingue la CH *descendante (ou divisive)* qui part de l'ensemble de tous les individus et les fractionne en un certain nombre de sous-ensembles, chaque sous-ensemble étant alors fractionné en un certain nombre de sous-ensembles, et ainsi de suite. Et la CH *ascendante (ou agglomérative)* qui part des individus seuls que l'on regroupe en sous-ensembles, qui sont à leur tour regroupés, et ainsi de suite. Pour déterminer quelles classes on va fusionner, on utilise le critère d'agrégation.

Dans la **classification non-hiérarchique**, les individus ne sont pas structurés de manière hiérarchique. Si chaque individu ne fait partie que d'un sous-ensemble, on parle de *partition*. Si chaque individu peut appartenir à plusieurs groupes, avec la probabilité p_i d'appartenir au groupe i, alors on parle de *recouvrement*.

2.3 Critère d'agrégation

Le critère d'agrégation permet de comparer les classes deux à deux pour sélectionner les classes les plus similaires suivant un certain critère. Les critères les plus classiques sont le plus proche voisin, le diamètre maximum, la distance moyenne et la distance entre les centres de gravités.

2.3.1 Plus proche voisin

La distance entre la classe C_p et la classe C_q est la plus petite distance entre un élément de C_p et un élément de C_q.

$$D(C_p, C_q) = \min\{dist(i,j); i \in C_p, j \in C_q\} \tag{12}$$

2.3.2 Diamètre maximum

La distance entre la classe C_p et la classe C_q est la plus grande distance entre un élément de C_p et un élément de C_q.

$$D(C_p, C_q) = \max\{dist(i,j); i \in C_p, j \in C_q\} \tag{13}$$

2.3.3 Distance moyenne

La distance entre la classe C_p et la classe C_q est la moyenne des distances entre les éléments de C_p et les éléments de C_q.

$$D(C_p, C_q) = \frac{\sum_{i,j}\{dist(i,j); i \in C_p, j \in C_q\}}{Card(C_p) \times Card(C_q)} \tag{14}$$

2.3.4 Distance entre les centres de gravité

Si G_p est le centre de gravité de la classe C_p et si G_q est le centre de gravité de la classe C_q alors la distance entre la classe C_p et la classe C_q est la distance entre leurs centres de gravités.

$$D(C_p, C_q) = dist(G_p, G_q)$$

Ce critère n'a de sens que si le calcul du centre de gravité possède lui-même un sens sur les données de l'étude.

2.4 Evaluation d'un système de classification

Nous présentons ici une méthode permettant d'évaluer une classification supervisée, et des techniques classiques pour mesurer et comparer des systèmes de classifications non-supervisées.

2.4.1 Corpus de test (cas supervisé)

Pour tester la qualité d'une procédure de classification supervisée, on sépare aléatoirement les éléments classés entre une base de référence(R) et une base de test(T). Ensuite, on détermine la procédure de classification C^f à partir des exemples de la base de référence. Puis, on utilise C^f pour retrouver la classe des éléments de la base de test. Enfin, on estime l'erreur de la procédure de classification.

Pour estimer le taux d'erreur TE d'une procédure de classification C^f, une méthode simple est de calculer le nombre d'éléments mal classés sur le nombre d'éléments à classer :

$$TE(C^f) = \frac{1}{card(T)} \sum_{t=1}^{card(T)} (C^f(\vec{d_t}) \neq C_{d_t}) \tag{15}$$

où C_{d_t} est la classe d'origine de d_t.

Dans les cas de classifications simples, on peut être amené à calculer l'erreur résultant d'une classification purement aléatoire C^a pour la comparer avec l'erreur faite par notre procédure C^f afin de vérifier la performance de notre système.

Soit P_k la fréquence (ou probabilité à priori) de la classe k dans la base de test, on appelle erreur TE_a du système aléatoire :

$$TE_a = 1 - \sum_{k=1}^{c} (P_k)^2 = 1 - \sum_{k=1}^{c} \left(\frac{card(C_k|T)}{card(T)} \right)^2 \tag{16}$$

où c est le nombre de classes et $card(C_k|T)$ est le nombre d'éléments de T qui sont dans la classe C_k.

L'erreur apparente $TE(C^f)$ est dépendante de l'échantillon considéré. Cependant, plus le nombre d'éléments de l'échantillon est grand, plus l'erreur mesurée tend vers l'erreur réelle de C^f.

2.4.2 Cas non-supervisé

Dans le cas non-supervisé, on peut évaluer la classification par rapport à certaines de ces caractéristiques. On distingue d'une part, les caractéristiques numériques : le nombre de classes obtenues, le nombre d'éléments par classe, le nombre moyen d'éléments par classe, l'écart-type des classes obtenues, et d'autre part, les caractéristiques sémantiques. Par exemple, si à un document est associé un ensemble de mots clés, la sémantique associée à une classe pourra se composer des mots les plus fréquents dans la classe.

Pour évaluer l'homogénéité du nombre d'images par classe, on peut utiliser la variance :

$$V = \sigma^2 = \frac{1}{c} \sum_{k=1}^{c} (card(C_k) - moy)^2 \tag{17}$$

où $moy = \frac{1}{c} \sum_{k=1}^{c} card(C_k)$ est le nombre moyen d'éléments par classe et c est le nombre de classes obtenues. L'écart-type $\sigma = \sqrt{V}$ permet d'exprimer la dispersion dans la même unité que la moyenne.

3 Présentation du corpus et mode opératoire

Le corpus est constitué de 665 photos de presse, mises à disposition par la société Editing dans le cadre du projet RNTL Muse[BLM02]. Les photos sont indexées textuellement par les documentalistes de cette société à partir des mots extraits d'un thésaurus structuré hiérarchiquement et stocké sous la forme d'un fichier XML. Les sujets dont elles traitent sont divers et variés, les tableaux 3 page 15 et 1 page 7 donnent un aperçu des mots présents dans le thésaurus et de leur répartition dans les images.

Nombre de mots dans le thésaurus	Profondeur maximale du thésaurus	Profondeur moyenne du thésaurus
1208	6	3.2
Nombre d'images dans le corpus	Nombre total de mots dans les images	Nombre total de mots différents dans les images
665	2005	193/1208

TAB. 1 – Résumé chiffré sur le corpus d'images et le thésaurus hiérarchique

A partir de ce corpus, nous allons extraire les indices textuels des fiches, puis les indices visuelles des images. Nous établirons ensuite une méthode permettant d'étudier la cohérence entre les deux.

3.1 Extraction des indices textuels

Les indices textuels associés aux images sont stockés dans des fiches XML qui suivent le schéma MPEG-7[MSS02]. Voici un exemple de fiches MPEG-7 très simplifié contenant les mot-clés *Téléphonie* et *Radio*.

```
<?xml version="1.0" encoding="UTF-8"?>
<mpeg7:Mpeg7 xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
             xmlns:mpeg7="http://www.mpeg7.org/2001/MPEG-7_Schema">
  <mpeg7:DescriptionMetadata>
    <mpeg7:LastUpdate>2002-10-2</mpeg7:LastUpdate>
    <mpeg7:PrivateIdentifier>BAR9501001C-1</mpeg7:PrivateIdentifier>
    <mpeg7:CreationTime>2002-10-2</mpeg7:CreationTime>
  </mpeg7:DescriptionMetadata>
  <mpeg7:ContentDescription xsi:type="ContentEntityType">
     <mpeg7:Creation>
       <mpeg7:Title>Développement du téléphone portable</mpeg7:Title>
       <mpeg7:KeywordAnnotation>
         <mpeg7:Keyword>Téléphonie</mpeg7:Keyword>
         <mpeg7:Keyword>Radio</mpeg7:Keyword>
       </mpeg7:KeywordAnnotation>
     </mpeg7:Creation>
  </mpeg7:ContentDescription>
  <mpeg7:ContentDescription xsi:type="ViewDescriptionType">
    <mpeg7:Image>
      <mpeg7:MediaUri>BAR9501001C-1.jpg</mpeg7:MediaUri>
    </mpeg7:Image>
  </mpeg7:ContentDescription>
</mpeg7:Mpeg7>
```

Pour extraire les indices textuels des fiches XML et pour extraire les mots du thésaurus, nous avons utilisé le package java org.w3c.dom qui permet de traduire un fichier XML en un 'arbre informatique'.

3.2 Extraction des indices visuels

Il existe de multiples façons d'extraire des indices visuels, nous avons mis en oeuvre, en collaboration avec Hervé Glotin, une méthode qui permet des calculs simples et rapides.

Nos indices visuels sont composés de 5 attributs :

- l'histogramme de la luminance(A_1),
- les 3 histogrammes des couleurs rouge(A_2), vert(A_3), bleu(A_4), normalisés par la luminance (indiquant donc les composantes absolues de chaque couleur),
- l'histogramme des directions des contours(A_5). Pour obtenir ce dernier histogramme, on commence par extraire les contours par la méthode des gradients maximum (méthode de Canny [Can86]). L'image 2(edges) de la figure 1 donne un exemple de matrice binaire de contours. Ensuite, on « fait passer » (convolution) une à une 6 matrices carrées de 7 pixels de côté sur la matrice de contours. Ces matrices contiennent des coefficients qui codent un segment dont la pente varie de $-\pi/2$ pour la première matrice à $\pi/3$ pour la dernière, par pas de $\pi/6$ avec une tolérance de $\pm\pi/12$. Ce système permet de détecter, de classer

et de compter les traits des contours selon leur pente. Par exemple, dans l'histogramme de direction de la figure 1, les segments de pente $\pi/2$(trait vertical) et π(trait horizontal) sont les plus représentés. Ces pentes caractérisent classiquement les bâtiments.

Ces attributs visuels sont extraits pour les images complètes (région globale, notée g). De plus, nous avons testé une méthode originale d'extraction de sous-images d'intérêt pour lesquelles nous calculons aussi les attributs visuels.

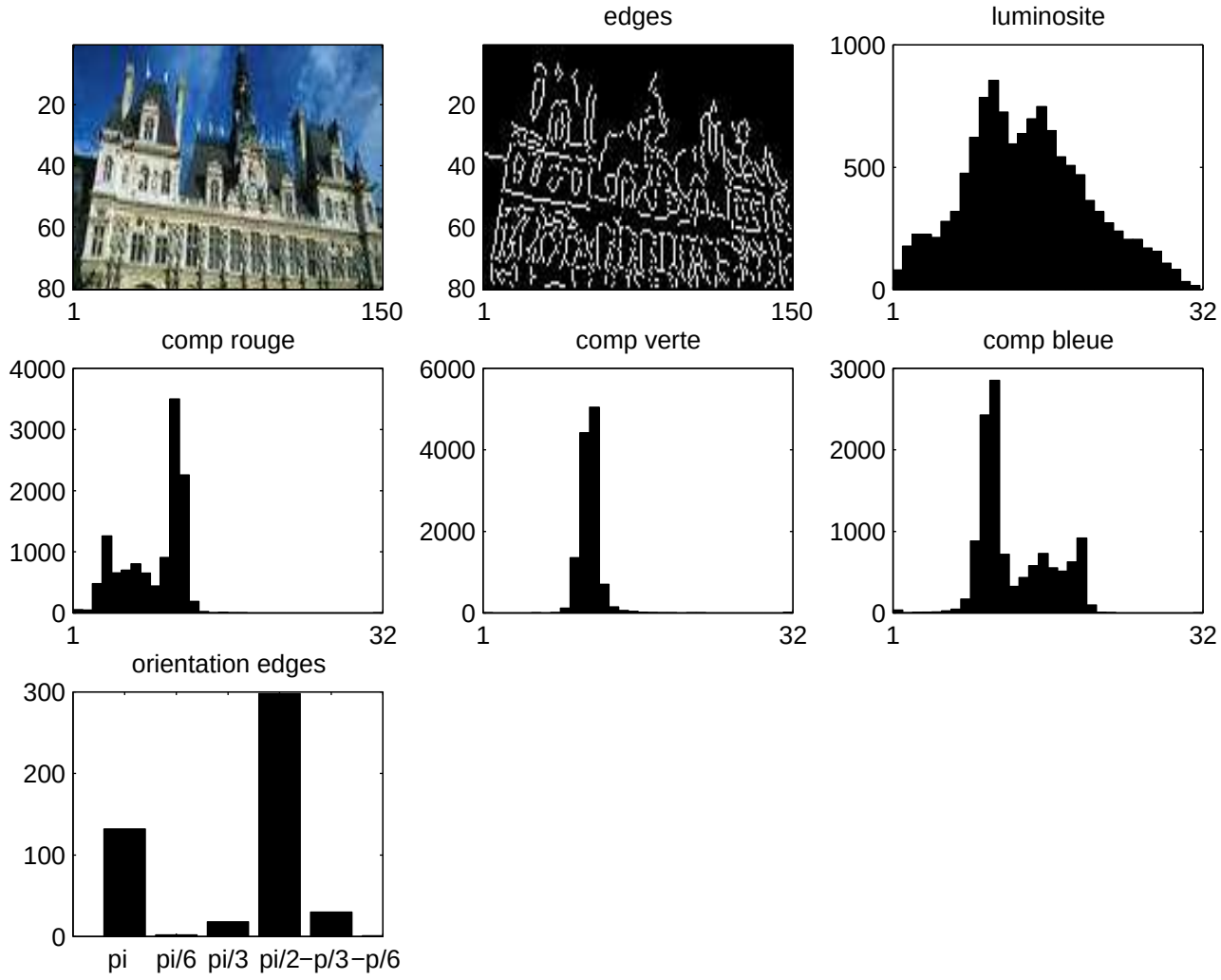


FIG. 1 – Indices visuels sous la forme d'histogrammes. Photo @Editing.

Pour chaque image, 4 sous-images sont détectées automatiquement. L'algorithme de détection commence par extraire les contours de l'image par la méthode de Canny comme précédemment. Puis, il fait la sommation de ces contours par région de dimension fixée. Nous avons choisi comme dimension une surface d'un quart de la surface de l'image globale. Ensuite, on extrait la région qui contient le plus de contours et on la soustrait de la matrice des contours. Enfin, la détection d'une nouvelle région d'intérêt est relancée sur la nouvelle matrice des contours. La figure 2 montre la détection automatique de 2 premières régions d'intérêts(ROI). On numérote ces régions de r_1 à r_4 selon leur ordre de détection.

L'intérêt de l'étude des histogrammes de couleurs de sous-images est de classer ensemble des images qui sont grossièrement différentes, mais qui possèdent des caractéristiques communes. Par exemple, détecter les images contenant un visage grâce à la couleur de la peau, sans être bruitée par le fond de l'image.

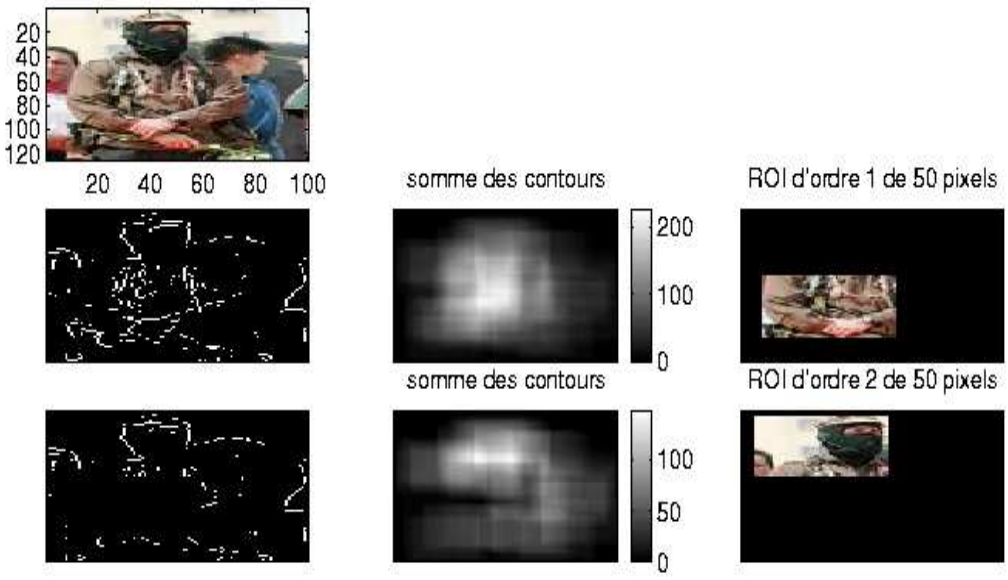


FIG. 2 – Sélection des 2 premières régions locales d'intérêts(ROI) d'une image par détection des contours par la méthode de Canny et par maximisation des sommes des contours par région. Photo ©Editing.

Au final, les indices visuels associés à l'image se présentent sous la forme de vecteurs de flottants (les histogrammes) qui permettent des calculs simples et rapides entre deux régions de l'image par simple mesure de similarité au sens DKL (section 1.3.2) des vecteurs. On peut se demander si faire des mesures de similarité entre l'histogramme d'une région d'intérêt et l'histogramme d'une image a un sens, mais les vecteurs sont normalisés et une sous-image n'est rien d'autre qu'une image.

3.3 Mode opératoire du système visuo-textuel

Nous allons construire un système de classification visuo-textuelle afin d'améliorer des recherches d'images posées avec peu de mot-clés, que nous rehausserons grâce à leur contenu visuel. Ainsi, à chaque image, on associe des descripteurs (ou indices) textuels et visuels. Puis, on les classe par classification ascendante hiérarchique afin d'obtenir un classement par rapport aux indices textuels seulement. La construction de la base de référence B_{Ref} est expliquée à la section 4.

Ensuite, on sépare la base obtenue en deux parties : une base d'exemples classés (sous-base de référence)B_{Ex} et une base de test B_{Test}. Pour cela, on choisit aléatoirement 50% des images de chaque classe de B_{Ref} pour constituer B_{Test}, les autres images constituant la sous-base de référence B_{Ex}. On connaît la classe des images de B_{Ex}, et on cherche à retrouver la classe de chaque image de B_{Test} par simple similarité au sens DKL avec les images de la base B_{Ex}. Pour évaluer, la performance de cette classification, on regarde alors le nombre d'images de B_{Test} qui sont bien classées. Les figures 3 et 4 résument la méthode.

En dernier lieu, nous estimons la probabilité d'appartenance d'une image à une classe à partir de l'information textuelle, visuelle ou par fusion des deux.

FIG. 3 – Schéma de la méthode de classification

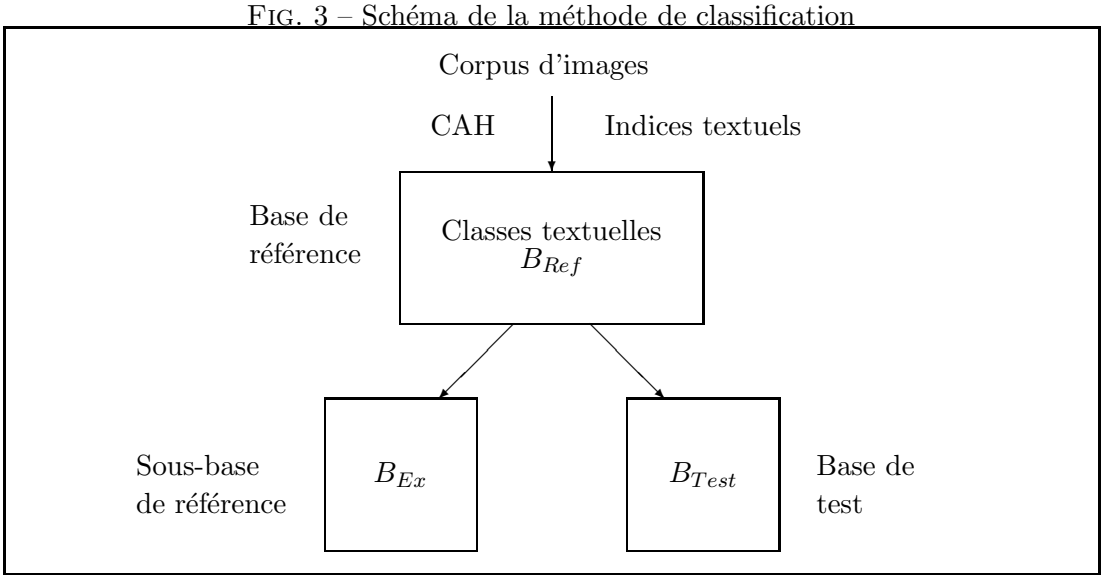
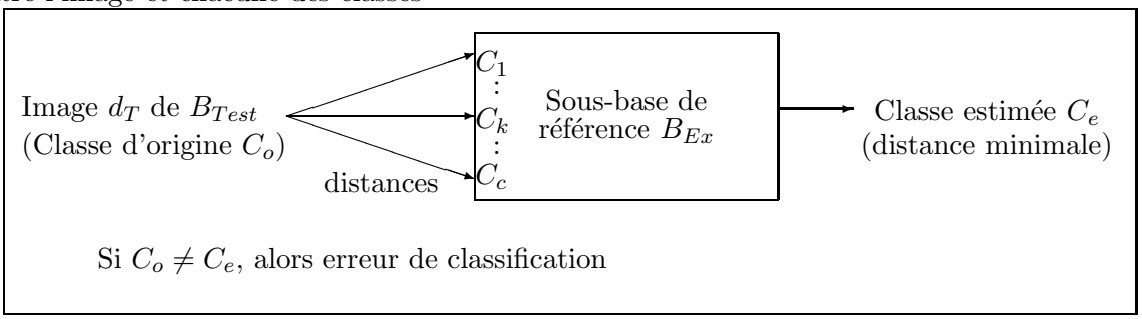


FIG. 4 – Classification d'une image de la base de test d_T par recherche de la distance minimale entre l'image et chacune des classes



4 Construction d'une base de référence par CAH

Dans un premier temps, il s'agit de construire une classification (non-supervisée) des images à partir de leur indexation textuelle uniquement. Cette classification constituera une base de référence qui permettra de mesurer les erreurs de classification, et par la suite, d'indexer de nouvelles images.

Pour réaliser cette classification, nous avons choisi d'utiliser la méthode classique de classification ascendante hiérarchique(CAH)(Lance et Williams, 1967) :

Algorithme Classification ascendante hiérarchique
Données :
E : ensemble de n éléments à classer
Tableau $n \times n$ des distances entre éléments
Variables :
C : ensemble des c classes
Début
$C \leftarrow \phi$
Pour chaque individu e de E **faire**
Créer une classe dans C contenant e
fin pour
Tant que $c > 1$ **faire**
Pour chaque couple (C_i, C_j) de classes de C **faire**
Calculer la distance entre C_i et C_j
pour le critère d'agrégation considéré
fin pour
Agréger les deux classes C_p et C_q de distance minimale
$c \leftarrow c - 1$
fin tant que
Fin

Pour mettre en oeuvre cette classification, il faut disposer de trois composantes :
(i) une représentation textuelle des images,
(ii) une mesure de similarité qui permet de comparer les images,
(iii) un critère d'agrégation qui permet de fusionner les classes.
Dans le cas standard, l'agrégation s'arrête lorsque tous les individus ont été rassemblés dans la même classe. Dans notre cas, le critère d'arrêt sera d'avoir obtenu un ensemble de classes représentatives du corpus d'images et suffisamment homogènes.

4.1 Représentation vectorielle du texte des images

Nous avons choisi de représenter(décrire, indexer) une image d_i par un vecteur $\vec{d_i}$ comme présenté à la section 1.2.2, car c'est une représentation efficace classiquement utilisée. A chaque mot-clé du thésaurus est associée une case du vecteur dans l'ordre d'apparition des mots. Une case est initialisée à 1 si le mot-clé appartient à l'image, à 0 sinon. On a donc $\omega_{j,i} \in \{0, 1\}$.

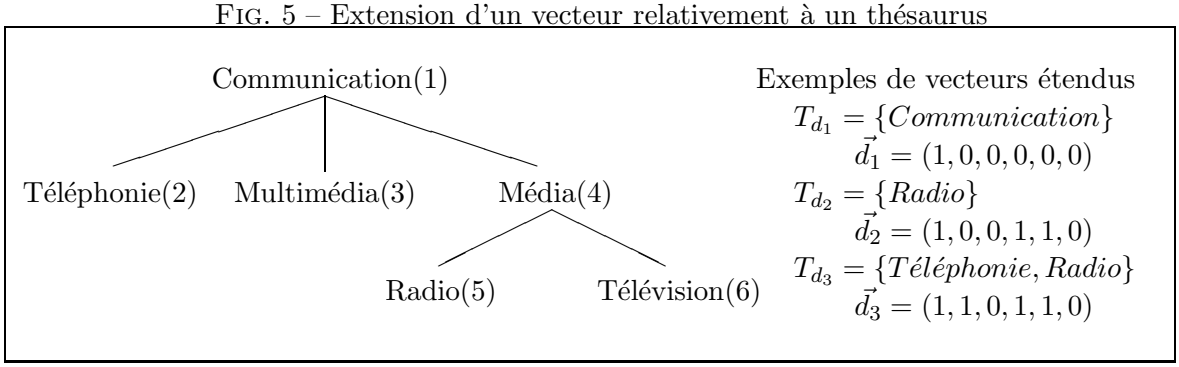
De plus, le thésaurus est structuré hiérarchiquement par une relation de généricité ($\prec$) qui implique que si une image est indexée par un mot-clé t_j et que $t_j \prec t_k$ alors cette image est aussi indexée par le mot-clé t_k. Il faut donc, comme dans [MCM02], étendre le vecteur $\vec{d_i} = (\omega_{1,i},\ \omega_{2,i},\ \ldots,\ \omega_{j,i},\ \ldots,\ \omega_{n,i})$ d'une image de façon à ce que

$$\forall j, k\ \in [1, n],\ \omega_{k,i} = 1\ si\ \omega_{j,i} = 1\ et\ t_j \prec t_k. \tag{18}$$

Cette méthode d'extension des vecteurs permet de raprocher des documents qui contiennent des termes 'frères' dans le thésaurus.

Exemple 4 *Considérons le thésaurus et l'indexation de l'image d_3 dans la figure 5. Le vecteur $\vec{d_3}$ initial est $(0,1,0,0,1,0)$. Puisque Téléphonie $\prec$ Communication, on a $\omega_{1,3}=1$, et puisque Radio $\prec$ Média $\prec$ Communication, on a $\omega_{4,3}=1$ et $\omega_{1,3}=1$. Le vecteur étendu de l'image d_3 est donc $(1,1,0,1,1,0)$. Les vecteurs étendus des images d_1 et d_2 sont obtenus de façon similaire.*

FIG. 5 – Extension d'un vecteur relativement à un thésaurus



4.2 Mesure de similarité

En recherche d'information(RI), on utilise souvent le cosinus défini en 1.3.3 entre le vecteur représentant le document d et le vecteur représentant la requête q comme expliqué dans la section 1.2.2. Mais dans notre cas, nous voulons mesurer la similarité entre deux documents d_k et d_l. De plus, compte tenu que les poids sont égaux à 0 ou à 1, on peut simplifier la formule du cosinus en :

$$sim(d_k,d_l) = cos(\vec{d_k},\vec{d_l}) = \frac{\sum_{j=1}^{n} \omega^*_{j,k,l}}{\sqrt{\sum_{j=1}^{n} \omega_{j,k}} \times \sqrt{\sum_{j=1}^{n} \omega_{j,l}}} \tag{19}$$

où $\omega_{j,k}$, $\omega_{j,l}$ et $\omega^*_{j,k,l} \in \{0,1\}$, et $\omega^*_{j,k,l}=1$ si les images d_k et d_l sont indexées par t_j, sinon $\omega^*_{j,k,l}=0$.

Pour la classification, c'est la distance entre deux images que l'on a besoin de connaître. Nous la calculons par la formule :

$$dist(d_k,d_l) = 1 - sim(d_k,d_l). \tag{20}$$

Deux images similaires ont une distance égale à 0 et deux images entièrement dissimilaires ont une distance égale à 1.

Exemple 5 *Si l'on considère les images d_1, d_2 et d_3 de la figure 5, on a $sim(d_1,d_2)=0.33$, $sim(d_2,d_3)=0.25$ et $sim(d_1,d_3)=0.25$.*

4.3 Paramétrages du critère d'agrégation et du critère d'arrêt

A chaque étape d'une classification ascendante hiérarchique, on agrège les deux classes C_p et C_q qui ont une distance $D(C_p,C_q)$ minimum. Il existe plusieurs formules pour calculer cette distance D dont les plus classiques sont définies à la section 2.3. Les distances par plus proche voisin, diamètre maximum et distance moyenne ont été expérimentées.

Le tableau 2 présente les meilleurs résultats que nous avons pu obtenir pour chacun des critères. Les classes obtenues avec les critères du plus proche voisin et de la distance moyenne sont sémantiquement homogènes, mais on observe que les classes les plus peuplées sont les plus attractives, et donc il y une grande hétérogénéité numérique (une grande classe qui contient la

plupart des images, et beaucoup de petites classes qui contiennent une seule image). Cette grande classe est peut-être due à la configuration de nos données, en effet, certains mots génériques comme 'portrait' sont présents dans beaucoup d'images. Le critère du diamètre maximum donne des classes numériquement plus homogènes, mais les images appartenant à une même classe sont sémantiquement trop différentes.

Résultats CAH	min	max	moy	écart-type	c	m
Plus proche voisin	1	238	13.3	33.7	50	665
Plus proche voisin filtré	8	238	41.2	65.9	10	412
Diamètre maximum	1	98	10.2	14.6	65	665
Diamètre maximum filtré	8	98	20.6	20.6	23	474
Diamètre maximum contraint 0.7	1	98	11.7	16.5	57	665
Diamètre maximum contraint 0.7 filtré	8	98	21.5	21.4	24	517

TAB. 2 – Résultat des meilleurs classifications ascendantes hiérarchiques obtenues pour chacun des critères (min : nombre minimal d'images dans une classe, max : nombre maximal d'images dans une classe, moy : nombre moyen d'images par classe, c : nombre de classes obtenues et m : nombre d'images totales). Le critère choisi pour la CAH est le diamètre maximum contraint 0.7 filtré.

Ces résultats étant peu satisfaisants, un compromis a alors été recherché entre la méthode du plus proche voisin et celle du diamètre maximum pour garder les avantages de l'une et de l'autre. Ce compromis a été d'utiliser le diamètre maximum, mais au lieu de prendre la distance maximale, nous avons pris la plus grande distance inférieure à un certain seuil (la contrainte CT).

On appelle diamètre maximum contraint de contrainte $CT \in [0,1]$ la distance entre la classe C_p et la classe C_q :

$$D(C_p, C_q) = \begin{cases} 0 & si\ \exists d_r, d_s\ dist(d_r, d_s) = 0 \\ max\{dist(d_r, d_s) < CT; d_r \in C_p,\ d_s \in C_q\} & si\ \exists d_r, d_s\ dist(d_r, d_s) < CT \\ min\{dist(d_r, d_s); d_r \in C_p,\ d_s \in C_q\} & sinon. \end{cases} \tag{21}$$

La première condition permet aux documents ayant exactement les mêmes descriptions de se retrouver dans la même classe. La deuxième élargit la recherche aux documents éloignés, mais pas trop. La dernière est pour les cas extrêmes ou toutes les distances entre deux images seraient plus grandes que la contrainte.

Nous avons déterminé de manière empirique que pour nos données les meilleurs résultats étaient obtenus pour $CT = 0,7$. En effet, ils sont meilleurs numériquement, car les classes sont plus homogènes que celles obtenues par le critère du plus proche voisin (écart-type 21.4<65.9). Et sémantiquement les images d'une même classe sont plus similaires que par le critère du diamètre maximum.

Pour le critère d'arrêt de la classification, nous avons déterminé de manière empirique que pour obtenir des classes représentatives, il fallait arrêter la classification lorsque la distance d'agrégation obtenue était de 0,55. On a ainsi obtenu 57 classes.

Pour obtenir des classes qui soient suffisamment représentatives, il faut que les classes aient certaines caractéristiques. Une classe doit avoir un nombre d'images n_{C_k} suffisant (fixé à 8) pour contenir assez d'informations visuelles, et les images qu'elle contient ne doivent pas être

indexées par les mêmes mot-clés pour avoir suffisamment de diversités textuelles. On exprime cela sous-forme de contraintes dont l'union forme un filtre. Chaque classe C_k doit vérifier :

$$\exists d_r, d_s \in C_k,\ dist(d_r, d_s) \neq 0\ et\ n_{C_k} \geq 8, \tag{22}$$

sinon elle est supprimée ainsi que les images qu'elle contient.

Après filtrage, nous obtenons 24 classes sémantiquement pertinentes et dont les images sont suffisamment bien réparties. Nous pouvons donc l'utiliser comme base de référence. Le tableau 3 donne un aperçu des mot-clés de chacune des classes obtenues et la figure 6 montre un exemple d'une classe et de ses images.

Classe	m_k	T_{f_1}	T_{f_2}	T_{f_3}
1	19	Mexique	Politique	Portrait
2	8	Israël	Judaïsme	Patrimoine
3	9	Constructeurs	Transport	Automobile
4	18	Contemporaine	Portrait	Rhône
5	29	Portrait	Armée de l'air	Aéronautique
6	8	Société	Famille	Enfant
7	18	Cameroun	Agriculture	Géographie physique
8	22	Municipalité	Portrait	Les Verts
9	12	Elevages	Santé	Police national
10	9	Portrait	Média	Administrations
11	10	Femme	Ouvriers	Industrie de précision
12	8	Région	Municipalité	Conseil régionaux
13	9	Communication	Télécommunications	Multimédia
14	8	Production	Travail	Alimentation
15	8	Israël	Liban	Urbanisation
16	28	Parti socialiste	Portrait	Municipalité
17	8	Multimédia	Start'up	Ouvriers
18	21	Jeux de société	Humain	Librairies
19	18	Problèmes sociaux	Conflits sociaux	Europe
20	48	Politique	Paris	Bourse
21	71	Bars et Café	Restauration rapide	Etats-Unis
22	12	Infrastructures routières	Innondation	Véhicules
23	98	Portrait	Municipalité	RPR-UMP
24	18	Justice	Portrait	Scandales politiques

TAB. 3 – Nombre (m_k) d'images par classe et liste des 3 premiers mot-clés les plus fréquents ($f_1 > f_2 > f_3$) dans chaque classe

5 Classification textuelle, visuelle et fusion

Nous venons de construire une base de référence pour notre corpus d'images, nous allons maintenant tester un système de classification automatique travaillant avec les indices visuels et/ou les indices textuels.

Pour cela, calculons d'abord le taux d'erreur du système pour une classification aléatoire. A partir de la formule 16, de la colonne 2 du tableau 3 et en considérant que nous avons pris 50% des images dans chaque classe pour notre base de test, nous obtenons 91.6%. Donc si nous obtenons un taux d'erreur proche de 91.6%, alors il n'y pas de correspondance entre les indices visuels et textuels.

FIG. 6 – Exemple de classe obtenue par notre CAH

Nous allons d'abord tester notre méthode en effectuant une classification supervisée à partir des indices textuels. Puis, nous testerons plusieurs classifications à partir des indices visuels toujours par référence aux classes textuelles. Enfin, nous fusionnerons les deux systèmes et analyserons les résultats.

5.1 Classification textuelle

Une première expérience consiste à tester la base de référence obtenue par CAH. Chaque classe C_k de B_{Ex} est représentée par un vecteur moyen textuel $\vec{C_k^t}^*$ normalisé obtenu en faisant la somme des vecteurs textuels des images qu'elle contient. La classe textuelle d'une image d_T de B_{Test} de vecteur textuel normalisé $\vec{d_T^t}^*$ est calculée par :

$$C^t(d_T) = \text{argmin}_{k\in\{1,2,\ldots,c\}} DKL(\vec{d_T^t}^*, \vec{C_k^t}^*). \tag{23}$$

Nous faisons alors deux tests : le premier en étendant les vecteurs textuels à l'aide du thésaurus comme expliqué à la section 4.1, le deuxième en utilisant des vecteurs non-étendus. Le tableau 4 donne les taux d'erreurs obtenus. Nous remarquons que lorsque les vecteurs sont étendus, les

Textuelle avec thésaurus	Textuelle sans thésaurus
1.17	13.72

TAB. 4 – Comparaison des taux d'erreurs textuelles (en %)

résultats donnent un taux d'erreurs très faible. La description des images et la procédure de classification utilisées sont efficaces. Si maintenant, nous nous plaçons dans le cas d'informations manquantes en n'étendant pas les vecteurs avec l'information du thésaurus, on observe une variation du taux d'erreurs non-négligeable que nous allons essayer de diminuer à l'aide des indices visuels.

5.2 Classification visuelle

Une deuxième expérience consiste à faire des classifications supervisées des images à partir des indices visuels seuls, mais toujours en référence aux classes textuelles. Pour cela, nous allons tester l'influence de certains paramètres. Un grand nombre de combinaisons possibles a été expérimenté pour choisir les meilleurs distances visuelles, nous présentons celles qui donnent les meilleurs résultats.

On note $\mathrm{DKL}_A(r_i, r_j)$ la distance DKL entre la région r_i de l'image d_T de B_{Test} et la région r_j de l'image d_E de B_{Ex} pour l'attribut A.

5.2.1 Distance par région

Nous commençons par calculer la distance entre les régions d'intérêts de niveaux égaux. La table 5 montre les résultats.

	DKL$(r1, r1)$	DKL$(r2, r2)$	DKL$(r3, r3)$	DKL$(r4, r4)$	DKL(g, g)
T.E. Rouge	81.17	79.21	81.17	82.35	**73.33**
T.E. Vert	83.13	**78.03**	86.66	80.78	78.43
T.E. Bleu	82.35	80.39	83.92	84.70	**74.50**
T.E. Luminance	80.39	81.17	81.56	83.52	**76.40**
T.E. Direction	**79.60**	81.56	80.00	84.31	85.49

TAB. 5 – Influence du choix de la région d'intérêt sur le Taux d'Erreur(T.E. en %) pour les différents attributs de l'image

On remarque que, en général, les distances sur les indices globaux sont meilleurs, sauf pour la direction où la région 1 donne de meilleurs résultats. En effet, la région 1 est celle qui contient le plus de contours, elle est donc la plus significative. Pour l'attribut vert, le bon résultat obtenu pour la région 2 s'explique par un artefact du aux données (une classe contenant plus de vert que les autres). Cet artefact pourra avoir des répercussions par la suite dont le lecteur ne tiendra pas compte. L'hypothèse de départ supposant que les régions locales les plus descriptives sont celles qui contiennent le plus de contour est vérifiée, car les régions 1 et 2 ont les plus faibles taux d'erreur.

5.2.2 Distances par fusion précoce des indices visuels

Pour un attribut A donné, chaque image possède 5 histogrammes (r1, r2, r3, r4 et g(r5)). Pour une image d_T de B_{Test} et pour une image d_E de B_{Ex}, il existe donc 5×5 distances entre régions de l'image possibles. Si l'on considère seulement les $L \in [1, 5]$ régions d'intérêt, il existe $L \times L$ distances entre régions de l'image possibles (si $L = 2$, $L^2 = 4$ et on ne considère que les distances $\mathrm{DKL}_A(r1, r1)$, $\mathrm{DKL}_A(r1, r2)$,$\mathrm{DKL}_A(r2, r1)$ et $\mathrm{DKL}_A(r2, r2)$). Nous allons définir une distance entre les indices visuels de deux images qui prenne en compte les meilleurs scores parmi ces distances. Pour les besoins du calcul de ces distances, on note $\mathrm{moymin_K}$ la fonction :

$$\mathrm{moymin_K} : \{\alpha_1, \alpha_2, \ldots, \alpha_M\} \rightarrow (\alpha_{min1} + \alpha_{min2} + \ldots + \alpha_{minK})/K \tag{24}$$

qui fait la moyenne arithmétique des K premières valeurs minimales.

Pour calculer la distance visuelle entre une image d_T de B_{Test} et une image d_E de B_{Ex}, on calcule les L^2 distances possibles entre 2 images et nous calculons la moyenne des N plus petites valeurs ($N \in [1, L^2]$), on obtient la distance :

$$\gamma_A(d_T, d_E) = \mathrm{moymin_N}(\{DKL_A(i, j); \forall i, j \in L\}). \tag{25}$$

Maintenant, si on considère la distance entre une image d_T de B_{Test} et la classe C_k, on calcule les distances entre d_T et les images d_{E_k} de C_k et on garde les I minimums dont nous calculons la moyenne pour obtenir la distance entre l'image d_T et la classe C_k :

$$\delta_A(d_T, C_k) = \text{moymin}_{\text{I}}(\{\gamma_A(d_T, d_{E_k}); \forall d_{E_k} \in C_k\}) \tag{26}$$

où d_{E_k} est un élément de la classe C_k de la base d'exemples et $I \in [1, \text{card}(\text{C}_\text{k})]$ est le nombre de valeurs minimales prises parmi les $\text{card}(\text{C}_\text{k})$ distances entre d_T et les éléments de la classe C_k possibles.

La classe visuelle de d_T pour l'attribut A est obtenue par :

$$C_A^v(d_T) = \text{argmin}_{k \in \{1,2,\ldots,c\}} \delta_A(d_T, C_k). \tag{27}$$

Cette méthode permet de rejeter les distances trop importantes (d_T très différente de d_E) qui pénaliseraient trop le système et permet de garder les meilleures distances qui donnent plus de probabilité d'être dans la bonne classe.

5.2.3 Résultats de la fusion précoce visuelle

Les tableaux 6, 7 et 8 donnent les taux d'erreur obtenus par cette méthode dite de « fusion précoce » des indices visuels en faisant varier les paramètres N, I et L.

N	1	2	3	4	5	6	7	8
T.E. Rouge	**71.76**	72.54	72.54	73.72	76.47	77.64	77.64	76.07
T.E. Vert	**76.07**	77.64	77.64	76.86	76.86	76.47	78.82	78.82
T.E. Bleu	77.64	**77.25**	79.60	80,00	79.60	81.56	81.96	81.96
T.E. Luminance	**77.64**	79.21	77.64	77.64	79.21	79.21	78.82	78.03
T.E. Direction	83.52	80.39	80.39	80,00	79.21	78.82	78.43	**76.86**

TAB. 6 – Taux d'Erreur(T.E. en %) pour différentes valeurs de N et pour les différents attributs par fusion précoce des indices visuels ($I = 4, L = 5$)

I	1	2	3	4
T.E. Rouge	75.68	74.50	**71.76**	**71.76**
T.E. Vert	79.60	78.03	76.86	**76.07**
T.E. Bleu	78.03	77.64	78.03	**77.25**
T.E. Luminance	79.21	78.03	**76.07**	77.64
T.E. Direction	84.70	78.03	**76.86**	**76.86**

TAB. 7 – Taux d'Erreur(T.E. en %) pour différentes valeurs de I, et pour les valeurs de N pour lesquels le taux d'erreur est le plus faible par fusion précoce des indices visuels des différents attributs ($L = 5$)

Le tableau 6 donne l'influence du paramètre N pour les valeurs de I et L donnant les meilleurs résulats. On remarque que le paramètre N a peu d'influence pour les attributs Rouge, Vert, Bleu et Luminance. Par contre, pour la direction, on observe une réelle amélioration du T.E. quand on prend N grand. Le tableau 7 montre qu'il vaut mieux regarder si l'image test est similaire à plusieurs images d'une même classe qu'à une seule. Enfin, dans le tableau 8, on remarque que la région d'intérêt 1 seul n'est pas suffisante($L = 1$) et que la région d'intérêt numéro 4 n'apporte finalement que peu d'informations, car les T.E. pour $L = 4$ sont moins bons

L	1	2	3	4	4+g
Dimension L^2	1	4	9	16	25
T.E. Rouge	81.17	78.82	76.07	76.07	**71.76**
T.E. Vert	83.13	78.82	**75.68**	79.60	76.07
T.E. Bleu	82.35	80.00	79.60	81.56	**77.25**
T.E. Luminance	80.39	79.60	78.03	**77.64**	**77.64**
T.E. Direction	79.60	78.03	**76.07**	76.47	76.86

TAB. 8 – Taux d'Erreur(T.E. en %) pour différentes valeurs de L, et pour les valeurs de N pour lesquels le taux d'erreur est le plus faible par fusion précoce des indices visuels des différents attributs ($I = 4$)

que pour $L = 3$. Et on remarque aussi que, pour $L = 5(4+g)$, les indices globaux apportent une nette amélioration du T.E., sauf dans le cas de la direction, ce qui était prévisible.

Si on compare ces résultats à ceux du tableau 5, on remarque une baisse de l'ordre de 5% à 10% du taux d'erreur sur les indices locaux, et une amélioration d'environs 2% sur les globaux. Donc notre méthode de fusion précoce apporte un gain non-négligeable. Cependant, elle a une mauvaise compléxité et nécessite un temps de calculs assez important.

5.3 Fusion tardive visuo-textuelle

Nous allons maintenant fusionner les indices textuels et visuels afin d'améliorer les résultats obtenus pour la classification textuelle.

Pour chaque image d_T et pour chaque classe C_k, on calcule la distance textuelle $DKL(\vec{d_T}^*, \vec{C_k}^*)$ comme expliqué à la section 5.1. Puis, on la normalise et on la complète à 1 pour estimer la probabilité d'appartenance $P^t_{d_T}(C_k)$ de l'image d_T à la classe C_k par rapport aux indices textuels :

$$P^t_{d_T}(C_k) = 1 - \frac{DKL(\vec{d_T}^*, \vec{C_k}^*)}{\sum_k DKL(\vec{d_T}^*, \vec{C_k}^*)}. \tag{28}$$

De même, on estime la probabilité d'appartenance $P^v_{d_T}(C_k)$ de l'image d_T à la classe C_k par rapport à l'attribut visuel A :

$$P^v_{d_T}(C_k|A) = 1 - \frac{\delta_A(d_T, C_k)}{\sum_k \delta_A(d_T, C_k)}. \tag{29}$$

On numérote de 1 à 5 les attributs visuels et on donne le numéro 6 à l'indice textuel. La probabilité d'appartenance $P^{v\vee t}_{d_T}(C_k)$ de l'image d_T à la classe C_k par fusion tardive des indices textuels et visuels est :

$$P^{v\vee t}_{d_T}(C_k) = \sum_{j=1}^{5} P^v_{d_T}(C_k|A_j) \times \omega'(A_j) + P^t_{d_T}(C_k) \times \omega'(A_6) \tag{30}$$

où $\omega'(A_j) = \frac{\omega(A_j)^p}{\sum_{i=1}^{6} \omega(A_i)^p}$, $\omega(A_j) = \frac{1-TE(j)}{\sum_{i=1}^{6} 1-TE(i)}$, $TE(j)$ est le taux d'erreur obtenu pour l'attribut A_j. Le paramètre p est déterminé de manière empirique.

La classe d'appartenance de chaque image d_T de B_{Test} est alors celle qui maximise cette probabilité (c'est le critère classique du « Maximum a Posteriori » (MAP)).

$$C^{v\vee t}(d_t) = \text{argmax}_{k\in\{1,2,...,c\}} P^{v\vee t}_{d_T}(C_k) \tag{31}$$

La figure 7 décrit les résultats obtenus pour la fusion de la classification textuelle sans thésaurus (T.E. 13.72%) et de plusieurs classifications visuelles. Le premier résultat (T+Vis[Locaux])

est obtenu à partir des meilleures classifications par fusion précoce des locaux ($L \in [1,4]$) uniquement. Le deuxième (T+Vis[Globaux]) considère les classifications sur les indices globaux uniquement. Le troisième (T+Vis[Locaux+Globaux]) utilise les meilleurs paramètres de fusion précoce des indices locaux et globaux ($L \in [1,5]$). Le dernier (T+Vis[Dir+Globaux]) prend en compte les globaux pour les attributs rouge, vert, bleu et luminance, et la direction locale calculée par DKL(r1,r1). Sur cette figure, on remarque que les locaux accélèrent le gain de classification par rapport à p, montrant donc que les poids $\omega(A_j)$ sont mieux adaptés que ceux des méthodes globales. On remarque aussi que les quatres méthodes tendent pour $p = 4$ vers le même résultat. Le tableau 9 donne le gain final que l'on peut espérer du rehaussement de la classification textuelle par la classifiation visuelle. Pour p grand ($p > 8$), toutes les méthodes convergent vers le T.E. textuel.

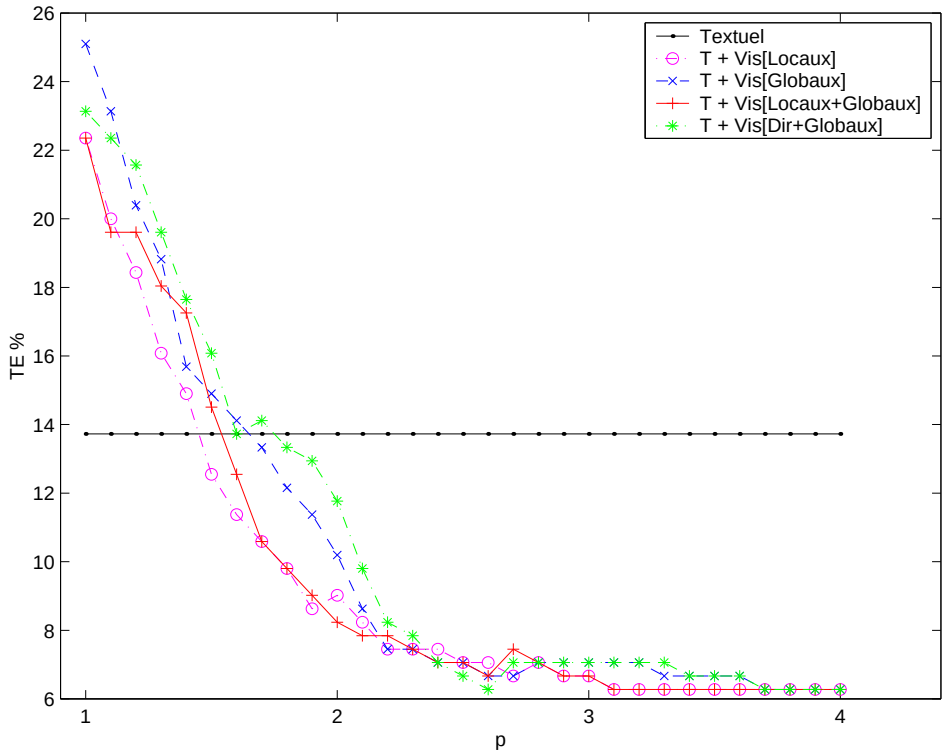


FIG. 7 – Influence de p sur le Taux d'Erreur(T.E. en %) pour la fusion tardive des probabilités textuelles (T) et visuelles (Vis) de différents indices visuels

Textuelle sans thésaurus	Fusion textuelle/visuelle	Gain
13.72	6.27	+54.3

TAB. 9 – Résultat final du rehaussement de la classification textuelle par la classification visuelle par fusion tardive (en %)

6 Discussion

Notre corpus n'étant que de 665 images, notre méthode doit être testée sur une base de données plus importantes afin d'affiner les résultats, et éviter les artefacts. Cependant, le cacule

des intervalles de confiance (tableau 10) montre que si la même étude était réalisée sur une autre base de données, les résultats finaux seraient assez proches de ceux obtenus. En effet, ces intervalles ne se chevauchent pas ($(13.72 - 4.22) > (6.27 + 2.98)$), donc ces résultats sont significatifs. De plus, étant donné la taille réduite de notre corpus, nous n'avons pas pu optimiser nos paramètres sur un ensemble de développement, mais leurs valeurs ne devraient pas trop varier.

	Aléatoire	Meilleur visuelle	Textuelle sans thésaurus	Fusion textuelle/visuelle
T.E.	91.60	71.16	13.72	6.27
Intervalle de confiance	±3.40	±5.56	±4.22	±2.98

TAB. 10 – Significativité de quelques Taux d'Erreur (en %) pour garantir que 95% des échantillons y sont contenus

De nombreux critères et paramètres restent encore à étudier pour améliorer la description visuelle. On peut citer l'influence de la taille et de la forme des régions d'intérêt. Une étude de l'intérêt des indices locaux par rapport aux locaux a déjà été faite dans [TGL03]. Une autre étude pourra également porter sur l'utilité d'utiliser les attributs Rouge, Vert, Bleu, Luminance et Direction. En effet, on pourrait très bien n'utiliser qu'un ou plusieurs de ces attributs. Cependant, à taux d'erreur identiques, des attributs ou méthodes différents apportent une information différente. Il est donc intéressant de combiner plusieurs méthodes et attributs. D'autres attributs comme la texture ou la forme pourraient être utilisés. D'autres méthodes de pondération $\omega(A_j)$ pourraient donner de meilleurs résultats (entropie des distributions...). Comme pour certaines images, la méthode des sous-images peut être inefficace, il faudrait une méthode automatique qui détermine pour qu'elles images il est intéressant de travailler avec des régions d'intérêt.

Au niveau de la complexité, la CAH est une méthode peu performante, il existe cependant des méthodes qui permettent de l'améliorer. La méthode par fusion précoce des indices visuels que nous avons utilisé est coûteuse en nombre d'opérations effectuées. En effet, pour calculer la distance entre deux images pour un attribut, il faut 25 opérations pour $L = 5$, et le gain de classification n'est que d'environ 2% par rapport à la distance entre les indices visuels qui se réalisent en une seule opération. Cette méthode de fusion précoce ne pourra être validée que sur un plus grand corpus.

Conclusion

Ce travail a montré qu'il existe un lien entre l'indexation textuelle et l'indexation visuelle d'une image. Ce lien nous permet d'améliorer de 54% la pertinence des résultats d'une recherche d'images par mot-clés privée d'un thésaurus. L'intérêt principale de cette méthode est qu'elle est simple et entièrement automatique. Ce système pourrait donc être mis en oeuvre sur des moteurs de recherches d'images (tel que Google). Il permettrait de sélectionner les images par rapport aux mots de la requête, puis de les filtrer et de les classer par rapport à leur contenu visuel.

Nous avons étudiés les indices visuels par rapport à des classes textuelles. Nous pourrions inverser l'expérience en considérant les indices textuels par rapport à des classes visuelles. Cette méthode permettrait par exemple de corriger une mauvaise indexation textuelle à l'aide du contenu visuel. Par exemple, si l'image d'un graphique sur la population ouvrière a été étiqueté automatiquement par 'femme' et 'ouvrière', une comparaison avec des classes visuelles représentant des femmes montrerait l'erreur d'indexation et permettrait d'enlever le mot 'femme'.

Références

[BK02] Marinette Bouet and Ali Khenchaf. Traitement de l'information multimédia : recherche de média image. *Ingénierie des systèmes d'information (RSTI série ISI-NIS)*, 7(5-6) : 65–90, 2002.

[BLM02] E. Bruno, J. Le Maitre, and E. Murisasco. Indexation et interrogation de photos de presse décrites en MPEG-7 et stockées dans une base de données XML. *Ingénierie des systèmes d'information (RSTI série ISI-NIS)*, 7(5-6) : 169–186, 2002.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

[Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6) : 679–698, 1986.

[MCM02] Jean Martinet, Yves Chiaramella, and Philippe Mulhem. Un modèle vectoriel étendu de recherche d'informations adapté aux images. *Actes du XXème Congrès INFORSID*, pages 337–348, 4-7 juin 2002.

[MM99] Wei-Ying Ma and B. S. Manjunath. Netra : A toolbox for navigating large image databases. *Multimedia Systems*, 7(3) :184–198, 1999.

[MM02] B.S. Manjunath and Wei-Ying Ma. Texture features for image retrieval. In V. Castelli and L. D. Bergman, editors, *Image Databases*, chapter 12, pages 313–344. John Wiley & Sons, 2002.

[MSS02] B.S. Manjunath, Philippe Salembier, and Thomas Sikora, editors. *Introduction to* MPEG-7. John Wiley & Sons, 2002.

[Nib93] W. Niblack. The QBIC project : qurerying images by content using color, texture and shape. *Proceedings SPIE : Storage and Retrieval for Image and Video Database*, pages 173–181, 1993.

[NMMB98] C. Nastar, M. Mitschke, C. Meilhac, and N. Boujemaa. Surfimage : a flexible content-based image retrieval system. In *The* 6^{th} *ACM International Multimedia Conference (MM'98)*, 1998.

[RJ76] S.E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Sciences*, 27(3) : 129–146, 1976.

[Rui97] Y. Rui. A relevance feedback architecture in content-based multimedia information retrieval systems. In *Proceedings IEEE Workshop Content-Based Access of Image and Video Libraries*, 1997.

[Sal71] G. Salton. *The SMART Retrieval System ; Experiments in Automatic Document Processing*. Englenwood Cliffs, Prenctice-Hall, New Jersey, 1971.

[SB88] G. Salton and C. Buckley. Term-weighting approaches in automatic retrieval. *Information processing and management*, 24(5) : 513–523, 1988.

[SC96] John R. Smith and Shih-Fu Chang. Visualseek : A fully automated content-based image query system. In *ACM Multimedia*, pages 87–98, 1996.

[SFW83] G. Salton, E.A. Fox, and H. Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11) : 1022–1036, 1983.

[SL68] G. Salton and M.J. Lesk. Computer evaluation of indexing and text-processing. *Journal of the ACM*, 15(1) : 8–36, 1968.

[Smi02] John R. Smith. Color for image retrieval. In V. Castelli and L. D. Bergman, editors, *Image Databases*, chapter 11, pages 285–312. John Wiley & Sons, 2002.

[TGL03] S. Tollari, H. Glotin, and J. Le Maitre. Mise en relation et fusion d'indices textuels et visuels pour une recherche d'images par le contenu. *Actes des 19ièmes Journées Bases de Données Avancées, article soumi*, 2003.