HAL
open science

# Traitement automatique des langues pour l'indexation d'images

Pierre Tirilly

## To cite this version:

Pierre Tirilly. Traitement automatique des langues pour l'indexation d'images. Interface homme-machine [cs.HC]. Université Rennes 1, 2010. Français. NNT : . tel-00516422

**HAL Id: tel-00516422**

**https://theses.hal.science/tel-00516422**

Submitted on 9 Sep 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N$^o$ d'ordre : 4139 ANNÉE 2010

**THÈSE / UNIVERSITÉ DE RENNES 1**
*sous le sceau de l'Université Européenne de Bretagne*

pour le grade de

**DOCTEUR DE L'UNIVERSITÉ DE RENNES 1**

*Mention : Informatique*

**École doctorale MATISSE**

présentée par


# Pierre Tirilly

préparée à l'unité mixte de recherche 6074 IRISA
Institut de Recherche en Informatique et Systèmes Aléatoires
IFSIC


---

# Traitement automatique des langues pour l'indexation d'images

**Thèse soutenue à l'IRISA**
**le 07/07/2010**

devant le jury composé de :

**Patrick GALLINARI**
Professeur Université Pierre et Marie Curie / président

**Mohand BOUGHANEM**
Professeur Université Paul Sabatier / rapporteur

**Philippe MULHEM**
Chargé de recherche CNRS / rapporteur

**Christophe GARCIA**
Ingénieur de recherche FT R&D / examinateur

**Patrick GROS**
Directeur de recherche INRIA / directeur de thèse

**Vincent CLAVEAU**
Chargé de recherche CNRS / co-directeur de thèse

N$^o$ d'ordre : 4139 ANNÉE 2010

**THÈSE / UNIVERSITÉ DE RENNES 1**
*sous le sceau de l'Université Européenne de Bretagne*

pour le grade de

**DOCTEUR DE L'UNIVERSITÉ DE RENNES 1**

*Mention : Informatique*

**École doctorale MATISSE**

présentée par


# Pierre Tirilly

préparée à l'unité mixte de recherche 6074 IRISA
Institut de Recherche en Informatique et Systèmes Aléatoires
IFSIC


---

# Traitement automatique des langues pour l'indexation d'images

**Thèse soutenue à l'IRISA**
**le 07/07/2010**

devant le jury composé de :

**Patrick GALLINARI**
Professeur Université Pierre et Marie Curie / président

**Mohand BOUGHANEM**
Professeur Université Paul Sabatier / rapporteur

**Philippe MULHEM**
Chargé de recherche CNRS / rapporteur

**Christophe GARCIA**
Ingénieur de recherche FT R&D / examinateur

**Patrick GROS**
Directeur de recherche INRIA / directeur de thèse

**Vincent CLAVEAU**
Chargé de recherche CNRS / co-directeur de thèse

## Remerciements

Je tiens tout d'abord à remercier Vincent Claveau et Patrick Gros de m'avoir donné l'opportunité de réaliser cette thèse sous leur direction. L'originalité de ces travaux doit beaucoup à leur vision du domaine de la recherche d'information multimédia. Leur confiance, leur patience et leur disponibilité ont été des éléments décisifs au bon déroulement de cette thèse. J'ai beaucoup appris à leur contact, tant en termes de technique que de philosophie de recherche, et je les en remercie sincèrement.

Je remercie aussi Mohand Boughanem et Philippe Mulhem d'avoir accepté d'être les rapporteurs de ces travaux, apportant, grâce à leurs expertises complémentaires, un point de vue complet sur les différentes facettes de cette thèse. Leurs remarques et questions ont été très enrichissantes. Je remercie également Patrick Gallinari qui m'a fait l'honneur de présider le jury, ainsi que Christophe Garcia qui a accepté de s'y joindre.

Rédiger un manuscrit de thèse est un effort qu'il est difficile de porter seul. Je remercie amicalement Émilie, Yann et Fabien pour leur relecture attentive de celui-ci, qui a permis d'en améliorer notablement la qualité. De même, je remercie tous les membres de l'équipe Texmex qui ont pu prendre part aux répétitions de soutenance : leurs remarques éclairées m'ont été plus qu'utiles.

Je remercie également l'ensemble de l'équipe Texmex, dont mes (nombreux) voisins de bureau successifs, pour leur accueil chaleureux et pour l'ambiance studieuse mais néanmoins décontractée qui règne dans les bureaux et lors des séminaires. Les longues conversations échangées avec Nguyen Khang Pham et François Poulet au sujet des mots visuels m'ont été très profitables. L'aide d'Hervé Jégou et de Laurent Amsaleg sur les questions de *clustering* m'a apporté un sérieux coup de pouce pour la mise en oeuvre des mes expérimentations. Le soutien technique des ingénieurs de l'équipe, Cédric Dufouil, Arnaud Dupuis et Sébastien Campion, a également été très appréciable. Enfin, j'adresse un grand merci à Loïc Lesage qui réalise chaque jour l'exploit de rendre les formalités administratives parfaitement indolores.

Je remercie enfin tous ceux, famille et amis, qui m'ont, de près ou de loin, accompagné durant cette thèse, ainsi que les professeurs qui, tout au long de ma scolarité, m'ont donné le goût des sciences et l'envie de chercher. J'ai une pensée tout particulière pour ma mère qui n'aura pas vu ce travail mené à terme, mais dont la foi inébranlable en ce que je fais m'accompagne toujours.

# Table des matières

**Introduction** **9**

**1 Indexation et recherche d'images : état de l'art** **13**
1.1 Recherche d'information : généralités . . . 13
1.1.1 Indexation des documents . . . 15
1.1.2 Formulation et indexation de la requête . . . 15
1.1.3 Phase de recherche . . . 15
1.1.4 Présentation des résultats . . . 15
1.1.5 Reformulation de la requête et retour de pertinence . . . 16
1.1.6 Évaluation des SRI . . . 16
1.1.6.1 Conditions de l'évaluation . . . 16
1.1.6.2 Mesures de performances . . . 17
1.1.6.3 Discussion sur l'évaluation des systèmes de recherche d'information . . . 19
1.2 Recherche d'images : spécificités . . . 19
1.2.1 Nature des descripteurs . . . 20
1.2.2 Formulation de la requête . . . 20
1.2.3 Le fossé sémantique . . . 22
1.3 Recherche d'images par le contenu . . . 22
1.3.1 Notion d'images similaires . . . 22
1.3.1.1 Recherche d'objets ou de scènes identiques . . . 23
1.3.1.2 Recherche d'images catégorisées . . . 23
1.3.1.3 Recherche de copies . . . 24
1.3.2 Description globale des images . . . 24
1.3.2.1 Descripteurs de couleurs . . . 25
1.3.2.2 Descripteurs de texture . . . 26
1.3.2.3 Descripteurs de forme . . . 27
1.3.2.4 Comparaison d'images par descripteurs globaux . . . 28
1.3.2.5 Bilan des approches globales . . . 31
1.3.3 Description locale des images . . . 31
1.3.3.1 Diviser les images en régions . . . 31
1.3.3.2 Descripteurs de régions d'image . . . 33
1.3.3.3 Comparaison d'images par régions . . . 34
1.3.3.4 Bilan des approches locales . . . 36
1.4 Recherche sémantique d'images . . . 36
1.4.1 Approches *bottom-up* et approches *top-down* . . . 37
1.4.2 Nature de l'information textuelle . . . 38
1.4.3 Annotation d'images . . . 38
1.4.3.1 Principaux modèles d'annotation . . . 39

1.4.3.2 Évaluation des systèmes d'annotation . . . . . 44
1.4.4 Fusionner les informations visuelles et textuelles . . . . . 45
1.4.4.1 Fusion précoce . . . . . 45
1.4.4.2 Fusion tardive . . . . . 46
1.4.4.3 Réhaussement d'une information par l'autre . . . . . 46
1.4.4.4 Conclusion sur la fusion des informations textuelles et visuelles . . . . . 47
1.5 Conclusion . . . . . 47

**2 Traitement automatique des langues, recherche d'information textuelle et recherche d'images 49**
2.1 Introduction au traitement automatique des langues . . . . . 49
2.1.1 Niveaux d'analyse linguistique . . . . . 50
2.1.1.1 Analyse morphologique . . . . . 50
2.1.1.2 Analyse syntaxique . . . . . 51
2.1.1.3 Analyse sémantique . . . . . 51
2.1.2 Approches classiques en TAL . . . . . 51
2.1.2.1 Approches symboliques . . . . . 51
2.1.2.2 Approches statistiques . . . . . 52
2.1.2.3 Approches mixtes . . . . . 52
2.1.3 Quelques applications classiques du TAL . . . . . 52
2.2 Recherche d'information textuelle . . . . . 53
2.2.1 Principe . . . . . 53
2.2.2 Description des textes . . . . . 53
2.2.3 Structure d'index . . . . . 54
2.2.4 Modèles de recherche d'information . . . . . 54
2.2.4.1 Modèles ensemblistes . . . . . 54
2.2.4.2 Modèles booléens . . . . . 55
2.2.4.3 Modèles vectoriels . . . . . 55
2.2.4.4 Modèles probabilistes . . . . . 55
2.2.4.5 Remarques sur les modèles de recherche d'information . . . 56
2.2.5 TAL et RI . . . . . 56
2.3 Des mots aux images . . . . . 57
2.3.1 TAL et recherche d'images par le contenu . . . . . 57
2.3.1.1 Décrire les images comme des textes : les mots visuels . . . 58
2.3.1.2 Quelles méthodes issues du texte pour l'indexation à base de mots visuels ? . . . . . 59
2.3.2 TAL et recherche sémantique d'images . . . . . 60
2.3.2.1 Corpus bimodaux existants . . . . . 60
2.3.2.2 Utilisation d'outils du TAL en indexation sémantique d'images . . . . . 60
2.4 Conclusion et axes de travail . . . . . 61

**3 Mesurer la pertinence des mots visuels 63**
3.1 Mots visuels et mots textuels : différences fondamentales . . . . . 63
3.1.1 Origine du vocabulaire . . . . . 63
3.1.2 Sens des mots visuels . . . . . 64
3.1.3 Longueur des documents . . . . . 65
3.1.4 Fréquence des mots dans les documents . . . . . 66

3.1.5 Requêtes . . . 67
3.2 *Stop-lists* . . . 68
3.2.1 *Stop-lists* basées sur la fréquence . . . 68
3.2.2 *Stop-lists* basées sur pLSA . . . 69
3.2.2.1 Principe de pLSA . . . 69
3.2.2.2 Constitution d'une *stop-list* grâce à pLSA . . . 70
3.3 Schémas de pondération . . . 70
3.3.1 Pondérations et modèle vectoriel . . . 70
3.3.1.1 Pondérations locales . . . 71
3.3.1.2 Pondérations globales . . . 72
3.3.1.3 Facteur de normalisation . . . 73
3.3.2 Pondérations et modèles probabilistes . . . 73
3.3.2.1 Lien entre modèles probabilistes et vectoriels . . . 74
3.3.2.2 Pondérations issues des modèles *Best Match* . . . 74
3.3.2.3 Pondérations issues des modèles *Divergence From Randomness* . . . 75
3.4 Pondérations pour la recherche d'images . . . 78
3.4.1 Pondération globale . . . 78
3.4.2 Pondération DFR . . . 79
3.5 Expérimentations . . . 79
3.5.1 Problèmes traités et données associées . . . 79
3.5.1.1 Recherche de scènes identiques . . . 79
3.5.1.2 Recherche d'objets catégorisés . . . 80
3.5.2 Protocole expérimental . . . 81
3.5.2.1 Vocabulaire visuel . . . 81
3.5.2.2 Requêtes . . . 81
3.5.2.3 Évaluation . . . 81
3.5.3 Expériences sur les *stop-lists* . . . 82
3.5.3.1 *Stop-lists* testées . . . 82
3.5.3.2 Résultats . . . 82
3.5.4 Expériences sur les distances . . . 83
3.5.4.1 Distances testées . . . 83
3.5.4.2 Résultats . . . 83
3.5.5 Expériences sur les pondérations . . . 84
3.5.5.1 Pondérations testées . . . 84
3.5.5.2 Résultats . . . 85
3.5.6 Discussion . . . 87
3.5.6.1 Effet de $k$ sur l'usage des distances $L_k$ . . . 87
3.5.6.2 Pondérations locales . . . 91
3.5.6.3 Pondérations globales . . . 92
3.5.7 Pondérations DFR . . . 93
3.5.7.1 Influence de la nature des requêtes . . . 93
3.5.7.2 Relation entre distances de Minkowski et pondérations . . . 94
3.6 Travaux connexes . . . 94
3.6.1 Mots visuels et distances . . . 94
3.6.2 Mots visuels et pondérations . . . 95
3.7 Conclusion . . . 95

**4 Modèles de langues et images** **99**
4.1 Modèles de langues . . . 99
4.1.1 Fonctionnement . . . 99
4.1.1.1 Modélisation des séquences de termes . . . 100
4.1.1.2 Lissage : principe . . . 101
4.1.1.3 Quelques stratégies de lissage . . . 102
4.1.2 Applications . . . 103
4.1.2.1 Modèles de langues et classification . . . 103
4.1.2.2 Autres applications . . . 104
4.2 Modèles de langues et mots visuels . . . 104
4.2.1 Problématique . . . 104
4.2.2 Modéliser les images comme des séquences de mots visuels . . . 105
4.2.2.1 Choix des axes . . . 105
4.2.2.2 Élimination des mots visuels redondants . . . 108
4.3 Expérimentations . . . 110
4.3.1 Protocole expérimental . . . 110
4.3.1.1 Données . . . 110
4.3.1.2 Vocabulaire visuel . . . 110
4.3.1.3 Performance measure . . . 110
4.3.1.4 *Baseline* . . . 110
4.3.1.5 Implémentation des modèles de langues . . . 110
4.3.2 Résultats . . . 111
4.3.2.1 Choix de l'axe . . . 111
4.3.2.2 Élimination des mots visuels redondants . . . 112
4.3.2.3 Choix de la longueur des $n$-grammes . . . 112
4.3.2.4 Choix du lissage . . . 113
4.3.2.5 Performances globales du système . . . 113
4.4 Travaux connexes . . . 116
4.4.1 Modèles de langues et indexation d'images . . . 116
4.4.2 Relations géométriques entre mots visuels . . . 116
4.5 Conclusion . . . 117

**5 Exploitation conjointe des textes et images** **119**
5.1 Données utilisées . . . 119
5.1.1 Description . . . 119
5.1.2 Intérêt de ces données . . . 120
5.2 Caractériser le fossé sémantique . . . 120
5.2.1 Principe . . . 122
5.2.2 Protocole expérimental . . . 122
5.2.2.1 Descripteurs visuels utilisés . . . 122
5.2.2.2 Recherche textuelle . . . 123
5.2.2.3 Calculs de corrélation . . . 123
5.2.3 Résultats et discussion . . . 124
5.3 Utilisation du texte pour annoter les images . . . 126
5.3.1 Association d'indices visuels et textuels de haut-niveau . . . 126
5.3.2 Les entités nommées comme indices textuels . . . 127
5.3.2.1 Systèmes de détection et catégorisation des entités nommées 127
5.3.2.2 Système utilisé . . . 127
5.3.3 Indices visuels associés aux entités nommées . . . 128

5.3.4 Annotation des images par les entités nommées . . . . . 129
5.3.4.1 Sélection des entités nommées candidates à l'annotation . . 129
5.3.4.2 Nombre d'entités nommées candidates retenues . . . . . 130
5.3.4.3 Cas d'ambiguité . . . . . 130
5.3.5 Expérimentations . . . . . 130
5.3.5.1 Détection des concepts visuels . . . . . 130
5.3.5.2 Vérité-terrain et mesure des performances . . . . . 131
5.3.5.3 Résultats et discussion . . . . . 131
5.4 Bilan . . . . . 132

**A Expériences sur les pondérations : détail des résultats 141**
A.1 Corpus utilisés . . . . . 141
A.2 Schémas de pondération . . . . . 141
A.3 Distances employées . . . . . 142
A.4 Mesures de performance . . . . . 142

**B Fonctionnement et évaluation du détecteur de logos 163**
B.1 Algorithme de détection . . . . . 163
B.1.1 Calcul du score de détection . . . . . 164
B.1.2 Complexité . . . . . 166
B.1.2.1 Phase d'apprentissage . . . . . 166
B.1.2.2 Phase de détection . . . . . 166
B.1.2.3 Réduction du nombre de mots . . . . . 166
B.1.2.4 Taille minimale des logos . . . . . 167
B.2 Évaluation . . . . . 167
B.2.1 Conditions expérimentales . . . . . 167
B.2.1.1 Données d'apprentissage . . . . . 167
B.2.1.2 Conditions d'évaluation . . . . . 167
B.2.1.3 Construction des mots visuels . . . . . 167
B.2.2 Résultats . . . . . 168
B.2.2.1 Taille du vocabulaire . . . . . 168
B.2.2.2 Taille minimale des logos . . . . . 168
B.2.2.3 Élimination des mots superflus . . . . . 168
B.2.2.4 Scores de détection . . . . . 169
B.2.2.5 Temps d'exécution . . . . . 170
B.3 Travaux connexes . . . . . 171

**Bibliographie 173**

**Table des figures 189**

**Liste des tableaux 191**

**Liste des algorithmes 193**

# Introduction

Toute production de documents implique, pour ne pas être vaine, d'avoir les moyens d'accéder efficacement à l'information produite. En ce sens, la recherche d'information, qui englobe le stockage, l'organisation et la restitution de l'information, est une discipline presque aussi ancienne que l'écriture. Mais c'est grâce à l'informatisation, qui a, à la fois, multiplié les moyens de produire de l'information et offert la possibilité d'automatiser l'accès à celle-ci, que la recherche d'information s'est développée en tant que science.

Les premiers systèmes de recherche d'information, apparus dans les années 1960, se limitaient, pour des raisons pratiques, à l'indexation de documents composés uniquement de texte, principalement de la documentation technique. S'imposant dans le monde professionnel au fur et à mesure de la numérisation de l'information, ils s'ouvrent au grand public, sous la forme des moteurs de recherche, à l'apparition d'Internet, qui offre aux particuliers l'accès à des quantités considérables d'information, mais aussi la possibilité de diffuser leurs propres contenus. Ce dernier aspect s'est particulièrement développé ces dernières années avec l'apparition de plate-formes comme les *blogs* qui facilitent la production de contenu, notamment textuel, sur le web. Ainsi, Google, le moteur de recherche le plus populaire, possédait un index de 26 millions de documents en 1998, puis 1 milliard en 2000, taille estimée aujourd'hui à plus de 25 milliards[1].

Parallèlement à cela, le traitement automatique des langues (TAL) s'est développé durant la même période, fournissant aux systèmes de recherche d'information des outils pour indexer les documents de manière plus riche et plus pertinente. Aujourd'hui, une grande majorité des moteurs de recherche intègrent des fonctionnalités issues du TAL (principalement la prise en compte des variantes - en genre, en nombre - des mots, mais aussi la désambiguisation des requêtes, la recherche à partir de synonymes...).

Bien que plus récent, le domaine de l'indexation d'images connaît une évolution assez similaire à celle de l'indexation de textes. Les premiers systèmes de recherche d'images apparaissent dans les années 1990, poussés par les travaux en vision par ordinateur, qui offrent des techniques pour analyser le contenu visuel des images, et l'augmentation de la puissance des machines, qui permet d'appliquer ces techniques assez efficacement pour traiter des collections d'images. Ces premiers systèmes se limitent néanmoins à des collections fermées et de taille restreinte, tant pour des raisons de disponibilité des données adéquates que de coût calculatoire des algorithmes employés. En parallèle, Internet se développe et propose déjà au grand public, à la fin des années 1990, un accès à de grandes quantités d'images stockées de manière décentralisée et non structurée. Mais c'est surtout la récente démocratisation des moyens numériques d'acquisition d'images (appareils photo numériques, désormais intégrés à tous les modèles de téléphones portables, mais aussi webcam et caméras numériques) qui fait exploser la quantité d'images à indexer, comme l'atteste

[1] Google a même annoncé en 2008 avoir recensé 1000 milliards de pages web (dont une grande partie, néanmoins, est générée automatiquement) : `http ://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html`.

le succès d'entreprises comme FlickR (4 milliards d'images hébergées) ou Facebook (10 milliards).

Cependant, malgré des évolution et objectif communs, la recherche d'images n'a que rarement tiré parti des travaux produits dans un cadre de recherche d'information textuelle, en dehors des principes généraux de la recherche d'information comme le schéma *requête* → *recherche dans une base* → *présentation des résultats* ou la notion de descripteurs de documents. Ce fait est d'autant plus étonnant qu'il est apparu rapidement que la manière la plus intuitive et la plus efficace de décrire le contenu des images est de leur adjoindre de l'information textuelle. Malgré cela, la recherche académique s'est principalement concentrée, dans un premier temps, sur les techniques de vision par ordinateur permettant de décrire et comparer le contenu visuel des images (description en termes de couleurs, textures, formes, dite de bas-niveau), pour pouvoir rechercher les images similaires à des requêtes ou de classer les images pour en organiser les collections. Cette description bas-niveau a ensuite été associée à des techniques d'apprentissage artificiel pour obtenir une description plus sémantique des images (description en termes d'objets ou de concepts, dite de haut-niveau), généralement sous forme mots-clefs.

Nos travaux se situent à l'intersection entre indexation d'images et TAL : nous explorons les possibilités d'intégration de méthodes de TAL dans les systèmes d'indexation et de recherche d'images. Nous souhaitons aborder ces systèmes sous un angle nouveau qui permette de les améliorer en tirant parti des méthodes qui ont montré leur efficacité pour la recherche d'information textuelle. Cette intégration peut se situer à deux niveaux, que nous étudions tous les deux. Le premier est celui de la description bas-niveau des images. Nous devons, pour travailler à ce niveau, utiliser une description initiale du contenu visuel des images qui soit suffisamment proche de la description des textes utilisée en recherche d'information pour permettre de lui appliquer des méthodes de TAL. Le second niveau est celui de la description haut-niveau des images. Nous souhaitons ici exploiter le TAL pour permettre de choisir des mots-clefs capables d'exprimer les concepts contenus dans les images. Pour cela, nous nous appuyons sur les textes qui accompagnent parfois les images (dans le cadre d'articles de presse, de pages web. . . ) : c'est à ces textes que nous appliquons des méthodes de TAL pour en extraire les mots-clefs recherchés.

Pour étudier les interactions possibles entre TAL et description bas-niveau des images, nous nous appuyons sur les travaux de Sivic et Zisserman [SZ03] : ceux-ci ont en effet proposé une description des images reposant sur des régions d'images élémentaires appelées mots visuels, par analogie avec les mots constituant les textes. Bien que l'approche qu'ils emploient s'inscrive dans un cadre général déjà existant, ce sont eux qui ont mis en avant la proximité entre ce type de description bas-niveau et la recherche d'information textuelle, tant par les méthodes utilisées (pondération des mots visuels, index inversé) que par le nom de leur système, *video-google*. C'est donc dans ce cadre que nous nous plaçons pour intégrer des méthodes de TAL à un système de recherche d'images. Pour cela, nous devons néanmoins prendre en compte le fait que, malgré certaines ressemblances, mots visuels et mots textuels restent intrinsèquement différents : les mots textuels s'inscrivent dans le cadre des langues naturelles, dont les propriétés sont exploitées par de nombreux outils de TAL, alors que les mots visuels sont purement artificiels. Seule une fraction des techniques de TAL sont donc applicables à la description du contenu visuel : celles se basant sur les propriétés statistiques des mots.

Cependant, cette gamme de méthode du TAL suffit à traiter deux problématiques majeures des mots visuels, qui sont également des problématiques essentielles de l'indexation

de textes : le choix des mots (ou des mots visuels) les plus pertinents pour décrire les textes (ou les images) et la prise en compte de relations de dépendance entre les mots, supposés indépendants dans les modèles traditionnels de recherche d'information. C'est donc sous ces deux angles complémentaires que nous plaçons nos travaux.

La question du choix des mots les plus pertinents a généré de très nombreux travaux autour des concepts de *stop-lists*, des listes qui regroupent les mots peu significatifs, et de pondération des termes, qui vise à accorder des poids plus importants aux mots les plus pertinents lors de la phase de recherche des documents. Ces techniques reposent essentiellement sur la distribution statistique des mots dans les documents et dans les collections de documents, ce qui les rend facilement transposables dans le cadre des mots visuels. Nous proposons ici une méthode de constitution de *stop-lists* pour les mots visuels, puis une étude détaillée des schémas de pondération classiques appliqués au cas des images, ainsi que de nouvelles pondérations .

La question des dépendances entre les mots peut également être abordée en recherche d'information d'un point de vue statistique. Ces dernières années ont en effet vu se multiplier les travaux sur l'utilisation des modèles de langues en recherche d'information. Ces modèles de langues sont des outils du TAL principalement développés dans le cadre de la reconnaissance de la parole. Ils se basent sur des séquences de mots plutôt que des mots indépendants, et permettent de prendre en compte des relations de dépendance locales entre mots. Nous proposons une nouvelle manière de décrire les images, qui permet de prendre en compte des relations de nature géométrique entre les mots visuels, que nous associons aux modèles de langues dans un cadre de classification d'images. Cette méthode de classification peut être étendue à la recherche d'images.

Pour notre étude de la description haut-niveau des images, les techniques de TAL peuvent être appliquées de manière plus naturelle car nous les appliquons à de textes dont nous pouvons exploiter à la fois les propriétés statistiques et linguistiques. Il faut cependant, pour cela, disposer de données offrant en parallèle des images et des textes de longueur suffisante, et non simplement des mots-clefs ou des légendes, comme c'est le cas dans la majorité des corpus utilisés pour évaluer les méthodes d'indexation sémantique d'images : nous avons donc d'abord cherché un corpus adapté au cadre que nous nous sommes fixé. De plus, il faut composer avec un problème majeur de l'indexation sémantique des images : le fossé sémantique, c'est-à-dire le fait que les descripteurs traditionnels ne parviennent pas à capturer le contenu sémantique des images, et ne peuvent donc pas être mis efficacement en relation avec du texte dans un processus d'annotation d'images. Il est donc illusoire d'espérer mettre au point un système d'annotation entièrement autonome qui associerait simplement des caractéristiques visuelles comme la couleur à des concepts quelconques tirés de textes.

Nous contournons cet écueil en nous concentrant sur des concepts spécifiques identifiables efficacement dans les images comme dans les textes. Nous proposons une méthode d'annotation d'images basée, du côté des textes, sur la détection d'entités nommées, qui nous permet d'extraire des textes des mots adaptés à l'annotation d'images, et, du côté image, à des techniques avancées de vision par ordinateur permettant de vérifier si les entités nommées détectées dans le texte ont leur équivalent dans l'image, pour vérifier si elles consituent des annotations pertinentes des images.

## Organisation du manuscrit

Ce mémoire s'organise d'abord autour de notre problématique générale d'application du TAL à l'indexation d'images, puis traite indépendamment des deux niveaux auxquels se situent nos contributions : la description bas-niveau et la description haut-niveau des images.

Dans le chapitre 1, nous présentons la problématique générale de l'indexation d'images et détaillons les points essentiels d'un système de recherche d'images. Puis nous proposons un état de l'art qui s'articule selon les deux aspects de l'indexation d'images présentés dans cette introduction. Dans un premier temps nous décrivons les systèmes de recherche basés sur une description bas-niveau des images, les descripteurs visuels qu'ils emploient et la manière dont ils sont mis en correspondance. Ensuite, nous décrivons les systèmes de recherche sémantique d'images, basés sur l'utilisation conjointe des descripteurs visuels, issus de la recherche par le contenu, et d'informations textuelles.

Le chapitre 2 présente le cadre de notre étude. Nous y présentons les principales notions de TAL et de recherche d'information textuelle, puis détaillons, pour chaque aspect de la recherche d'images, recherche par le contenu et recherche sémantique, le cadre dans lequel nous nous plaçons pour pouvoir introduire des techniques de TAL : représentation des images en mots visuels et utilisation des ressources textuelles accompagnant les images.

Dans le chapitre 3, nous nous intéressons à la manière de sélectionner les mots visuels le plus pertinents pour décrire des images. Nous proposons une méthode d'élimination des mots visuels les moins significatifs, puis étudions l'influence de différentes méthodes de pondération des mots visuels utilisées traditionnellement en recherche d'information textuelle, ainsi que de nouvelles pondérations que nous proposons. Enfin, nous étudions également l'influence des distances de Minkowski dans ce cadre de recherche d'images, qui s'avère être équivalente à celel des pondérations.

Dans le chapitre 4, nous cherchons à dépasser les limites de l'hypothèse d'indépendance des termes dans le cas des mots visuels en modélisant les images à l'aide de modèles de langues. Nous évaluons cette modélisation sur une tâche de classification, problématique connexe de celles de la recherche d'images.

Dans le chapitre 5, nous nous intéressons au lien entre les images et les textes qui les accompagnent, dans le cadre d'articles de presse. Nous caractérisons d'abord le fossé sémantique qui existe entre les descripteurs visuels et les descripteurs textuels des images, puis nous proposons une méthode d'annotation d'images basée sur les entités nommées extraites des textes des articles, que nous associons à des concepts visuels de haut-niveau détectés dans les images.

Enfin, nous dressons un bilan des résultats obtenus et les replaçons dans le cadre général de l'association entre recherche d'images et TAL. Puis nous proposons quelques perspectives ouvertes par nos travaux, directement liées à ceux-ci ou ayant une portée plus générale.

# Chapitre 1

# Indexation et recherche d'images : état de l'art

Dans ce premier chapitre, nous faisons un état des lieux des recherches effectuées jusqu'à présent en indexation et recherche d'images. Tout d'abord, nous présentons les principes généraux de la recherche d'information, qui peuvent s'appliquer à tout système indépendamment du type de document manipulé (textes, images, pages web, vidéos...). Puis nous exposons les principales spécificités des systèmes de recherche d'images par rapport à ce cadre général. Enfin, nous décrivons les principaux travaux existant en matière de recherche d'images. Nous divisons ces travaux en deux approches majeures, la recherche d'images par le contenu et la recherche sémantique d'images. Cette séparation correspond aux deux axes que suivent nos travaux et que nous avons déjà évoqués en introduction. Il faut néanmoins noter que ces deux approches ne sont pas indépendantes l'une de l'autre. En particulier, la recherche sémantique d'images s'appuie très largement sur les travaux de recherche d'images par le contenu : elle lui emprunte notamment ses descripteurs d'images, bien qu'elle cherche également à dépasser les limites de ceux-ci.

## 1.1 Recherche d'information : généralités

L'objectif d'un système d'indexation et de recherche d'information (SRI) est, étant donné un ensemble de documents (ou *corpus*), de permettre à ses utilisateurs d'accéder le plus rapidement possible aux documents qui correspondent à leurs besoins d'information, et uniquement ceux-ci. Les techniques d'indexation manuelle (classement par auteur ou par mots-clefs par exemple), qui ont commencé à apparaître en même temps que les bibliothèques, nécessitent un travail manuel long et fastidieux. Dans les années 1950 sont apparus les premiers SRI automatiques, dans lesquels les tâches d'indexation et de recherche des documents sont confiés à la machine. Les premiers documents traités étaient des documents constitués uniquement de texte, mais les principes fondamentaux de la recherche d'information sont les mêmes qu'il s'agisse de textes, d'images, de vidéos, de documents audio, ou encore de documents mêlant ces différents types d'information.

L'architecture globale d'un SRI est présentée en figure 1.1. On distingue une étape hors-ligne (l'indexation des documents), qui peut être réalisée à l'avance et une fois pour toutes, des phases en-ligne, pour lesquelles le système est en interaction directe avec l'utilisateur et doit donc répondre dans des délais acceptables.

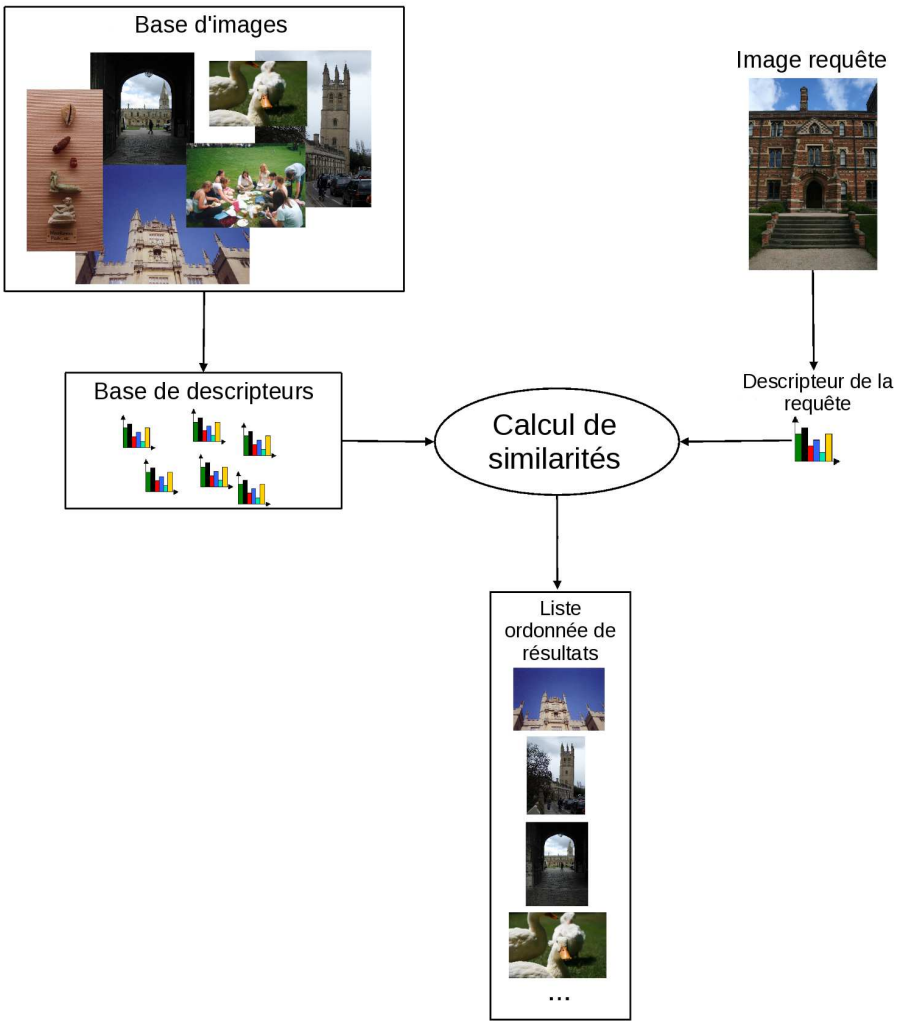



FIG. 1.1 – Un système de recherche d'images classique.

### 1.1.1 Indexation des documents

Le contenu de chaque document à indexer est représenté par la machine sous forme d'un *descripteur*, qui est ensuite stocké. Le but d'un descripteur est d'offrir une représentation du document qui soit à la fois compacte, pour permettre de manipuler de grandes quantités de descripteurs rapidement lors des phases de recherche, et fidèle au sens porté par le document, de telle sorte que deux documents de sens similaires soient associés à des descripteurs similaires, pour qu'une recherche de documents basée sur les descripteurs produise des résultats de recherche pertinents. La phase d'indexation met également en œuvre des *structures d'indexation* spécifiques : ce sont des structures de données qui permettront d'accéder rapidement aux descripteurs intéressants lors d'une phase de recherche. Par exemple, dans le cas des documents textuels, chaque document est représenté typiquement par l'ensemble des mots qui le composent, qui est censé représenter le sens du document ; ainsi, deux documents partageant les mêmes mots sont considérés comme ayant des sens proches.

### 1.1.2 Formulation et indexation de la requête

L'utilisateur fournit au système une requête censée refléter son besoin d'information. La requête est ensuite traduite sous forme d'un ou plusieurs descripteurs similaires à ceux extraits des documents, pour pouvoir comparer requête et documents lors de la phase de recherche. Bien que la formulation d'une requête conforme au besoin d'information de l'utilisateur repose essentiellement sur ce dernier, il est possible de l'assister dans cette tâche, par exemple en lui proposant automatiquement des requêtes proches de la sienne ou en l'invitant à choisir une formulation plus précise si sa requête initiale est ambiguë.

### 1.1.3 Phase de recherche

Le ou les descripteurs extraits de la requête sont comparés aux descripteurs des documents pour fournir, pour chaque document, un score de pertinence exprimant la similarité entre le document courant et la requête. Le nombre de documents dans le corpus étant potentiellement très grand (plusieurs milliards dans le cas du *web*), cette étape peut demander beaucoup de temps et une comparaison linéaire de la requête à chaque document n'est pas envisageable dans la plupart des cas. Pour contourner ce problème, on utilise lors de la phase d'indexation des structures de données qui permettent d'accéder directement à un sous-ensemble de documents susceptibles d'être pertinents, éliminant ainsi d'emblée la majorité des documents du corpus, sans qu'il ne soit nécessaire d'accéder à leur descripteur et de calculer leur similarité vis-à-vis de la requête.

### 1.1.4 Présentation des résultats

Une fois que les documents résultats ont été obtenus, il faut les présenter à l'utilisateur. L'interface la plus répandue est une simple liste des documents dans l'ordre de pertinence décroissante à la requête, en fonction des scores obtenus lors de la phase de recherche (Google, Exalead). Il existe de nombreux autres types d'interfaces, destinés à faciliter non seulement la lecture des résultats, mais aussi la navigation au sein des résultats et de documents connexes, l'exploration des résultats par thématique... On pourra se référer à [BCD08] pour un aperçu très complet des interfaces en RI.

### 1.1.5 Reformulation de la requête et retour de pertinence

Après avoir pris connaissance des résultats proposés, l'utilisateur peut, si les résultats ne le satisfont pas, reformuler sa requête pour obtenir des résultats plus pertinents ou plus spécifiques à un aspect particulier de son besoin d'information. Certains systèmes permettent également de pratiquer un *retour de pertinence* (ou *bouclage de pertinence*, ou encore *relevance feedback*) pour prendre en compte le jugement de l'utilisateur sur la qualité des résultats fournis. L'utilisateur indique au système si les documents obtenus en réponse à sa requête étaient ou non pertinents ; une nouvelle recherche est alors effectuée en tenant compte de ces informations supplémentaires, soit en modifiant la formulation initiale de la requête, soit en jouant sur le calcul des scores.

### 1.1.6 Évaluation des SRI

L'évaluation des SRI est une étape cruciale dans l'élaboration de tels systèmes, puisqu'elle permet de savoir si le système fonctionne correctement ou non. Un SRI performant doit respecter deux critères :

- il doit être rapide (*efficiency*) : les opérations en ligne (calcul du descripteur de la requête, recherche dans la base de descripteurs et présentation des résultats) doivent paraître quasi-immédiates à l'utilisateur (typiquement, en un temps de l'ordre de la seconde). La phase d'indexation, bien que s'effectuant hors-ligne, peut aussi être considérée comme importante, notamment si le corpus indexé est appelé à changer régulièrement (comme les pages web, par exemple) ;
- il doit être pertinent (*effectiveness*) : il doit, pour une requête donnée, renvoyer tous les documents pertinents, et uniquement les documents pertinents. Mesurer cet aspect des performances d'un SRI nécessite de fixer une définition précise de la notion de pertinence, de disposer de documents, de requêtes et de la *vérité-terrain* (jugements de pertinence des documents pour chaque requête) associée, et enfin de définir des mesures qui permettent de refléter les performances globales du système.

Ainsi, l'évaluation de la rapidité des SRI peut se faire assez simplement (mesure des temps d'exécution pour les phases d'indexation et de recherche), mais l'évaluation de leur pertinence nécessite de définir un cadre et des mesures d'évaluation précis. Les deux sous-sections suivantes détaillent d'une part, les conditions habituelles de l'évaluation de la pertinence des SRI, et, d'autre part, les mesures classiques utilisées pour cette évaluation.

#### 1.1.6.1 Conditions de l'évaluation

Évaluer un SRI suppose de disposer de données pour mener à bien cette évaluation. Ces données comprennent :

- un corpus de documents ;
- un ensemble de requêtes ;
- une vérité-terrain, c'est-à-dire, pour chaque couple requête-document, un jugement indiquant si le document répond de manière pertinente à la requête. Cette vérité-terrain doit être mise au point manuellement, par des humains. La notion de pertinence étant en partie subjective, il existe toujours des biais dans la manière dont sont obtenues les vérités-terrain.

Pour pouvoir évaluer de manière simple les SRI à partir de telles données, il faut également poser des hypothèses qui vont restreindre la notion de pertinence et les propriétés des données de test (modèle de Cranfield [Cle67]) :

- hypothèse de jugement total : pour toute requête, on peut dire si chaque document est pertinent ou non. Il n'existe jamais de document dont la pertinence à la requête est inconnue ;
- hypothèse de jugement binaire : un document est soit pertinent pour une requête donnée, soit non pertinent. Il n'y a pas de notion de "document partiellement pertinent" ;
- hypothèse d'absence de mémoire : un document pertinent l'est toujours, même si un autre document similaire a été obtenu plus haut dans la liste des résultats ;
- hypothèse d'absence d'additivité : deux documents non pertinents mais qui, une fois réunis, fourniraient une réponse à la requête, ne peuvent être considérés comme pertinents. Ces deux dernières hypothèses peuvent être vues comme les deux pans d'une notion d'*indépendance entre les documents* : le jugement de pertinence d'un document à une requête ne dépend jamais des autres documents du corpus.

#### 1.1.6.2 Mesures de performances

Les mesures de performances des SRI doivent permettre de répondre aux deux questions suivantes :
- les documents retournés par le système sont-ils pertinents ?
- le système retourne-t-il tous les documents pertinents ?

Pour cela, on dispose de deux mesures princpales : la *précision* et le *rappel* [VR77]. Dans les définitions suivantes, on note :
- $\Psi_i$ : l'ensemble des documents pertinents pour la requête $i$ ;
- $\Delta_i$ : l'ensemble des documents retournés par le SRI pour la requête $i$.

**Précision** Elle indique la proportion de documents pertinents parmi les documents donnés en résultat. Ceci répond donc à la première question posée plus haut : les documents retournés sont-ils pertinents ? La précision $P$ d'un SRI, calculée sur $N$ requêtes, est :

$$P = \frac{1}{N} \sum_{i=1}^{N} \frac{|\Psi_i \cap \Delta_i|}{|\Delta_i|}$$

**Rappel** Il indique la proportion de document pertinents retrouvés par rapport à la totalité des documents pertinents. Il répond donc à la seconde question : le système a-t-il oublié des documents pertinents ? Le rappel $R$ d'un SRI, calculé sur $N$ requêtes, est :

$$R = \frac{1}{N} \sum_{i=1}^{N} \frac{|\Psi_i \cap \Delta_i|}{|\Psi_i|}$$

Les notions de rappel et de précision sont complémentaires : elles représentent les deux aspects de la qualité des SRI, aspects qui sont liés entre eux. En effet, un SRI qui privilégie la précision aura tendance à retourner moins de documents, y compris un nombre inférieur de documents pertinents, diminuant ainsi le rappel. Inversement, un système qui privilégie le rappel aura tendance à renvoyer un plus grand nombre de documents, pour s'assurer de renvoyer tous les documents pertinents, et risque donc d'avoir une précision moindre.

De plus, les mesures de rappel et de précision calculées sur l'ensemble des documents retournés ne sont pas nécessairement représentatives de la qualité du SRI telle qu'elle sera perçue par l'utilisateur. Ce dernier consultera généralement les premiers documents retournés (dont les scores étaient les meilleurs à l'issue de la phase de recherche), mais pas

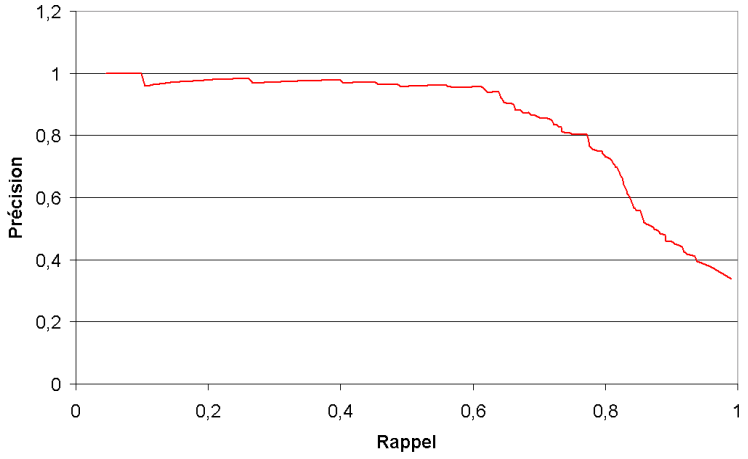



FIG. 1.2 – Exemple de courbe rappel-précision.

les derniers, le système pouvant en retourner des centaines. De plus, les premiers documents retournés auront de fortes chances d'être pertinents, contrairement aux derniers. Pour outrepasser cet inconvénient, on a recours aux DCV.

**DCV** Les DCV (pour *Document Cut-off Values*) indiquent le nombre de documents pris en compte pour le calcul de la performance d'un SRI. Ainsi, des rappels et précisions calculés à une DCV de 50 sont calculés sur les 50 premiers documents retournés uniquement. Ce type de limitation permet d'obtenir des mesures plus locales de performance d'un SRI, et également plus fidèles au ressenti de l'utilisateur.

Le choix de la DCV influe naturellement sur les scores obtenus : plus on utilise un nombre important de documents pour évaluer les performances, plus on risque d'obtenir un rappel élevé et une précision faible, et inversement. Pour avoir un aperçu global des performances des SRI, on utilise une courbe rappel-précision.

**Courbe rappel-précision** Les courbes rappel-précision donnent une image de l'évolution conjointe du rappel et de la précision. Elles sont obtenues en calculant le taux de précision pour différentes valeurs de rappel données, valeurs obtenues en faisant varier la DCV. On utilise généralement 11 points de rappel, par pas de 10% de 0 à 100%.

Il existe également des mesures de performances définies en complémentarité du rappel et de la précision, qui reflètent les mêmes aspects des performances des SRI, comme par exemple le silence $(1-R)$ ou le bruit $(1-P)$. Enfin, il existe des mesures uniques reflétant à la fois le rappel et la précision d'un système, par exemple les *F-mesures* et la *Mean Average Precision (MAP)*.

**F-mesure** Les F-mesures sont des moyennes harmoniques pondérées entre rappel et précision :

$$F_\alpha = \frac{(1+\alpha^2).P.R}{\alpha^2.P+R} \text{ avec } \alpha \geq 0$$

Le paramètre $\alpha$ permet d'attribuer plus ou moins de poids à la précision. Quand $\alpha = 1$, rappel et précision ont un poids identique.

**MAP** La MAP (pour *Mean Average Precision*, précision moyenne non interpolée) est calculée comme la moyenne sur l'ensemble des requêtes des *précisions moyennes* (AP, *Average Precision*) obtenues pour chaque requête. L'AP est elle-même définie comme étant la moyenne des précisions obtenues après chaque document pertinent retrouvé, soit, pour la requête $i$ :

$$\mathrm{AP}_i = \frac{1}{|\Psi_i|} \sum_{d \in (\Delta_i \cap \Psi_i)} P(\mathrm{rang}(d))$$

où $P(r)$ désigne la précision calculée sur les $r$ premiers documents retrouvés. La MAP est donc définie ainsi :

$$\mathrm{MAP} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{AP}_i$$

Elle permet d'avoir une mesure globale de la précision du système, qui prend en compte à la fois le fait que des documents pertinents n'aient pas été retournés et le fait que des documents pertinents se retrouvent ou non en fin de classement des documents retournés.

#### 1.1.6.3 Discussion sur l'évaluation des systèmes de recherche d'information

Les principes d'évaluation présentés ici sont sujets à discussion, tant du point de vue des conditions expérimentales mises en œuvre dans le modèle de Cranfield que de la capacité des mesures d'évaluation à déterminer de manière fiable si un système est meilleur qu'un autre. Néanmoins, de nombreux travaux montrent le bien-fondé de cette démarche d'évaluation, au moins lorsqu'il s'agit de comparer les performances relatives de plusieurs systèmes de recherche d'information. Ainsi, E. Voorhees a montré expérimentalement que le fait que la notion de pertinence soit subjective[1] n'est pas une entrave à la fiabilité de l'évaluation (les performances relatives des systèmes sont les mêmes quand les jugements de pertinence changent) [Voo02]. De même, Buckley et Voorhees ont montré que les mesures d'évaluation classiques, si elles sont effectivement sujettes à des erreurs d'appréciations, sont globalement fiables si le nombre de requêtes de test employé est suffisant (typiquement, supérieur à 50) [BV00]. J. Zobel ajoute que, plus que le choix de la mesure d'évaluation, c'est surtout la significativité statistique des résultats qui compte [Zob98]. Bien que ce cadre d'évaluation et les remarques à son sujet soient issus du domaine de la recherche d'information textuelle, ils nous semblent adaptés à n'importe quel type de documents, et donc aux images. Cependant, la notion de pertinence est beaucoup plus floue dans le cadre des images, et, au-delà des simples appréciations subjectives, elle peut donner lieu à des tâche sde recherche d'images très différentes. Nous aborderons ce point dans la section 1.3.1 page 22.

## 1.2 Recherche d'images : spécificités

Un système de recherche d'images est avant tout un système de recherche d'information classique, il répond donc à toutes les caractéristiques évoquées dans la section précédente. Néanmoins certaines spécificités, dues à la nature des documents , le différencient des SRI classiques, initialement développés pour les documents textuels. Ces spécificités sont liées à la nature des documents manipulés : des images.

[1] C'est une critique classique de l'évaluation des systèmes de recherche d'information : celle-ci ne serait pas fiable car se basant sur des jugements de pertinence statiques, alors que la notion de pertinence est totalement subjective.

### 1.2.1 Nature des descripteurs

Informatiquement, une image est représentée par une matrice de pixels, composants élémentaires des images (*pixel : picture element*). À chaque pixel correspond la couleur de l'image à l'emplacement correspondant, couleur représentée par une (dans le cas des images en niveau de gris) ou plusieurs (dans le cas des images en couleurs) valeurs entières. Un image est donc, du point de vue de la machine, une matrice d'entiers, ou un ensemble de matrices d'entiers (figure 1.3). Cette représentation brute des images ne peut pas être utilisée telle quelle pour l'indexation d'images, elle ne respecte pas les propriétés nécessaires à un bon descripteur évoquées en Section 1.1.1 :

- compacité du descripteur : une image moyenne contient de plusieurs centaines de milliers à plusieurs millions de pixels et nécessite un espace de stockage important (de 500 Ko à plusieurs Mo) ;
- expressivité : la matrice de pixels brute n'est pas représentative du contenu visuel de l'image. Deux images représentant un même objet peuvent correspondre à deux matrices de pixels complètement différentes, pour peu que le fond, les conditions d'illumination, la position de l'objet... changent (figure 1.4). Il n'est donc pas possible de déterminer si deux images sont similaires ou non en les comparant directement pixel par pixel.

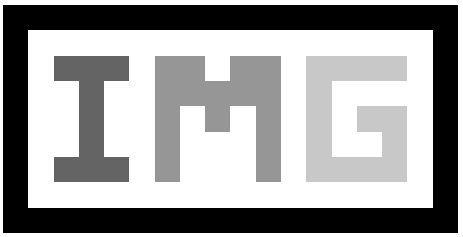
Il est donc nécessaire d'extraire des images des caractéristiques (ou *features*) qui permettent de décrire le contenu de l'image plus efficacement que la simple matrice de pixels. On peut envisager ces caractéristiques de deux points de vue, qui correspondent à deux niveau d'interprétation des images :

- niveau visuel ou physique : il s'agit à ce niveau de caractéristiques représentant les propriétés physiques des images : couleur(s) dominante(s), luminosité, présence de textures données... Ces caractéristiques ne correspondent pas directement à des objets ou des concepts particuliers (par exemple, une pomme) mais uniquement à une description visuelle que l'on pourrait en faire (objet rouge et de forme ronde). Le contenu de l'image n'est pas, ou peu, interprété (définition des couleurs, des caractéristiques de textures...). On parle dans ce cas de *descripteurs de bas-niveau* ;
- niveau sémantique : à ce niveau, on décrit directement les images en fonction des objets (visage, voiture...) ou des concepts (nuit, réunion...) qui les composent. On obtient donc une description directement liée au sens porté par l'image, et non plus à son simple aspect visuel. À ce niveau, le contenu physique des images est fortement interprété, pour obtenir une description du sens de l'image, et non plus de son aspect visuel. On parle alors de *descripteurs de haut-niveau*.

### 1.2.2 Formulation de la requête

Dans un SRI classique, l'utilisateur formule sa requête en langage naturel (généralement, une phrase ou des mots-clefs), qui est également le constituant de base des documents. Il y a donc une correspondance directe entre la manière dont est formulée la requête et celle dont sont formulés les documents. Dans le cas des images, on distingue généralement deux cas de figure :

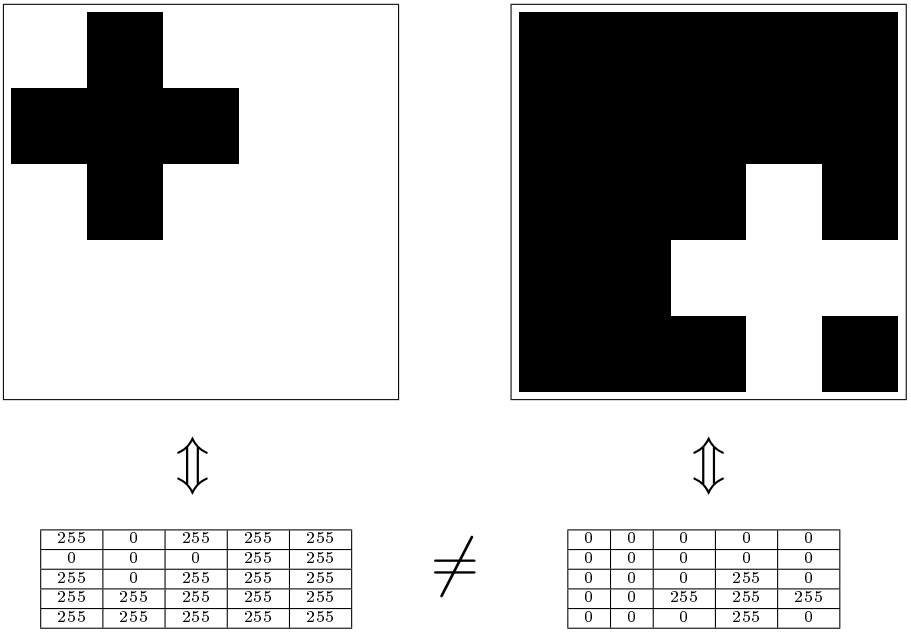
- l'utilisateur fournit une image dont il souhaite obtenir d'autres images qui lui sont similaires. On parle de requête par l'exemple (*QBE, Query By Example*). On se situe ainsi dans le domaine de la recherche d'images par le contenu : disposant d'une image, on peut se contenter de chercher des images similaires sur la base de descripteurs de bas-niveau. Ce cas de figure, néanmoins, ne correspond pas à scénario d'utilisation très réaliste, car cela présuppose que l'utilisateur possède déjà un modèle de l'image




$\Updownarrow$

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 0 |
| 0 | 255 | 100 | 100 | 100 | 255 | 150 | 150 | 255 | 150 | 150 | 255 | 200 | 200 | 200 | 200 | 255 | 0 |
| 0 | 255 | 255 | 100 | 255 | 255 | 150 | 150 | 150 | 150 | 150 | 255 | 200 | 255 | 255 | 255 | 255 | 0 |
| 0 | 255 | 255 | 100 | 255 | 255 | 150 | 255 | 150 | 255 | 150 | 255 | 200 | 255 | 200 | 200 | 255 | 0 |
| 0 | 255 | 255 | 100 | 255 | 255 | 150 | 255 | 255 | 255 | 150 | 255 | 200 | 255 | 255 | 200 | 255 | 0 |
| 0 | 255 | 100 | 100 | 100 | 255 | 150 | 255 | 255 | 255 | 150 | 255 | 200 | 200 | 200 | 200 | 255 | 0 |
| 0 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

FIG. 1.3 – Une image en niveau de gris et la matrice de pixels lui correspondant.




| | | | | |
|---|---|---|---|---|
| 255 | 0 | 255 | 255 | 255 |
| 0 | 0 | 0 | 255 | 255 |
| 255 | 0 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 255 | 0 |
| 0 | 0 | 255 | 255 | 255 |
| 0 | 0 | 0 | 255 | 0 |

FIG. 1.4 – Deux images de croix ayant des matrices de pixels totalement différentes.

qu'il cherche, ce qui n'est généralement pas le cas ;
- l'utilisateur fournit une requête en langage naturel, sous forme d'une ou plusieurs phrases ou de mots-clefs. Ce mode d'interrogation est beaucoup plus naturel pour l'utilisateur, mais suppose que le SRI soit capable de mettre en relation le contenu visuel de l'image et les mots du langage naturel, autrement dit d'associer automatiquement à une image les concepts qui y sont représentés. On se situe donc ici plutôt dans le domaine de l'indexation sémantique d'images, avec utilisation de descripteurs de haut-niveau.

### 1.2.3 Le fossé sémantique

L'expression *fossé sémantique* (ou *semantic gap*) désigne la difficulté à établir une relation entre les descripteurs bas-niveau que l'on peut extraire d'une image, et les descripteurs haut-niveau que l'on souhaiterait en extraire. Il n'existe en effet pas de bijection entre ces deux types de descripteurs : certains traits visuels peuvent être communs à plusieurs objets (un extincteur et une tomate sont tous les deux rouges), de même qu'un même objet peut se présenter sous des aspects visuels différents (une Porsche, une 2CV et un Hummer sont tous des automobiles, quelle que soit leur couleur). Il semble donc difficile de se reposer uniquement sur des descripteurs visuels simples comme la couleur pour évaluer la similarité entre deux images ou déterminer si un objet précis se situe dans une image.

## 1.3 Recherche d'images par le contenu

La recherche d'images par le contenu designe les méthodes qui permettent de retrouver, à partir d'une image requête, des images au contenu visuel similaire, en se basant donc sur des descripteurs de bas-niveau uniquement. On espère qu'en se basant uniquement sur ces descripteurs simples, on pourra retrouver des images dont la sémantique sera également proche de celle de la requête. Cette tâche nécessite, avant toute chose, de définir la notion d'images similaires. Ensuite nous présenterons les descripteurs bas-niveau couramment utilisés en recherche d'images par le contenu.

### 1.3.1 Notion d'images similaires

La notion d'images similaires n'est pas définie de manière stricte. Elle est en réalité très subjective. En effet, deux utilisateurs distincts soumettant une même image requête à un système de recherche d'images peuvent avoir des attentes de résultats très différents. Supposons que la requête soit une photographie d'une 2CV devant des arbres, l'un des utilisateurs attendra peut-être uniquement des images de 2CV, indépendemment du décor de fond, alors que le second préfèrera des images de voitures, quelles qu'elles soient, mais uniquement sur fond d'arbres. Il est donc très difficile de définir formellement la notion de similarité entre images, et donc de pertinence des résultats à une requête. Or cette notion de pertinence est essentielle à l'évaluation des systèmes de recherche d'images, et doit respecter certaines propriétés (énoncées en section 1.1.6.1).

Plutôt que de d'être abordé frontalement, le problème est contourné en définissant des tâches précises à résoudre. En effet, le problème de l'évaluation des SRI se pose aussi sous l'angle de la disponibilité de données d'évaluation. Dans le cas des images, aux différents jeux de données disponibles correspondent une ou plusieurs tâches à résoudre, avec une vérité-terrain associée. Ainsi, la notion d'images similaires est définie au cas par cas, en fonction de la tâche étudiée, et non de manière globale. De plus, très souvent, la tâche étudiée n'est pas décrite explicitement dans les travaux de recherche d'images, mais uniquement

imposée de manière implicite par la nature des données employées. Voici quelques-unes des tâches classiques en recherche d'images par le contenu, et quelques corpus de données qui leur sont associés.

#### 1.3.1.1 Recherche d'objets ou de scènes identiques

Une première tâche classique en recherche d'images est la recherche de scènes (ou d'objets) identiques. Il s'agit, étant donnée une image requête d'une scène ou d'un objet donné (par exemple, la tour Eiffel), de retrouver les images contenant la même scène ou le même objet, indépendamment des changements d'échelle, de luminosité ou de point de vue. Dans ce cas, la requête contient l'unique représentation de l'objet à retrouver, le SRI doit donc être capable de différencier la scène recherchée de scènes différentes, mêmes proches, mais ne requièrt pas de capacité de généralisation lui permettant de retrouver des objets de fonction identique (*la tour de Pise* par exemple). Parmi les corpus utilisés pour ce type de tâche on peut citer les corpus Kentucky [NS06], Oxford [PCI+07] et Holidays [JDS08]. La figure 1.5 donne un exemples d'images similaires selon ce principe.

FIG. 1.5 – Images similaires au sens de la recherche de scènes identiques (corpus Kentucky).

#### 1.3.1.2 Recherche d'images catégorisées

Une autre tâche habituellement traitée en recherche d'images par le contenu est la recherche d'objets appartenant à une catégorie donnée. Dans ce cas, on considère qu'à une requête contenant un objet donné (par exemple une voiture), le SRI devra renvoyer des images contenant un ou des objets de la même catégories, même d'aspect visuel différent (des voitures, indépendamment de leur modèle ou de leur couleur). Un SRI traitant ce problème devra donc être capable d'identifier des caractéristiques discriminantes des objets considérés, mais également de généraliser suffisamment pour identifier toutes les variantes possibles de l'objet recherché. Les corpus utilisés pour cette tâche sont généralement des corpus issus du domaine de la classification d'images, problématique assez proche de celle posée ici. On peut par exemple citer les corpus Caltech-101 [FFFP07] et Caltech-256 [GHP07], proposant des images classées respectivement selon 101 et 256 catégories.

FIG. 1.6 – Images similaires au sens de la recherche d'images catégorisées (corpus Caltech-101).

Contrairement à la recherche de scènes identiques où la notion de similarité correspond à une réalité sémantique (la scène est la même ou non), certaines frontières entre catégories peuvent être plus floues, bien qu'au sens du corpus qui les proposent, ces catégories doivent être considérées comme différentes. Caltech-101, par exemple, distingue les catégories *crocodile* et *crocodile head*, alors que l'une inclue clairement l'autre.

#### 1.3.1.3 Recherche de copies

Une dernière tâche, qui peut être apparentée à la recherche d'images par le contenu, est la détection de copies. Ici on ne cherche clairement pas à identifier la contenu de l'image, mais juste à retrouver, dans un corpus donné, les images qui sont des copies, partielles ou entières, modifiées ou non, de l'image requête. Dans ce cas, le système considéré devra être très discriminant, robuste aux modifications de l'image, mais ne devra pas généraliser. Les corpus dédiés à cette tâche peuvent être générés automatiquement en appliquant diverses combinaisons de transformations (redimensionnement, occultations, compression...) à des images quelconques. Nous ne nous intéresserons pas à cette tâche dans la suite du manuscrit car elle constitue un cas à part d'indexation d'images.

Nous avons présenté ici les trois manières principales de considérer la notion de similarité entre images, qui déterminent généralement la notion de pertinence pour les systèmes de recherche d'images et leur évaluation. Cette notion de pertinence peut néanmoins aller au-delà de cette simple notion de similarité entre images. Ainsi, certains considèrent qu'un système ne doit pas uniquement fournir des résultats similaires à la requêtes, mais également faire en sorte que ces résultats soient les plus variés possibles, pour ne pas proposer à l'utilisateur uniquement des versions quasi-identiques d'une même image. Ce type de critère de pertinence est par exemple pris en compte dans les dernières campagnes d'évaluation CLEF [TDFT+09].

### 1.3.2 Description globale des images

Un descripteur global est un descripteur qui encode les caractéristiques d'une image dans son ensemble, sans délimiter de régions. Il ne tient donc pas compte du fait qu'une image contienne un fond qui n'est pas nécessairement utile de décrire, ou différents objets qu'il pourrait être judicieux de décrire séparément. Il se présente sous la forme d'une vecteur numérique, dont la dimension et le contenu dépendent du ou des types de caractéristiques visuelles représentées. Ces vecteurs numériques présentent l'avantage d'être facilement manipulables par l'outil informatique et d'offrir une large palette d'outils mathématiques pour

estimer leur proximité, donc la similarité entre les images qu'ils représentent. On distingue généralement trois types de descripteurs globaux, qui peuvent être combinés entre eux pour obtenir un descripteur plus complet.

#### 1.3.2.1 Descripteurs de couleurs

Les descripteurs de couleurs permettent, comme leur nom l'indique, de décrire les couleurs constituant l'image. Avant de chercher à décrire ces couleurs, il faut s'interroger sur la manière dont les couleurs sont représentées par la machine. Une couleur est représentée sous la forme d'un vecteur dans un *espace de couleurs*. Il existe plusieurs espaces de couleurs, donc plusieurs manières de décrire les couleurs. On distingue deux types d'espaces de couleurs : les espaces perceptuels et les espaces non-perceptuels. Les espaces perceptuels sont construits de sorte à ce que, si l'on compare deux couleurs, en calculant une distance entre leur vecteurs, cette distance soit proportionelle à la différence entre les couleurs telle qu'elle est perçue par l'œil humain normal. Les espaces perceptuels les plus utilisés sont HSV et Lab. L'espace non-perceptuel le plus couramment utilisé est RGB, pour *Red, Green, Blue* (ou RVB, Rouge, Vert, Bleu), espace classiquement utilisé par le matériel informatique. Bien que les espaces perceptuels paraissent *a priori* plus adaptés à la recherche d'images par le contenu, l'expérience tend à montrer qu'ils n'améliorent pas significativement les performances des SRI, tout en nécessitant des conversions qui peuvent s'avérer coûteuses en temps de calcul (Lim *et al.* rapportent qu'à vitesse de calcul égale, HSV est l'espace offrant la meilleure précision, cependant le gain reste très faible [LL03]).

Une fois l'espace de couleurs choisi, il faut déterminer une manière de décrire les couleurs apparaissant dans les images. Il peut s'agir de décrire la (ou les) couleur(s) dominante(s) de l'image (moyenne de couleurs et moments) ou la répartition des différentes couleurs (histogramme de couleurs).

**Moyenne et moments** La moyenne et les moments de couleurs fournissent une représentation très compacte des couleurs, très rapide à calculer, mais aussi peu discriminante. Les moments d'ordre 2 représentent la variance d'une des composantes de l'espace de couleurs, ou la covariance de deux de ces composantes. Les moments d'ordres supérieurs permettent de définir d'autres grandeurs comme l'asymétrie ou la kurtosis. Les moyennes et moments constituent les descripteurs de couleurs les plus élémentaires et sont donc peu performants pour la recherche d'images [MZ98].

**Histogrammes de couleurs** Un histogramme de couleurs est obtenu en comptant, pour chaque couleur, le nombre de pixels de cette couleur contenus dans l'image. Le nombre de couleurs possibles dans une image pouvant être très élevé (jusqu'à $2^{24}$, soit plusieurs milliards), l'espace des couleurs est d'abord quantifié en $n$ couleurs possibles (on parle de *bins*), typiquement en divisant chaque composante en $\frac{n}{3}$ parties. L'histogramme est ensuite normalisé par la taille de l'image. L'histogramme ne représente alors plus la quantité de chaque couleur dans l'image, mais la proportion de chaque couleur. Il devient alors robuste aux changements d'échelles. L'équation donnant l'histogramme $H$ de couleurs d'une image $I$ de taille $M \times N$ est :

$$H(c) = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \delta(I(i,j), c), \ \forall c \in \mathcal{C}_n \tag{1.1}$$

où $\mathcal{C}_n$ désigne l'espace de couleurs quantifié en $n$ bins, $I(i,j)$ la couleur du pixel de coordonnées $(i,j)$ et $\delta$ est le symbole de Kronecker défini ainsi :

$$\delta(x,y) = \begin{cases} 1 \text{ si } x = y \\ 0 \text{ sinon} \end{cases}$$

L'histogramme de couleurs donne donc une description globale de l'image, où toutes les parties de celle-ci ont la même importance. Certaines améliorations permettent cependant d'ajouter de l'information locale à l'histogramme de couleurs. Les histogrammes pondérés (équation 1.2) attribuent à chaque pixel $(i,j)$ un poids $w(i,j)$ contrôlant sa contribution à l'histogramme de couleurs. Les poids peuvent par exemple correspondre au laplacien de l'image, ce qui donne un poids important aux pixels se situant sur des contours, et diminue la contribution des pixels situés dans des zones uniformes. Une autre amélioration possible est de calculer des histogrammes locaux à différentes parties de l'image, puis de fusionner ces histogrammes pour obtenir un unique descripteur. Il existe plusieurs stratégies pour effectuer cette fusion. Une comparaison des différents types d'histogrammes de couleurs est disponible dans [BBV02].

$$H(c) = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} w(i,j).\delta(I(i,j),c), \ \forall c \in \mathcal{C}_n \tag{1.2}$$

**Autres descripteurs de couleurs** Nous pouvons citer deux autres descripteurs de couleurs pour les SRI. Le *Color Coherence Vector* (CCV) est un histogramme double, dans lequel on sépare les pixels présents dans des zones uniformes de l'image (dits *pixels cohérents*) des pixels situés dans des zones non-uniformes (dits *non-cohérents*). Les CCV offrent de meilleures performances que les histogrammes de couleurs sous certaines conditions (images majoritairement uniformes ou majoritairement texturées) [MZ98]. Les corrélogrammes de couleurs décrivent la répartition spatiale des couleurs en calculant les corrélations entre pixels des différentes couleurs à des distances données. Ce descripteur pouvant être coûteux à calculer, une version réduite ne prenant en compte que les corrélations entre pixels de même couleur, appelée auto-corrélogramme, a également été proposée [HKM+97]. Ce descripteur offre des performances comparables aux autres descripteurs de couleurs, histogrammes et CCV [MZ98].

#### 1.3.2.2 Descripteurs de texture

La notion de texture n'est pas définie de manière formelle. Elle désigne un certain nombre de propriétés d'homogénéité de l'image remarquables localement et qui ne sont pas uniquement liées à la couleur ou à la luminosité [SC96]. Caractériser la texture d'une image équivaut à caractériser la présence ou non de certains motifs répétés dans l'image (cercles, rayures...), ainsi que leur propriétés (taille, orientation...). On distingue couramment trois types de descripteurs de texture : les descripteurs perceptuels, les descripteurs statistiques et les filtres.

**Descriptions perceptuelles** Les descripteurs perceptuels cherchent à exprimer les propriétés des textures de la même manière que le ferait un humain. La description perceptuelle des textures la plus courante utilise les caractéristiques de Tamura [TMY78]. Ces caractéristiques ont été retenues à la suite d'expériences psychologiques sur la perception humaine des textures. Il y en a six : l'échelle, le constraste, la directionnalité, la tendance, la régularité et la rugosité. Ce type de caractéristiques peut sembler intéressant pour comparer

le contenu visuel des images car il correspond directement à la manière dont l'humain les perçoit. Cependant, les méthodes mises en œuvre pour calculer automatiquement ces caractéristiques n'ont donné que des résultats limités pour la recherche d'images dans des données réelles [HR04].

**Descriptions statistiques** Les descripteurs statistiques de texture, proposés par Haralick [Har79] se basent sur les cooccurrences entre les valeurs des pixels des images réduites à des niveaux de gris. À partir de ces cooccurrences, il est possible d'extraire différentes valeurs caractéristiques, telles que l'énergie, l'entropie, le constraste ou l'homogénéité, qui constituent un vecteur descripteur de la texture d'une image. Bien que montrant de bonnes performances pour la catégorisation de données artificielles (images uniquement composées d'une texture), ces descripteurs ont montré un intérêt plus limité dans un cadre de recherche d'images réelles [HR04].

**Filtres** Les filtres appliqués à une image permettent de faire ressortir certaines particularités comme la présence de motifs, caractéristiques des textures, à certaines orientations et échelles données. Typiquement, si un motif correspondant à l'échelle et à l'orientation du filtre considéré existe dans l'image, le filtre, à chaque pixel, produira des valeurs de réponse élevées pour les pixels correspondant à ce motif. Un descripteur par filtre peut ainsi être calculé de la manière suivante :

1. choisir une série de filtres à considérer (type, orientation, échelle) ;
2. pour chaque filtre, calculer la moyenne et l'écart-type de la réponse du filtre à chaque pixel de l'image. Ces deux valeurs permettent de décrire de manière compacte la présence de motifs correspondant au filtre utilisé (moyenne), et la régularité de cette présence (écart-type) ;
3. construire un vecteur descripteur en concaténant les moyennes et écarts-types obtenus pour chaque filtre.

On obtient ainsi un descripteur compact (vecteur de taille 2 x le nombre d'orientations x le nombre d'échelles) reflétant la présence régulière ou non de certains motifs dans l'image. Parmi les différents types de filtres existants, les filtres de Gabor sont les plus utilisés et les plus performants. Une discussion sur les différents paramètres des filtres de Gabor a été effectuée par Chen *et al.* [CLZ04]. Une comparaison du descripteur basé sur l'énergie décrit ici, et d'autres descripteurs dérivés des filtres de Gabor a été proposée par Grigorescu *et al.* [GPK02]. Le comparatif des différents descripteurs de textures (caractéristiques de Tamura, cooccurrences, filtres de Gabor) sur des données réelles réalisé par Howarth et Rüger conclue que les filtres de Gabor sont les descripteurs les plus efficaces pour la recherche d'images par le contenu [HR04].

#### 1.3.2.3 Descripteurs de forme

Commes les descripteurs de texture, les descripteurs de forme s'attachent à représenter des structures remarquables dans une image, à la différence que celles-ci ne sont pas nécessairement répétées de manière homogène dans tout ou partie de l'image. En particulier, les descripteurs de forme cherchent à caractériser la structure globale de l'image, en se basant sur la représentation des principaux contours qui y apparaissent. Ainsi, pour prendre un exemple trivial, un descripteur de forme représentera l'image d'un rectangle comme étant une image contenant deux arêtes horizontales et deux arêtes verticales. Nous décrivons ici deux descripteurs de forme intéressants : les histogrammes de contours, tels qu'ils ont été

intégrés dans la norme MPEG-7 [MOVY01], et le descripteur GIST, qui a montré récemment de bonnes performances pour la recherche d'images, en termes de vitesse comme de pertinence [DJS+09].

**Histogrammes de contours** Les histogrammes de contours décrivent la répartition locale des contours dans l'image. L'image est d'abord divisée en une grille régulière. Puis, pour chaque région ainsi obtenue, différents types de contours sont extraits par l'application de filtres aux pixels de la région ; 5 filtres différents sont utilisés pour en extraire 5 types de contours (verticaux, horizontaux, diagonaux selon les deux orientations possibles et non-directionnels). Les occurrences de chaque type de contours sont ensuite comptées pour générer un histogramme par région. Enfin, les histogrammes de chaque région sont concaténés en un descripteur global. Le descripteur global obtenu intègre donc des informations géométriques sur l'image, chaque sous-partie du descripteur correspondant à une région précise de l'image.

**Descripteur GIST** Le descripteur GIST a été proposé par Oliva et Torralba [OT01], qui basent leurs travaux sur la manière dont la vision humaine perçoit la structure générale d'une scène. Il apparaît dans leurs expériences que l'aspect général d'une image est perçu de manière assez grossière, indépendamment des nombreux détails qui peuvent y apparaître. Ainsi, le fait qu'une image soit particulièrement floue empêche d'en percevoir les détails, mais permet néanmoins de comprendre la structure globale de l'image. Ils proposent donc de calculer un descripteur sur des images réduites en imagettes carrées, d'une taille comprise entre 32x32 pixels et 128x128 pixels. Les images réduites sont ensuite divisées en une grille régulière de 4 régions de hauteur et 4 régions de largeur. Enfin, un descripteur est calculé pour chacune des 16 régions obtenues. Ce descripteur est basé sur des histogrammes d'orientation de gradients, également très utilisés pour la description locale des images (voir section 1.3.3.2), qui permettent de capturer de manière compacte mais néanmoins précise la forme globale d'une région d'image en caractérisant l'orientation des différents contours qui y apparaissent. Les descripteurs des différentes régions sont ensuite concaténés pour obtenir un vecteur décrivant l'image dans sa globalité. Ici aussi, cette étape implique la présence d'informations géométriques dans le descripteur, puisque chaque sous-partie du descripteur correspondra à une région donnée de l'image. Les descripteurs GIST ont été évalués de manière approfondie dans des tâches de recherche de scènes identiques et de détection de copies par Douze *et al.*, qui les comparent à des approches locales de description des images [DJS+09].

#### 1.3.2.4 Comparaison d'images par descripteurs globaux

Une fois que les images ont été décrites sous forme de vecteurs numériques, il reste à comparer les vecteurs entre eux pour établir si les images sont similaires ou non. On dispose pour cela de différentes fonctions qui permettent d'associer à deux vecteurs un score reflétant leur similarité. Parmi ces fonctions, on distingue parfois les *mesures de similarité* des *mesures de dissimilarité*. Les mesures de similarité regroupent les fonctions qui fournissent un score qui est d'autant plus grand que les vecteurs sont similaires (par exemple, le cosinus entre les vecteurs) ; à l'inverse, une mesure de dissimilarité fournira un score d'autant plus grand que les vecteurs sont différents (c'est le cas, par exemple, de la distance euclidienne). Dans la pratique, ces mesures répondent à un même objectif : obtenir des scores reflétant la similarité des documents à la requête, pour fournir à l'utilisateur une liste de documents-résultats classés et/ou filtrés. Certaines mesures de similarité et dissimilarité

sont de plus équivalentes en termes d'ordre induit, c'est-à-dire qu'elles produisent des listes ordonnées de résultats identiques pour toute requête (Lesot *et al.* proposent de telles équivalences pour certaines distances courantes [LRD09]). Dans la suite de ce manuscrit, nous parlerons uniquement de mesures de similarité, qu'il s'agisse de mesures de similarité ou de dissimilarité.

Les mesures de similarité utilisées en recherche d'images sont pour la plupart des distances au sens mathématique du terme, ces dernières offrant des propriétés intéressantes pour ce cadre applicatif. Rappelons tout d'abord la définition d'une distance :

**Définition 1.1** *L'application $d : E \times E \rightarrow \mathbb{R}$ est une distance sur $E$ si et seulement si :*

1. $\forall x, y \in E, d(x, y) = d(y, x)$ *(symétrie)*
2. $\forall x, y \in E, d(x, y) = 0 \Leftrightarrow x = y$ *(séparation)*
3. $\forall x, y, z \in E, d(x, z) \leq d(x, y) + d(y, z)$ *(inégalité triangulaire)*

La propriété de symétrie garantit la cohérence des résultats en assurant que si le descripteur d'une image $d$ se situe à une distance donnée du descripteur de l'image requête $q$, cette distance sera identique si $d$ est utilisé comme requête. La notion de similarité entre deux images ne dépend ainsi pas du choix de l'une ou l'autre comme requête. La propriété de séparation garantit que la distance entre deux descripteurs identiques sera nulle. La distance entre deux images identiques sera donc également nulle, en revanche il n'y a pas de garantie que deux images différentes soient séparées d'une distance non-nulle, car il est possible d'extraire des descripteurs identiques à partir d'images différentes. La propriété d'inégalité triangulaire peut quant à elle être exploitée pour accélérer le processus de recherche : il est possible de ne calculer qu'une sous partie des distances nécessaires à une recherche exhaustive dans la base des descripteurs en éliminant d'emblée les descripteurs qui sont assurés, par inégalité triangulaire, de n'être pas suffisamment proches de la requête [BFM+96].

Dans la suite de cette section, nous présentons quelques mesures de similarité classiques utilisées en recherche d'information et, plus précisemment, en recherche d'images. Les descripteurs d'images étant des vecteurs numériques, toutes ces distances seront des applications du type $\mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}$, $N$ étant la dimension des vecteurs descripteurs. Une présentation plus complète des mesures de similarité entre vecteurs numériques peut être trouvée dans [LRB09].

**Distances de Minkowski** Les distances de Minkowski (équation 1.3) sont des distances définies sur l'espace vectoriel normé $\mathbb{R}^N$. Parmi celles-ci seules trois sont utilisées couramment : la distance $L_1$ (ou distance de Manhattan), la distance $L_2$ (ou distance euclidienne) et la distance $L_\infty$ (ou distance de Tchebychev).

$$d_{L_k}(x, y) = \left( \sum_{i=1}^{N} (x_i - y_i)^k \right)^{\frac{1}{k}}, \ \forall x, y \in \mathbb{R}^N \text{ et } k \geq 1 \tag{1.3}$$

**Distances fractionnelles** Les distances fractionnelles sont des dérivés des distances de Minkowski dans les cas ou le paramètre $k$ est inférieur à 1. Ce ne sont pas des distances au sens mathématique car elles ne respectent pas l'inégalité triangulaire. Néanmoins, elles offrent des propriétés intéressantes dans les espaces de grandes dimensions [AHK01], et donc pour la recherche d'images [HR05] :

- les distances fractionnelles limitent l'amplitude des distances locales[2]. Cela permet de mieux prendre en compte la contribution de chaque dimension à la distance globale et ainsi de limiter les effets de la malédiction de la dimension ;
- les distances fractionnelles gèrent mieux la présence de bruit dans les données. Néanmoins, au delà d'une certaine quantité de bruit dans les données, toutes les distances tendent à être équivalentes, indépendamment de la valeur de $k$.

**Similarité du cosinus** La similarité du cosinus (équation 1.4) est surtout utilisée en RI textuelle. Elle consiste à considérer l'angle formé entre deux vecteurs comme une distance les séparant. Elle est donc intrinsèquement insensible à la norme des vecteurs (*cf.* le dénominateur de l'équation 1.4), ce qui peut lui conférer, lorsqu'elle est utilisée avec certains descripteurs, des propriétés d'invariance. Par exemple, la similarité du cosinus utilisée pour comparer des histogrammes de couleurs est invariante aux changements d'illumination. De plus, lorsque les vecteurs sont normés avec une norme 2, la similarité du cosinus est strictement équivalente à la distance $L_2$ en termes d'ordre induit (voir équation 1.5).

$$d_{cos}(x,y) = \frac{\sum_{i=1}^{N} x_i.y_i}{||x||_2||y||_2} \tag{1.4}$$

$$d_{L_2}(x,y)^2 = 2 \cdot (1 - d_{cos}(x,y)) \tag{1.5}$$

**Distances statistiques** La distance la plus courante issue des statistiques est la distance du $\chi^2$, qui permet de mesurer l'adéquation entre des données empiriques et une loi de probabilité. Dans le cas de la recherche d'images, la distance du $\chi^2$ entre deux descripteurs $x$ et $y$ se note :

$$d_{\chi^2}(x,y) = \sum_{i=1}^{N} \frac{(x_i - m_i)^2}{m_i} \text{ où } m_i = \frac{x_i + y_i}{2} \tag{1.6}$$

Elle représente donc l'écart entre les données observées (vecteur requête $x$) et la distribution attendue (moyenne des deux vecteurs $x$ et $y$) [LSR+08].

**Distances issues de la théorie de l'information** Les distances issues de la théorie de l'information ont initialement pour objectif de mesurer la divergence entre des distributions de probabilités. Elles sont néanmoins applicables à des vecteurs numériques quelconques à condition que ceux-ci respectent la condition suivante :$\sum_{i=1}^{N} x_i = 1$. Cette condition peut être respectée en normalisant les vecteurs avec la norme 1. La distance de ce type la plus utilisée n'en est pas une au sens strict car elle ne respecte pas la propriété de symétrie : il s'agit de la divergence de Kullback-Liebler (équation 1.7). Il est néanmoins possible d'en obtenir une distance au sens strict, la distance de Kullback-Liebler (équation 1.8). Parmi les distances issues de la théorie de l'information, on peut également citer la distance de Jensen-Shannon (équation 1.9), aussi connue sous le nom de divergence de Jeffrey [RPTB01].

$$\text{div}_{KL}(x,y) = \sum_{i=1}^{N} x_i \log(\frac{x_i}{y_i}) \tag{1.7}$$

$$d_{KL}(x,y) = \frac{1}{2}(\text{div}_{KL}(x,y) + \text{div}_{KL}(y,x)) \tag{1.8}$$

[2]Nous appellons distance locale une distance calculée sur une seule dimension des vecteurs.

$$d_{JS}(x,y) = \frac{1}{2}\mathrm{div}_{KL}(x, \frac{x+y}{2}) + \frac{1}{2}\mathrm{div}_{KL}(\frac{x+y}{2}, y) \tag{1.9}$$

**Performances des distances** Il existe plusieurs travaux visant à déterminer la ou les meilleures distances pour la recherche d'images par le contenu [RPTB01, KCB03, Zha03, HR05, CC05, LSR+08]. Cependant, la majorité d'entre eux repose soit sur des données complètement artificielles (la collection de textures Brodatz [Bro66]), soit sur des données issues des catalogues d'images Corel, qui présentent de nombreux biais [MMMP02].

#### 1.3.2.5 Bilan des approches globales

Les descripteurs globaux d'images présentent l'avantage d'être rapides à calculer et d'offrir une représentation compacte des images (en général, ce sont des vecteus de quelques centaines de dimensions). En revanche, ils décrivent les images de manière très rudimentaire, ce qui ne permet pas de les différencier très efficacement. En particulier, toutes les informations locales sur les images (uniformité de certaines régions, distribution des couleurs dans l'image) sont perdues, ce qui limite beaucoup leur pouvoir d'expression. Ainsi, il n'est pas rare de trouver deux images dont les contenus semblent intuitivement différents mais dont les descripteurs globaux sont identiques. Ce manque d'expressivité pousse à se diriger vers des descriptions d'images qui permettent de capter plus d'informations locales, plus spécifiques.

### 1.3.3 Description locale des images

Pour pallier les limitations des descriptions globales, il est possible d'adopter une description locale des images. Plutôt que d'être décrite sous forme d'un unique vecteur numérique, chaque image est d'abord divisée en un ensemble de régions, puis chaque région est représentée par un descripteur similaire aux descripteurs globaux. L'avantage de ce type de description est qu'elle prend en compte des aspects locaux du contenu des images, comme la présence de plusieurs objets par exemple, ou de parties d'objets. Représenter les images sous formes d'ensembles de descripteurs de régions plutôt que d'un descripteur global unique pose deux problèmes :

- quel découpage de l'image en régions adopter ? (section 1.3.3.1)
- comment comparer des ensembles de descripteurs plutôt que de simples descripteurs ? (section 1.3.3.3)

#### 1.3.3.1 Diviser les images en régions

Utiliser une description locale des images nécessite en premier lieu de découper l'image en régions ou d'en extraire les régions les plus intéressantes. Un tel découpage peut être réalisé *a priori*, c'est-à-dire de manière identique pour toutes les images, ou bien être réalisé en fonction du contenu de l'image, pour éviter de séparer des régions d'un même objet, ou de regrouper des objets différents sous une même région. Voici les méthodes les plus courantes pour obtenir des régions à partir d'une image.

**Découpage en blocs** Le découpage en blocs est la manière la plus simple de séparer une image en régions, mais également la plus frustre. Il s'agit simplement de découper l'image selon une grille dont la taille et la forme est fixée à l'avance. On peut néanmoins distinguer deux types d'approches :

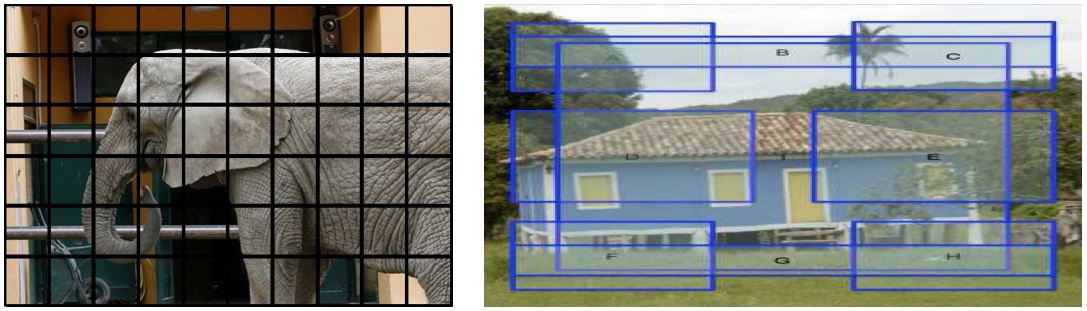



FIG. 1.7 – Deux façons de découper une image en blocs (l'image de droite est empruntée à [TDFT+08]).

FIG. 1.8 – Une image et une segmentation possible de celle-ci (images empruntées à [BF01]).

- le découpage en carrés de petite taille ($<$ 100 pixels de côté) : il s'agit ici d'isoler de petits éléments d'image, pour ensuite en identifier le contenu, par accumulation de ces éléments. Pour être plus robustes aux changements d'échelle, certains systèmes utilisent plusieurs découpages, en carrés de tailles différentes. Pour pallier au fait que le découpage est arbitraire, certains systèmes extraient des carrés se chevauchant ;
- le découpage en zones exploitant la géométrie des images : dans ce cas l'image est divisée en régions en fonction de zones que l'on peut identifier généralement dans les images, par exemple le haut, le centre et le bas de l'image. L'objectif est d'utiliser des informations *a priori* sur la prise de vue des images pour faciliter l'identification de leur contenu. On considère ainsi que le zone centrale de l'image contient l'essentiel de l'information, ou encore que le ciel a de très fortes chances de se trouver en haut de l'image plutôt qu'en bas.

**Segmentation** La segmentation d'une image consiste à extraire de l'image des régions correspondant aux objets effectivement présents dans l'image, ou à des parties de ces objets. Contrairement au découpage en blocs, le choix des régions n'est pas réalisé *a priori* mais est guidé par le contenu visuel de l'image. La figure 1.8 montre un exemple d'une image et d'une segmentation possible de cette image. Le problème de la segmentation d'images a commencé à être traité dans les années 1980 et il existe aujourd'hui des dizaines d'algorithmes automatiques ou semi-automatiques de segmentation, néanmoins aucun d'entre eux ne fournit de résultats optimaux (au regard des régions souhaitées, définies manuellement). On pourra se référer à [LM01] et [FMnR+02] pour un aperçu des principaux algorithmes. Les algorithmes de segmentation les plus utilisés pour la recherche d'images sont *blobworld* [CTB+99], *Normalized Cuts* [SM00] et *Mean Shift* [CM02]. Parmi ceux-ci, *Normalized Cuts* permet d'obtenir les meilleurs résultats sur une tâche d'annotation d'images [BDG+03].

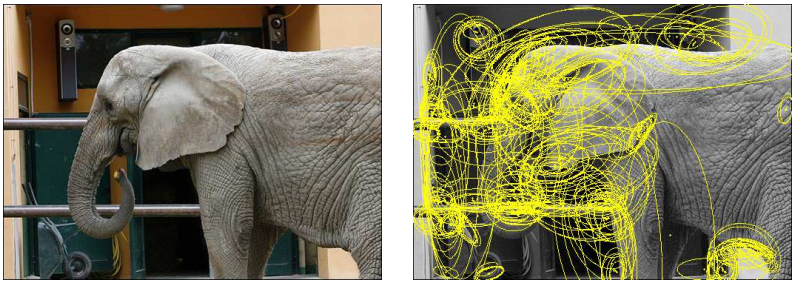


FIG. 1.9 – Détection de régions d'intérêt sur une image (détecteur *Hessian-Affine*).

**Régions d'intérêt** Les régions d'intérêt sont des régions d'image dont on considère qu'elles portent une information sur le contenu de l'image. Il peut s'agir de régions où se situent des changements (angles par exemple) ou de régions particulièrement uniformes, selon le type d'information recherché. Les détecteurs de régions d'intérêt détectent généralement des zones elliptiques dans l'image, des cercles, ou des régions qui seront, par approximation, ramenées à une forme elliptique. On attend de ces détecteurs qu'ils extraient des régions répétables, c'est-à-dire qu'ils extraient les mêmes régions à partir d'objets similaires. La détection doit donc être invariante aux transformations géométriques (rotations, changements d'échelle, transformations affines) et aux changements de luminosité. La figure 1.9 donne un exemple de détection de points d'intérêt dans une image à l'aide du détecteur *Hessian-Affine*.

Parmi les détecteurs qui détectent les zones "changeantes", on peut citer les détecteurs *Harris-Affine*, *Hessian-Affine* ou *Difference of Gaussian* (*DoG*). Le détecteur *MSER* est quant à lui le plus couramment utilisé pour extraire des régions uniformes. On peut trouver un comparatif des principaux détecteurs de régions d'intérêt dans [MTS+05].

#### 1.3.3.2 Descripteurs de régions d'image

Tous les descripteurs globaux d'images peuvent évidemment être utilisés pour décrire des régions d'image, cependant, certains seront plus pertinents pour la description de régions qu'ils ne l'étaient pour décrire des images complètes, et inversement. De plus, certains descripteurs de forme sont spécifiques au cas des régions d'image. Voici un aperçu des principales différences en fonction des types de descripteurs.

**Couleur** Les histogrammes de couleurs sont adaptés aux découpages en grandes régions qui exploitent la géométrie des images, parce que ces régions contiennent encore des informations visuelles assez complexes [TDFT+08]. En revanche, dans le cas d'un découpage en grille fine, ce genre de descripteur n'a que peu d'intérêt, car l'information visuelle est très locale et a peu de chance de contenir beaucoup de couleurs différentes. Des descripteurs plus compacts intégrant la couleur moyenne et les écarts-types des différents canaux sont alors plus adaptés. Il en va de même pour les régions segmentées qui ne contiennent en principe qu'un objet ou qu'une partie d'objet, limitant ainsi les différentes couleurs possibles [BDF+03b]. Ceci est d'autant plus vrai que la plupart des méthodes de segmentation s'appuient sur les informations de couleurs pour délimiter les régions, limitant ainsi de fait la variabilité des couleurs au sein de chaque région. Les autres descripteurs de couleurs, comme les corrélogrammes, sont utilisés de manière marginale pour décrire des régions d'image.

**Texture** Les descripteurs de texture utilisés pour les descriptions locales sont les mêmes que pour les descriptions globales des images. En particulier, les méthodes basées sur les filtres comme les filtres de Gabor sont très prisées [DI07], les descripteurs de type cooccurrences ou caractéristiques de Tamura étant beaucoup plus rares dans ce contexte.

**Forme** Deux différences majeures existent, l'une concernant les régions issues d'une segmentation, l'autre les régions d'intérêt. Les régions segmentées n'ont pas toutes la même forme et il peut donc être intéressant de décrire cette forme pour caractériser les objets qu'elles représentent. Plusieurs mesures peuvent être utilisées pour caractériser de telles formes, comme, par exemple, la proportion de l'image couverte par la région, le rapport entre l'aire et le carré du périmètre ou le moment d'inertie par rapport au centre [BDF+03b]. Les régions d'intérêt, quant à elles, ont toujours été historiquement décrites par des descripteurs robustes, de même que les détecteurs qui leurs sont associés, aux transformations géométriques (rotations, transformations affines, échelle) et de luminosité. Comme la détection de régions d'intérêt, le calcul de tels descripteurs s'effectue sur l'image réduite à l'information de luminance (niveaux de gris). Les invariants différentiels [SM97] ont été parmi les premiers descripteurs de ce type, suivi par les SIFT (*Scale-Invariant Feature Transform*) [Low99], puis les SURF (*Speeded-Up Robust Features*) [BTVG08]. Parmi ceux-ci, les SIFT et leurs variantes sont clairement les plus répandus aujourd'hui, en raison de leurs bonnes propriétés [MS05]. Ils sont basés sur un histogramme d'orientations des gradients, quantifiées selon huit directions et calculées dans seize positions au sein de la région d'intérêt, fournissant ainsi un descripteur de 128 dimensions. Le descripteur SIFT ressemble donc au descripteur de forme GIST, ce dernier étant en réalité une adaptation du descripteur SIFT à la description d'images globales. Certaines variantes de SIFT ont été proposées, notamment PCA-SIFT [KS04], qui propose un descripteur plus compact par application d'une Analyse en Composantes Principales (PCA, *Principal Component Analysis*) à un ensemble de descripteurs et conservation des axes les plus significatifs, ou encore des variantes couleurs de SIFT, qui intègrent des informations de couleurs en appliquant la méthode de SIFT aux canaux de couleur de l'image (HSV par exemple [BZM06]) plutôt qu'à la luminance seule. Ces descripteurs de régions d'intérêt sont aussi parfois calculés sur des images divisées en grille, éventuellement en grilles de différentes échelles : on parle alors généralement d'échantillonnage dense de régions d'intérêt [JT05].

#### 1.3.3.3 Comparaison d'images par régions

Comparer des images par régions n'est pas aussi simple que de comparer des descripteurs globaux car le nombre de régions décrivant les images n'est pas nécessairement fixe. Il est donc nécessaire de mettre en œuvre des techniques qui permettent de comparer les régions d'image entre elles, puis d'obtenir une mesure de similarité globale en fonction des similarités entre régions. On peut distinguer deux approches, la comparaison exhaustive et la quantification des régions, plus le cas particulier où le nombre de régions par image est constant quelle que soit l'image.

**Cas particulier des découpages réguliers** Les découpages réguliers (en blocs) produisent un nombre identique de régions pour toutes les images. Une fois chaque région décrite par un vecteur, il est possible d'obtenir, en concaténant ces vecteurs les uns aux autres, un vecteur descripteur similaire à une description globale des images. Les images peuvent alors être comparées en utilisant une des mesures de similarités disponibles pour la comparaison de descripteurs globaux. On remarquera qu'avec ce schéma de mise en cor-

respondance, chaque région d'une image n'est comparée qu'à son homologue dans l'autre image. Il existe donc une forte contrainte sur la position géométrique des régions qui rend ce type d'approches sensible aux transformations géométriques comme les rotations.

**Comparaison exhaustive** Une méthode pour comparer des images par régions consiste à définir une mesure de similarité pour des ensembles de descripteurs plutôt que pour des descripteurs simples. La solution la plus courante consiste à mettre en place un système de vote [SM97] qui consiste à ordonner les images résultats en fonction du nombre de descripteurs qu'elles partagent avec l'image requête. La procédure est la suivante :

1. soient $n$ images $I_1, I_2, \ldots, I_n$, chaque image $I_i$ étant représentée par $k_i$ descripteurs $\{V_1^i, V_2^i, \ldots, V_{k_i}^i\}$ ;
2. soient une distance entre vecteurs $d(V_j^i, V_{j'}^{i'})$ et un seuil $d_{max}$ sur cette distance ;
3. soit une image requête $I_{req}$ dont on extrait $k_{req}$ descripteurs $\{V_1^{req}, V_2^{req}, \ldots, V_{k_{req}}^{req}\}$ :
   (a) $\forall j \in [1; k_{req}],\ \forall i \in [1; n],\ \forall j' \in [1; k_i]$, :
   si $d(V_j^{req}, V_{j'}^i) < d_{max}$, alors ajouter une voix à l'image $I_i$ ;
   (b) ordonner les résultats par ordre décroissant du nombre de voix.

Deux types d'améliorations peuvent être apportées à ce système :
- des méthodes de recherche accélérée des descripteurs dans la base : la recherche d'un descripteur local parmi l'ensemble des descripteurs peut être très longue lorsque l'on indexe un grand nombre d'images, il peut donc être utile d'accélérer cette recherche grâce à des structures de données et méthodes de recherche adaptées, dans un cadre de recherche exacte (*kd-tree* ou *VA-file*, par exemple [AG01]) ou de recherche approximative (*Best-bin-first* [Low04], *NV-tree* [LAJA08]...). De plus, il a été montré que lorsque la dimension des descripteurs est élevée, les structures de données classiques ne permettent pas d'améliorer la vitesse de recherche par rapport à une recherche exhaustive, il est donc nécessaire d'avoir recours à des techniques de recherche approximative ;
- des méthodes d'élimination des *outliers* : les descripteurs locaux représentant une image peuvent contenir de nombreux descripteurs atypiques (*outliers*), que l'on ne retrouve habituellement pas dans les objets contenus dans cette image (descripteurs issus du fond de l'image par exemple). Il peut être utile d'éliminer automatiquement de tels descripteurs. Lowe propose par exemple d'éliminer le descripteur le plus proche d'un descripteur requête si sa distance à la requête n'est pas significativement inférieure à celle des autres descripteurs [Low04]. D'autres techniques plus perfectionnées ont également été proposées, comme l'utilisation d'une transformée de Hough [Low04] ou de l'algorithme RANSAC [WJY08].

**Quantification des régions** Une autre méthode consiste à effectuer une quantification des descripteurs locaux de manière à obtenir un ensemble restreint de descripteurs locaux possibles, généralement appelé *vocabulaire visuel* (*visual vocabulary*, *visual codebook*). Chaque descripteur dans l'espace quantifié, autrement dit chaque entrée du vocabulaire visuel, est appelé mot visuel. Une image peut alors être décrite par l'ensemble des mots visuels qui la compose plutôt que par chaque descripteur local intégral, ou encore par un vecteur des occurrences de chaque mot visuel de l'image. On obtient ainsi un descripteur unique pour chaque image, à la manière des descripteurs globaux, descripteur construit par accumulation d'éléments locaux de l'image. Cette approche permet de combiner à la

fois la forte expressivité des descripteurs locaux et la faible complexité des descripteurs globaux pour la phase de recherche. On leur prête également un pouvoir de généralisation plus élevé que les approches par comparaison d'ensembles de descripteurs locaux en raison de la phase de quantification qui rendrait les descripteurs plus robustes aux faibles variations. Ce type d'approche est particulièrement prisé avec les descripteurs locaux de type SIFT, en conjonction avec un détecteur de régions d'intérêt ou un découpage de l'image en grille régulière, depuis 2003 et les travaux de Sivic et Zisserman [SZ03], puis de Csurka *et al.* [CDF+04]. Néanmoins, on peut trouver plusieurs travaux antérieurs et similaires à la nature des descripteurs près, tels que ceux de Mori *et al.* [MTO99] (image en grille et descripteurs de couleurs et texture), Varma et Zisserman [VZ05] (basés sur les *textons*, mots visuels construits à base d'un descripteur de texture) ou encore de Barnard *et al.* [BDF+03b] (segmentation et descripteurs de couleurs, texture et forme). Chen *et al.* ont montré que l'approche de Sivic et Zisserman est plus efficace que celle de Barnard *et al.* en raison de la très grande expressivité des descripteurs locaux utilisés [CHS09]. Enfin, on peut remarquer de fortes similitudes entre ce type de description et certains descripteurs globaux comme les histogrammes de couleurs ou les histogrammes d'orientations d'arêtes, la description par quantification de régions n'étant finalement qu'un histogramme de descripteurs locaux. La différence se situe uniquement dans la nature des descripteurs locaux qui servent à construire l'histogramme, beaucoup plus expressifs que la couleur d'un pixel isolé par exemple.

#### 1.3.3.4 Bilan des approches locales

Les approches locales, beaucoup plus expressives que les approches globales, permettent généralement d'obtenir de meilleures performances. En particulier, les approches par quantification de régions connaissent un grand succès ces dernières années, notamment celles utilisant des descripteurs locaux de type SIFT, car ces derniers capturent des informations visuelles très discriminantes. Comme nous le verrons au chapitre 2, ce type d'approche permet de plus d'obtenir une description des images qui présente beaucoup de points communs avec la description des textes habituellement utilisée en recherche d'information textuelle. C'est donc ce type de représentation que nous utiliserons pour le premier axe de nos travaux, l'utilisation de méthodes de TAL pour la recherche d'images par le contenu.

## 1.4 Recherche sémantique d'images

Nous décrivons ici les travaux en lien avec le second axe de ce mémoire, l'utilisation de techniques du TAL pour la recherche sémantique d'images. Nous appelons *recherche sémantique d'images* l'ensemble des techniques qui visent à donner une description plus riche des images que les simples descripteurs de bas-niveau. Ces techniques s'attachent à identifier le contenu sémantique des images (présence d'objets, de personnes, de concepts précis) plutôt que leur aspect visuel. Il est globalement admis que cette description sémantique passe par l'utilisation de mots pour décrire les images, à la place ou en complément des descripteurs bas-niveau, de même qu'une personne aura tendance à décrire une image en citant les objets qui y apparaissent (un ballon, le ciel), puis éventuellement en précisant sa description par des éléments visuels (un ballon rouge, le ciel bleu) qu'en décrivant uniquement les propriétés visuelles de l'image (un rond rouge, un à-plat bleu).

Parmi les méthodes proposées pour réaliser une description sémantique des images, on peut distinguer d'une part les techniques d'annotation d'images, qui visent à associer automatiquement des mots-clefs aux images en fonction de caractéristiques de bas-niveau

qui en sont extraites, et d'autres part les techniques de recherche texte-image, qui associent dans la description des images des descripteurs de bas-niveau et des mots-clefs extraits des textes associés aux images (dans des pages web par exemple). Dans les deux cas, ces techniques nécessitent pour fonctionner des données qui associent du texte et de l'image : les techniques d'annotation nécessitent une phase d'apprentissage souvent gourmande en données, les techniques de recherche texte-image nécessitent quant à elles de disposer de textes associés aux images pour en calculer un descripteur mixte.

### 1.4.1 Approches *bottom-up* et approches *top-down*

Une première distinction classique en recherche sémantique d'images est de séparer les techniques dites *bottom-up* des techniques dites *top-down* [HSL+06]. Les techniques *bottom-up* se basent uniquement sur les données disponibles (images, textes, mots-clefs) pour mettre en place leur processus de recherche ou d'annotation. Ces techniques seront décrites plus précisemment dans la suite de cette partie. Les techniques *top-down*, quant à elles, reposent sur des connaissances disponibles *a priori*. Ces connaissances peuvent être de différentes natures :

- métadonnées liées à la création de l'image : ce sont des renseignements sur les conditions dans lesquelles ont été créées les images : coordonnées GPS, date et heure, distance focale de l'appareil, auteur, etc. ;
- connaissances génériques : certaines ressources généralistes sur la langue peuvent être exploitées pour créer, enrichir ou désambiguiser les informations textuelles. L'outil de ce type le plus courant est Wordnet [Fel98], un réseau sémantique généraliste pour la langue anglaise, utilisé dans plusieurs travaux d'annotations ou de recherche d'images [BJ03a, HSWW03, KNA+04, PG08] ;
- connaissances liées à un domaine : l'annotation des images peut être guidée par des classifications ou hérarchies de connaissances standards utilisées dans le domaine d'expertise couvert par les images. On dispose de telles connaissances spécifiques dans les domaines de l'art [KSA+08] ou de la médecine [LCLG08], par exemple.

Lorsque des connaissances *a priori* sont utilisées, elles le sont généralement dans le cadre d'une approche mixte, c'est à dire reposant sur les données pour établir des annotations, mais utilisant également les connaissances *a priori* pour sélectionner les informations textuelles, lever des ambiguités ou rapprocher certains termes (par exemple, les termes synonymes) [PG08]. Certains auteurs cherchent également à utiliser des données sous forme d'ontologies construites automatiquement, comme dans les travaux de Wang *et al.*, mais leur phase de construction de l'ontologie n'est pas totalement automatisée et nécessite donc une intervention manuelle [WLC08].

Le principal inconvénient de l'utilisation de connaissances *a priori* est le coût élevé d'obtention de ces connaissances, car elles sont généralement constituées entièrement à la main. Si l'on utilise des connaissances spécialisées, chaque changement de domaine d'application implique de créer manuellement une nouvelle base de connaissances adaptée. Si l'on utilise des bases génériques, on se touve généralement confronté à de nombreuses ambiguités dues au fait qu'un mot peut avoir des sens différents d'un domaine de spécialité à un autre. Par exemple, WordNet propose 5 contextes différents pour le mot anglais *car* ; aussi certains travaux choisissent-ils de se limiter à un sous-ensemble de WordNet, se limitant de fait à un domaine particulier [WLC08].

Compte-tenu du coût d'obtention des ressources extérieures, mais également du faible nombre de travaux en exploitant (à l'exception notable de Wordnet), nous nous concentrerons principalement sur les méthodes basées sur les données (*bottom-up*) dans la suite de

ce manuscrit.

### 1.4.2 Nature de l'information textuelle

La seconde distinction possible en recherche sémantique d'images concerne la nature des données textuelles accompagnant les images, données dont on extrait les mots-clefs ou syntagmes qui serviront à annoter et rechercher les images. Ces données textuelles peuvent se présenter sous trois formes différentes :

- un ensemble de mots-clefs : les images sont associées à un ou plusieurs mots-clefs décrivant leur contenu visuel et/ou conceptuel. Les couples (image, mots-clefs) peuvent servir de données d'apprentissage pour acquérir automatiquement les relations entre images complètes et mots-clefs, pour découvrir les relations entre des régions d'image et les mots-clefs, ou uniquement servir de base à un système de recherche. Les données de ce type les plus répandues sont les données des bases d'images Corel [BDF+03b] ;
- une description textuelle de l'image : les images sont accompagnées par une description plus ou moins détaillée de leur contenu, exprimée en langage naturel. Il n'y a plus d'association explicite entre termes d'annotation et images, il est alors nécessaire d'interpréter le contenu textuel pour isoler, parmi les termes contenus dans la description, les termes d'annotation décrivant effectivement l'image et, éventuellement, des relations entre ceux-ci. Parmi les données de ce type, on peut citer les couples (image, légende) récupérés sur les sites d'actualités en ligne (souvent *Yahoo ! News*), qui offrent une description concise des images (une ou deux phrases) et sont souvent utilisés pour les travaux d'annotation [DM07, BBE+04]. On retrouve également des corpus plus spécialisés, comme les dictionnaires d'œuvres d'art [KSA+08], ou encore le corpus de scènes de crimes utilisé par Pastra *et al.* [PSW03], qui offre des descriptions très précises du contenu des images (*une table haute au centre de la pièce*, *un couteau posé sur la table*, etc) ;
- un texte illustré par une ou plusieurs images : les images illustrent le contenu textuel du document, sans que celui-ci ne décrive spécifiquement l'image ou ne s'y réfère explicitement. Il ne s'agit donc plus ici de faire simplement la part entre les mots correspondant à du contenu visuel et les autres, mais aussi d'isoler les parties décrivant les illustrations, ou l'intention du texte que l'on retrouve dans les illustrations. Ce genre de données est beaucoup moins courant que les autres dans les travaux d'annotation, bien qu'il corresponde à un cas courant dans les applications réelles (pages web par exemple). Parmi les données de ce type, il y a les articles de presse ([JT09] utilise de courts extraits d'articles, [FL08] des articles complets en parallèle avec les légendes des images), les articles de l'encyclopédie libre Wikipedia [FTATG08] ou encore les données de l'ancienne campagne d'évaluation ImagEval [TG07].

La nature des données textuelles a évidemment un impact sur les méthodes d'annotation utilisées. Lorsque les données textuelles se résument à des ensembles de mots-clefs, le problème se réduit à l'apprentissage ou la découverte de relations entre les descripteurs visuels des images et les mots-clefs. Ce problème, bien que formulé simplement, peut néanmoins se révéler très difficile à résoudre quand la complexité des données traitées augmente.

### 1.4.3 Annotation d'images

La problématique de l'annotation d'images est la suivante : comment, étant donnée une image dont le contenu sémantique est inconnu, lui associer automatiquement une description de nature textuelle, sous forme d'une liste de mots-clefs ? Cette tâche peut être attaquée selon deux axes, illustrés par la figure 1.10 :

FIG. 1.10 – Une image annotée globalement [ZLX09] et une image annotée localement [BDF03a].

- l'annotation globale : annotation des images dans leur ensemble à l'aide de mots-clefs (on parle parfois de modèle d'auto-annotation) ;
- l'annotation locale : annotation de régions d'image à partir d'un mot-clef spécifique (on parle parfois de modèle de correspondance).

Bien que les travaux d'annotation prennent souvent le problème suivant l'un de ces deux axes, ces derniers peuvent s'avérer complémentaires. En effet, si certains mots-clefs décrivent des objets précis (*sable*, *ciel*, *eau* par exemple), d'autres sont plus adaptés à la description d'une scène dans son ensemble (*plage* par exemple, qui décrit des images dont certaines régions correspondent aux mots-clefs précédents *sable*, *ciel* et *eau*). Cette complémentarité peut amener à vouloir raisonner sur les régions, ou mots-clefs associés au régions, pour déduire des annotations globales à partir d'annotations locales. Le choix de l'axe employé aura en revanche une influence sur le choix des descripteurs employés pour représenter les images. En effet, si réaliser des annotations globales laisse le choix d'employer des descripteurs globaux ou locaux, annoter des régions nécessite évidemment d'avoir recours à une description locale. Plus précisemment, les travaux traitant d'annotation d'images par régions ont recours à un découpage des images en grille ou par segmentation, mais jamais par régions d'intérêt. On peut voir deux explications à cette tendance : les descriptions par région d'intérêt sont apparues assez récemment, après les premiers travaux d'annotation, et, surtout, elles ne couvrent pas la totalité de l'image, ce qui rend difficile une annotation complète de l'image et le regroupement des régions en fonction de la sémantique de l'image.

#### 1.4.3.1 Principaux modèles d'annotation

Nous présentons ici les principaux modèles pour annoter des images, c'est-à-dire associer automatiquement des descripteurs d'images à des mots-clefs. Les modèles sont regroupés par catégories, indépendamment de la stratégie d'annotation, globale ou locale. Quelle que soit la stratégie adoptée, tous ces modèles nécessitent de disposer de données d'apprentissage, c'est-à-dire d'images déjà annotées sur lesquelles se baser pour apprendre les relations existant entre les descripteurs et les mots-clefs. Ces données d'apprentissage peuvent elles-mêmes être annotées de manière globale ou locale, sans que cela n'ait nécessairement d'influence sur la stratégie finalement adoptée. En effet, si certains modèles, comme les modèles à classifieurs, nécessitent de disposer d'images d'apprentissage annotées localement pour apprendre à annoter des régions d'image, d'autres, comme les modèles génératifs, permettent d'inférer automatiquement des annotations locales à partir d'images d'apprentissage annotées globalement.

**Classification supervisée** La classification supervisée est une branche de l'intelligence artificielle, plus précisément de l'apprentissage artificiel, dont l'objectif est de permettre de classer des données en deux catégories (classification binaire) ou plus (classification multi-classes). Si l'on note $\mathcal{C}$ l'ensemble des classes possibles et $\mathcal{X}$ le domaine des données en entrée, on peut considérer un classifieur supervisé comme étant une fonction $f : \mathcal{X} \rightarrow \mathcal{C}$, de paramètres $w$, telle que :

$$\forall x \in \mathcal{X}, c_i \in \mathcal{C}, x \in c_i \Leftrightarrow f(x; w) = c_i \tag{1.10}$$

Les paramètres $w$ sont appris à partir de données étiquetées (ou données d'apprentissage), c'est-à-dire de données dont la classe est connue, lors de la phase d'apprentissage. L'objectif de cette phase est de déterminer les paramètres $w$ qui permettront de classer sans erreur les données dans le cas général, c'est-à-dire des données différentes de celles de l'ensemble d'apprentissage (on parle de faculté de généralisation du classifieur). La forme générale de la fonction $f$ est quant à elle déterminée par le type de classifieur utilisé. Par exemple, un réseau de neurones pourra définir une fonction linéaire par parties, et un Séparateur à Vaste Marge (SVM, *Support Vector Machine*) définira une fonction non-linéaire en passant par une projection dans un espace de plus grande dimension que l'espace $\mathcal{X}$ (voir [CMK02] pour une description détaillée des principaux classifieurs). En pratique, il n'est généralement pas possible de trouver une fonction $f$ réalisant un partitionnement parfait des données dans le cas général, ni même des données d'apprentissage, on se contente alors d'une approximation de $f$, $\tilde{f}$, minimisant les erreurs de classification.

La formulation de l'annotation d'images comme un problème de classification n'est pas directe. En effet, dans un cadre classique de classification, on chercherait à attribuer une classe unique à chaque image, hors plusieurs mots-clefs sont généralement nécessaires pour en décrire le contenu. La solution généralement adoptée [LCW03, LGC03, CGSW03, JLZZ04, FSC04, GCL05, DI07, FBC08, QH07, WL08, LMS+09, WHY09] consiste à entraîner des classifieurs binaires qui déterminent chacun la présence ou l'absence d'un concept donné dans les images, ces concepts étant ensuite mis en correspondance avec les termes d'annotation à proprement parler, de façon directe (1 concept équivaut à un mot-clef) ou non (plusieurs concepts impliquent un mot-clef, ou plusieurs mots-clefs représentent un concept). Cette approche permet, à partir d'images d'apprentissage annotées globalement, d'annoter des images complètes, mais pas d'inférer des annotations de régions d'image. Pour réaliser l'annotation de régions d'image par un classifieur, il faut disposer de données d'apprentissages préalablement annotées par régions, comme dans les travaux de Cusano *et al.* [CCS04] et Town et Sainclair [TS00]. Certains auteurs ajoutent à cette étape d'annotation des régions une étape supplémentaire visant à exploiter les concepts élémentaires ainsi détectés (par exemple, *sable* et *mer*) pour inférer des annotations d'ordre plus général (*plage*) [GWL06, VS07]. Malgré tout, ces deux dernières approches sont fortement limitées par le coût excessif de l'annotation manuelle d'images par régions. Une autre limite de cette approche est que le fait de considérer des images complètes en entrée des classifieurs rend difficile l'identification de concepts n'occupant qu'un espace limité dans les images, y compris en ayant recours à des ensembles de descripteurs locaux (la plupart des descripteurs locaux d'une image étiquetée positive s'avérant en effet des exemples d'apprentissage négatifs dans les faits). Certaines méthodes d'apprentissage permettent de gérer la présence d'un fort bruit dans les données d'entrée, comme par exemple l'apprentissage d'instances multiples (MIL, *Multiple-Instance Learning*) utilisé par Yang *et al.* [YDH06], puis Qi *et al* [QH07].

Les classifieurs les plus utilisés pour réaliser l'annotation d'images sont les SVM, en raison de leurs excellentes performances pour les tâches de classification en général. On le

trouve sous sa forme classique [JLZZ04, CCS04, GWL06], ou sous des formes légèrement modifiées : intégration d'apprentissage MIL [YDH06, QH07], d'active learning [FBC08], de co-apprentissage (*co-training*) [FSC04], de scores de confiances [LGC03]... On retrouve également des *Bayes Points Machines* [CGSW03], des réseaux de neurones [TS00], ou des arbres de décision [WL08]. Les classifieurs de type bayésien (utilisé par [WHY09] par exemple) et modèles de Markov cachés (HMM, *Hidden Markov Models*, utilisés par [LCW03, GIK05] par exemple), qui reposent également sur une phase d'optimisation de leur paramètres à partir de données d'apprentissage, se situent plutôt dans la catégorie des modèles génératifs, étudiée plus bas. Le classifieur des $k$ plus proches voisins ($k$-NN, *$k$-nearest neighbors*), quant à lui, correspond au cas de la propagation de mots-clefs, traité dans le paragraphe suivant.

**Propagation de mots-clefs** L'idée de départ de la propagation de mots-clefs est que des images proches visuellement doivent partager la même sémantique, et donc que les termes d'annotation pertinents pour les décrire sont les mêmes. Le principe de base de la propagation de mots-clefs consiste donc, pour annoter une nouvelle image, à effectuer une recherche sur des critères visuels dans un ensemble d'images annotées, puis à attribuer à cette image tous les termes d'annotation associés aux premières images retournées par le système, ou les termes d'annotations communs à ces images. Bien qu'elle repose essentiellement sur des données d'apprentissage, comme les modèles d'annotation à base de classifieurs et les modèles génératifs, cette méthode se distingue néanmoins de ceux-ci par son aspect local par rapport aux données d'apprentissage : dans le cas de classifieurs et de modèles génératifs, l'objectif est de définir un modèle unique pour chaque mot-clef ou concept, modèle qui devra donc prendre en compte les variations visuelles du concept, qui sont potentiellement nombreuses ; à l'inverse, le modèle par propagation de mots-clefs ne considérera, lors de l'annotation d'une image, qu'une des représentations visuelles possibles de chaque concept à la fois, celle qui figure effectivement dans l'image à annoter. La propagation de mots-clefs se rapproche néanmoins d'un classifieur particulier, le $k$-NN[3]. La différence majeure se situe dans le fait que le classifieur $k$-NN ne "propage" qu'une classe unique, alors que la propagation de mots-clefs s'intéresse à des annotations généralement multiples, ce qui donne plus de latitude dans le choix des annotations à propager, en recoupant par exemple le nombre d'occurrences de chaque annotation parmi les $k$ voisins, ou en considérant les distances des voisins à l'image requête. Certains auteurs considèrent également que si l'on considère autant de plus proches voisins qu'il y a d'exemples d'apprentissage, toute approche se basant sur un ensemble d'apprentissage se réduit à faire de la propagation de mots-clefs, à partir de ces données d'apprentissage (Jing *et al.* [JLZZ04], par exemple, nomme propagation de mots-clefs le fait d'utiliser des SVM). Cette assimilation peut sembler néanmoins excessive compte-tenu des différences évoquées ci-dessus. Les différentes approches par propagation de mots-clefs se distinguent par :

- la méthode utilisée pour rechercher les plus proches voisins : Hare *et al.* comparent par exemple une distance classique et l'utilisation de l'analyse de la sémantique latente (LSA, *Latent Semantic Analysis*), alors que Guillaumin *et al.* utilisent une distance optimisée sur les données ;
- la représentation des données : certains auteurs représentent les proximités entre images par un graphe [THL+06, LDXE08], d'autres représentent les relations entre images et mots-clefs par un réseau sémantique [ZS01] ;
- l'utilisation conjointe de propagation de mots-clefs avec d'autres méthodes : par

[3]Classifieur $k$-NN : classifieur qui attribue à un objet donné en entrée la classe majoritaire parmi les $k$ exemples d'apprentissage qui en sont les plus proches au sens d'une distance donnée.

exemple une classification par SVM [LMS+09], le retour de pertinence [THL+06, ZS01] ;
- les critères de propagation des mots-clefs : ils peuvent être élémentaires (propagation systématique de $n$ mots-clefs) [HL05, TL06]) ou plus fins (prise en compte de la distance des images à l'image requête par exemple) [GMVS09], et prendre en compte des relations entre mots-clefs (inclusion et exclusion entre concepts) [ZLX09].

Bien qu'ayant reçu peu d'attention au départ, les modèles par propagation de mots-clefs ont montré récemment qu'ils pouvaient surpasser les meilleurs modèles génératifs ou par classifieurs [MPK08, GMVS09]. Cette efficacité serait due à leur capacité à considérer indépendamment les différentes représentations visuelles des concepts d'annotation, évoquée ci-dessus. Une contrepartie à cela pourrait être une grande sensibilité à la quantité et la représentativité des données d'apprentissage, bien qu'aucune étude ne soit encore disponible à ce sujet.

**Modèles génératifs** Les modèles génératifs sont des modèles probabilistes représentant le lien entre un terme d'annotation $w$ et une image $I$ (respectivement, une région d'image $r$) par la probabilité $\Pr(w|I)$ (resp. $\Pr(w|r)$) que cette image (resp. cette région) génère $w$. Depuis les premiers modèles génératifs proposés par Barnard et Forsyth [BF01], qui s'inspiraient notamment des travaux de Hofmann sur la modélisation des textes [Hof98], de nombreux modèles de ce type ont été proposés. Ces modèles se distinguent les uns des autres par plusieurs aspects :
- les variables cachées introduites dans l'estimation des probabilités $\Pr(w|I)$ ou $\Pr(w|r)$, et les relations de dépendance que ces variables cachées entretiennent avec les variables visibles du système (termes d'annotation, descripteurs visuels). Ces variables cachées représentent généralement une notion de "concepts" présents dans les images et conditionnant les termes d'annotation ou les descripteurs visuels de ces images, les concepts conditionnant l'occurrence des mots-clefs et les concepts conditionnant les descripteurs visuels pouvant être les mêmes ou non, selon la modélisation adoptée ;
- les distributions de probabilités utilisées pour modéliser les concepts, les termes d'annotation et les descripteurs visuels, ainsi que les paramètres de ces distributions. Les descripteurs visuels sont généralement modélisés par des mélanges de distributions gaussiennes, et les termes d'annotation par des mélanges de distributions multinomiales, les paramètres de ces distributions étant conditionnés par les variables cachées du système. Les concepts sont modélisés par des distributions de Dirichlet ou des distributions multinomiales, selon les auteurs.

Les paramètres des différentes distributions sont estimés en utilisant l'algorithme E.M. (*Expectation-Maximization*) [DLR77]. Le nombre de variables cachées et les relations qu'elles entretiennent entre elles permettent de modéliser plus ou moins finement les associations entre termes d'annotation et descripteurs visuels (bijection entre termes et descripteurs, correspondance entre termes et ensembles de descripteurs...), mais la multiplication des variables cachées rend l'estimation des paramètres plus difficile.

Une propriété intéressante de certains de ces modèles est leur capacité d'apprendre, à partir d'images annotées, non seulement les relations entre images et mots-clefs, mais aussi entre régions d'image et mots-clefs, bien que ne disposant pas au départ d'annotations par régions. Ces modèles sont évalués sur des collections d'images Corel, dont les limitations sont connues [MMMP02], à l'exception de Jeon et Manmatha qui proposent une évaluation de leur système sur des données réelles [JM04], ou de Feng et Lapata qui évaluent certains de ces systèmes sur des données réelles (articles de presse accompagnés d'une photographie),

après un prétraitement des textes et légendes des images qui permet de limiter le nombre de termes considérés comme termes d'annotation [FL08].

Parmi les modèles probabilistes les plus célèbre, on peut citer les modèles de traductions proposés par Barnard *et al.* [BDF03a], les modèles basés sur pLSA (une variante probabiliste de LSA proposée par Hofmann [Hof99]) de Monay et Gatica-Perez [MGP04], les modèles basés sur l'allocation latente de Dirichlet (LDA, *Latent Dirichlet Allocation*) de Blei et Jordan [BJ03b] ou les modèles *Cross-Media Relevance Model* [JLM03] et *Continuous Relevance Model* [LMJ03] de Jeon *et al.*.

**Projection dans un espace de similarités** Le principe de la projection dans un espace de similarités est le suivant : disposant de deux modalités de description des images, l'une textuelle et l'autre visuelle, chacune étant définie dans un espace distinct (typiquement, l'espace des termes d'annotation pour la modalité textuelle, et l'espace des descripteurs visuels pour la modalité visuelle), l'objectif est de définir un nouvel espace commun, parfois appelé espace de similarités, ainsi que, pour chaque modalité, une fonction de projection de l'espace d'origine dans ce nouvel espace, de telle sorte que les proximités entre les éléments projetés dans l'espace commun corresponde à des proximités sémantiques entre les éléments initiaux. Cela nécessite de :

1. disposer de données annotées servant de référence en termes de proximités sémantiques à obtenir dans l'espace final. On se situe donc dans un cadre d'apprentissage supervisé ;
2. définir la forme générale prise par les fonctions de projection (transformation linéaire ou non par exemple) ;
3. déterminer les paramètres optimaux des fonctions de projection, c'est-à-dire les paramètres qui permettent de s'approcher au mieux des proximités sémantiques de référence telles qu'elles sont définies dans les données d'apprentissage. Cette étape se réduit à la résolution d'un problème d'optimisation numérique où l'on cherche les paramètres minimisant l'écart entre les proximités théoriques et celles obtenues en pratique.

Une fois les fonctions de projection connues, les termes d'annotation d'une image correspondent aux termes les plus proches du descripteur visuel de l'image dans l'espace de similarités. Inversement, il est possible d'illustrer un mot-clef ou un texte par les images qui lui sont proches dans l'espace de similarités. Cette stratégie partage avec la classification supervisée le fait d'optimiser des fonctions à partir de données étiquetées. Elle s'en distingue par la nature des fonctions obtenues : alors que l'objectif de la classification supervisée sera de trouver une fonction séparatrice des classes, l'objectif ici consiste à obtenir deux fonctions de projection complémentaires. La modélisation du problème d'annotation se rapproche ainsi plus de celle utilisée dans des modèles génératifs comme le modèle GM-Mixture [BJ03b], qui utilise une variable cachée représentant les concepts communs entre les modalités textuelle et visuelle. Les outils de classification supervisée peuvent néanmoins être utilisés pour définir les fonctions de projection et en estimer les paramètres optimaux. Ainsi, Carkacioglu *et al.* utilisent des réseaux de neurones pour estimer leurs fonctions de projection [CV02]. Liu *et al.* modélisent quant à eux les fonctions de projection comme des transformations linéaires [LQL+05]. Jiang *et al.* proposent un méthode proche de la projection dans un espace de similarités [JT09].

**Annotation de photos de personnes** Un cas particulier d'annotation d'images est l'annotation de photos par le nom des personnes y apparaissant. Ce problème a d'abord

été évoqué par Srihari *et al.* pour les images [SB94] et Satoh *et al.* pour la vidéo [SNK99]. Ce problème est généralement traité dans un cadre d'annotation de collections d'images personnelles [Zha03] ou d'images de presse [BBE+04, PCL08]. Dans le premier cas, des images annotées par des noms de personnes sont fournies ; dans le second cas, les images sont accompagnées de légendes dont les noms sont extraits par un processus élémentaire de traitement automatique des langues (patrons adaptés à la forme des légendes). Le problème consiste alors à identifier, à partir des images annotées globalement, quels visages correspondent à quels noms, puis, éventuellement, à annoter de nouvelles images inconnues du système. Ces systèmes fonctionnent généralement par détection des visages, puis par *clustering* des visages détectés, pour regrouper les visages selon les personnes, puis par analyse des cooccurrences avec les noms utilisés pour annoter les images globalement. Les performances de ces systèmes dépendent donc en majeure partie des performances des détecteurs et surtout des descripteurs de visages, pour pouvoir reconnaître et regrouper les différentes occurrences d'un même visage. Il s'avère que ces systèmes de reconnaissance de visage fonctionnent encore assez mal sur des données réelles comme les images de presse : Pham *et al.* reportent ainsi que les groupes de visages obtenus automatiquement ne correspondent qu'à 33% avec leur vérité-terrain [PCL08]. Ces approches sont néanmoins intéressantes car elles proposent, contrairement aux autres approches d'annotation, de travailler sur des indices visuels et textuels spécifiques, de haut-niveau, dont la correspondance est établie, plutôt que sur des descripteurs bas-niveau dont l'aspect sémantique est très limité. C'est sur ce type d'approche que se base la méthode d'annotation que nous proposons au chapitre 5.

#### 1.4.3.2 Évaluation des systèmes d'annotation

Plutôt que d'évaluer les performances des systèmes d'annotation en fonction de la performance d'un système de RI reposant sur les mots-clefs utilisés pour annoter les images, on peut vouloir mesurer directement la pertinence des annotations. Ces deux propositions sont en réalité similaires : si l'on effectue une recherche portant sur un des mots-clefs utilisés pour annoter les images, la liste résultat obtenue sera la liste des images annotées par ce mot-clef, qui sera comparée, à des fins d'évaluation, à la liste des images qui auraient dûes être annotées par ce mot-clef, d'après la vérité-terrain. Il est donc possible d'utiliser directement les mesures de rappel et de précision disponibles pour évaluer les systèmes de RI, décrites en section 1.1.6.2, avec l'interprétation suivante :

- précision : proportion d'images correctement annotées par un mot-clef par rapport à l'ensemble des images annotées par ce mot-clef ;
- rappel : proportion d'images correctement annotées par un mot-clef par rapport à l'ensemble des images qui auraient dû l'être.

En calculant des précisions et rappels moyens sur l'ensemble des mots-clefs possibles, on obtient un aperçu des performances d'un système d'annotation.

Certains auteurs [BDF+03b, Tol06] ajoutent un score supplémentaire à ces mesures de rappel et de précision, le score normalisé moyen ($E_{NS}$, *Average Normalized Score*), défini ainsi pour $N$ images de test :

$$E_{NS} = \frac{1}{N} \sum_{i=1}^{N} \frac{|\Psi_i \cap \Delta_i|}{|\Delta_i|} - \frac{|\Delta_i - \Psi_i|}{|\Omega - \Delta_i|} \tag{1.11}$$

où $\Psi_i$ est l'ensemble des mots-clef associés automatiquement à l'image $I_i$, $\Delta_i$ l'ensemble des mots-clefs associés à l'image $I_i$ d'après la vérité-terrain, et $\Omega$ l'ensemble des mots-clefs pris en compte par le système. Pour une image donnée, le score normalisé prend pour valeur 1 quand sont retrouvés tous les mots-clefs pertinents et uniquement ceux-ci, -1 quand le

système attribue tous les mots-clefs non-pertinents et uniquement ceux-ci, et 0 si tous les mots-clefs sont associés à l'image, ou aucun. La prise en compte de l'ensemble des mots-clefs gérés par le système permet de favoriser les systèmes permettant de prendre en compte un grand nombre de mots-clefs : à précision égale, un système qui prend en compte plus de mots-clefs qu'un autre aura un score $E_{NS}$ plus important, car il avait plus de chances de commettre des erreurs.

Un autre score parfois utilisé est la justesse (Acc, *Accuracy*) d'annotation [BDF+03b, Tol06], définie comme un rappel des annotations pour une image $I_i$ donnée :

$$\mathrm{Acc}_i = \frac{|\Psi_i \cap \Delta_i|}{|\Delta_i|} \tag{1.12}$$

Moyennée sur un ensemble de $N$ images, la justesse permet d'obtenir un score de prédiction moyen (WPS, *Word Prediction Score*) [BDF+03b, Tol06] :

$$\mathrm{WPS} = \frac{1}{N} \sum_{i=1}^{N} Acc_i \tag{1.13}$$

### 1.4.4 Fusionner les informations visuelles et textuelles

Outre l'annotation d'images, l'autre grande approche de l'indexation sémantique d'images consiste à utiliser non plus les données mêlant texte et image pour apprendre à annoter de nouvelles images, mais à exploiter directement les deux médias conjointement à des fins de recherche d'information ou, parfois, de catégorisation. L'idée qui motive ce choix est que les informations issues du texte et celles issues des descripteurs visuels ne sont pas redondantes mais complémentaires, et qu'il faut donc les utiliser conjointement plutôt que de chercher des relations entre elles, comme le font les systèmes d'annotation. Le défi consiste alors à fusionner les deux types d'informations disponibles, le texte et l'image, afin de tirer parti au mieux de ces deux sources d'information. La difficulté majeure ici se situe dans le fait que les représentations adoptées pour décrire chacun de ces médias sont généralement très différentes. Ainsi les images sont généralement décrites par des vecteurs denses, de distribution supposée gaussienne[4], alors que le texte est décrit par des vecteurs creux de très haute dimension, ou par des mélanges de distributions supposées multinomiales. Ces différences provoquent naturellement des difficultés lorsqu'il s'agit de fusionner ces descriptions dans un même système de recherche. Bien que beaucoup moins nombreux que les travaux disponibles en annotation d'images, il existe un certain nombre de travaux dans cette direction, dont on peut extraire trois axes majeurs pour attaquer ce problème : la fusion précoce, la fusion tardive, et l'utilisation d'une des modalités pour réhausser les résultats obtenus par l'autre.

#### 1.4.4.1 Fusion précoce

Les méthodes de fusion précoce consistent à trouver un descripteur contenant conjointement les informations visuelles et textuelles, puis à utiliser ce descripteur dans un cadre classique de recherche d'information ou de catégorisation, selon l'objectif recherché. Les travaux reposant sur cette approche sont plutôt rares. On peut néanmoins citer l'approche de Zhou *et al.*, qui repose simplement sur la concaténation des vecteurs visuels et textuels, vecteurs qu'ils corrigent au fur et à mesure des interactions avec l'utilisateur par un processus de retour de pertinence [ZH02]. L'autre approche majeure, adoptée par Zhao *et*

[4]Cette supposition n'a cependant jamais été vérifiée en pratique.

*al.* [ZG02], puis par Pham *et al.* [PCL08], consiste à concaténer les vecteurs représentant chaque modalité pour obtenir un vecteur unique par document, puis à effectuer une analyse de la sémantique latente pour projeter ces vecteurs dans un nouvel espace, de dimension plus réduite, unifiant les informations visuelles et textuelles. Dans leurs expériences, Zhao *et al.* montrent que la concaténation simple des vecteurs de chaque modalité diminue les performances du système par rapport à l'utilisation du texte seul, alors que l'utilisation de LSA pour fusionner les modalités permet d'améliorer les résultats.

#### 1.4.4.2 Fusion tardive

Les méthodes de fusion tardive, partant du principe que les modalités textuelles et visuelles ont trop peu en commun, proposent de fusionner les résultats de recherche plutôt que les descripteurs. Cette fusion peut être opérée à deux niveaux :

- fusion des distances : une distance est calculée selon chaque modalité, visuelle et textuelle, puis un score global est calculé en fonction de ces deux distances. Les deux distances utilisées n'ont pas besoin d'être identiques. La méthode la plus simple pour fusionner des distances consiste à réaliser une interpolation linéaire :

  $$d_f(d_i, d_j) = \alpha.d_V(V(d_i), V(d_j)) + (1 - \alpha).d_T(T(d_i), T(d_j))$$

  où $d_f(d_i, d_j)$ désigne la distance finale entre les documents $d_i$ et $d_j$, $d_V(V(d_i), V(d_j))$ la distance ente les descripteurs visuels et $d_T(T(d_i), T(d_j))$ la distance entre les descripteurs textuels. $\alpha$ est un facteur qui pondère l'importance relative des modalités dans la distance finale, mais aussi de normalisation si les distances textuelle et visuelle ne sont pas elle-même normalisées. L'interpolation linéaire des distances a par exemple été utilisée par Tollari *et al.* pour rechercher des documents mêlant texte et image [TG07], et par Lin *et al.* [LCC04] ;
- fusion des classements : un classement des résultats est calculé selon chaque modalité, puis les classements sont fusionnés pour obtenir le classement final. Bien qu'il existe de nombreuses manières de fusionner des listes ordonnées de résultats [CELM07], la méthode généralement employée est celle de de Borda [dB81], qui consiste à attribuer à chaque document des scores en fonction de ses positions dans les différents classements (typiquement, le premier document reçoit un score de $n$ et le dernier et $n$-ième un score de 1), puis à calculer le classement final en fonction de la somme de ces scores. Cette approche est par exemple utilisée par Fakeri *et al.* [FTATG08] pour la recherche de documents mêlant texte et image. Une autre méthode de fusion, très différente de celles évoquées dans [CELM07], est celle utilisée par Rui *et al.* [RYWL07] dans un contexte d'annotation d'images et qui combine classements et distances dans le cadre de la théorie de Dempster-Shafer [Sha76].

#### 1.4.4.3 Réhaussement d'une information par l'autre

Cette méthode consiste à effectuer d'abord une recherche selon une modalité donnée, puis à réordonner les résultats de cette recherche selon la seconde modalité. Elle a été proposée par Tollari *et al.* qui réalisent d'abord un classement des documents en fonction des descripteurs textuels, puis reclassent les résultats obtenus selon des critères de recherche visuelle [Tol05, FTATG08]. Popescu et Grefenstette utilisent une approche similaire pour la recherche d'images sur internet, avec une phase préalable d'extension de requêtes à l'aide de Wordnet [PG08]. Une approche similaire a été utilisée par Zhu *et al.* dans un cadre de classification d'images, en inversant les modalités : un premier classifieur SVM sélectionne des catégories potentielles pour l'image requête, puis un second SVM choisit la catégorie

définitive à partir du texte qui a été détecté dans l'image [ZYC06]. Cette méthode est également proposée dans son cadre général par Ah-Pine *et al.* sous le nom de retour de pertinence trans-média (*transmedia relevance feedback*) [APBC+09].

#### 1.4.4.4 Conclusion sur la fusion des informations textuelles et visuelles

Aucune étude complète mettant en compétition les différentes méthodes de fusion n'étant disponible, et les données des différentes études disponibles étant différentes, il n'est pas possible de conclure en faveur d'une méthode plutôt que d'une autre. Plusieurs de ces études [TG07, FTATG08, ZG02, PCL08] mettent néanmoins en avant un point essentiel : ce sont les descripteurs textuels qui apportent la majorité de l'information au système de recherche d'information, les informations visuelles ne permettant d'apporter qu'une amélioration modeste aux performances globales. En particulier, Tollari et Glotin font une étude détaillée du rapport entre information textuelle et information visuelle dans le cadre d'une fusion basée sur une combinaison linéaire des distances visuelle et textuelle. Ce résultat met en évidence la faiblesse des descripteurs d'images classiques lorsqu'il s'agit de rechercher des images au contenu variable et complexe, comme évoqué en 1.3.1.2.

## 1.5 Conclusion

Nous avons d'abord donné, dans ce chapitre, des notions de base de recherche d'information et de recherche d'images, puis nous avons dressé un état des lieux des travaux en recherche d'images, que nous avons organisé en fonction des deux axes que nous avons choisi de suivre pour notre étude : la recherche d'images par le contenu, d'abord, puis la recherche sémantique d'images.

Globalement, il ressort des résultats actuels de recherche d'images que les descripteurs globaux ne permettent pas de capter suffisamment de détails dans les images pour permettre d'effectuer d'obtenir des résultats qui soient vraiment pertinents lorsque l'on s'intéresse à de grandes collections d'images très variées. Ce sont plutôt les approches locales qui sont plébiscitées à l'heure actuelle, et plus particulièrement les approches qui se basent sur la quantification de descripteurs globaux. Comme nous allons le montrer dans le chapitre suivant, la représentation des images fournie par ce type d'approche partage avec la représentation classique du texte de nombreux points communs, c'est donc vers celles-ci que nous allons nous orienter pour appliquer des techniques de TAL à la recherche d'images par le contenu.

La recherche sémantique d'images, quant à elle, motive de très nombreux travaux, essentiellement en annotation d'images. Les approches actuelles d'annotation, basées sur un usage intensif de techniques d'apprentissage artificiel et de descripteurs de bas-niveau, se heurtent au problème supposé du fossé sémantique, qui justifierait que leurs performances restent limitées, malgré l'usage de grandes quanités de données d'apprentissage, qui sont obtenues au prix d'une intervention humaine coûteuse. Les travaux fusionnant texte et descripteurs de bas-niveau corroborent cette observation, en montrant que ce sont avant tout les données textuelles qui permettent de retrouver les images pertinentes, et que les informations visuelles sont utiles mais ne jouent qu'un rôle limité dans le processus de recherche. Nos travaux visent à mieux exploiter l'information textuelle, en lui appliquant des méthodes de TAL, pour obtenir une description des images qui soit vraiment fidèle à leur contenu visuel. Nous nous appuierons pour cela plutôt sur des descripteurs de haut-niveau qui permettent de caractériser certains éléments visuels des images de manière plus précise, et donc de contourner le problème du fossé sémantique.

# Chapitre 2

# Traitement automatique des langues, recherche d'information textuelle et recherche d'images

Comme nous l'avons expliqué en introduction, l'indexation d'images, bien qu'ayant été étudiée bien plus tardivement que l'indexation de textes et les travaux en TAL qui lui sont associés, tire peu partie de ces disciplines. Nous proposons donc d'exploiter des méthodes du TAL dans le contexte de l'indexation et de la recherche d'images. Plus précisemment, nous nous intéressons à chacun des deux axes principaux de la recherche d'images : la recherche d'images par le contenu et la recherche sémantique d'images. Pour cela, il est nécessaire, dans un premier temps, de fixer précisément pour chacun de ces axes le cadre dans lequel il est possible d'avoir recours à des méthodes du TAL, ainsi que les familles de méthodes qui sont adaptées à chacun des cadres de travail ainsi définis. Ainsi, nous devons, pour appliquer des méthodes de TAL à la recherche d'images par le contenu, disposer d'une représentation des images similaire à celle employée pour les textes. Et, pour introduire des méthodes de TAL en recherche sémantique d'images, nous devons disposer, en plus d'un corpus d'images, de données textuelles suffisantes que nous exploiterons à l'aide d'outils du TAL.

Dans ce chapitre, nous donnons d'abord quelques éléments de TAL, ainsi qu'un aperçu des bases de recherche d'information textuelle, et situons les apports du TAL à celle-ci. Nous définissons ensuite les deux principaux cadres de travail que nous avons adoptés pour introduire des méthodes de TAL dans la recherche d'images, l'un pour la recherche d'images par le contenu, l'autre pour la recherche sémantique d'images. Enfin, nous présentons les 3 axes que nous avons choisi d'étudier et qui correspondent respectivement aux trois chapitres suivants de cet ouvrage.

## 2.1 Introduction au traitement automatique des langues

Le traitement automatique des langues se situe à l'intersection de la linguistique et de l'informatique. Son principal objet est la mise en place d'outils informatiques capables d'analyser de manière efficace et pertinente des contenus exprimés en langage naturel[1].

[1] Nous plaçons cette description dans un contexte de texte écrit, bien qu'il existe, bien sûr, des interactions entre TAL et étude des contenus parlés, dans le cadre de la reconnaissance de la parole notamment. Cet aspect particulier sera tout de même évoqué dans la section 2.1.3 page 52 sur les principales applications du TAL.

En effet, la représentation classique d'un texte par un ordinateur se limite à une séquence de codes numériques qui correspondent, par convention, à des caractères écrits (lettres, ponctuation...). Cette représentation du texte, qui ne connaît ni notion de phrase, ni même de mot, ne permet pas de prendre en compte l'extrême complexité du langage naturel, complexité qui s'exprime par des concepts comme la synonymie ou la polysémie. À titre d'exemple, le simple fait d'isoler les différents mots d'une phrase est un problème non trivial, sujet à plusieurs ambiguïtés : comment différencier automatiquement, par exemple, l'apostrophe de *l'université* de celle d'*aujourd'hui* ? Il est donc nécessaire d'apporter à tout programme ayant pour objectif d'analyser, d'interpréter ou de générer du contenu en langage naturel des connaissances linguistiques suffisantes pour manipuler ce contenu, pour dépasser la simple notion de chaîne de caractères. Ces connaissances peuvent se situer à différents niveaux d'analyse du contenu textuel, et être acquises selon différentes méthodes, plus ou moins automatiques.

### 2.1.1 Niveaux d'analyse linguistique

L'analyse des mots constituant un texte peut se faire à différents niveaux, selon que l'on s'intéresse au mot seul ou en relation avec d'autres mots, à son rôle grammatical ou sémantique. Chacun de ces niveaux correspond à des propriétés particulières des mots. Nous décrivons ici trois niveaux d'analyse linguistique qui sont exploités en TAL et dont les apports sont utilisés en recherche d'information.

#### 2.1.1.1 Analyse morphologique

L'analyse morphologique s'intéresse principalement à la variation de graphie des mots. En effet, un même mot, ayant un sens donné, peut être orthographié de différentes manières en fonction de son emploi dans la phrase (temps des verbes, accord en genre et nombre des adjectifs...), de même que des mots peuvent avoir des sens proches parce que formés à partir d'une même racine. L'analyse morphologique a pour objectif de permettre de rapprocher des occurrences de mots proches en faisant abstraction de ces variations. On distingue habituellement deux types de variations morphologiques distinctes :

- les variations flexionnelles : elles regroupent toutes les variations liées à la position du mot dans la phrase (personne, temps et mode pour les verbes ; genre et nombre pour les adjectifs ; nombre pour les noms). Les lemmatiseurs permettent de regrouper les variantes flexionnelles d'un mot sous une forme normalisée (masculin singulier pour les adjectifs, infinitif pour les verbes...) appelé lemme ;
- les variations dérivationnelles : elles désignent les variations obtenues lors de la construction d'un mot à partir d'un autre, par ajout de préfixes ou de suffixes (*reconstruire*, *construction*). Elles peuvent être prises en charge par un racineur (ou *stemmer*) qui regroupe les variantes dérivationnelles sous une forme unique, la racine, qui n'est pas nécessairement un mot existant, contrairement au lemme (par exemple, *constru* pourra être une racine de *reconstruire* et *construction*)[2].

Les deux outils présentés ici n'ont pas le même emploi, d'une part parce qu'ils ne prennent pas en charge les mêmes variations, mais également parce qu'ils n'ont pas les mêmes contraintes d'usage. Un lemmatiseur nécessite de connaître le rôle grammatical des mots dans la phrase, et de posséder un dictionnaire pour retrouver les lemmes ; ses performances sont donc sujettes aux erreurs d'étiquetage grammatical et de mots inconnus. Les racineurs ont un fonctionnement plus grossier, par suppression de suffixes (et préfixes

[2]En supprimant les variations dérivationnelles, le stemmeur supprime également les variations flexionnelles.

éventuellement) à partir d'un nombre souvent restreint de règles prédéfinies ; ils sont donc sujets aux erreurs d'approximation (sous-racinisation et sur-racinisation).

#### 2.1.1.2 Analyse syntaxique

L'analyse syntaxique s'intéresse au rôle du mot au sein de la phrase et aux relations existant entre les différents mots d'une phrase. On peut citer par exemple deux champs importants de l'analyse syntaxique :

- l'étiquetage grammatical (ou syntaxique) des mots qui consiste à attribuer à chaque mot d'une phrase la catégorie grammaticale (verbe, conjonction de coordination, sujet, complément d'objet direct...) qui lui correspond. Cet étiquetage est souvent couplé avec une analyse morphologique des mots sous forme d'une analyse morpho-syntaxique ;
- l'identification des collocations, qui désignent au sens large tout couple ou ensemble de mots qui apparaissent ensemble plus souvent que le hasard ne le voudrait, et ont un sens différent selon qu'on les considère ensemble ou séparément.

#### 2.1.1.3 Analyse sémantique

L'analyse sémantique s'intéresse au sens des mots, ainsi qu'aux relations existant entre les mots du point de vue de leur sens. Il peut d'agir d'étudier les différents sens des mots (polysémie), ou des relations d'équivalence (synonymie), hiérarchiques (hyperonymie, hyponymie)... Les connaissances et relations sémantiques peuvent se caractériser de différentes manières, par exemple en exploitant le contexte des mots (hypothèse de Harris [HGR+89]), ou en se basant sur leur organisation syntaxique [CSFB03].

### 2.1.2 Approches classiques en TAL

Il existe en TAL différentes manières d'acquérir des connaissances linguistiques sur le texte que l'on traite. Ces différentes approches peuvent généralement être appliquées à n'importe quelle tâche du TAL, quelque soit le niveau d'analyse auquel on se situe. On peut différencier trois approches principales, la dernière étant en fait une combinaison des deux premières approches.

#### 2.1.2.1 Approches symboliques

Les approches symboliques reposent uniquement sur des connaissances *a priori* sur la langue, exprimées sous formes de règles (*si un verbe se termine en -ez, il est à la deuxième personne du pluriel*) ou de patrons syntaxiques (le patron *[Nom à Verbe]* permet de retrouver *pince à épiler*, *table à langer*...), d'ontologies... L'avantage d'avoir recours à ce type de méthodes est qu'elles fournissent des résultats interprétables (puisqu'issus de connaissances linguistiques) et permettent de prendre en compte les exceptions et les cas rares, pourvu qu'ils fassent partie des connaissances de base. L'acquisition manuelle de connaissances linguistiques symboliques peut s'avérer très coûteuse et présente des difficultés de mise à jour de ces connaissances (évolutions de la langue, spécialisation à certains domaines...), c'est pourquoi certains travaux s'intéressent au moyen d'acquérir autoamtiquement de telles connaissances, en utilisant par exmeple la méthodologie proposée par M. Hearst [Hea92].

#### 2.1.2.2 Approches statistiques

Les approches statistiques, quant à elles, reposent sur l'utilisation de critères numériques (fréquence des mots, cooccurrences entre mots...) pour acquérir des connaissances linguistiques à partir d'un corpus de textes donné. En analyse syntaxique, ce genre d'approches peut, par exemple, être employé pour découvrir automatiquement des collocations, en utilisant des critères de cooccurrence entre mots (test du $\chi^2$, information mutuelle...) qui permettent de détecter des ensembles de mots apparaissant ensemble de manière significative. En analyse sémantique, les approches statistiques peuvent aussi être employées, selon l'hypothèse de Harris qui postule que des mots sémantiquement proches ont généralement des contextes similaires ; il est donc possible de trouver des critères numériques qui permettent de découvrir des relations sémantiques entre mots ou de constituer des classes sémantiques. Contrairement aux approches symboliques, les approches statistiques ont l'avantage de ne nécessiter que peu de connaissances linguistiques *a priori*, ce qui leur confère une grande généricité (portabilité vers d'autres domaines, d'autres langues...). En revanche, ces approches ne sont adaptées qu'à la découverte de connaissances linguistiques suffisamment fréquentes pour être détectées : elles ne fonctionnent pas pour les exceptions et les cas qui ne se produisent pas suffisamment dans le corpus utilisé. De plus, elles peuvent fournir des résultats peu intuitifs ou difficiles à interpréter, contrairement aux approches symboliques qui ne fournissent que des résultats linguistiquement cohérents.

#### 2.1.2.3 Approches mixtes

Les approches mixtes combinent une approche symbolique et une approche statistique. Cette combinaison peut être réalisée en utilisant successivement chacune des deux approches. Le système XTRACT de F. Smadja [Sma93], par exemple, détecte des collocations en utilisant des critères purement statistiques, puis en filtrant les résultats non pertinents à l'aide de patrons syntaxiques définis manuellement. Le système ACABIT proposé par B. Daille [Dai93], à l'inverse, réalise cette même tâche en utilisant une détection symbolique dont les résultats sont filtrés sur des critères numériques. L'intérêt des approches mixtes est qu'elles profitent de la grande automaticité des approches statistiques, réduisant ainsi l'intervention humaine nécessaire, tout en fournissant des résultats interprétables d'un point de vue linguistique.

### 2.1.3 Quelques applications classiques du TAL

Le TAL est potentiellement présent dans toute application en lien avec le langage naturel, écrit ou parlé. Voici quelques applications classiques dans lesquelles le TAL joue un rôle majeur. Ces applications constituent toutes des domaines de recherche à part entière, souvent connexes de la recherche d'information textuelle ou multimédia.

**Systèmes de questions-réponses** : les systèmes de questions-réponses ont le même objectif que les systèmes de recherche d'information : permettre à leurs utilisateurs d'accéder à l'information contenu dans un corpus de documents. La différence se situe dans la forme que prennent la requête et la réponse à cette requête : alors qu'un système de recherche d'information accepte des requêtes de forme quelconque (mots-clefs ou phrase), et fournit en réponse à cette requête une liste de documents considérés comme pertinents, un système de question-réponse n'accepte que des requêtes sous forme de questions, auxquelles il fournit une réponse unique, adaptée à la question posée. Ces systèmes évitent ainsi à l'utilisateur la tâche potentiellement fastidieuse de localisation de l'information désirée au

sein des documents réponses. Ils sont donc intimement liés au TAL, car ils ont besoin d'analyser le contenu de la réponse pour en saisir le sens, et de formuler une réponse en langage naturel qui soit correctement formulée.

**Traduction automatique** : l'objectif de cette discipline est de produire des systèmes capables de traduire automatiquement des textes d'une langue donnée vers une autre langue. Cela nécessite d'avoir des connaissances à la fois sur les langues source et cible de la traduction, pour comprendre le texte en entrée et formuler un texte correct en sortie, et sur les relations existant entre ces langues, pour effectuer la traduction proprement dite. Le TAL constitue donc un élément central de tels systèmes. Traduction automatique et recherche d'information peuvent se rejoindre dans des systèmes de recherche d'information translingue, dont l'objectif est de fournir, pour une requête exprimée dans une langue donnée, des résultats exprimés dans des langues différentes.

**Reconnaissance et synthèse de la parole** : les systèmes manipulant des données parlées peuvent également tirer parti du TAL. Les systèmes de reconnaissance de la parole peuvent en effet exploiter des connaissances linguistiques pour choisir, parmi leurs hypothèses de transcription, celle qui est la plus correcte vis-à-vis de la langue. Les systèmes de synthèse de la parole ont également besoin de ce type de connaissances pour analyser le texte en entrée et générer la suite de phonèmes adaptée.

## 2.2 Recherche d'information textuelle

Nous présentons dans cette section les principes de base de la recherche d'information textuelle, car c'est dans un cadre similaire à celui-ci que l'on cherche à se placer pour pouvoir ensuite appliquer des techniques de TAL à la description textuelle des images. Nous donnons également un aperçu des apports du TAL à la recherche d'information textuelle.

### 2.2.1 Principe

Le principe de base de la recherche d'information textuelle est identique à celui de la recherche d'images présenté dans la figure 1.1 page 14 : les documents (des textes) et les requêtes (formulées par l'utilisateur sous forme de mots-clefs ou en langage naturel) sont représentées sous forme de descripteurs, puis une mesure de similarité est utilisée pour fournir à l'utilisateur une liste de résultats, si possible par ordre de pertinence.

### 2.2.2 Description des textes

Les textes sont généralement décrits comme des ensembles de termes d'indexation. Ces termes d'indexation peuvent être de deux types :

- des termes simples, c'est-à-dire des mots seuls (*trompette*, *éléphant*, *manger*...). Ces mots sont isolés par une phase de segmentation en fonction de séparateurs spécifiques (espaces, ponctuation). Comme nous l'avons déjà évoqué, cette segmentation n'est pas forcément triviale ;
- des termes complexes, c'est-à-dire des groupes de mots (*canne à pêche*, *pince à épiler*...). Ces termes complexes sont extraits du corpus segmenté à l'aide d'outils du TAL.

Le type de terme choisi joue un rôle important dans les performances des systèmes de recherche d'information. Les termes simples, génériques mais peu précis, favorisent le rappel. Á l'inverse, les termes complexes, moins génériques, favorisent la précision des systèmes.

Il est de plus possible de sélectionner, parmi l'ensemble des termes d'indexation extraits du corpus, ceux qui sont les plus pertinents pour décrire les documents. Ainsi, les mots grammaticaux (articles, pronoms...) sont généralement éliminés car ils ne portent pas d'information sur le contenu sémantique des documents. Ces mots, dits vides, sont regroupés dans des listes spécifiques nommées *stop-lists*. Il existe deux approches principales pour déterminer quels sont les mots vides :

- une approche statistique, proposée par Luhn [Luh58], qui part du principe que les termes qui sont très fréquents ou très rares au sein d'un document décrivent mal ce document. Au niveau de chaque document, les termes les plus fréquents et les moins fréquents sont donc éliminés (cette approche est décrite plus en détails dans la section 3.2.1 page 68 ;
- une approche symbolique, qui consiste à énumérer manuellement les termes de la langue qu'il convient d'éliminer. Notons que, même si cette approche repose uniquement sur une intervention manuelle, elle est la plus utilisée car les listes, relativement courtes (quelques centaines de termes), n'ont besoin de n'être définies qu'une fois pour une langue donnée, et il en existe déjà pour de nombreuses langues. De plus, cette méthode permet d'obtenir des résultats beaucoup plus pertinents que l'approche statistique.

### 2.2.3 Structure d'index

Le recherche d'information textuelle utilise une structure d'index très efficace : les fichiers inversés. Un fichier inversé contient, pour chaque terme d'indexation, la liste des documents contenant ce terme. Il permet donc d'obtenir directement, pour une requête donnée, la liste des documents contenant au moins un terme de la requête, qui sont les documents susceptibles d'être pertinents. En effet, les modèles classiques de recherche d'information se basent sur les termes communs aux documents et à la requête pour calculer les scores de similarité et ordonner les résultats, il est donc inutile de prendre en compte les documents ne partageant aucun terme d'indexation avec la requête, car ils se voient toujours attribuer le score minimal. Les requêtes étant souvent très courtes (2,4 mots en moyenne sur le web [Sav08], au plus quelques dizaines si la requête est exprimée sous forme de phrase(s)), cette structure d'index permet d'éliminer rapidement une très grande partie des documents, et donc d'obtenir des temps de réponse très courts.

### 2.2.4 Modèles de recherche d'information

Les modèles de recherche d'information définissent précisément un type de descripteur à partir de l'ensemble des termes utilisés pour décrire les documents, et une manière de comparer les descripteurs pour obtenir la liste des résultats les plus similaires à la requête. Nous citons juste dans cette partie les modèles les plus classiques, ceux dont nous nous servirons dans nos travaux seront décrits plus en détail par la suite.

#### 2.2.4.1 Modèles ensemblistes

Les modèles ensemblistes se basent directement sur l'ensemble des termes décrivant les documents. L'utilisation d'un descripteur sous forme d'ensemble implique que seule

l'apparition des termes dans les documents soit prise en compte, toute information de fréquence sur les termes d'indexation est donc ignorée. Les scores de similarité utilisés pour classer des documents sont des mesures de similarité entre ensembles (indice de Dice, indice du Jacquart...). Ces scores sont néanmoins monotones les uns par rapport aux autres car ils se basent tous sur la même grandeur de base, le nombre de termes en commun entre la requête et le document ; ils fournissent donc tous des classements similaires.

#### 2.2.4.2 Modèles booléens

Le modèle booléen considère les documents comme des conjonctions logiques de termes. Les requêtes sont quant à elles représentées à l'aide des trois opérateurs logiques classiques, la conjonction, la disjonction et la négation. Ce modèle offre donc une grande expressivité dans l'expression des requêtes, car il est possible par exemple d'imposer la présence de deux termes simultanément, ou encore de refuser la présence d'un terme donné dans les documents réponses. La liste des résultats est obtenue par des opérations ensemblistes (intersection, union, différence) entre les ensembles de documents correspondant à chaque terme présent dans la requête. Il n'est donc pas possible avec ce modèle d'obtenir des scores de pertinence, et donc de classer les documents réponses.

Une extension possible du modèle booléen standard est le modèle booléen flou. Il ne représente plus les termes en fonction de leur présence absence seule dans les documents, mais en termes de degré d'appartenance (tel que défini en logique floue) des termes aux documents. Ceci permet à la fois d'assigner aux termes des degrés d'appartenance caractérisant leur importance au sein des documents (un terme fréquent, par exemple, décrit *a priori* mieux un document qu'un terme n'apparaissant qu'une seule fois), mais également de calculer des scores de similarité entre documents et requêtes, à l'aide des opérateurs classiques de logique floue.

#### 2.2.4.3 Modèles vectoriels

Le modèle vectoriel de Salton [SWY75] est le modèle le plus employé en recherche d'information. Il représente les documents et requêtes comme des vecteurs de fréquence des termes d'indexation. Ces vecteurs sont de très grande dimension (autant de dimensions que de termes d'indexation dans le corpus), et très creux (car peu de termes distincts, en regard du nombre de termes possibles, sont présents dans chaque document). Les vecteurs représentant documents et requêtes sont ensuite comparés à l'aide de mesures de similarité algébriques (distance de Minkowski, cosinus entre vecteurs...) pour obtenir des scores de similarité. L'intérêt de ce modèle est de pouvoir prendre en compte l'importance des termes dans les documents, puisqu'à chaque terme correspond une valeur donnée dans le vecteur. Cette valeur peut être la fréquence du terme dans le document, mais aussi une valeur reflétant son poids dans le document. Ces pondérations des termes ont été très étudiées, et de nombreux schémas de pondération existent, exploitant la fréquence des termes dans les documents, mais aussi leur répartition dans le corpus et la taille des documents. En revanche, ce modèle ne permet pas d'imposer des conjonctions entre termes ou des négations de termes comme le modèle booléen.

#### 2.2.4.4 Modèles probabilistes

Les modèles probabilistes expriment l'importance des termes dans les documents en termes de probabilité d'appartenance du terme au document, et la notion de similarité des documents à une requête en termes de probabilité de pertinence du document à cette

requête. L'enjeu de ces modèles se situe dans la manière dont sont estimées ces probabilités. Les modèles probabilistes fournissent souvent de très bons résultats ; le modèle de recherche d'information de référence actuellement est un modèle probabiliste, le modèle BM25 de Robertson [RWB+96]. De la même manière que le modèle vectoriel, les modèles probabilistes permettent de prendre en compte l'importance des mots et de classer les documents par pertinence, mais ne permettent pas d'effectuer des requêtes complexes utilisant des conjonctions et des négations de mots-clefs.

#### 2.2.4.5 Remarques sur les modèles de recherche d'information

Les deux questions majeures posées par ces modèles de recherche d'information sont :
- comment attribuer automatiquement aux termes des poids reflétant leur importance dans les documents ?
- comment obtenir une mesure de similarité reflétant la notion de pertinence ?

Ces deux questions sont intimement liées puisque l'amélioration des descripteurs de documents permet naturellement d'obtenir des scores de similarité plus fidèles au contenu des documents. C'est la première de ces deux questions qui a été la plus discutée dans la littérature, plus particulièrement dans le cas du modèle vectoriel et dans le cas des modèles probabilistes.

De plus, les modèles de recherche d'information présentés ici fonctionnent tous sous l'hypothèse d'indépendance des termes, c'est-à-dire qu'ils considèrent que les termes apparaissent dans les documents indépendamment les uns des autres. Cette hypothèse montre une certaine efficacité en pratique mais n'est en réalité pas vérifiée car certains termes utilisés ensemble prennent un sens différent de celui qu'ils ont lorsqu'ils sont utilisés indépendamment : par exemple, l'expression *tailler une bavette* n'a rien à voir avec le champ sémantique de la boucherie, auquel appartiennent pourtant les termes *tailler* et *bavette* pris seuls. Il existe également d'autres relations de nature sémantique entre termes : un terme est par exemple équivalent à ses synonymes, leurs occurrences dans un document dépendent donc les unes des autres, et l'importance du concept qu'ils représentent dans le document dépend des tous ces termes à la fois. Un des enjeux de la recherche d'information est donc de dépasser cette hypothèse d'indépendance des termes. Plusieurs modèles développés dans ce sens ont été proposés, dont le modèle d'analyse de la sémantique latente (*Latent Semantic Analysis*, *LSA*) [DDH90], le modèle vectoriel généralisé (*Generalized Vector Space Model*, *GVSM*) [WZW85] ou les modèles de langues (*Language Models*, *LM*) [PC98].

### 2.2.5 TAL et RI

Le TAL a bien sûr été mis à contribution pour pallier les limites des modèles de recherche d'information. Les techniques de TAL servent à acquérir des connaissances linguistiques qui peuvent ensuite être intégrées aux systèmes de recherche d'information de deux manières distinctes :
- lors de la phase d'indexation des documents et des requêtes : les connaissances sont intégrées directement aux descripteurs de documents et de requêtes (par exemple, représenter les documents comme des ensembles de concepts plutôt que de termes d'indexation pour intégrer des relations sémantiques au système). Cette nouvelle description des documents peut ensuite être utilisée dans le cadre des modèles de recherche d'information habituels ;
- lors de la phase de recherche des résultats : on utilise les connaissances linguistiques lors de la phase de sélection et de classement des documents résultats, en ajoutant à

la requête des termes qui lui sont liés d'après les connaissances linguistiques dont on dispose (on parle d'extension de requêtes).

Il est possible d'intégrer ainsi des connaissances linguistiques correspondant à n'importe quel niveau d'analyse (morphologique, syntaxique ou sémantique). Voici quelques exemples de connaissances linguistiques couramment employées :

- niveau morphologique : il est possible d'utiliser une analyse morphologique des termes pour s'abstraire des variations flexionnelles et/ou dérivationnelles des termes. L'approche courante consiste à utiliser un racineur pour remplacer les termes de l'index par leur racine. Ceci permet de gagner en efficacité (le nombre de termes d'indexation possibles diminue), mais également de gagner en rappel en mettant en relation des occurrences d'un même mot qui étaient à l'origine considérées comme des termes d'indexation distincts ;
- niveau syntaxique : l'intérêt principal de ce niveau d'analyse est qu'il permet l'acquisition de termes complexes qui peuvent être employés à la place des termes simples dans l'index pour améliorer la précision des systèmes ;
- niveau sémantique : les connaissances sémantiques peuvent être employées pour désambiguïser les termes polysémiques, identifier des concepts au sein des documents ou étendre les requêtes avec des termes sémantiquement liés aux termes de la requête.

F. Moreau a étudié les corrélations entre les différentes connaissances linguistiques que l'on peut ajouter aux systèmes de recherche d'information et a montré que l'ajout de telles connaissances permet effectivement d'améliorer les performances des systèmes [Mor06].

## 2.3 Des mots aux images

Les méthodes de recherche d'information textuelle et de TAL peuvent être utilisées pour l'indexation d'images à condition de se placer dans un cadre de travail propice à ces méthodes. Nous présentons ici les deux deux cadres distincts que nous utilisons et qui permettent, pour l'un, d'utiliser des méthodes de TAL pour la recherche d'images par le contenu, et, pour l'autre, d'utiliser de telles méthodes dans un contexte de recherche sémantique d'images.

### 2.3.1 TAL et recherche d'images par le contenu

Pour utiliser des méthodes de TAL dans un cadre de recherche d'images par le contenu, il est nécessaire de disposer d'une description des images qui ne soit pas numérique, comme le sont les vecteurs qui servent généralement de descripteurs d'images, mais symbolique, le contenu textuel étant lui-même un ensemble de symboles. Sivic et Zisserman [SZ03] ont proposé une méthode pour décrire les images de manière symbolique, comme un ensemble de régions d'image élémentaires qu'ils ont nommées *mots visuels* (*visual words*), en référence aux mots, composants élémentaires des textes.

Cette manière de décrire les images entre dans le cadre de la description et comparaison d'images par quantification de descripteurs locaux, que nous avons déjà évoqué dans la section 1.3.3.3 page 34. Bien qu'ils n'aient pas été les seuls à exploiter cette approche, Sivic et Zisserman ont été les premiers (avec Zhu *et al.* [ZRZ02]) à mettre en avant ses points communs avec la recherche d'information textuelle et à lui appliquer les principes de cette dernière.

#### 2.3.1.1 Décrire les images comme des textes : les mots visuels

Le système proposé par Sivic et Zisserman est illustré par la figure 2.1. Il se compose de deux étapes, la création d'un vocabulaire visuel, puis la description des images à partir des mots visuels de ce vocabulaire.

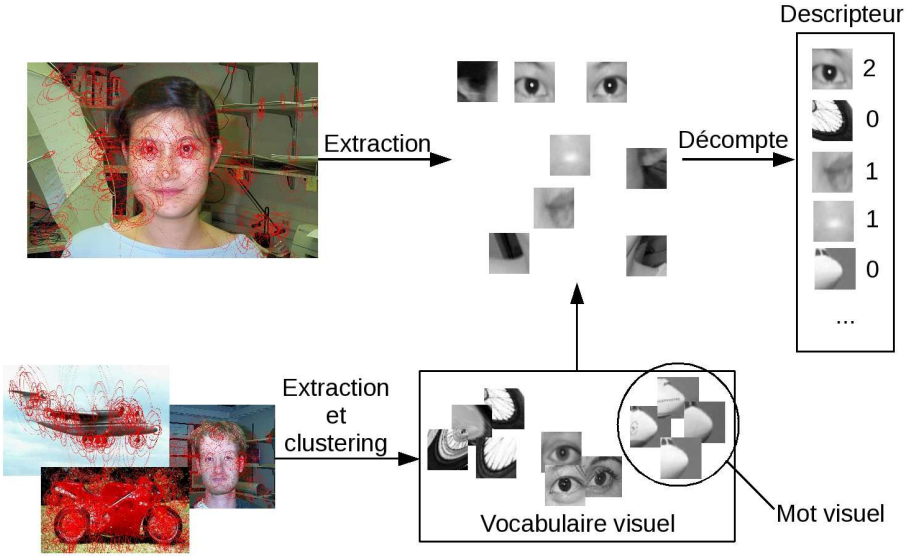



FIG. 2.1 – Processus de construction d'un vocabulaire visuel et de description d'une image comme un ensemble de mots visuels.

**Création du vocabulaire visuel** : le vocabulaire visuel est l'ensemble des mots visuels utilisés pour décrire les images, de même que le vocabulaire d'un corpus rassemble l'ensemble des mots contenus dans les textes de ce corpus. La différence majeure se situe dans le fait que le vocabulaire d'un corpus de textes est issu directement du contenu des textes[3], alors qu'un vocabulaire visuel est construit de manière complètement artificielle. Le processus de création d'un vocabulaire visuel est le suivant :

1. détection de régions d'intérêt dans un échantillon d'images (ou échantillonnage de régions d'intérêt détectées dans un corpus complet). Description de ces régions d'intérêt par des descripteurs numériques (voir section 1.3.3.2 page 33 pour des détails sur les descripteurs locaux) ;
2. *clustering* des descripteurs locaux en $k$ clusters. Chaque cluster correspond à un mot visuel donné (voir figure 2.1). La taille du vocabulaire, $k$, est fixée arbitrairement, à moins que l'algorithme de clustering utilisé ne permette de déterminer automatiquement une valeur optimale pour $k$[4].

**Description des images** : chaque image peut ensuite être décrite par l'ensemble des mots visuel qui y apparaissent, de la manière suivante :

1. des régions d'intérêt sont détectées dans l'image, puis décrites à l'aide de descripteurs locaux ;

[3] À quelques prétraitements près, comme l'utilisation d'un racineur ou la détection de termes complexes remplaçant les termes de base.

[4] À notre connaissance, tous les travaux exploitant des vocabulaires visuels fixent $k$ arbitrairement. Les algorithmes capables de choisir le nombre de clusters optimal sont en général très coûteux et ne garantissent pas de passer à l'échelle de millions de descripteurs à grande dimension.

2. chacun de ces descripteurs est associé au mot visuel du vocabulaire dont il est le plus proche (en termes de distance au centroïde du cluster). Chaque descripteur extrait de l'image correspond ainsi à une occurrence d'un mot visuel donné dans l'image. Cette dernière est donc représentée par un ensemble de mots visuels ;
3. généralement, on construit un vecteur de fréquences des mots visuels dans l'image, pour se placer dans le cadre classique du modèle vectoriel en recherche d'information. On parle également souvent de sac de mots visuels (*bag of visual words*), par analogie avec l'expression sacs de mots utilisée en recherche d'information textuelle.

La représentation en sacs de mots visuels a deux avantages majeurs :
- elle se base sur des descripteurs locaux qui sont très discriminants car ils décrivent des régions d'intérêt, ce qui permet de prendre en compte des informations locales (contrairement aux descripteurs globaux), et fournissent une représentation très précise de ces régions ;
- elle utilise une phase de quantification de ces descripteurs locaux qui permet d'obtenir plus de robustesse et des facultés de généralisation.

#### 2.3.1.2 Quelles méthodes issues du texte pour l'indexation à base de mots visuels ?

Bien que la forme globale de la description des images par mots visuels soit calquée sur la représentation symbolique des textes, tous les outils issus du TAL ne sont pas adaptés aux mots visuels. En effet, la plupart de ces outils se basent sur une approche symbolique ou mixte, qui utilise un grand nombre de connaissances linguistiques *a priori* définies manuellement. Dans le cas des images, nous ne disposons d'aucune connaissance de ce type sur les mots visuels que nous manipulons, ou leur organisation dans l'image. Il faut donc se restreindre aux approches purement statistiques que le TAL nous offre. De plus, les mots visuels ne s'organisent pas dans le document de la même manière que les mots textuels : alors que ces derniers forment des séquences (phrases), les mots visuels ont une disposition géométrique spécifique dans les deux dimensions du plan que constitue l'image. Cet aspect doit également être pris en compte pour utiliser des méthodes de TAL dans ce contexte.

Quelques outils de la recherche d'information textuelle ou du TAL sont utilisés avec les mots visuels, notamment :
- les fichiers inversés : les fichiers inversés pour les mots visuels ont été introduits par Sivic et Zisserman dans le cadre de la recherche d'objets dans des vidéos. Certaines limites de cette structure d'indexation sont ensuite apparues, notamment parce que les requêtes visuelles (images) contiennent beaucoup plus de mots que les requêtes employées en texte. Certains travaux proposent des améliorations des fichiers inversés prenant en compte ces limitations [JHS07] ;
- l'extraction de termes complexes : certains auteurs ont proposé de regrouper des mots visuels en fonction des relations de cooccurrence qu'ils entretiennent, pour obtenir des *syntagmes visuels* [ZWG06, YWY07]. Cette idée est similaire à la détection de collocations, problématique classique du TAL ;
- les pondérations : la pondération tf. idf a été utilisée dès les premiers travaux de Sivic et Zisserman sur les mots visuels [SZ03], puis repris dans la grande majorité des travaux qui ont suivi. Néanmoins, il n'y a jamais eu d'étude précise attestant de l'utilité de cette pondération dans le cas général.

### 2.3.2 TAL et recherche sémantique d'images

La recherche sémantique d'images a pour objectif de fournir une description plus riche des images que les descripteurs de bas-niveau utilisés en recherche par le contenu, que l'on suppose généralement être insuffisants pour capter le sens contenu dans les images (voir section 1.2.3 page 22). Elle s'appuie pour cela sur des informations textuelles, qui pourront être associées automatiquement aux images (voir annotation d'images, section 1.4.3 page 38) ou utilisées en complément des descripteurs de bas-niveau (voir fusion texte-image, section 1.4.4 page 45). L'ajout de ressources textuelles à la description des images est naturellement propice à l'utilisation de méthodes du traitement automatique des langues pour améliorer la qualité de la description textuelle des images.

#### 2.3.2.1 Corpus bimodaux existants

Il existe plusieurs corpus mêlant de l'information textuelle à des images dans les travaux existants. Ces corpus peuvent se classer en différentes catégories en fonction de la forme que prend l'information textuelle qu'ils contiennent.

**Mots-clefs :** il existe de nombreux corpus croisant images et mots-clefs d'annotation. Le plus connu d'entre eux est certainement le corpus d'images COREL, que nous avons déjà évoqué en section 1.4.2 page 38. Bien que permettant l'usage de ressources linguistiques telles que les ontologies, le fait de ne disposer que de mots-clefs isolés rend ces corpus peu propices à l'utilisation de méthodes de TAL.

**Légendes d'images :** plusieurs corpus proposent des images accompagnées par de courtes légendes. C'est par exemple le cas du corpus IAPR TC-12 [GCMD06] ou des corpus de presse utilisés par différents auteurs [BBE+04, DM07, JM04]. Nous pouvons y ajouter le corpus de scènes de crimes utilisé par Pastra *et al.* car, bien que contenant des textes plus longs que les autres, ces textes se limitent également à une description du contenu de l'image [PSW03]. Ces corpus proposent un problème d'annotation simplifié du fait que le contenu du texte accompagnant les images se limite à leur description.

**Textes illustrés :** certains corpus, plus rares, contiennent des textes complets que les images viennent illustrer. C'est le cas, par exemple, du corpus d'article de presse utilisé par Jiang *at al.* [JT09], et c'était le cas du corpus de la campagne d'évaluation ImagEval qui n'est plus disponible actuellement [TG07]. Ces corpus sont néanmoins de taille restreinte (moins de 1000 documents pour les deux que nous avons cités). Ce type de corpus propose à la fois un problème d'annotation réaliste, les images étant souvent insérées dans des textes (articles de presse, articles scientifiques, sites web...), et un cadre propice à l'usage de méthodes de TAL.

Nous voyons donc qu'il existe de nombreux jeux de données croisant de l'information textuelle et des images, malheureusement très rares sont ceux qui proposent un cadre qui soit à la fois réaliste, d'échelle convenable et propice à l'usage du TAL.

#### 2.3.2.2 Utilisation d'outils du TAL en indexation sémantique d'images

Certains outils du traitement automatique des langues ont déjà été utilisés dans un cadre d'indexation, de recherche ou d'annotation d'images. En voici les principaux.

**Ontologies :** les ontologies proposent des ressources linguistiques organisées en fonction des relations qu'elles entretiennent entre elles (hyperonymie, hyponymie, synonymie...). Parmi les ontologies[5] utilisées en recherche d'images, la plus célèbre est sans conteste Wordnet [Fel98], qui a une vocation généraliste, mais on trouve aussi des ontologies spécialisées, comme UMLS pour la médecine [LLCL07] ou AAT (the Art and Architecture Thesaurus) pour l'art [HSWW03, KSA+08]. Ces ontologies ont été utilisées dans différents cadres : prise en compte de relations linguistiques entre mots-clefs [PSW03, LCC+08], extension de requêtes [PG08] ou encore sélection de termes pertinents à l'annotation d'images au sein de la légende accompagnant chacune [HSWW03].

**Analyseurs morpho-syntaxiques :** les analyseurs morpho-syntaxiques ont été utilisés dans certains travaux d'annotation d'images [PSW03, KSA+08]. Leur rôle essentiel dans ce cadre a été de permettre la sélection des termes d'annotation en fonction de leur catégorie grammaticale, les mots appartenant à certaines de ces catégories (noms, adjectifs) étant plus pertinents pour décrire du contenu visuel que ceux d'autres catégories (adverbes, pronoms).

**Détection d'entités nommées :** la détection d'entités nommées a été utilisée dans certains travaux pour associer des noms de personnes et des visages [BBE+04, PCL08]. On notera toutefois que dans ces travaux, les entités nommées ne sont pas extraites des textes de manière générique mais à l'aide de patrons spécifiques à la syntaxe particulière des phrases des légendes décrivant les images.

## 2.4 Conclusion et axes de travail

Les travaux présentés dans ce chapitre mettent en avant la possibilité d'utiliser des techniques de TAL pour l'indexation d'images. En particulier, nous avons décrit un cadre de travail précis pour chacun des deux grands axes de la recherche d'images que nous avions identifiés au chapitre 1 : la recherche par le contenu et la recherche sémantique. Les chapitres suivants décrivent nos contributions à chacun de ces axes, dans les cadres de travail fixés ici.

**TAL et recherche d'images par le contenu :** nous explorons certains des apports possibles des travaux en recherche d'information textuelle aux systèmes de recherche d'images basés sur le formalisme des mots visuels. Nous avons choisi deux axes qui sont des problématiques phares pour la communauté du texte :

1. le choix des termes d'indexation pertinents (chapitre 3) : nous proposons une étude de l'impact des méthodes classiques de sélection des termes (*stop-list*, pondérations) sur les systèmes de recherche d'images utilisant les mots visuels ;
2. le dépassement de l'hypothèse d'indépendance des termes (chapitre 4) : nous avons donné en section 2.3.1.2 les raisons pour lesquelles cette hypothèse semble particulièrement inadaptée au cadre des mots visuels. Nous proposons donc de prendre en compte les relations existant entre les mots visuels et proposons pour cela d'utiliser les modèles de langues, outil classique du traitement automatique des langues qui soulève depuis plusieurs années un intérêt croissant dans la communauté de la recherche d'information.

[5] Il s'agit plus souvent de thésaurus, mais ils sont considérés comme des ontologies dans la communauté de la recherche d'images.

**TAL et indexation sémantique d'images :** nous nous interrogeons sur la pertinence des descripteurs visuels courants pour des applications réelles, puis nous utilisons des outils du TAL pour tirer parti de l'information textuelle accompagnant les images afin d'en extraire des annotations pertinentes. Nous utilisons pour cela un corpus d'articles de presse qui nous place dans un cadre de travail propice à l'utilisation de méthodes de TAL et réaliste en terme d'application (chapitre 5).

# Chapitre 3

# Mesurer la pertinence des mots visuels

Dans ce chapitre, nous nous intéressons au premier axe de notre étude sur l'utilisation de méthodes de TAL pour la recherche d'images à partir de mots visuels : comment identifier les mots visuels les plus pertinents pour décrire les images ? Nous présentons d'abord quelques différences fondamentales existant entre les mots visuels et les mots textuels, qui expliquent que les propriétés statistiques de ces deux types de mots ne soient pas nécessairement les mêmes. Nous proposons ensuite une méthode de constitution de *stop-lists* pour la recherche d'images que nous comparons à la méthode classique basée sur la fréquence employée par Sivic et Zisserman [SZ03]. Nous explorons ensuite l'usage des pondérations issues de la recherche d'information textuelle dans le cadre des mots visuels, et proposons de nouvelles pondérations qui nous semblent adaptées à notre problème. Nous décrivons ensuite les expérimentations que nous avons menées pour valider l'usage des *stop-lists* et des pondérations présentées, ainsi que des expérimentations sur les distances de Minkowski et la relation qu'elles entretiennent avec les pondérations. Enfin, nous situons nos contributions par rapport aux travaux de la littérature qui leur sont proches, puis nous concluons.

## 3.1 Mots visuels et mots textuels : différences fondamentales

Bien que les représentations en sacs de mots visuels se veuillent inspirées de la représentation classique des documents textuels (un ensemble de mots décrit un document), il existe de nombreuses différences entre ces deux représentations, différences dont il faut prendre compte pour adapter des techniques issues du traitement des documents textuels au cas des images. Nous détaillons dans cette section les différences entre ces deux représentations qui nous semblent les plus fondamentales.

### 3.1.1 Origine du vocabulaire

Une première différence notable est la manière dont le vocabulaire utilisé pour décrire les documents est obtenu. Dans le cas du texte, les documents étant exprimés en langage naturel, le vocabulaire est imposé naturellement par les mots existant dans la langue ou les langues utilisées. De plus, on dispose de connaissances linguistiques *a priori*, apportées par l'homme, et qui permettent d'effectuer des traitements efficaces sur ce vocabulaire, comme la racinisation ou l'utilisation de listes de mots vides (*un*, *une*, *celui*, *elle*...). À l'inverse, un vocabulaire visuel est créé artificiellement à partir des images brutes. Ce processus fait

intervenir de nombreux paramètres, il est donc possible de produire de nombreux vocabulaires différents à partir d'un même ensemble d'images. Voici les principaux paramètres :

**Le détecteur de régions d'intérêt** Le détecteur de régions d'intérêt utilisé a une influence majeure sur le vocabulaire puisqu'il détermine quelles régions d'images seront utilisées pour construire le vocabulaire. Comme indiqué en section 1.3.3.1, il existe différents détecteurs qui vont produire différents types de régions, régions homogènes ou hétérogènes selon les cas, et donc produire différents types de vocabulaires. Il est de plus possible de combiner différents détecteurs, comme le font Sivic et Zissermann, qui combinent un détecteur *Harris-Affine* et un détecteur *MSER* [SZ03].

**Descripteur de régions** Le type de descripteur utilisé a également une influence sur le vocabulaire, car la similarité entre les régions d'intérêt qui seront regroupées lors de la phase de *clustering* dépendra principalement de la qualité des descripteurs employés (expressivité et robustesse aux transformations géométriques et de luminosité).

**Algorithme de clustering** L'algorithme de clustering employé est, après le descripteur de régions d'intérêt, le second point clé de la constitution de mots visuels homogènes, car c'est lui qui détermine comment seront regroupés les descripteurs formant un même mot visuel. C'est d'ailleurs probablement le paramètre des vocabulaires visuels qui a reçu le plus d'attention [LJ09, NS06, PCI+07], en raison de la difficulté à trouver un algorithme qui soit à la fois précis et rapide.

**Taille du vocabulaire** La taille du vocabulaire, enfin, a une importance primordiale sur la qualité de celui-ci. Si le nombre de mots visuels est trop restreint, il y a un risque que chaque mot visuel n'englobe des descripteurs trop différents, et donc que le vocabulaire obtenu ne soit pas suffisamment discriminatif pour séparer les images pertinentes des images non-pertinentes. *A contrario*, si le vocabulaire est trop grand, le système de recherche perdra en robustesse et son coût calculatoire pourra devenir intenable. Le problème principal ici est que la plupart des algorithme de clustering, et en particulier les algorithmes les plus basiques, qui sont les plus utilisés dans ce contexte en raison de leur coût calculatoire limité, ne déterminent pas automatiquement le nombre de *clusters* optimal en fonction des données, nombre qui doit donc être fixé manuellement, par essais successifs. Néanmoins, il existe une tendance lourde dans les tâches de recherche d'images pour l'utilisation de vocabulaires de très grande taille (jusqu'à un million de mots visuels), qui donnent généralement les meilleurs résultats [NS06, PCI+07].

### 3.1.2 Sens des mots visuels

Les mots visuels et les mots textuels ont des sens très différents. Alors qu'un mot textuel ne correspond généralement qu'à un concept donné (par exemple, *maison*, *ordinateur* ou *démocratie*), un mot visuel ne correspond qu'à une partie d'un objet, et potentiellement à différentes parties d'objets complètement différents. Plusieurs mots visuels sont donc nécessaires pour décrire un objet. Cette différence est essentielle car les modèles classiques de recherche d'information reposent sur l'hypothèse d'indépendance des mots (*word independance assumption*), c'est-à-dire qu'ils considèrent que les mots apparaissent dans un document indépendamment les uns des autres. Cette hypothèse est raisonnable dans le cas des documents textuels, quoiqu'elle se révèle incorrecte dans certains cas (par exemple, *Maison Blanche* a un sens différent des mots *maison* et *blanche* pris séparément). Dans

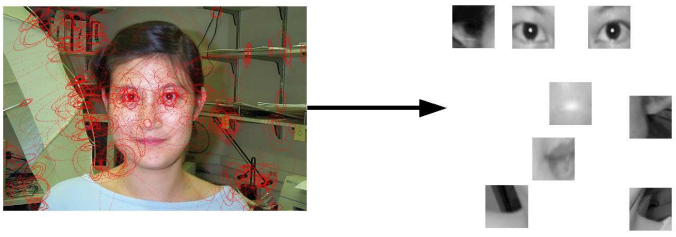


FIG. 3.1 – Visage représenté par un ensemble de mots visuels.

le cas des mots visuels, cette hypothèse est clairement irréaliste. La figure 3.1 montre clairement qu'un concept (ici, un visage) est représenté par un ensemble de mots visuels.

Le tableau 3.1 illustre également les limites de cette hypothèse : nous avons compté, sur un corpus comprenant 6 catégories d'images, le nombre de catégories dans lesquelles apparaît chaque mot visuel. Nous observons que la quasi-totalité des mots visuels apparaît dans toutes les catégories, et seulement 4 d'entre eux sont spécifiques à une catégorie donnée. Ceci confirme clairement l'inexactitude de l'hypothèse d'indépendance des termes dans le cas des images : les mots visuels ne sont pas spécifiques au concept représenté par une image, c'est l'ensemble des mots de cet image qui est spécifique à ce concept. Néanmoins, dans ce chapitre, puisque nous nous intéressons à l'usage de modèles classiques de recherche d'information textuelle, nous travaillerons sous cette hypothèse, comme le font la grande majorité des travaux de recherche d'images basés sur les mots visuels. Nous chercherons néanmoins à dépasser cette hypothèse dans le chapitre 4.

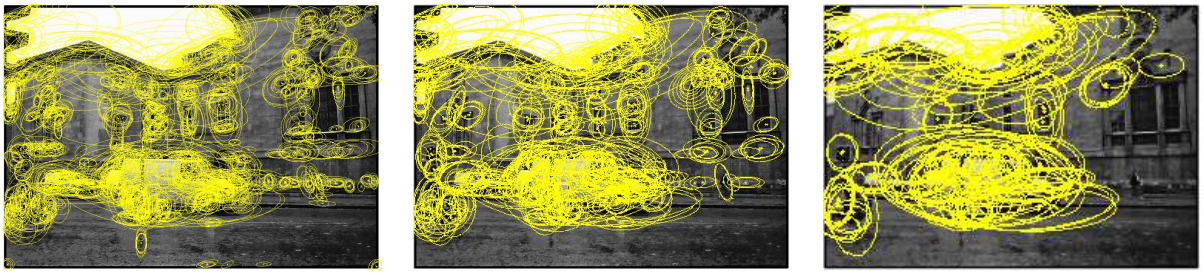
| Nombre de catégories | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Nombre de mots visuels | 4 | 3 | 11 | 15 | 81 | 6442 |

TAB. 3.1 – Nombre de mots visuels en fonction du nombre de catégories où ils apparaissent (données Caltech6).

### 3.1.3 Longueur des documents

La longueur d'un document désigne couramment le nombre d'occurrences de mots qui le composent. En recherche d'information textuelle, il est communément admis que plus un document est long, plus les fréquences des mots qui y apparaissent seront élevées, et plus il pourra contenir de mots différents. Cela a conduit à de nombreux travaux sur la normalisation de la fréquence des termes dans les documents, pour pouvoir s'abstraire de la longueur de ces derniers. Dans le cas des images cependant, la relation entre la fréquence des mots visuels dans l'image, sa taille en mots visuels et son contenu n'est pas aussi simple. En effet, le nombre d'occurrences de mots visuels dans une image peut dépendre de plusieurs facteurs :

- la taille de l'image en entrée : plus une image est grande (en pixels), plus elle contient de détails, donc plus de régions d'intérêt pourra y être détectées. Cette tendance est illustrée sur la figure 3.2 qui indique, pour une même image que nous avons échantillonnée à trois tailles différentes, le nombre de régions qui y sont détectées par un détecteur *Hessian-Affine* avec les paramètres par défaut ;
- le contenu visuel de l'image : le contenu de l'image joue un rôle essentiel sur le nombre de mots visuels qui y apparaissent. En effet, plus une image sera encombrée, plus les détecteurs y découvriront de régions d'intérêt. Inversement, plus une image

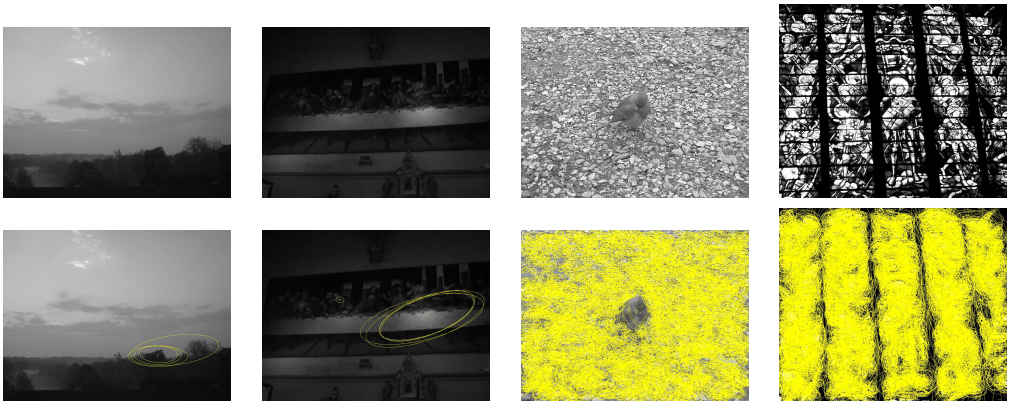


1024 x 720 pixels
2966 régions

512 x 360 pixels
1139 régions

256 x 180 pixels
288 régions

FIG. 3.2 – Nombre de régions d'intérêt détectées en fonction de la taille des images.



6 regions | 11 regions | 19,003 regions | 25,995 regions

FIG. 3.3 – Détection d'un nombre variable de régions d'intérêt sur des images de taille comparable.

contiendra de zones uniformes, et moins il sera possible d'y détecter des régions d'intérêt, celles-ci étant déterminées par les forts changements de niveaux de gris. De même, les images particulièrement sombres, sur-exposées ou floues contiendront également beaucoup moins de régions d'intérêt. Nous illustrons ce phénomène par la figure 3.3 qui montre quatre images de taille initiale comparable et sur lesquelles des nombres très différents de régions d'intérêt sont détectés par le détecteur *Hessian-Affine*. Dans la première image, peu de régions sont détectées en raison de la grande uniformité du contenu. Dans la seconde image, c'est l'obscurité ambiante qui est en cause. Dans les deux autres images, au contraire, un nombre très important de régions d'intérêt sont détectées, en raison du fond très encombré pour la troisième, et de la quantité de détails présents sur la quatrième ;

– les propriétés du détecteur de régions d'intérêt utilisé : les détecteurs n'extraient pas tous, pour une même image, le même nombre de régions d'intérêt. De plus, il est généralement possible de paramétrer leur sensibilité pour extraire un nombre plus ou moins grand de régions.

### 3.1.4 Fréquence des mots dans les documents

Dans les documents textuels, la fréquence des termes au sein d'un document (ou fréquence locale) est ordinairement interprétée ainsi : plus un terme apparaît dans un document, plus il est caractéristique de ce document, et donc plus il est intéressant de le prendre

FIG. 3.4 – Exemple d'une région détectée plusieurs fois sur une image de moto.

en compte pour décrire ce document. En ce qui concerne les images, la fréquence des mots visuels dépend, comme la taille des documents, de plusieurs facteurs :

- de la même manière que la taille en pixels d'une image va influencer le nombre de régions, et donc de mots visuels, qui y seront détectés, la taille d'un objet dans une image va influencer le nombre de mots visuels décrivant cet objet, car les détails de cet objet seront plus nombreux. En un sens, cela correspond à l'idée prédominant dans le cas des textes : plus l'objet aura d'importance dans l'image, plus les mots le décrivant seront nombreux. Cependant, le changement d'échelle et, surtout, l'apparition de détails pourront également provoquer l'apparition de mots visuels qui n'y seraient pas si l'objet occupait une surface plus limitée de l'image. Cependant, le fait qu'un mot visuel soit particulièrement fréquent dans une image ne signifie pas que l'objet qu'il représente soit significatif du sens de l'image. Ainsi, dans la troisième image de la figure 3.3, les nombreux galets du fond rassemblent la très grande majorité des régions d'intérêt, et provoqueront donc une très forte fréquence locale des mots visuels représentant ce type de régions, alors que peu de régions sont detectées sur l'oiseau, qui représente pourtant l'information principale de l'image ;
- la fréquence des mots visuels dépend aussi des propriétés des détecteurs de régions d'intérêt. Ainsi, un phénomène remarquable est la tendance des détecteurs courants à détecter des régions similaires plusieurs fois, ainsi que le montre la figure 3.4 ;
- la fréquence dans une image d'un mot visuel correspondant à un objet donné dépend également du nombre de fois que cet objet apparaît dans l'image. Ainsi, si un objet apparaît deux fois, les fréquences des mots le représentant seront doubles. Plus généralement, un mot visuel donné pouvant apparaître dans deux objets différents, la fréquence d'un mot visuel donné dans une image dépend du nombre d'occurrences d'objets décrits par ce mot visuel dans l'image. En plus de cela, cette fréquence sera aussi affectée par les éventuelles occultations partielles subies par l'objet ;
- enfin, le vocabulaire aura également une influence sur la fréquence des mots visuels dans les images : plus le vocabulaire sera grand, moins il y aura d'occurrences de chaque mot visuel dans la collection, et *a fortiori* dans chaque image. Néanmoins, ce phénomène est relativement indépendant des précédents et peut être négligé une fois la taille du vocabulaire déterminée.

### 3.1.5 Requêtes

Enfin, la nature des requêtes varie fortement entre un système de recherche d'information textuelle et un système de recherche d'images basé sur les mots visuels. Plus particulièrement, la taille (au sens du nombre d'occurrences de mots constituant la requête) diffère. En recherche textuelle, les requêtes sont courtes (typiquement, une question conte-

nant quelques dizaines de mots, par exemple dans les campagnes d'évaluation TREC[1]) ou très courtes (2 mots en moyenne dans le cas des requêtes sur Internet [Sav08]). Dans le cas des moteurs de recherche d'images par le contenu, les requêtes sont des images, c'est-à-dire des documents complets ; ces requêtes seront donc composées d'un nombre de mots bien plus importants. Cette différence de taille entre les requêtes a ainsi des conséquences non négligeables sur les systèmes d'indexation et de recherche :

- les fichiers inversés perdent de leur efficacité. En effet, le fait que la requête soit plus longue va forcer le système à explorer une plus grande portion de la base d'images, le rendant d'autant plus lent. Certains auteurs ont tenté de limiter cet effet en améliorant la structure des fichiers inversés [JHS07], d'autres l'ont contourné en traitant des requêtes courtes, ce qui peut constituer un cas d'utilisation réaliste dans lequel l'utilisateur spécifie la région de l'image requête qu'il souhaite rechercher, sans toutefois être acceptable pour tous les cas de figure. En particulier, les fichiers inversés étaient présentés comme étant particulièrement efficaces dans les travaux fondateurs de Sivic et Zissermann car ceux-ci ne considéraient comme requêtes que des régions d'image de taille restreinte, contenant donc peu de mots visuels [SZ03] ;
- il faut considérer une normalisation de la requête, ainsi que des poids équivalents pour les termes de la requête et ceux des documents, ce qui n'est pas toujours le cas dans les modèles de recherche textuelle. Ce n'est en particulier pas vrai dans les modèles probabilistes, qui nécessitent donc d'être adaptés pour être utilisés sur des images (voir section 3.3.2.1).

## 3.2 *Stop-lists*

Dans cette section, nous présentons une méthode de constitution de *stop-lists* basée sur pLSA et adaptée à la recherche d'images catégorisées à l'aide des mots visuels. L'utilisation de stop-lists est classique en recherche d'information textuelle, comme nous l'avons expliquée en section 2.2.2 : elle permet d'améliorer les performances des systèmes à la fois du point de vue de la pertinence des résultats et des temps de réponse, en supprimant du vocabulaire d'indexation les termes les moins pertinents pour décrire les mots visuels. Les méthodes de constitution de *stop-lists* n'ont pas été étudiées pour les mots visuels, à l'unique exception, à notre connaissance, des travaux de Sivic et Zisserman [SZ03] qui utilisent la méthode classique d'élimination des mots vides basée sur la fréquence des mots dans la collection. Nous décrivons ici cette méthode, puis notre méthode basée sur pLSA.

### 3.2.1 *Stop-lists* basées sur la fréquence

Le principe des *stop-lists* basées sur la fréquence des mots dans la collection provient de l'hypothèse de Luhn, elle-même issue des observations de Zipf sur la fréquence des mots [VR77]. Ce dernier avait observé que les fréquences des termes dans un document ne sont pas aléatoires mais suivent globalement la propriété suivante : $\mathit{frequence} * \mathit{rang} = \mathit{constante}$. Cette propriété indique que les fréquences des différents mots suivent une évolution régulière : la fréquence du second mot le plus représenté vaut la moitié de celle du premier, la fréquence du troisième le tiers, et ainsi de suite. Luhn a ajouté à cette observation l'hypothèse suivante : les mots dont la fréquence dans le document est très élevée ou très faible sont de peu d'importance et peuvent être éliminés. Il y a deux idées derrière cette hypothèse : les mots très fréquents représentent des mots courants, peu spécifiques et donc décrivant mal les documents ; les mots les moins fréquents, quant à eux, ne sont d'aucune

[1] Text REtrieval Conference : `http ://trec.nist.gov`.

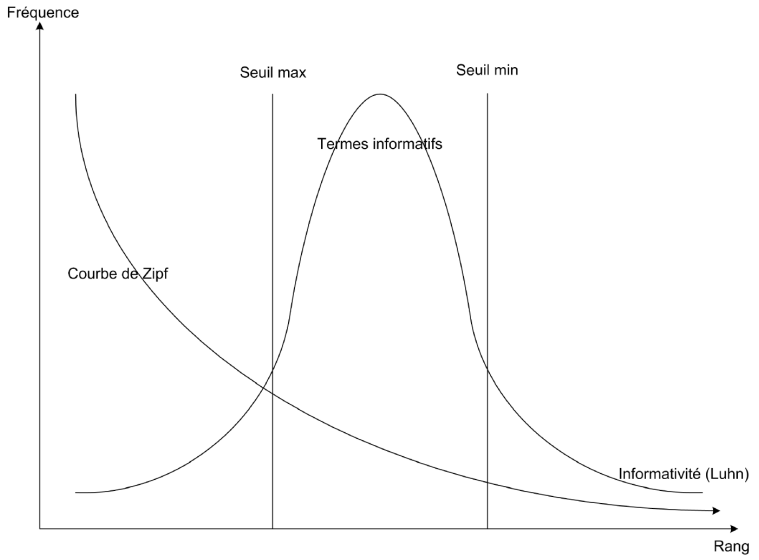



FIG. 3.5 – Importance des termes dans un document selon l'hypothèse de Luhn (schéma adapté de [Mor06]).

aide car ils ont peu de chances d'apparaître dans d'autres documents et donc d'être utiles lors des calculs de similarité (ceci élimine notamment les fautes de frappe ou les néologismes). Luhn considère que les mots de fréquence moyenne sont les plus significatifs (voir figure 3.5), et propose donc d'éliminer tous les mots dont la fréquence dans le document se situe au-delà d'un seuil de fréquence supérieur donné ou en-deça d'un seuil inférieur donné. Cette approche est adoptée par Sivic et Zisserman, mais à l'échelle du corpus et non des documents : ils éliminent les 5% de mots visuels les plus fréquents et les 10% les moins fréquents du corpus (seuils choisis de manière empirique) [SZ03]. Les raisons qui justifient ce choix sont similaires à celles de l'approche de Luhn : les mots visuels apparaissant de très nombreuses fois dans le corpus représentent des informations visuelles trop génériques pour être exploitées (les reflets lumineux, par exemple) tandis que les mots très rares ne sont pas utiles lors des calculs de similarité.

### 3.2.2 *Stop-lists* basées sur pLSA

Nous proposons ici une méthode d'élimination des mots visuels plus fine que le recours à de simples listes de fréquence, dans un contexte de recherche d'images catégorisées. Cette méthode nous permet d'obtenir pour chaque mot un score indiquant s'il est pertinent pour décrire chaque catégorie d'images, et donc d'éliminer les mots qui ne sont utilies à la description d'aucune catégorie.

#### 3.2.2.1 Principe de pLSA

pLSA (pour probabilistic Latent Semantic Analysis) [Hof99] est une extension probabiliste de LSA, une méthode d'analyse non supervisée des données textuelles qui peut être utilisée dans un contexte de classification de textes ou d'indexation [DDH90]. Son objectif est de faire émerger des thèmes (ou des concepts) des documents, chaque thème étant représenté par un ensemble de mots, et chaque document par un ensemble de thèmes. Le

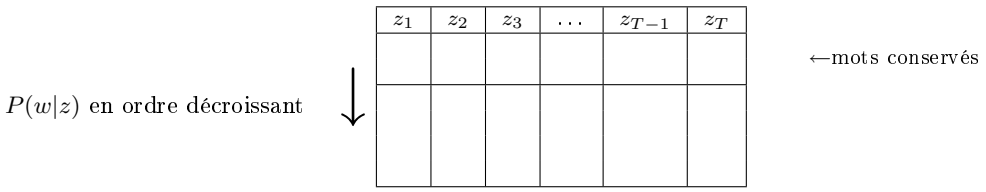



FIG. 3.6 – Sélection de mots visuels basée sur pLSA.

modèle probabiliste de pLSA est le suivant :

$$\Pr(w, d) = \prod_{i=1}^{T} \Pr(w|z_i) \Pr(z_i|d) \tag{3.1}$$

où $\Pr(w, d)$ désigne la probabilité conjointe du document $d$ et du mot $w$, $\Pr(w|z_i)$ la probabilité d'occurrence du mot $w$ dans le thème $z_i$, $\Pr(z_i|d)$ la probabilité d'occurrence et $T$ le nombre de thèmes utilisé. Les valeurs de $\Pr(w, d)$ sont calculées à partir de la matrice contenant les fréquences des mots dans un sous-ensemble du corpus, puis les probabilités $\Pr(w|z_i)$ et $\Pr(z_i|d)$ sont estimées en utilisant l'algorithme *Expectation-Maximization* [DLR77]. Le paramètre $T$ doit quant à lui être fixé manuellement.

#### 3.2.2.2 Constitution d'une *stop-list* grâce à pLSA

Le principe de la constitution de *stop-lists* est la suivante : si un mot a une probabilité $\Pr(w|z_i)$ faible pour tous les thèmes $z_i$, alors il n'est pertinent pour décrire aucun des thèmes, et donc aucun des documents. Il peut être éliminé. Ce principe est illustré par la figure 3.6. Pour procéder à cette élimination, il faut choisir un seuil qui permette de déterminer automatiquement la quantité de mots à conserver. Il est difficile de fixer arbitrairement un seuil sur les probabilités car celui-ci dépendrait du nombre de mots considérés et de la manière dont les masses de probabilité sont réparties. Nous choisissons plutôt de garder les mots en fonction de leur rang : nous conservons tous les mots qui apparaissent parmi les $\frac{n}{T}$ mots les plus probables dans au moins un thème. Cette méthode est plutôt adaptée à la recherche d'images catégorisées car il faut pouvoir faire apparaître un nombre donné de thèmes de manière fiable, nombre qui correspond alors au nombre de catégories possibles.

## 3.3 Schémas de pondération

Dans cette section, nous décrivons les principales pondérations que nous avons utilisées dans un cadre de recherche d'images et les replaçons dans leur cadre initial, en fonction du type de modèle de recherche d'information dont elles sont issues, vectoriel ou probabiliste. Le tableau 3.2 indique les notations que nous utiliserons dans cette partie et dans le reste de ce chapitre.

### 3.3.1 Pondérations et modèle vectoriel

Les pondérations sont une des caractéristiques majeures du modèle vectoriel introduit par Salton [Sal71]. Contrairement aux modèles précédents, ensemblistes et booléens, le modèle vectoriel permet de caractériser non seulement la présence ou l'absence de termes dans les documents, mais également leur importance relative pour décrire le contenu du document : on attribue à chaque terme $t_i$ d'un document $d_j$ un poids $w_{ij} \in \mathbb{R}$ dont la valeur sera d'autant plus grande que le terme $t_i$ caractérise correctement $d_j$. Cette propriété

| | |
|---|---|
| $t_i$ | $i$-ième terme (ou mot) |
| $d_j$ | $j$-ième document |
| $N$ | Nombre total de documents |
| $\mathrm{df}_i$ | Nombre de documents contenant le terme $t_i$ |
| $\mathrm{tf}_{ij}$ | Nombre d'occurrences du terme $t_i$ dans le document $d_j$ |
| $\overline{\mathrm{tf}_i}$ | Fréquence moyenne de $t_i$ |
| $\mathrm{CF}_i$ | Nombre d'occurrences du terme $t_i$ dans la collection |
| $\mathrm{CF}^*$ | Nombre total d'occurrences de mots dans la collection |
| $\mathrm{dl}_j$ | Longueur de $d_j$ (nombre d'occurrences de mots dans $d_j$) |
| $\mathrm{dl}_{\mathrm{avg}}$ | Longueur moyenne des documents de la collection |

TAB. 3.2 – Notations utilisées dans ce chapitre.

permet ainsi de décrire le contenu des documents beaucoup plus finement et de mieux estimer leur pertinence vis-à-vis des requêtes. Les poids employés dans le modèle vectoriel sont basés sur des observations empiriques plutôt que sur des modèles élaborés, et reposent en particulier sur trois observations couramment acceptées :

1. *plus un terme est présent dans un document, mieux il décrit ce document.* Cette observation, déjà évoquée dans la section 3.1.4, est issue des premiers travaux sur la description statistique des documents, par Luhn ;
2. *plus un terme apparaît dans un grand nombre de documents, moins il est spécifique à ces documents.* Ainsi, dans un corpus contenant des documents sur l'informatique, le terme *ordinateur* apparaîtra probablement dans un grand nombre de documents, et ne leur sera pas très spécifique, tandis que le terme *cryptographie* n'apparaîtra que dans les documents relatifs à la *sécurité informatique*, auxquels il sera très spécifique. Cette seconde observation est dûe aux travaux de Sparck-Jones [SJ72] ;
3. plus un document est long, plus la fréquence de chaque mot dans ce document sera élevée. Prendre en compte la fréquence brute (nombre d'occurrences) des termes dans les documents aura donc pour effet de donner plus de poids aux documents longs.

Ces trois observations ont conduit, d'une manière plus générale, à considérer les poids $w_{ij}$ sous trois aspects complémentaires :

1. **une pondération locale** $l_{ij}$ qui caractérise l'importance du terme $t_i$ dans le document $d_j$ ;
2. **une pondération globale** $g_i$ qui caractérise la spécificité du terme $t_i$ dans la collection de documents considérée ;
3. **un facteur de normalisation** $n_j$ qui normalise les poids des termes en fonction de la taille des documents $d_j$, de sorte à s'abstraire de cette dernière.

Le poids $w_{ij}$ de chaque terme $t_i$ au sein du document $d_j$ peut alors être obtenu par une combinaison de ces trois facteurs :

$$w_{ij} = l_{ij}.g_i.n_j \tag{3.2}$$

Nous détaillons dans la suite de cette section les options existantes pour chacun de ces facteurs.

#### 3.3.1.1 Pondérations locales

Nous détaillons ici quelques-unes des pondérations locales les plus courantes en recherche d'information dans le cadre d'un modèle vectoriel.

**Fréquence :** c'est la fréquence brute $\mathrm{tf}_{ij}$ (tf : *term frequency*) du terme $t_i$ dans le document $d_j$, c'est-à-dire son nombre d'occurrences.

**Logarithme de fréquence :** l'utilisation d'un logarithme permet de réduire l'importance des très hautes fréquences, pour laisser un certain poids aux termes plus rares. En effet, il peut arriver que, pour une requête constituée de plusieurs mots, un document contenant un mot de la requête de manière très fréquente et aucun des autres, soit favorisé par rapport à un document contenant tous les mots de cette requête avec une fréquence faible. Cette pondération permet de rectifier ce problème [BSA92].

$$l_{ij} = \begin{cases} 0 & \text{si } \mathrm{tf}_{ij} = 0 \\ 1 + \log(\mathrm{tf}_{ij}) & \text{sinon} \end{cases} \tag{3.3}$$

**Fréquence normalisée augmentée :** comme le logarithme de fréquence, la fréquence normalisée augmentée permet de limiter l'influence des plus hautes fréquences. Elle intègre explicitement un terme relatif à la présence du terme $t_i$ dans le document $d_j$, et un autre relatif à sa fréquence. Ce dernier intègre de plus une normalisation par rapport au terme le plus fréquent du document. Le paramètre $a$ permet de contrôler l'importance relative accordée à la présence du terme par rapport à la fréquence. Cette pondération locale est issue du système de recherche d'information SMART de Salton [SB88].

$$l_{ij} = \begin{cases} 0 & \text{si } \mathrm{tf}_{ij} = 0 \\ a + (1-a) \cdot \frac{\mathrm{tf}_{ij}}{\max_{t_k \in d_j}(\mathrm{tf}_{kj})} & \text{sinon} \end{cases} \tag{3.4}$$

**Pondération binaire :** la pondération binaire ne prend en compte que la présence du terme dans le document, et élimine toute information de fréquence.

$$l_{ij} = \begin{cases} 1 & \text{si } \mathrm{tf}_{ij} > 0 \\ 0 & \text{sinon} \end{cases} \tag{3.5}$$

**Fréquence au carré :** la fréquence au carré permet, contrairement au logarithme de fréquence et à la fréquence normalisée augmentée, de mettre en avant les termes de haute fréquence.

$$l_{ij} = \mathrm{tf}_{ij}^2 \tag{3.6}$$

#### 3.3.1.2 Pondérations globales

Détaillons à présent les principales pondérations globales couramment utilisées dans un cadre vectoriel.

**Fréquence documentaire inverse :** la fréquence documentaire inverse (idf, *Inverse Document Frequency*) est l'adaptation directe de l'observation empirique selon laquelle un mot est d'autant plus spécifique à un document qu'il apparaît dans peu de documents de la collection. L'idf d'un terme $t_i$ est défini de sorte à être inversement proportionnel à $\mathrm{df}_i$, le nombre de documents contenant $t_i$.

$$g_i = \log\left(\frac{N}{\mathrm{df}_i}\right) \tag{3.7}$$

idf **au carré :** l'idf élevé au carré creuse encore plus l'écart d'importance entre termes fréquents dans la collection et termes rares dans la collection. Il permet en principe de favoriser la précision du système [CMS07].

$$g_i = \left[\log\left(\frac{N}{\mathrm{df}_i}\right)\right]^2 \tag{3.8}$$

#### 3.3.1.3 Facteur de normalisation

Le facteur de normalisation a, dans le modèle vectoriel, un rôle double :

1. il permet de s'abstraire de la taille du document, comme évoqué précédemment, en considérant des fréquences des termes relatives plutôt qu'absolues. Cette normalisation peut également être intégrée à la pondération locale, comme dans le cas de la fréquence normalisée augmentée (équation 3.4) ;
2. il permet d'obtenir des vecteurs de même norme, et donc de produire des distances entre vecteurs comparables les unes aux autres pour établir le classement des résultats. À ce titre, le facteur de normalisation doit être choisi de manière cohérente avec la distance utilisée. À une distance de Minkowski $L_k$, on associe donc la norme $||\cdot||_k$ associée, définie ainsi pour un vecteur $d_j = (w_{1j}, w_{2j}, \ldots, w_{mj})$ :

$$n_j = ||d_j||_k = \left(\sum_{i=1}^{m} w_{ij}^k\right)^{\frac{1}{k}} \tag{3.9}$$

La similarité du cosinus intègre d'elle-même une normalisation $||\cdot||_2$, et est, à ce titre, équivalente à une distance $L_2$ entre vecteurs normalisés (comme indiqué en section 1.3.2.4).

### 3.3.2 Pondérations et modèles probabilistes

La recherche d'information probabiliste a été introduite par Robertson dans les années 1970. Elle modélise, dans un premier temps, le processus de recherche d'information dans un cadre probabiliste et complètement générique, c'est-à-dire que, bien que cette théorie n'ait été directement utilisée, à notre connaissance, que dans le cadre des documents textuels, ses résultats fondamentaux pourraient être appliqués à la recherche d'images, de vidéos, de séquences sonores ou de tout autre type de document. L'objectif des modèles probabilistes est de modéliser la pertinence des documents à une requête sous forme d'une *probabilité de pertinence* (alors qu'elle l'était sous forme de distance dans le modèle vectoriel). Le résultat fondamental de la recherche d'information probabiliste est le principe de classement probabiliste (PRP, *Probability Ranking Principle*) établi par Robertson en 1977 [Rob77], exprimé ainsi :

**Probability Ranking Principle :** *If retrieved documents are ordered by decreasing probability of relevance on the data available, then the system's effectiveness is the best to be gotten for the data* (Si les documents retrouvés sont ordonnés par probabilité de pertinence décroissante en fonction des données disponible, alors l'efficacité[2] du système est optimale pour ces données).

Une fois ce principe posé, deux remarques doivent être prises en compte avant de mettre en œuvre un système de recherche d'information probabiliste :

[2]Efficacité en termes de pertinence, telle que mesurée par le rappel et la précision.

- étant donné que la finalité de tout système n'est pas d'obtenir des valeurs de pertinence absolues, mais d'établir un classement des documents par score de pertinence, il n'est pas indispensable de s'en tenir à des probabilités au sens strict du terme. Les scores de pertinence peuvent donc être des probabilités aussi bien que n'importe quel autre score obtenu par une transformation de ces probabilités, à condition que la transformation appliquée préserve l'ordre initialement induit par les probabilités ;
- le problème majeur reste de déterminer les probabilités de pertinence des documents. Pour le texte, plusieurs modèles ont été proposés, qui proposent de représenter la distribution des termes de différentes manières. Ces modèles font référence.

Dans la suite de cette section, nous présentons d'abord le lien qui peut être établi entre les modèles vectoriels et les modèles probabilistes, puis nous présenterons les deux principales familles de modèles probabilistes et les pondérations qu'il est possible d'en extraire.

#### 3.3.2.1 Lien entre modèles probabilistes et vectoriels

Les modèles probabilistes tels qu'ils sont généralement présentés peuvent être vus comme des cas particuliers de modèles vectoriels. En effet, nous avons observé que les scores de pertinence probabilistes entre une requête $q$ et un document $d_j$ dépendent des mots en commun entre la requête et le document, et sont généralement exprimés sous la forme suivante :

$$PM(d_j, q) = \sum_{t_i \in q} q_i.w(\mathrm{tf}_{ij}) \tag{3.10}$$

où $q_i$ désigne la présence du terme $t_i$ dans la requête, et $w(\mathrm{tf}_{ij})$ le score probabiliste associé au terme $t_i$ dans le document $d_j$. Cette formulation générale est en fait équivalente à un produit scalaire entre un vecteur représentant les fréquences des termes d'indexation dans $q$ et des fréquences pondérées des termes dans le document $d_j$. Les scores probabilistes peuvent donc être ramenés à des cas particuliers de modèles vectoriels où la similarité entre documents est calculée à l'aide de la similarité du cosinus, à deux différences près :

- les vecteurs ne sont pas normalisés explicitement. La normalisation de la requête n'est en effet pas considérée comme nécessaire car, en recherche textuelle, les requêtes sont courtes et contiennent généralement les termes une seule fois. La normalisation des fréquences des termes du document est réalisée à travers le score probabiliste associé à chaque terme ;
- il n'y a pas de pondération de la requête, pour la raison évoquée ci-dessus.

Pour adapter des modèles probabilistes aux modèles vectoriels, et en particulier au cas des images, il suffit donc de considérer les scores probabilistes associés aux termes comme de nouveaux schémas de pondération, que l'on appliquera aux requêtes comme aux documents. Les vecteurs peuvent ensuite être comparés en utilisant n'importe quelle distance classique du modèle vectoriel, après normalisation adaptée à cette distance.

#### 3.3.2.2 Pondérations issues des modèles *Best Match*

Le modèle probabiliste le plus employé est la famille de mesures dites *Best Match* (BM), proposées par Robertson. Plus particulièrement, le meilleur de ces modèles (d'après les évaluations de TREC [RWB+96]), BM25, constitue la référence en matière de modèle pour la recherche d'information textuelle. La score de pertinence probabiliste de BM25

entre une requête $q$ et un document $d_j$ est souvent défini ainsi :

$$\mathrm{BM25}(d_j, q) = \sum_{t_i \in q} \left[ \frac{\mathrm{tf}_{ij} * (k_1 + 1)}{K + \mathrm{tf}_{ij}} \cdot \mathrm{tf}_{iq} \cdot \log \left( \frac{(r_{iq} + 0.5)(N - \mathrm{df}_i - R_q + r_{iq} + 0.5)}{(R_q - r_{iq} + 0.5)(\mathrm{df}_i - r_{iq} + 0.5)} \right) \right] \tag{3.11}$$

où $R_q$ désigne le nombre de documents pertinents à la requête $q$ dans la collection, $r_{iq}$ le nombre de documents pertinents qui contiennent le terme $t_i$ et $k$ est défini de la manière suivante :

$$K = k_1((1 - b) + b \cdot \frac{\mathrm{dl}_j}{\mathrm{dl}_{\mathrm{avg}}}) \tag{3.12}$$

$k_1$ et $b$ sont des constantes. Nous utilisons par la suite les valeurs par défaut de ces constantes : $k_1 = 1.2$ et $b = 0.75$.

Conformément à ce que nous décrivions en section 3.3.2.1, il est possible d'extraire une pondération de ce score :

$$w_{ij} = \frac{\mathrm{tf}_{ij} * (k_1 + 1)}{K + \mathrm{tf}_{ij}} \cdot \log \left( \frac{(r_{iq} + 0.5)(N - \mathrm{df}_i - R_q + r_{iq} + 0.5)}{(R_q - r_{iq} + 0.5)(\mathrm{df}_i - r_{iq} + 0.5)} \right) \tag{3.13}$$

Ce poids est composé de deux parties distinctes. La première dépend de la fréquence locale du terme $t_i$ dans le document $d_j$, elle peut donc être assimilée à une pondération locale :

$$l(t_i, d_j) = \frac{\mathrm{tf}_{ij} * (k_1 + 1)}{K + \mathrm{tf}_{ij}} \tag{3.14}$$

Cette première partie est dérivée d'un modèle probabiliste de la fréquence des termes dans les documents, le modèle 2-Poisson de Harter [SJWR00a]. Ce modèle représente la distribution des termes dans les documents comme un mélange de deux distributions de Poisson : l'une représentant la fréquence des termes pertinents pour décrire le document, l'autre celle des termes non-pertinents [Har75]. Ce poids intègre une normalisation en fonction de la taille du document.

La seconde partie du poids issu de BM25 peut être vue comme un poids global pour les termes, car il n'implique aucune grandeur relative au document courant. Il est cependant spécifique à la requête $q$ car il nécessite de connaître les documents qui lui sont pertinents :

$$g_{iq} = \log \left( \frac{(r_{iq} + 0.5)(N - \mathrm{df}_i - R_q + r_{iq} + 0.5)}{(R_q - r_{iq} + 0.5)(\mathrm{df}_i - r_{iq} + 0.5)} \right) \tag{3.15}$$

Ce poids est directement dérivé du PRP [SJWR00b], il est donc théoriquement optimal, mais nécessite de posséder des données de pertinence sur les données. Quand aucune donnée de pertinence n'est disponible, il se réduit à la formulation suivante, indépendante de la requête :

$$g_{iq} = g_i = \log \left( \frac{N - \mathrm{df}_i + 0.5}{\mathrm{df}_i + 0.5} \right) \tag{3.16}$$

Cette formulation est très proche de la formulation empirique du poids global idf présenté précédemment. Les principes théoriques qui le sous-tendent confirment le bien-fondé du poids idf. Ce poids est parfois nommé idf *probabiliste* (souvent sans les "+0.5", originellement ajoutés pour estimer correctement des probabilités de pertinence).

#### 3.3.2.3 Pondérations issues des modèles *Divergence From Randomness*

Les modèles de divergence de l'aléatoire (DFR, *Divergence From Randomness*) sont des mesures de similarités basées sur des modèles probabilistes de fréquence des termes, proposés par Amati et Van Rijsbergen [AVR02]. Ces modèles sont basés sur deux hypothèses :

1. les termes sont distribués aléatoirement dans la collection de documents. Leur fréquence au sein de chaque document suit un *modèle aléatoire* (*Randomness model*), indépendamment du fait qu'ils soient de bons descripteurs du document ou non ;
2. quand un terme est un descripteur pertinent d'un document, sa fréquence dans ce document est plus élevée que dans les autres. Les termes pertinents pour décrire un document peuvent donc être identifiés en ce que leur fréquence *diverge du modèle aléatoire.*

Dans ces modèles, les scores probabilistes des mots, et donc, par extension, leurs poids dans un contexte vectoriel, sont considérés comme étant le produit de deux sources d'information, $\mathrm{Inf}_1$ et $\mathrm{Inf}_2$, elles-mêmes dérivées des probabilités $\mathrm{Prob}_1$ et $\mathrm{Prob}_2$, respectivement :

$$\begin{aligned} w(t_i, d_j) &= \mathrm{Inf}_1(t_i, d_j).\mathrm{Inf}_2(t_i, d_j) \\ &= -\log_2(\mathrm{Prob}_1(t_i, d_j)).(1 - \mathrm{Prob}_2(t_i, d_j)) \\ &= -\log_2(\mathrm{Prob}_1(t_i, d_j)^{1-\mathrm{Prob}_2(t_i, d_j)}) \end{aligned} \tag{3.17}$$

La probabilité $\mathrm{Prob}_1$ correspond à la probabilité pour que le terme $t_i$ ait la fréquence $\mathrm{tf}_{ij}$ dans le document $d_j$ selon le modèle aléatoire choisi. Cette probabilité dépend des propriétés de la collection uniquement, elle s'apparente donc à un score global. La source d'information $\mathrm{Inf}_1$ qui en dérive s'appelle le *contenu informatif* (*informative content*) du terme dans le document : il correspond à la quantité d'information sur le document fourni par ce texte. La seconde source d'information, $\mathrm{Inf}_2$, correspond au *gain d'information* (*information gain*) obtenu si l'on considère que $t_i$ est pertinent pour décrire le document $d_j$. Il ne dépend donc pas de la collection mais du document dans lequel $t_i$ apparaît, et correspond donc à une pondération locale. Il est complémentaire de la probabilité $\mathrm{Prob}_2$ qui, elle, modélise le risque de choisir $t_i$ pour décrire $d_j$.

Amati et Van Rijsbergen ont proposé trois modèles aléatoires différents, et deux approximations possibles pour calculer ces modèles qui ne sont pas calculables de manière exacte.

**Le modèle de Bernoulli :** il modélise la fréquence d'un terme comme une succession de tirages aléatoires indépendants dans l'ensemble des termes.
Le modèle de $\mathrm{Prob}_1$ est :

$$\mathrm{Prob}_1(\mathrm{tf}_{ij}) = \left[ \begin{pmatrix} \mathrm{CF}_i \\ \mathrm{tf}_{ij} \end{pmatrix} \left(\frac{1}{N}\right)^{\mathrm{tf}_{ij}} \left(\frac{N-1}{N}\right)^{\mathrm{CF}_i - \mathrm{tf}_{ij}} \right] \tag{3.18}$$

Les deux approximations de $\mathrm{Inf}_1$ qui en découlent sont :

P :

$$\begin{aligned} &\mathrm{Inf}_1(\mathrm{tf}_{ij}) = \mathrm{tf}_{ij}\,.\log_2\left(\frac{\mathrm{tf}_{ij}}{\lambda}\right) + \left(\lambda + \frac{1}{12.\,\mathrm{tf}_{ij}} - \mathrm{tf}_{ij}\right).\log_2(e) + 0.5\log_2(2\pi.\,\mathrm{tf}_{ij}) \\ &\text{avec } \lambda = \frac{\mathrm{CF}_i}{N} \end{aligned} \tag{3.19}$$

D :

$$\begin{aligned} &\mathrm{Inf}_1(\mathrm{tf}_{ij}) = \mathrm{CF}_i\,.D(\phi_{ij}, p) + 0.5\log_2(2\pi.\,\mathrm{tf}_{ij}(1-\phi_{ij})) \\ &\text{avec } \phi_{ij} = \frac{\mathrm{tf}_{ij}}{\mathrm{CF}_i},\ p = \frac{1}{N} \\ &\text{et } D(\phi_{ij}, p) = \phi_{ij}.\log_2\left(\frac{\phi_{ij}}{p}\right) + (1-\phi_{ij}).\log_2\left(\frac{1-\phi_{ij}}{1-p}\right) \end{aligned} \tag{3.20}$$

**Le modèle de Bose-Einstein :** il s'inspire d'un modèle aléatoire utilisé à l'origine en sciences physiques.
Le modèle de $\mathrm{Prob}_1$ est :

$$\mathrm{Prob}_1(\mathrm{tf}_{ij}) = \frac{(\mathrm{CF}_i - \mathrm{tf}_{ij} + 1) \cdot \ldots \cdot \mathrm{CF}_i \cdot (N-1)}{(N + \mathrm{CF}_i - \mathrm{tf}_{ij} - 1) \cdot \ldots \cdot (N + \mathrm{CF}_i - 1)} \tag{3.21}$$

Les deux approximations de $Inf_1$ qui en découlent sont :

G :

$$\mathrm{Inf}_1(\mathrm{tf}_{ij}) = -\log_2\left(\frac{1}{1+\lambda}\right) - \mathrm{tf}_{ij} \cdot \log_2\left(\frac{\lambda}{1+\lambda}\right) \text{ avec } \lambda = \frac{\mathrm{CF}_i}{N} \tag{3.22}$$

Be :

$$\begin{aligned} \mathrm{Inf}_1(\mathrm{tf}_{ij}) = \quad & -\log_2(N-1) - \log_2(e) + f(N + \mathrm{CF}_i - 1, N + \mathrm{CF}_i - \mathrm{tf}_{ij} - 2) \\ & - f(\mathrm{CF}_i, \mathrm{CF}_i - \mathrm{tf}_{ij}) \\ \text{avec} \quad & f(n,m) = (m + 0.5) \cdot \log_2\left(\tfrac{n}{m}\right) + (n-m) \cdot \log_2(n) \end{aligned} \tag{3.23}$$

**Le modèle de fréquence documentaire inverse :** il s'inspire de la notion de fréquence documentaire inverse classique en recherche d'information et déjà présentée dans le cadre de la pondération idf.
Le modèle de $\mathrm{Prob}_1$ est :

$$\mathrm{Prob}_1(\mathrm{tf}_{ij}) = \left(\frac{\mathrm{df}_i + 0.5}{N+1}\right)^{\mathrm{tf}_{ij}} \tag{3.24}$$

Les deux approximations de $\mathrm{Inf}_1$ qui en découlent sont :

In :

$$\mathrm{Inf}_1(\mathrm{tf}_{ij}) = \mathrm{tf}_{ij} \cdot \log_2\left(\frac{N+1}{\mathrm{df}_i + 0.5}\right) \tag{3.25}$$

$\mathrm{In}_e$ :

$$\begin{aligned} \mathrm{Inf}_1(\mathrm{tf}_{ij}) &= \mathrm{tf}_{ij} \cdot \log_2\left(\tfrac{N+1}{n_e + 0.5}\right) \\ \text{avec } n_e &= N \cdot \left(1 - \left(\tfrac{N-1}{N}\right)^{\mathrm{CF}_i}\right) \end{aligned} \tag{3.26}$$

Amati et Van Rijsbergen ont accompagné ces trois modèles aléatoires de deux modèles de gain d'information. Ces modèles représentent le gain d'information obtenu lorsque l'on ajoute une occurrence d'un terme donné à sa fréquence dans un document. Nous présentons ici ces deux modèles très succintement (voir [AVR02] pour les détails).

**Modèle de Laplace :** Ce modèle repose sur la loi de succession de Laplace, qui permet de calculer la probabilité qu'un événement qui s'est déjà produit se reproduise, ce qui correspond à notre situation : nous cherchons la probabilité qu'un terme soit choisi aléatoirement une fois de plus qu'il ne l'a déjà été. Ce modèle permet de calculer $\mathrm{Inf}_2$ de la manière suivante :

$$\mathrm{Inf}_2(\mathrm{tf}_{ij}) = \frac{1}{\mathrm{tf}_{ij} + 1} \tag{3.27}$$

| | |
|---|---|
| H0 | $\mathrm{tf}_{ij}$ |
| H1 | $\mathrm{tf}_{ij} \cdot \frac{\mathrm{dl}_{\mathrm{avg}}}{\mathrm{dl}_j}$ |
| H2 | $\mathrm{tf}_{ij} \cdot \log_2(1 + \frac{\mathrm{dl}_{\mathrm{avg}}}{\mathrm{dl}_j})$ |

TAB. 3.3 – Normalisations pour les poids DFR.

**Modèle de Bernoulli :** Ce second modèle calcule le gain d'information comme le rapport entre la probabilité d'avoir une fréquence $\mathrm{tf}_{ij}$ donnée et la probabilité d'avoir $\mathrm{tf}_{ij}+1$, ces deux probabilités étant modélisées par des processus de Bernoulli. La formule ainsi obtenue pour $\mathrm{Inf}_2$ est alors :

$$\mathrm{Inf}_2(\mathrm{tf}_{ij}) = \frac{\mathrm{CF}_i + 1}{\mathrm{df}_i \cdot (\mathrm{tf}_{ij} + 1)} \tag{3.28}$$

Chacun de ces deux modèles de gain d'information peut être utilisé de manière indifférente avec n'importe lequel des modèles aléatoires décrits précédemment. Ces modèles DFR sont de plus complétés par une normalisation de la fréquence des termes $\mathrm{tf}_{ij}$. En effet, ces modèles ne prennent pas en compte la longueur des documents de manière intrinsèque, il est donc nécessaire d'utiliser une fréquence modifiée $\mathrm{tf}'_{ij}$ qui permette de prendre en compte les variations de fréquence dûes aux variations de longueur des documents. Le tableau 3.3 présente les trois normalisations qui ont été proposées par Amati et Van Rijsbergen [AVR02].

## 3.4 Pondérations pour la recherche d'images

Nous proposons ici deux nouvelles pondérations qui nous semblent adaptés au cas de la recherche d'images à base de mots visuels. L'un est une pondération globale définie sur des critères purement empiriques, l'autre se place dans le cadre des modèles DFR.

### 3.4.1 Pondération globale

La pondération globale que nous proposons se base sur deux grandeurs distinctes :

1. la fréquence documentaire inverse (idf) : nous prenons en compte la fréquence documentaire inverse idf car les mots visuels appartenant au fond des images et ne représentant donc ni un objet représentatif du sens de l'image, ni une partie d'un tel objet, sont susceptibles d'apparaître dans un grand nombre d'images. L'idf permet donc de réduire l'importance de tels mots visuels dans la représentation des images ;
2. la fréquence locale moyenne : les objets contiennent souvent des parties répétées (les roues d'une voiture, les yeux d'un visage, les fenêtres d'un bâtiment...) ou suivent une certaine symétrie. Ceci se caractérise par une répétition des mêmes mots visuels au sein d'une image. Nous pouvons donc considérer qu'un mot visuel dont la fréquence est élevée dans de nombreuses images sera plus pertinent qu'un mot visuel qui apparaît généralement seul, qui sera plutôt caractéristique du fond aléatoire des images. Nous proposons donc de prendre en compte la fréquence moyenne par image des mots visuels dans notre pondération : une forte fréquence moyenne étant caractéristique d'un mot pertinent pour décrire les images. Comme les images ne contenant pas le mot visuel considéré viendraient baisser artificiellement sa fréquence moyenne, sans rapport avec le sens que nous souhaitons donner à cette moyenne, celle-ci n'est pas calculée sur la collection complète mais uniquement sur l'ensemble des documents contenant ce mot.

La pondération globale que nous proposons combine ces deux grandeurs par un produit :

$$g_i = \overline{\mathrm{tf}_i} \cdot \log\left(\frac{N}{\mathrm{df}_i}\right) = \frac{\mathrm{CF}_i}{\mathrm{df}_i} \cdot \log\left(\frac{N}{\mathrm{df}_i}\right) \tag{3.29}$$

### 3.4.2 Pondération DFR

Nous proposons également un modèle aléatoire hypergéométrique pour les pondérations DFR. Contrairement aux modèles de Bernoulli qui considèrent un document comme une succession de tirages aléatoires d'un terme parmi tous les termes de la collection, un modèle hypergéométrique représente un document comme un unique tirage aléatoire de plusieurs mots. Le modèle de Bernoulli construit un document par accumulation de termes. Plus le document contient de termes, plus il y a de tirages successifs et plus la fréquence de chaque terme aura de chances d'être élevée, ce qui correspond bien aux hypothèses classiques utilisées pour les documents textuels. À l'inverse, le modèle hypergéométrique semble plus adapté au cas des images : comme nous l'avons expliqué précédemment, un objet contenu dans une image correspond à un ensemble de mots visuels donné, il convient donc de tirer aléatoirement ces mots visuels simultanément plutôt que successivement, ce que permet le modèle hypergéométrique. Le modèle hypergéométrique de $\mathrm{Prob}_1$ est défini ainsi :

$$\mathrm{Prob}_1 = \frac{\begin{pmatrix} \mathrm{CF}_i \\ \mathrm{tf}_{ij} \end{pmatrix} \begin{pmatrix} \mathrm{CF}^* - \mathrm{CF}_i \\ \mathrm{dl}_j - \mathrm{tf}_{ij} \end{pmatrix}}{\begin{pmatrix} \mathrm{CF}^* \\ \mathrm{dl}_j \end{pmatrix}} \tag{3.30}$$

$\mathrm{Inf}_1$ peut donc être calculé de la manière suivante :

$$\begin{aligned} \mathrm{Inf}_1 = \; & \log((\mathrm{tf}_{ij})!) + \log((\mathrm{CF}_i - \mathrm{tf}_{ij})!) + \log((\mathrm{dl}_j - \mathrm{tf}_{ij})!) \\ & + \log((\mathrm{CF}^* - \mathrm{CF}_i - \mathrm{dl}_j + \mathrm{tf}_{ij})!) + \log(\mathrm{CF}^*!) - \log(\mathrm{CF}_i!) \\ & - \log((\mathrm{CF}^* - \mathrm{CF}_i)!) - \log(l!) - \log((\mathrm{CF}^* - \mathrm{dl}_j)!) \end{aligned} \tag{3.31}$$

Pour évaluer les logarithmes de factorielles, nous utilisons l'approximation de Ramanujan qui permet une approximation plus juste que l'approximation de Stirling utilisée par Amati et Van Rijsbergen :

$$\log(k!) \approx k\log(k) - k + \frac{k(1 + 4k(1 + 2k))}{6} + \frac{\log(\pi)}{2} \tag{3.32}$$

## 3.5 Expérimentations

### 3.5.1 Problèmes traités et données associées

Pour estimer la pertinence des pondérations et distances de manière suffisamment générale, nous avons choisi plusieurs collections d'images pour réaliser nos expérimentations. Ces collections correspondent à deux types de problèmes différents, déjà détaillés en section 1.3.1, page 22 : la recherche de scènes identiques et la recherche d'objets catégorisés. Nous détaillons ici les collections de données utilisées.

#### 3.5.1.1 Recherche de scènes identiques

La recherche de scènes identiques est sans aucun doute, dans le cadre de la recherche d'images, le problème le plus traité dans la littérature par les modèles en sac de mots visuels. Il existe par conséquent plusieurs corpus de données qui lui correspondent. Nous avons choisi parmi ceux-ci les deux qui nous semblaient les plus utilisés.

**Corpus Kentucky** Le corpus Kentucky, proposé par Nister et Stewenius [NS06], contient 2550 scènes distinctes, représentées chacune par 4 images, soit un total de 10200 images. La vérité-terrain étant disponible pour la totalité des images, il est possible d'utiliser n'importe laquelle d'entre elles comme requête. Les scènes représentées sont variées, peuvent être prises en extérieur ou en intérieur et présenter tous types d'objets, naturels (par exemple, une plante verte) comme artificiels (par exemple, une maison).

**Corpus Oxford** Le corpus Oxford, proposé par Zisserman et *al.* [CPS+07], se concentre sur la recherche de monuments de la ville d'Oxford. Le corpus contient donc des photos de bâtiments d'Oxford, noyées parmi d'autres images de contenu varié (personnes, animaux...), pour un total de 5063 images. La vérité-terrain fournit 55 images requêtes, correspondant à 11 monuments distincts d'Oxford, et les jugements de pertinence associés. Ces jugements de pertinence sont classés en plusieurs catégories : *good*, *OK* ou *junk*, quand l'objet est présent, en fonction de la proportion du bâtiment qui est visible, et *absent* s'il n'apparaît pas sur l'image. Comme nous ne considérons que des jugements de pertinence binaires, nous considérons comme pertinente toute image où le bâtiment recherché apparaît.

#### 3.5.1.2 Recherche d'objets catégorisés

Le problème de la recherche d'images catégorisées a quant à lui été beaucoup moins traité avec des systèmes à base de mots visuels, bien que l'on trouve quelques travaux à ce sujet également [ZWG06]. Il n'existe pas de corpus dédié à cette tâche, mais il est possible d'exploiter des corpus destinés à évaluer les tâches de catégorisation d'images. Parmi ces corpus, nous avons choisi de porter notre attention sur les corpus Caltech (du *California Institute of Technology*), qui font référence en matière de catégorisation d'images basée sur les mots visuels. Dans ce genre de corpus, chaque image est associée à une catégorie donnée. Pour l'évaluation, on considère donc comme image pertinente toute image appartenant à la même catégorie que la requête.

**Caltech-6** Nous avons constitué le corpus Caltech-6 à partir de 6 catégories d'images fournies par Caltech et communément utilisées : les avions (*airplanes*, 1074 images), les voitures vues de dos (*cars_rear*, 1155 images), les motos (*motorbikes*, 826 images), les visages (*faces*, 450 images), les guitares (*guitars*, 1025 images) et les fonds (*backgrounds*, 885 images). Le corpus contient un total de 5415 images.

**Caltech-101** Le corpus Caltech 101 [FFFP07] est un corpus de référence très utilisé en catégorisation d'images. Il contient 8697 images, réparties en 101 catégories très variées (véhicules : avions, voitures, motos..., animaux : éléphants, flamants roses, crocodiles..., objets variés : guitares, montres...). Ce corpus est particulièrement difficile, pour plusieurs raisons :

- les catégories peuvent contenir des images d'aspect visuel très différent, par exemple des dessins et des photographies ;
- certaines catégories sont très proches l'une de l'autre visuellement (par exemple, les guitares et les mandolines) ;
- les catégories sont très déséquilibrées : le nombre d'images par catégorie va de 31 à plus de 800.

Ce corpus de données est clairement le corpus le plus difficile traité dans ces expérimentations.

### 3.5.2 Protocole expérimental

#### 3.5.2.1 Vocabulaire visuel

Nous indiquons ici les paramètres utilisés pour construire notre vocabulaire visuel. Dans tous les cas, il a été construit sur un sous-ensemble d'images selectionné aléatoirement dans chaque corpus.

**Détecteur de régions d'intérêt :** nous utilisons le détecteur *Hessian-Affine* proposé par Mikolajcjyk *et al.* [MS04]. Il présente d'excellentes propriétés [MTS+05] et est utilisé dans de nombreux travaux basés sur les mots visuels [JDS08, CPS+07].

**Descripteur de régions d'intérêt :** nous utilisons les descripteurs SIFT. Ils fournissent des résultats parmi les meilleurs pour la mise en correspondance d'images [MS05] et sont clairement les plus utilisés pour la recherche d'images basée sur les mots visuels [SZ03, CDF+04, JDS08, CPS+07].

**Algorithme de clustering :** nous utilisons l'algorithme *k-means* hiérarchique proposé par Nister et Stewenius [NS06]. Il ne fournit pas de clusters optimaux, ce qui limite la performance globale du système, mais il est très efficace en termes de temps de calcul.

**Taille du vocabulaire :** le tableau 3.4 indique les tailles de vocabulaire employées pour chaque corpus. Ces tailles ont été choisies par validation sur quelques requêtes choisies aléatoirement et différentes des requêtes employées par ailleurs. Notons que l'algorithme de clustering que nous utilisons permet de spécifier certains paramètres influençant le nombre de clusters obtenus mais pas de choisir un nombre précis de clusters, ce qui explique que les tailles des vocabulaires obtenus ne soient pas des comptes ronds.

| Corpus | Taille du vocabulaire |
|---|---|
| Caltech-6 | 6556 |
| Caltech-101 | 61687 |
| Kentucky | 19545 |
| Oxford | 117151 |

TAB. 3.4 – Taille du vocabulaire employé pour chaque ensemble de données.

#### 3.5.2.2 Requêtes

Les requêtes sont toutes des images entières, nous leur appliquons donc les mêmes pondérations qu'aux images de la base. Les requêtes sont choisies aléatoirement parmi les images de chaque corpus, à l'exception du corpus Oxford pour lequel les requêtes sont celles fournies par la vérité-terrain. Le tableau 3.5 indique le nombre de requêtes utilisé pour chaque corpus.

#### 3.5.2.3 Évaluation

Nous mesurons les performances de nos systèmes en termes de précision et de MAP (voir section 1.1.6.2 pour une description de ces mesures de performances). Ici, comme nous effectuons une comparaison exhaustive des documents réponses et des requêtes, le rappel n'est pas important : il évolue de la même manière que la précision, à DCV égale. En

| Corpus | Nombre de requêtes |
|---|---|
| Caltech6 | 200 |
| Caltech101 | 200 |
| Kentucky | 300 |
| Oxford | 55 |

TAB. 3.5 – Nombre de requêtes utilisées pour chaque ensemble de données.

revanche, la MAP apporte un indice intéressant pour cette étude : elle reflète le fait que des documents pertinents se retrouvent ou non en fin de classement. Les DCV considérées sont choisies en fonction du nombre de documents pertinents : leurs valeurs sont plus grandes pour les corpus Caltech pour lesquels les images pertinentes peuvent être particulièrement nombreuses. De plus, compte tenu du fait que le corpus Kentucky ne contient que 4 images pertinentes par requête, les résultats donnés sur ce corpus pour des DCV supérieures à 4 seront peu significatifs.

Nous avons, de plus, vérifié si les différences entre les résultats obtenus étaient statistiquement significatives. Nous avons utilisé pour cela le test de Wilcoxon, avec un seuil de *p-value* fixé à 0.1.

### 3.5.3 Expériences sur les *stop-lists*

#### 3.5.3.1 *Stop-lists* testées

Nous avons comparé l'approche de construction des *stop-lists* adaptée de l'approche de Luhn avec l'approche que nous proposons, sur le corpus Caltech-6. Pour l'approche classique, nous avons testé différents seuils d'élimination des mots, aussi bien pour le seuil inférieur que le seuil supérieur : 1%, 5%, 10%, 15%.Les vecteurs utilisés pour la phase de recherche ont été pondérés avec une pondération tf. idf classique et comparés à l'aide d'une distance $L_1$ (après normalisation apropriée).

#### 3.5.3.2 Résultats

Le tableau 3.6 contient les résultats obtenus en termes de précision, de MAP, ainsi que la taille du vocabulaire obtenu après éliminations des mots vides. Les *stop-lists* basées sur la fréquence sont indiquées avec la valeur utilisée pour le seuil supérieur et pour le seuil inférieur : la proportion de mots éliminée correspond donc au double du seuil indiqué.

| *Stop-list* | Taille du vocabulaire | P10 | P20 | P50 | P100 | MAP |
|---|---|---|---|---|---|---|
| Aucune | 6556 | 0.779 | 0.722 | 0.644 | 0.583 | 0.389 |
| fréquence - 1% | 6431 | 0.779 | 0.719 | 0.641 | 0.583 | 0.389 |
| fréquence - 5% | 5905 | 0.767 | 0.715 | 0.640 | 0.580 | 0.387 |
| fréquence - 10% | 5249 | 0.487 | 0.585 | 0.592 | 0.553 | 0.374 |
| fréquence - 15% | 4593 | 0.302 | 0.500 | 0.552 | 0.530 | 0.363 |
| pLSA - $k = 6$ | 4214 | 0.771 | 0.713 | 0.642 | 0.580 | 0.385 |

TAB. 3.6 – Résultats des expériences sur les *stop-lists*.

Les résultats montrent que, contrairement aux résultats obtenus par Sivic et Zisserman sur leur application particulière (détection d'objets dans un nombre limité de vidéos), l'utilisation de *stop-lists* basées sur la fréquence n'est pas souhaitable dans un cas générique de recherche d'images catégorisées, même sur un corpus restreint : les résultats obtenus sont

systématiquement inférieurs à ceux obtenus avec un vocabulaire complet, y compris en ne supprimant que très peu de mots. Lorsque le nombre de mots éliminés devient important, la baisse de précision est très importante, en particulier sur les 10 premiers documents, qui sont les plus importants car les premiers consultés par l'utilisateur.

Les résultats obtenus par notre méthode de construction de *stop-lists* sont très supérieurs à ceux obtenus à partir de la méthode classique, à taille de vocabulaire équivalente. On parvient à éliminer environ un tiers du vocabulaire tout en n'accusant qu'une perte de performances limitée. Il est probable qu'en éliminant moins de mots visuels à l'aide de cette technique on parvienne à n'obtenir aucune perte de précision, voire un gain. Le problème est qu'il est difficile de contrôler le nombre de mots visuels éliminés. Il serait possible de conserver les mots visuels qui se situent parmi les $\alpha.\frac{n}{k}$ mots les plus probables pour chaque thème, avec $\alpha > 1$, pour conserver plus de mots visuels, mais cela ne donne qu'un contrôle très limité sur la quantité de mots visuels conservés. En effet, comme nous sélectionnons les mots apparaissant dans un moins un thème, il est possible que les mots supplémentaires sélectionnées pour un thème donné appartiennent déjà aux mots déjà conservés au niveau d'un autre thème.

Globalement, nous observons ici que l'utilisation de *stop-lists* n'est pas favorable à la recherche d'images catégorisées, car, si elle peut permettre d'obtenir de meilleurs temps de réponse en diminuant la taille du vocabulaire visuel, elle diminue également systématiquement les performances du systèmes en terms de pertinence des résultats, même en adoptant une méthode de choix des mots vides élaborée comme celle que nous avons proposée. Ce comportement est opposé à celui obtenu dans le domaine textuel, où l'utilisation de *stop-lists* permet généralement d'obtenir des gains de performances non négligeables, en plus de réduire la complexité de la phase de recherche.

### 3.5.4 Expériences sur les distances

#### 3.5.4.1 Distances testées

Nous avons testé les distances de Minkowski $L_k$, avec $k$ compris entre 0.01 et 3. Les propriétés de ces distances, et en particulier des distances fractionnelles (distances $L$ telles que $k < 1$), sont données en section 1.3.2.4, page 28. Ces distances ont été testées avec une pondération tf. idf des vecteurs, conformément à l'usage dans la plupart des travaux [SZ03, NS06, JHS07], et une normalisation adaptée à chaque distance.

#### 3.5.4.2 Résultats

La figure 3.7 présente les résultats obtenus en fonction de différentes valeurs de $k$ testées, pour différentes mesures de performances. Les principales observations que l'on peut en tirer dans un premier temps sont :

- sur les corpus Caltech-6 et Kentucky, les résultats montrent (précision et MAP pour Caltech-6, MAP uniquement pour Kentucky pour lequel la précision est ici moins significative) clairement que les valeurs élevées de $k$ fournissent les plus mauvais résultats. Les performances optimales du système se situent autour d'une valeur de $k = 0,75$. Ces résultats sont discutés dans la section 3.5.6.1 ;
- le corpus Caltech-101 a un comportement assez similaire aux corpus Caltech-6 et Kentucky, mais les différences de performances observées sont beaucoup moins significatives. Ce résultat est également discuté dans la section 3.5.6.1 ;
- la tendance générale pour le corpus Oxford est parfaitement contraire à celle observée sur les autres corpus : les plus mauvais résultats sont dûs à des valeurs de $k$ faibles, et

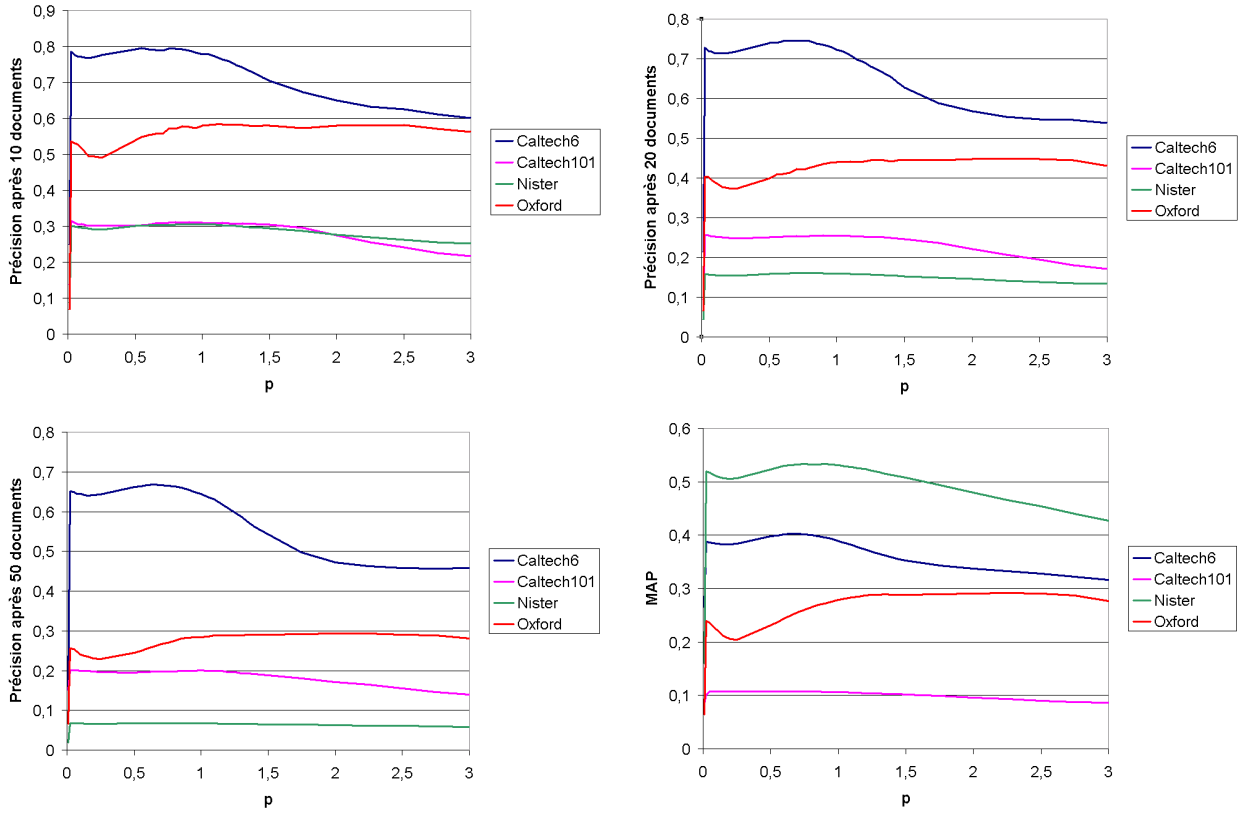



FIG. 3.7 – Influence de $k$ sur les performances des distances $L_k$.

les meilleurs obtenus pour $k \approx 2$. Ces résultats sont discutés dans la section 3.5.6.1.

### 3.5.5 Expériences sur les pondérations

#### 3.5.5.1 Pondérations testées

Les expériences effectuées sur les pondérations peuvent être regroupées en deux parties :

1. les expériences dont les pondérations sont basées sur la combinaison d'une pondération locale et d'une pondération globale ;
2. les expériences dont les pondérations sont basées sur les modèles DFR, en considérant les différentes combinaisons de sources d'informations et de normalisations possibles.

En effet, combiner des pondérations classiques et des pondérations DFR n'a pas de sens, chaque modèle DFR formant un tout.

**Pondérations locales et globales** Les pondérations locales utilisées sont les pondérations classiques du modèle vectoriel présentées en section 3.3.1.1, auxquelles on ajoute la pondération locale extraite de la formule probabiliste BM25 (section 3.3.2.2) et une pondération locale (en fait une normalisation par la taille du document) inspirée des normalisations utilisées avec les modèles DFR. Le tableau 3.7 résume les différentes pondérations locales utilisées. Ces dernières sont utilisées en association avec une pondération globale. Les pondérations globales sont : la pondération constante (équivalente à ne pas utiliser de pondération globale), l'idf, l'idf au carré (voir 3.3.1.2), l'idf probabiliste adapté de la formule BM25 (voir section 3.3.2.2), la pondération que nous avons proposée en section 3.4.1

et qui combine tf moyen et idf, et une version de cette dernière élevée au carré. Ces pondérations globales sont résumées dans le tableau 3.8. L'association d'une pondération locale $l_a$ et d'une pondération globale $g_b$ est noté $l_a g_b$ par la suite et est calculée comme le produit des deux pondérations : $l_a g_b(t_i, d_j) = l_a(t_i, d_j).g_b(t_i)$. Les différentes pondérations ainsi obtenues sont testées avec les distances $L_1$ et $L_2$, après normalisation adaptée des vecteurs.

| Identifiant | Description | Équation |
|---|---|---|
| $l_1(t_i, d_j)$ | Fréquence du terme tf | $\mathrm{tf}_{ij}$ |
| $l_2(t_i, d_j)$ | Logarithme de la fréquence | Équation 3.3 |
| $l_3(t_i, d_j)$ | Fréquence normalisée augmentée | Équation 3.4 |
| $l_4(t_i, d_j)$ | Facteur binaire | Équation 3.5 |
| $l_5(t_i, d_j)$ | Normalisation de type DFR | $\mathrm{tf}_{ij} \cdot \frac{\mathrm{dl}_{\mathrm{avg}}}{\mathrm{dl}_j}$ |
| $l_6(t_i, d_j)$ | tf au carré | Équation 3.6 |
| $l_7(t_i, d_j)$ | tf de BM25 | Équation 3.14 |

TAB. 3.7 – Pondérations locales d'un terme $t_i$ dans le document $d_j$.

| Identifiant | Description | Équation |
|---|---|---|
| $g_0(t_i)$ | Aucun poids | 1 |
| $g_1(t_i)$ | Fréquence Documentaire Inverse (idf) | Équation 3.7 |
| $g_2(t_i)$ | idf probabiliste | Équation 3.16 |
| $g_3(t_i)$ | idf au carré | Équation 3.8 |
| $g_4(t_i)$ | tf moyen * idf | Équation 3.29 |
| $g_5(t_i)$ | (tf moyen * idf) au carré | $[\overline{\mathrm{tf}_i} \log(\frac{N}{df_i})]^2$ |

TAB. 3.8 – Pondérations globales pour un terme $t_i$.

**Pondérations issues des modèles DFR** Nous avons pris en compte toutes les combinaisons possibles entre un modèle aléatoire (voir tableau 3.9), un modèle de gain d'information (voir tableau 3.5.5.1) et une normalisation de $\mathrm{tf}_{ij}$ (voir tableau 3.3). Chaque combinaison est représentée par la suite par une notation de la forme $XYZ$, où $X$ désigne le modèle aléatoire, $Y$ le modèle de gain et $Z$ la normalisation employés. Chaque pondération ainsi obtenue est testée avec les distances $L_1$ et $L_2$, ainsi que la mesure de similarité DFR classique définie par Amati et Van Rijsbergen de la manière suivante [AVR02] :

$$S(q, d_j) = \sum_{t_i \in q} \mathrm{tf}_{iq} \, .w(t_i, d_j) \tag{3.33}$$

où $w(t_i, d_j)$ est le poids DFR du terme $t_i$ dans le document $d_j$.

#### 3.5.5.2 Résultats

Les résultats de ces expérimentations sur les pondérations sont résumés en termes de gains par rapport à une pondération $l_1 g_0$[3] de base dans les figures 3.8 et 3.10 pour la distance $L_1$, et les figures 3.9 et 3.11 pour la distance $L_2$. Le tableau 3.11 donne également un aperçu des meilleures améliorations qu'il est possible d'obtenir pour chaque corpus.

[3] Nous nous situons par rapport à cette pondération qui est la plus simple possible, puisqu'elle correspond à l'utilisation la la fréquence des termes uniquement.

| Identifiant | Modèle pour $\text{Prob}_1$ | Approximation de $\text{Inf}_1$ |
|---|---|---|
| P | Bernoulli | Équation 3.19 |
| D | Bernoulli | Équation 3.20 |
| G | Bose-Einstein | Équation 3.22 |
| Be | Bose-Einstein | Équation 3.23 |
| In | Fréquence Documentaire Inverse | Équation 3.25 |
| $\text{In}_e$ | Fréquence Documentaire Inverse | Équation 3.26 |
| HG | Hypergéométrique | Équation 3.31 |

TAB. 3.9 – Modèles aléatoires testés et leurs approximations.

| Identifiant | $\text{Inf}_2$ |
|---|---|
| L | Équation 3.27 |
| B | Équation 3.28 |

TAB. 3.10 – Modèles de divergence

.

Les résultats détaillés de ces expérimentations, avec les résultats des tests de significativité statistique, sont regroupés dans l'annexe A.

Les deux observations principales que l'on peut formuler à partir de ces résultats sont :

- le gain de performance maximal que l'on peut obtenir en utilisant une pondération adaptée, par rapport à l'usage d'une pondération standard tf ($l_1 g_0$) est très limité, et pas toujours significatif statistiquement. Le tableau 3.11 montre que le gain obtenu n'excède 10% que dans le cas du corpus Oxford. Ce gain est particulièrement réduit dans le cas du corpus Caltech-101, corpus le plus difficile à traiter ;
- il n'y a pas de pondération optimale pour toutes les données, la meilleure pondération varie en fonction des corpus et des mesures de performances considérées. Il y a donc peu de chances qu'une de ces pondérations se révèle optimale sur tout nouveau jeu de données.

Nous discutons de l'effet des pondérations locales dans la section 3.5.6.2, et de l'effet des pondérations globales dans la section 3.5.6.3. Le cas particulier du corpus Oxford est quant à lui traité dans la section 3.5.7.1.

Concernant les pondérations issues des modèles DFR et leur association avec des distances dans un cadre vectoriel, deux remarques peuvent être faites :

- la différence entre les mesures de similarité DFR et l'utilisation de pondérations DFR avec la distance $L_2$ semble montrer que les requêtes ne doivent pas nécessairement être pondérées ni normalisées, contrairement à ce que nous avions proposé précédemment (voir section 3.1.5). Cela est dû au fait que, la mesure de similarité DFR n'étant pas normalisée, elle n'est pas strictement identique à une distance $L_2$. Nous avons néanmoins pu vérifier sur quelques expériences que, lors de l'utilisation d'une distance de Minkowski quelconque, la perte de performances était bien plus importante lorsque la requête n'était pas pondérée ni normalisée ;
- les scores obtenus avec les pondérations DFR sont sensiblement équivalents à ceux obtenus avec les autres pondérations testées. Ils sont, de plus, souvent très proches entre eux, et la différence est souvent non significative statistiquement (voir annexe A). Certains points de ces résultats sont néanmoins discutés en section 3.5.7.

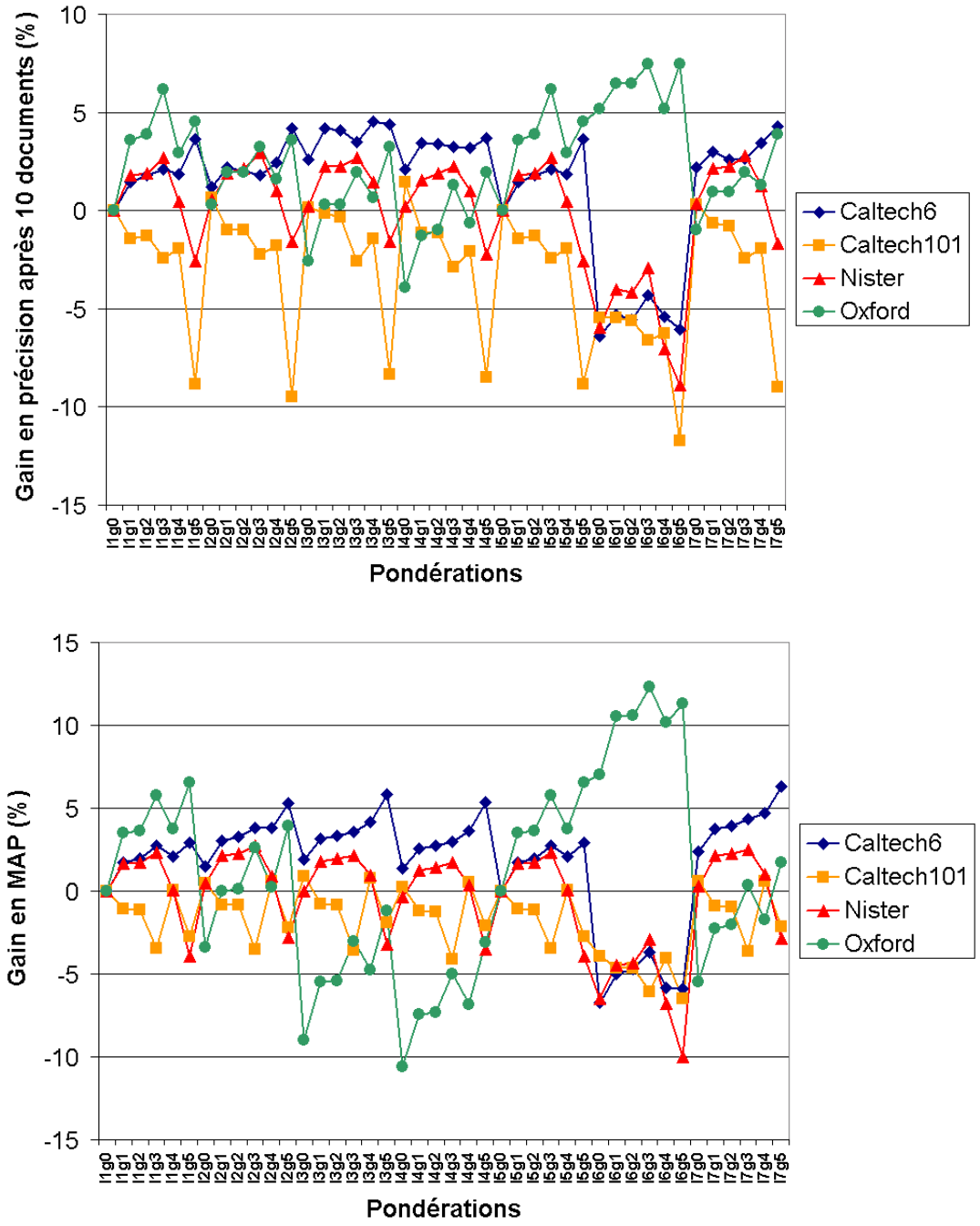



FIG. 3.8 – Gain de performances d'un système utilisant les pondérations des Tableaux 3.7 and 3.8 et la distance $L_1$, par rapport à la pondération de base $l_1g_0$.

### 3.5.6 Discussion

#### 3.5.6.1 Effet de $k$ sur l'usage des distances $L_k$

Pour le corpus Caltech-6 et Kentucky, l'effet du paramètre $k$ des distances de Minkowski est cohérent avec les observations d'Aggarwal *et al.* [AHK01] et, surtout, celles de Howarth et Rüger [HR05] : les valeurs faibles de $k$ améliorent la performance des systèmes de recherche d'images, jusqu'à un certain seuil en-deça duquel les performances s'écroulent.

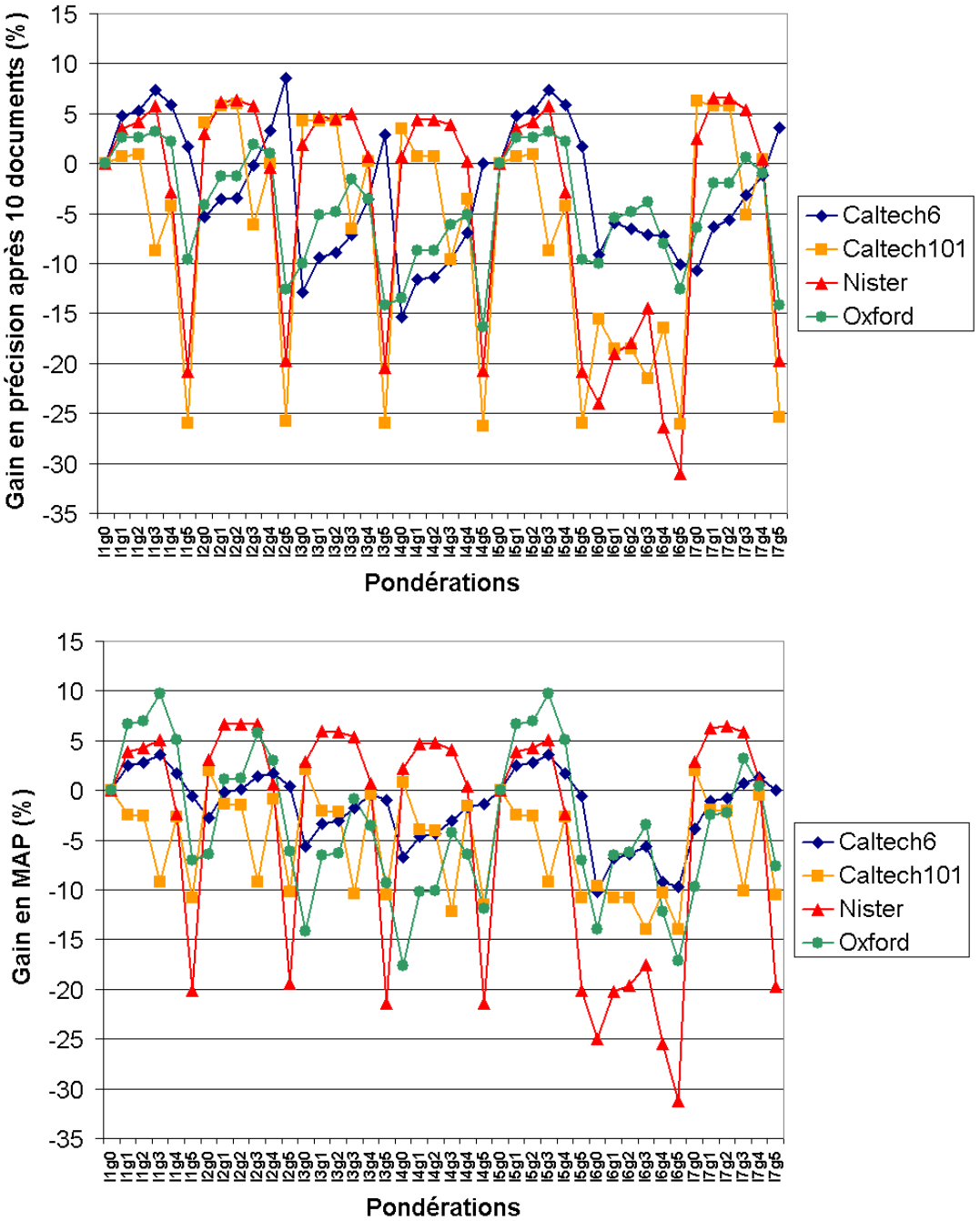



FIG. 3.9 – Gain de performances d'un système utilisant les pondérations des Tableaux 3.7 and 3.8 et la distance $L_2$, par rapport à la pondération de base $l_1g_0$.

Cependant, ces auteurs n'expliquent pas ce comportement. Dans nos expériences, la valeur optimale pour $k$ est $0,75$. L'influence de $k$ sur les distances $L_k$ est la suivante : les grandes valeurs de $k$ donnent plus d'importance aux distances locales (distances calculées au niveau d'une seule dimension des vecteurs) alors que des valeurs faibles donnent plus d'importance au seul fait que la valeur d'une dimension d'un vecteur soit nulle ou non, indépendemment de la fréquence (voir figure 3.12). Le fait que les performances s'écroulent en-deça d'un certain seuil montre qu'il faut trouver un bon compromis entre le fait de comparer les

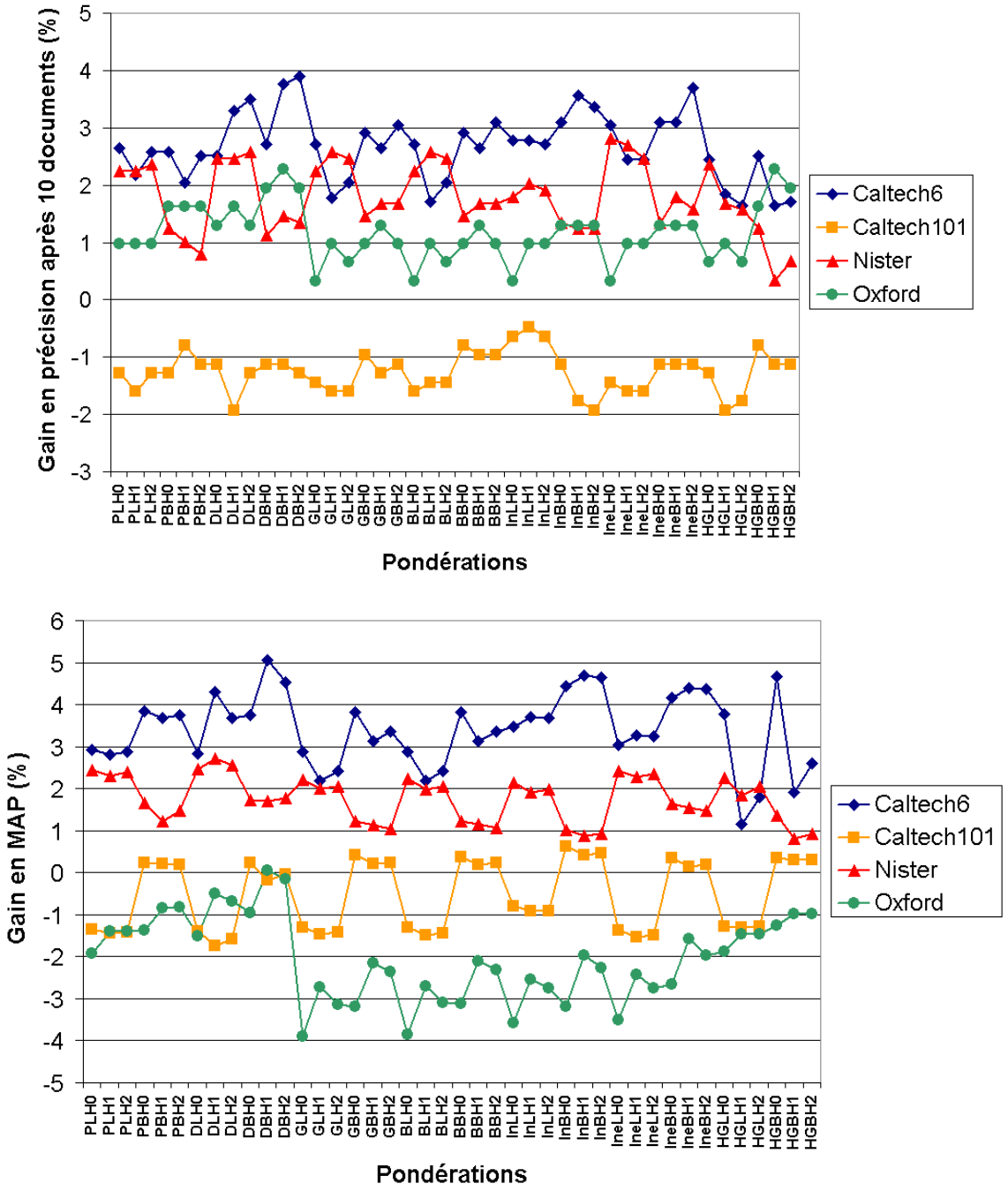



FIG. 3.10 – Performances d'un système utilisant les pondérations DFR et la distance $L_1$, par rapport à la pondération de base $l_1 g_0$.

fréquences des termes et le fait de comparer uniquement leur présence.

Pour Caltech-101, les distances telles que $k < 1$ apportent un gain beaucoup plus limité que dans le cas des corpus Caltech-6 et Kentucky. Une première différence majeure entre Caltech-101 et les autres corpus tient à la taille du vocabulaire : celui utilisé pour Caltech-101 est beaucoup plus grand, et produit donc des vecteurs descripteurs beaucoup plus creux (à nombre de mots visuels par image égal). Cependant, Howarth *et al.* ont montré que les distances fractionnelles permettent un gain de performances plus important sur

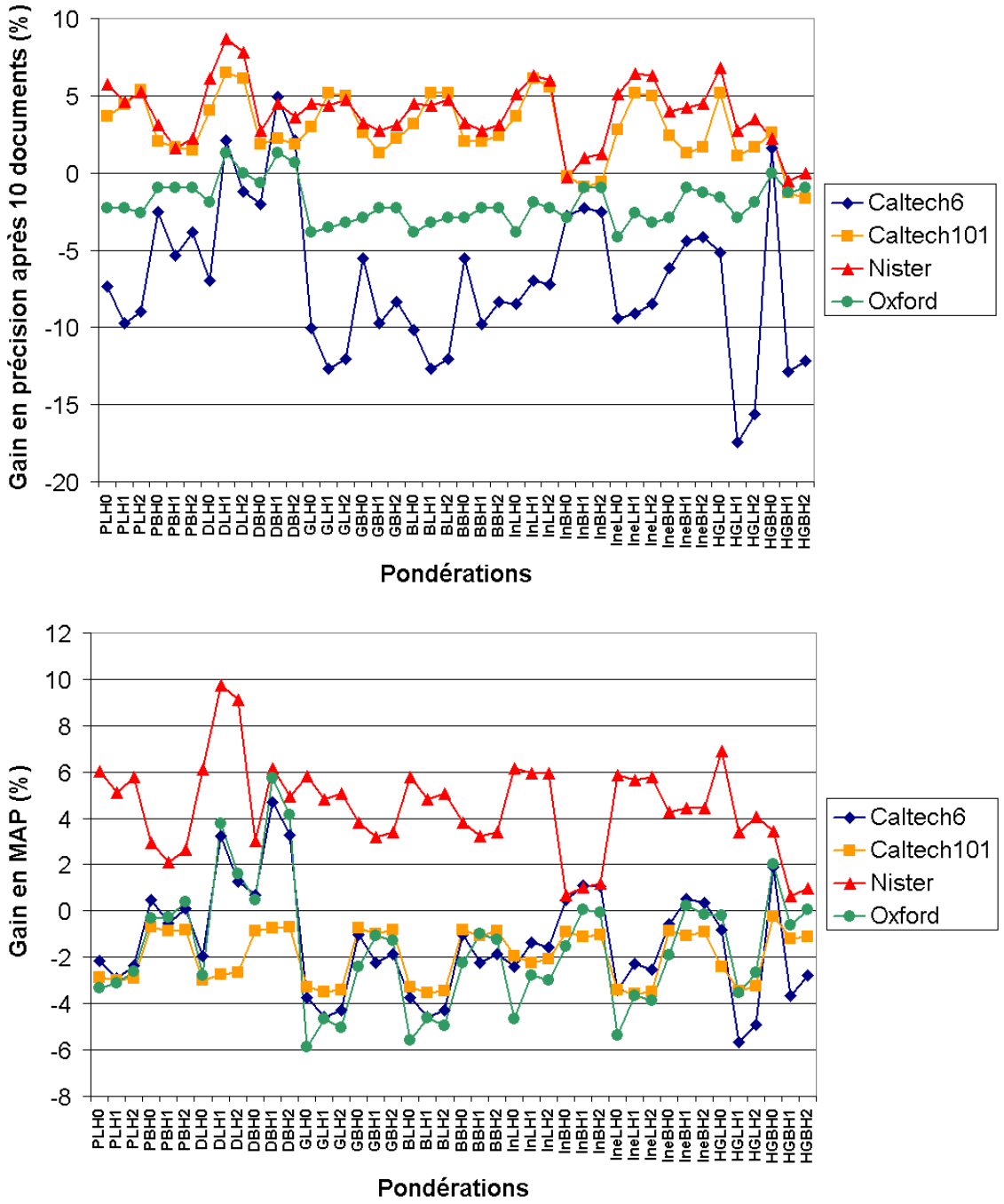



FIG. 3.11 – Performances d'un système utilisant les pondérations DFR et la distance $L_2$, par rapport à la pondération de base $l_1g_0$.

des vecteurs creux que sur des vecteurs denses [HR05], le fait que les vecteurs soient plus creux ne justifie donc pas les résultats obtenus sur Caltech-101. La seconde différence est que le vocabulaire obtenu est beaucoup plus bruité. La taille du vocabulaire, les fortes variations entre les différentes images d'une même catégorie et les ressemblances existant entre certaines catégories font que la probabilité pour qu'un descripteur soit assigné à un autre mot visuel que celui qui aurait été souhaitable est beaucoup plus élevée que sur les autres corpus. Les vecteurs décrivant les images contiennent donc beaucoup de bruit, ce qui

| | p@4 | p@5 | p@10 | p@20 | p@50 | p@100 | MAP |
|---|---|---|---|---|---|---|---|
| Caltech6 | N/A | N/A | +4.3% | +6.3% | +8.0% | +8.5% | +6.3% |
| Caltech101 | N/A | N/A | +1.5% | +1.1% | +0.9% | +0.7% | +0.9% |
| Kentucky | +3.3% | N/A | N/A | N/A | N/A | N/A | +2.7% |
| Oxford | N/A | +3.0% | +7.5% | +9.3% | +10.0% | N/A | +12.3% |

TAB. 3.11 – Amélioration maximale des performances par rapport au poids de référence $l_1 g_0$, en utilisant la distance $L_1$.

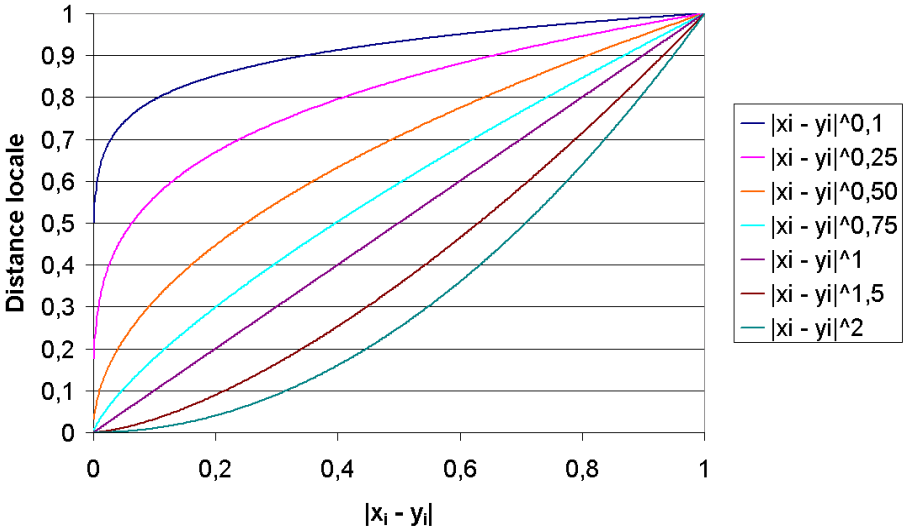



FIG. 3.12 – Importance accordée à la distance locale en fonction de la valeur de $k$.

tend à rendre les distances $L_k$ équivalentes, quelle que soit la valeur de $k$, comme l'avaient déjà observé Aggarwal *et al.* [AHK01]. Malgré cela, le vocabulaire employé nous permettait, lors de nos tests pour choisir la taille du vocabulaire, d'avoir de meilleures performances globales qu'avec des vocabulaires de plus petites tailles : il est donc préférable, malgré les inconvénients évoqués ci-dessus, d'utiliser un grand vocabulaire, ce qui nous rapproche d'un cas de mise en correspondance directe des descripteurs locaux, que d'utiliser un vocabulaire plus petit en espérant obtenir des facultés de généralisation liées aux regroupements plus larges des descripteurs locaux qui résultent d'un faible nombre de *clusters*.

#### 3.5.6.2 Pondérations locales

La meilleure pondération locale dépend fortement du corpus utilisé. Si l'on emploie une distance $L_1$, on observe que :

- pour Caltech-6, $l_3$ et $l_7$ sont les meilleures pondérations locales, $l_4$ venant ensuite ;
- pour Caltech-101, toutes les pondérations sont équivalentes, les différences observées n'étant pas statistiquement significatives (voir annexe A), à l'exception de $l_6$ qui donne les pires résultats ;
- pour Kentucky, $l_2$ et $l_7$ sont les meilleures ;
- pour Oxford, c'est $l_6$ qui est la meilleure, suivie par $l_5$ et $l_1$ qui donnent des résultats équivalents. Les autres pondérations réduisent les performances du système.

La figure 3.13 présente la moyenne et l'écart-type des fréquences des termes dans les documents où ils apparaissent, pour chaque corpus. Pour Caltech-6 et Kentucky, la fréquence moyenne des mots visuels se situe entre 1 et 2 pour la grande majorité des mots

visuels, et augmente rapidement pour les quelques autres. De plus, plusieurs mots ont une fréquence égale à 1, mais très peu d'entre eux sont des hapax[4] (voir tableau 3.12). Cette distribution des fréquences explique l'efficacité de pondérations locales comme $l_3$, $l_7$ ou $l_2$ : en réduisant la fréquence des mots visuels les plus fréquents, elles réduisent également les distances locales au niveau de ces mots visuels, et évitent ainsi que ces dernières ne représentent l'essentiel de la distance globale. Elles permettent ainsi aux mots visuels de fréquence faible de conserver une influence sur la distance globale calculée entre les documents. Cet effet est cohérent avec les résultats déjà observés sur les distances. Dans le corpus Caltech-101, la plupart des mots visuels ont une fréquence moyenne comprise entre 1 et 1,2, et quelques mots visuels une fréquence légèrement plus élevée. De plus, il y a proportionnellement beaucoup plus de mots visuels ayant une fréquence égale à 1, dont un tiers environ sont des hapax (voir tableau 3.12). Cette distribution des fréquences peut s'expliquer notamment par la taille du vocabulaire : le nombre de mots visuels étant important, il est logique que moins de descripteurs soient assignés à chacun. Néanmoins, cet effet peut également être lié au bruit dans l'assignation des descripteurs, déjà évoqué plus haut. Ces fréquences expliquent le manque d'effet de la plupart des pondérations locales. En effet, leur effet est plus limité sur les basses fréquences que sur les hautes fréquences. En particulier, quand $\mathrm{tf}_{ij} = 1$, les pondérations $l_1$, $l_2$, $l_4$ et $l_6$ sont parfaitement identiques. Malgré cela, $l_6$ fait chuter les résultats car, de même que la distance $L_2$, cette pondération donne plus d'importance aux quelques mots visuels ayant une fréquence $\mathrm{tf}_{ij} > 2$, au détriment de la majorité des mots visuels. Les fréquences moyennes sur le corpus Oxford suivent un modèle assez similaire à celles de Caltech-101, avec un écart-type plus important. Ce corpus génère cependant des résultats contradictoires qui sont traités dans la section 3.5.7.1.

| | Hapax | $\overline{\mathrm{tf}_i} = 1$ |
|---|---|---|
| Caltech-6 | 0 | 17 |
| Caltech-101 | 3021 | 9790 |
| Kentucky | 56 | 358 |
| Oxford | 1434 | 4206 |

TAB. 3.12 – Comparaison entre hapax et mots visuels tels que $\overline{\mathrm{tf}_i} = 1$.

#### 3.5.6.3 Pondérations globales

Comme pour les pondérations locales, il est difficile de déterminer quelle est la meilleure pondération globale :

- pour Caltech-6, ce sont les pondérations $g_4$ et $g_5$ que nous avons proposées qui donnent les meilleurs résultas ;
- pour Kentucky, l'idf au carré $g_3$ est le meilleur, mais la différence avec $g_1$ et $g_2$ est minime ;
- pour Caltech-101, les résultats sont similaires pour toutes les pondérations, et, souvent, ne pas utiliser de pondération globale reste le meilleur choix ;
- pour Oxford, ce sont $g_3$ et $g_5$ qui fournissent les meilleurs résultats.

D'une manière générale, aucune pondération globale n'est meilleure que toutes les autres, bien que l'idf au carré $g_3$ permette d'obtenir souvent de bonnes performances. Nos pondérations incluant la fréquence moyenne et l'idf fournissent également de bons résultats sur deux des corpus testés, Caltech-6 et Oxford. Cela tient à la nature des requêtes utilisées, et est discuté dans la section 3.5.7.1.

[4]Mots n'apparaissant qu'une fois dans la collection.

Le résultat le plus important ici est obtenu sur le corpus Caltech-101 : on voit que, dans le cas de corpus très variés, difficiles, comme Caltech-101, il est préférable de n'utiliser aucune pondération globale. Cela peut s'expliquer par la difficulté, quand l'assignation des mots visuels est très bruitée, d'exploiter l'information statistique sur les mots visuels pour en déduire leur importance dans la collection. Ce bruit lors de la phase de quantification peut être dû aussi bien aux faiblesses en grandes dimensions de algorithmes de *clustering* qu'à la grande variabilité, dans ce type de données, entre des descripteurs similaires sémantiquement (c'est-à-dire représentant les mêmes parties d'objets d'une même classe). Ce résultat va à l'encontre de l'habitude d'utiliser une pondération globale de type idf, systématique chez la plupart des auteurs.

### 3.5.7 Pondérations DFR

Les résultats obtenus avec les pondérations inspirées des modèles DFR sont très proches les uns des autres et ne sont généralement pas statistiquement significatifs. Les meilleurs résultats ont été obtenus avec l'approximation $D$ du modèle de Bernoulli, ainsi qu'avec les modèles basés sur la fréquence documentaire inverse, In et $\mathrm{In}_e$. Le modèle DLH1 donne les meilleurs résultats sur le corpus Kentucky, alors que c'est plutôt le modèle de divergence $B$ qui donne les meilleurs résultats sur les autres corpus. Il n'y a donc ici pas vraiment de modèle DFR qui soit optimal dans tous les cas, bien que certains se détachent plus clairement que ce n'était le cas parmi les pondérations testées précédemment (combinaisons d'une pondération locale et d'une globale). Néanmoins, les meilleurs résultats obtenus par ces dernières sont toujours meilleurs que les résultats obtenus avec les pondérations DFR. En plus de ces conclusions générales, nous pouvons faire deux remarques quant au modèle hypergéométrique que nous avons proposé :

- les meilleurs résultats obtenus l'ont été en utilisant la normalisation neutre de $\mathrm{tf}_{ij}$ H0. Ceci s'explique par le fait que ce modèle, contrairement aux autres, prend en compte la longueur des documents, ce qui rend la normalisation de $\mathrm{tf}_{ij}$ superflue ;
- bien qu'il soit plus séduisant en théorie, le modèle hypergéométrique offre des résultats similaires au modèle de Bernoulli. Ceci peut s'expliquer par le fait que ces deux modèles peuvent se comporter de manière équivalente sous certaines conditions, en particulier lorsque le nombre de termes sélectionné à chaque tirage (ici, la longueur du document $\mathrm{dl}_j$) est négligeable par rapport au nombre de termes qu'il est possible de tirer (ici, $CF*$), ce qui est le cas dans ces expériences.

#### 3.5.7.1 Influence de la nature des requêtes

Les résultats obtenus sur le corpus Oxford sont complètement opposés à ceux obtenus sur les autres corpus :

- les distances $L_k$ les meilleures sont celles avec un paramètre $k$ plutôt élevé (autour de 2) ;
- la pondération locale $l_6$ fonctionne mieux que les autres, sous distance $L_1$, alors qu'elle donne incontestablement les plus mauvais résultats dans les autres corpus.

Ces résultats montrent que ce corpus a des propriétés bien différentes des autres. Plus précisément, les requêtes utilisées sont particulièrement spécifiques. La figure 3.14 présente les onze bâtiments apparaissant dans les 55 requêtes du corpus. Le point commun à toutes ces requêtes est qu'elles possèdent un grand nombre d'éléments répétés (fenêtres, arches, décorations...). Ces éléments vont nécessairement provoquer la détection de nombreux mots visuels similaires dans l'image. Ces mots visuels à très haute fréquence sont particulièrement pertinents pour décrire ces requêtes, à l'inverse des autres corpus où, comme

nous l'avons montré précédemment, il est préférable de réduire les fréquences de mots visuels élevées. Ceci explique le bon comportement obtenu par la pondération $l_6$ qui donne encore plus d'importance à ces mots visuels spécifiques, ainsi que celui de la distance $L_2$ qui donne plus d'importance aux distances locales élevées générées par ces mots visuels de fréquence élevée. Ceci peut expliquer également, dans une certaine mesure, les résultats obtenus avec les pondérations globales $g_4$ et $g_5$, qui donnent également plus d'importance à ces mots visuels. Le même phénomène existe, dans une moindre mesure, dans le corpus Caltech-6 dont certaines catégories d'objets possèdent des parties répétées (roues des motos, yeux des visages, phares des voitures), comme l'attestent les bons résultats obtenus par les pondérations $g_4$ et $g_5$. Dans ce dernier corpus, cependant, les résultats obtenus avec les pondérations locales et distances démontrent néanmoins que cet effet est limité, et que ces mots répétés ne sont pas les seuls à avoir de l'importance dans la description des images.

Le résultat essentiel ici est l'influence de la nature des requêtes sur le choix des paramètres du système. Les propriétés des mots visuels les plus pertinents varient en effet en fonction du contenu visuel de ces requêtes, ce qui rend difficile toute optimisation de ce type de système de recherche d'images dans le cas général. Ce résultat remet également en cause l'habitude de nombreux auteurs de n'évaluer leurs travaux que sur un jeu de données, dont les requêtes sont de plus, en général, d'une nature similaire.

#### 3.5.7.2 Relation entre distances de Minkowski et pondérations

Comme nous l'avons déjà évoqué dans les discussion précédentes, les résultats suggèrent une relation entre les différents types de pondérations utilisés et les différentes distances de Minkowski : les pondérations qui limitent les hautes fréquences des termes tendent à donner des résultats similaires aux distances $L_k$ dont le paramètre $k$ est faible, alors que celles qui favorisent les hautes fréquences des termes se comportent comme les distances ayant un $k$ élevé. Ceci s'explique au niveau des distances locales : un paramètre $k$ peu élevé diminue les distances locales élevées, de même que limiter les hautes fréquences des termes limitera l'amplitude de ces distances locales, et inversement. Il est donc possible d'approcher une distance avec un paramètre $k$ réel par une pondération adéquate. Ceci pourrait permettre de contourner un des inconvénients majeurs des distances fractionnelles (distances $L_k$ telles que $k < 1$) : le non-respect de l'inégalité triangulaire [AHK01]. Au lieu d'utiliser une distance fractionnelle, il est ainsi possible d'obtenir des résultats similaires en utilisant une pondération adaptée et une distance $L_1$, ce qui permet d'utiliser malgré tout les techniques de recherche rapide de plus proches voisins reposant sur la propriété d'inégalité triangulaire des distances [BFM+96].

## 3.6 Travaux connexes

### 3.6.1 Mots visuels et distances

Le problème du choix de distance pour la recherche d'images à base de mots visuels a été peu traité dans la littérature. Nister et Stewenius ont comparé les distance $L_1$ et $L_2$ uniquement, sur leur corpus Kentucky [NS06]. Leur conclusion était que $L_1$ se comportait mieux, sans qu'ils n'expliquent ce résultat. Les autres travaux sur les distances consistent plutôt à définir de nouvelles distances, adaptées à certaines spécificités des mots visuels. Ainsi, Jégou *et al.* [JDS08] ont proposé une distance qui exploite l'étape d'assignation des descripteurs aux mots visuels pour affiner la mise en correspondance de ces derniers. Jiang *et al.* [JN09] ont également proposé une distance se basant sur ce type de propriétés.

Enfin, Jégou *et al.* [JHS07] ont également proposé une distance qui repose sur certaines propriétés d'asymétrie de la recherche par plus proches voisins.

### 3.6.2 Mots visuels et pondérations

Les travaux au sujet des pondérations des mots visuels sont plus nombreux que ceux abordant le choix d'une distance, mais rares sont ceux qui ont cherché à appliquer les pondérations habituelles de la recherche d'information aux mots visuels. À notre connaissance, seuls Jiang *et al.* ont proposé un comparatif entre la pondération binaire et la pondération tf. idf, qu'ils ont toutes deux comparées avec une nouvelle pondération qu'ils ont proposée, qui effectue un assignement flou des descripteurs entre leurs deux mots visuels les plus proches. Ils se sont placés pour cela successivement dans le cadre de la catégorisation d'images [JNY07], puis de la recherche d'images [YJHN07]. Philbin *et al.* ont proposé des travaux similaires, qui exploitent la phase d'assignation des descripteurs aux mots visuels pour affiner la mise en correspondance de ceux-ci [PCI+08]. Enfin, plus récemment, Chen *et al.* [CHS09] ont proposé une pondération intégrant des informations géométriques issues d'une segmentation de l'image basée sur la couleur. Parmi ces travaux, aucun n'avait abordé ce problème sous l'angle des fréquences intra-documentaires et extra-documentaires des mots visuels, en exploitant vraiment les travaux existant en recherche d'information textuelle, comme nous l'avons fait ici. Á notre connaissance, seul Jégou *et al.* [JDS09] ont étudié la distribution statistique des mots visuels avec des outils provenant de l'indexation de textes ; ils ont, en effet, étudié le phénomène connu dans la communauté du texte sous le nom de *burstiness*. Cet effet désigne le fait qu'un mot qui est apparu souvent dans un texte a une plus forte probabilité d'y réapparaître, phénomène que l'on rencontre également en image. Ils corrigent donc leur score de similarité entre images en cherchant à réduire l'influence de ces mots visuels spécifiques. Leurs résultats rejoignent les nôtres, qui montrent également que réduire les plus grandes fréquences de mots visuels améliorent les résultats. Cependant, nos résultats montrent en plus qu'exploiter ces hautes fréquences peut aussi être bénéfique, en fonction de la nature de la requête.

## 3.7 Conclusion

Nous avons étudié dans ce chapitre l'intérêt des *stop-lists* et des pondérations pour la recherche d'images basée sur les mots visuels, ainsi que des différentes distances de Minkowski. La principale conclusion de cette étude est qu'il n'existe pas de pondération ni de distance optimales pour tous types d'images, mais que le type de requête pris en compte influe fortement sur la pondération ou la distance à employer. Ces résultats remettent en cause les habitudes de la majorité des auteurs qui considèrent généralement que la pondération tf. idf doit être employée dans tous les cas et qui ne testent souvent leurs systèmes que sur un seul corpus d'images dont les requêtes sont homogènes.

Une étude plus approfondie des résultats permet néanmoins de mettre au jour d'autres points intéressants :

- l'existence d'un lien entre le choix d'une distance de Minkowski et le choix d'une pondération ;
- les limites inhérentes aux systèmes à base de mots visuels, dûes à la phase de quantification des descripteurs qui semble apporter du bruit à la description des images, et non de la robustesse comme il est souvent supposé. Ce résultat recoupe les conclusions d'autres travaux comme ceux de Boiman *et al.* sur la catégorisation d'images à base de mots visuels [BSI08], et éclaire les bonnes performances des méthodes affinant la

phase de quantification des descripteurs [JN09, JDS08, PCI+08]. Cette conclusion pourrait être confirmée par la comparaison des performances d'un système de recherche d'images à base de mots visuels et d'un système utilisant des descripteurs non quantifiés.

Les méthodes de TAL que nous avons utilisées ici ne permettent pas d'améliorer d'une manière générale les systèmes de recherche d'images à base de mots visuels, mais mettent à jour des propriétés intéressantes des mots visuels, des pondérations et des distances. Il apparaît notamment que le choix d'une pondération locale (ou d'une distance) adéquate dépend essentiellement de la nature du contenu visuel de la requête considérée. Ceci ouvre une perspective intéressante : comment choisir automatiquement la pondération locale (ou la distance) adaptée en fonction de la requête ? Le même problème se pose dans le cas des pondérations globales : si ces dernières sont particulièrement inefficaces pour les corpus catégorisés comme Caltech-101, c'est peut-être parce qu'il n'existe pas un ensemble de pondérations optimal pour le corpus mais pour chaque catégorie. De tels ensembles de pondérations, adaptés aux catégories, pourraient être obtenus grâce à la formulation initiale de l'idf probabiliste, qui inclut des informations de pertinence [SJWR00b]. Ici encore, le problème du choix de la pondération adaptée à la requête se pose.

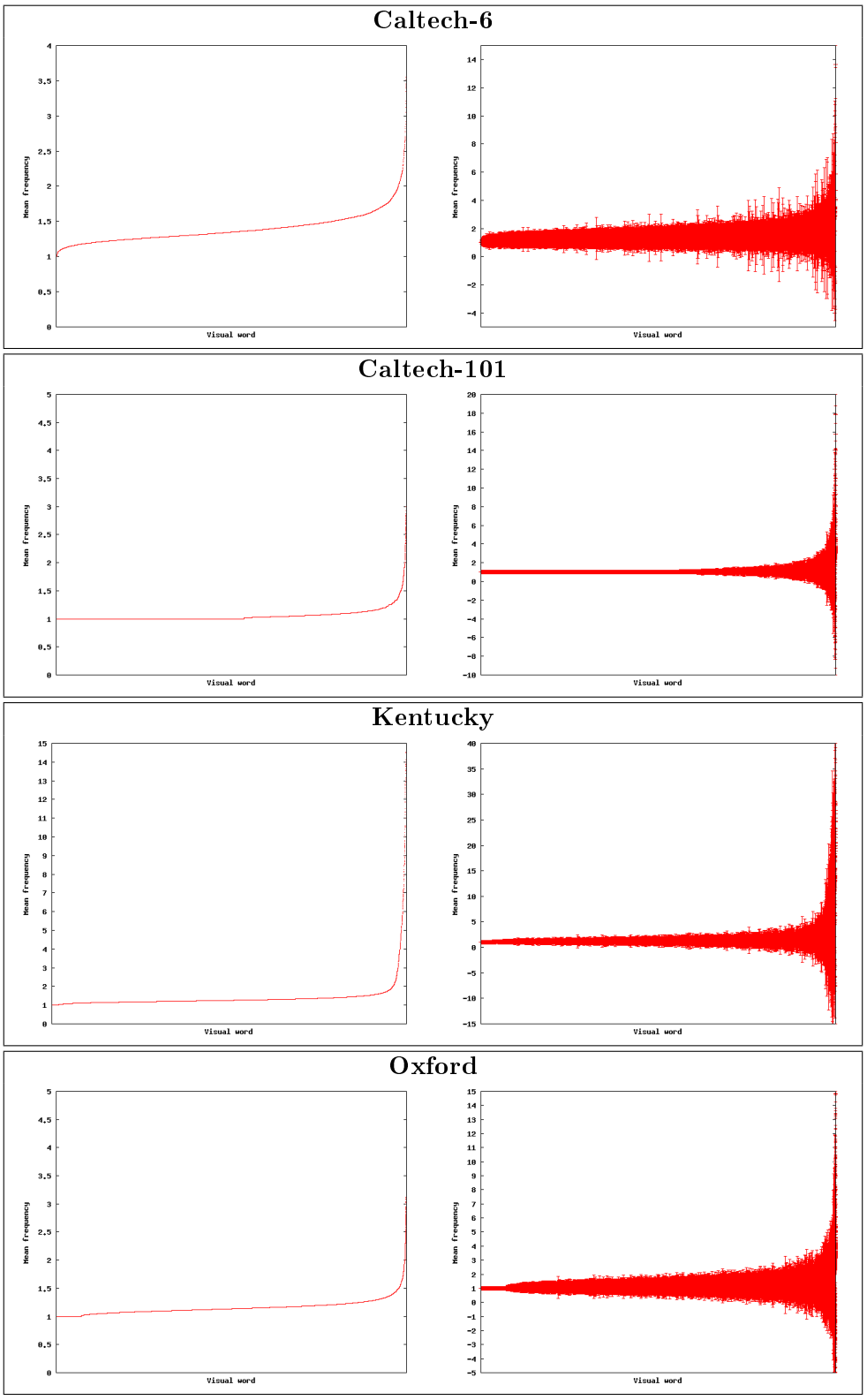



FIG. 3.13 – Fréquence moyenne et écart-type de fréquence des mots visuels.

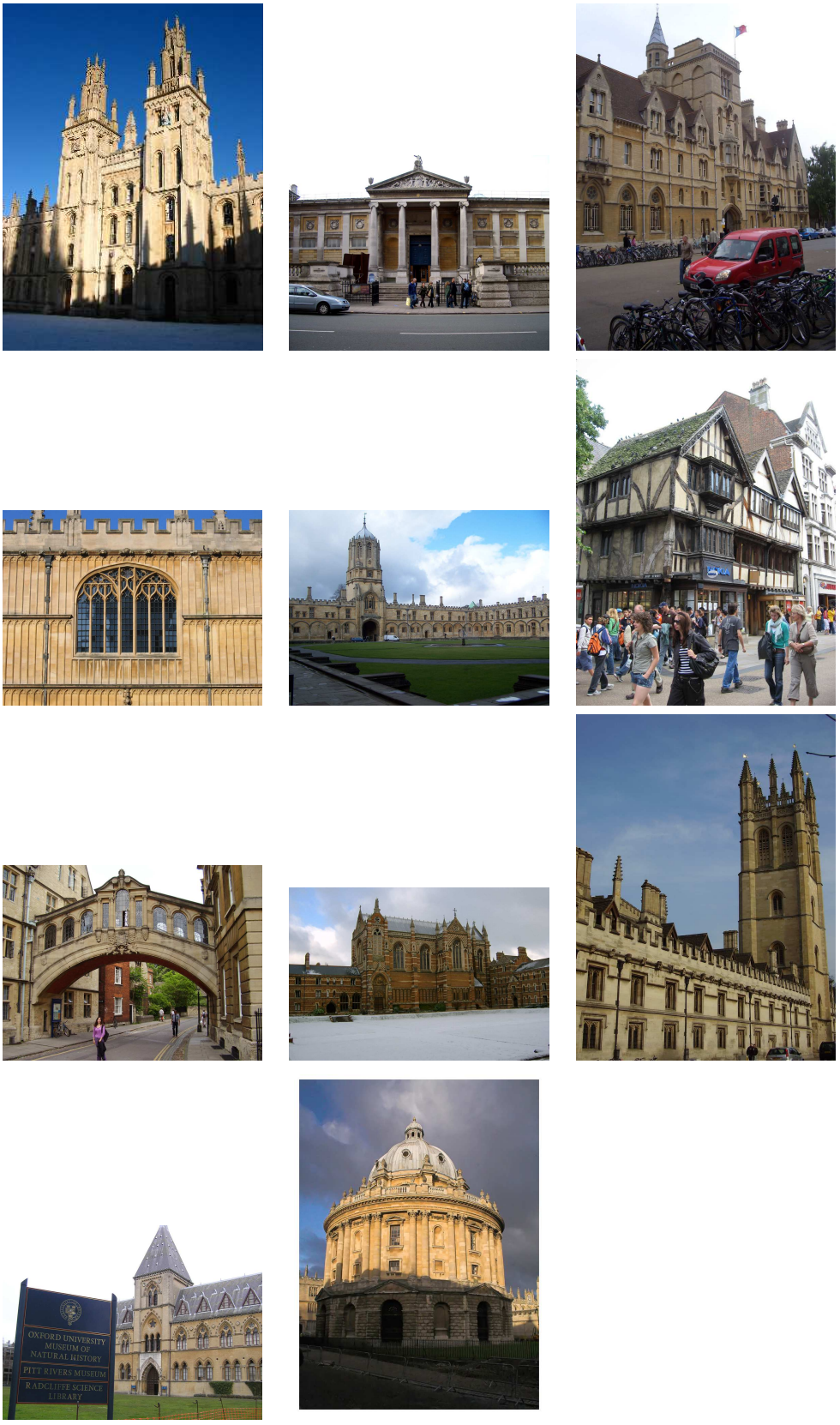



FIG. 3.14 – Images requêtes de la base d'images Oxford. Les bâtiments recherchés contiennent de nombreux éléments répétés (fenêtres et éléments architecturaux divers) qui leur sont caractéristiques.

# Chapitre 4

# Modèles de langues et images

Les modèles de recherche d'information que nous avons étudiés dans le chapitre précédent reposent tous sur l'hypothèse d'indépendance des termes, qui ne donnent de sens qu'aux mots pris séparément. Cette hypothèse, bien que souvent employée, est réductrice et ne permet pas de prendre en compte toute la complexité des langues naturelles qui contiennent de nombreux syntagmes, regroupements de mots dont le sens est différent du sens de chacun de ses composants pris isolément. Ainsi, par exemple, le syntagme *Maison Blanche* ne désigne pas simplement une habitation de couleur blanche mais également la résidence officielle des présidents des États-Unis d'Amérique. Comme nous l'avions évoqué dans la section 3.1.2 page 64, adopter cette hypothèse est particulièrement préjudiciable lorsqu'il s'agit de représenter les images, car, dans la quasi-totalité des cas, un objet ne correspond pas à un mot visuel unique mais à un ensemble de mots visuels ; nous avions de plus montré que chacun des mots visuels de cet ensemble n'était lui-même pas spécifique à un type d'objet donné mais pouvait apparaître dans différents objets, indistinctement (voir tableau 3.1 page 65). Dans ce chapitre, nous cherchons à aller au-delà de l'hypothèse d'indépendance des termes dans le cadre de la représentation des images à base de mots visuels (second des deux axes de notre étude sur l'utilisation du TAL dans le contexte des mots visuels, comme nous l'avions annoncé au chapitre 2). Nous nous appuyons pour cela sur le formalisme des modèles de langues, qui est un outil classique du traitement automatique des langues permettant de modéliser les langages naturels non pas comme des ensembles de mots indépendants, mais comme des séquences de mots inter-dépendants. Comme nous l'avions précisé en section 2.3.1.2 page 59, le fait que les mots visuels s'organisent dans un plan en deux dimensions et non sous forme de séquences nous pousse d'abord à représenter les images sous forme de séquences adaptées à l'usage des modèles de langues : les phrases visuelles.

Ce chapitre s'organise ainsi : nous situons d'abord les bases du fonctionnement des modèles de langues, puis nous décrivons notre méthode pour constituer des phrases visuelles et nous validons notre approche expérimentalement dans un cadre de classification d'images. Enfin, nous nous situons par rapport à quelques travaux proches de ceux-ci, puis concluons.

## 4.1 Modèles de langues

### 4.1.1 Fonctionnement

L'objectif des modèles de langues est d'obtenir une représentation des phrases qui tienne compte de relations de dépendance entre les mots adjacents, ce qui permet de considérer des

groupes de mots qui peuvent avoir un sens différents de leur constituants pris séparément. Ainsi, dans la phrase *Un volcan entre en éruption au sud de l'Islande*, un modèle de langues sera capable de modéliser l'expression *entre en éruption*, où le mot entre prend un sens différent de celui qu'il a habituellement, ou encore *sud de l'Islande*, qui est plus précis que les mots *sud* et *Islande* pris séparément.

#### 4.1.1.1 Modélisation des séquences de termes

Un modèle de langues associe à chaque terme $w_i$ d'une séquence de $n$ termes $s = w_1 w_2 \ldots w_i \ldots w_k$ la probabilité que le terme $w_i$ apparaisse précédé de la séquence de termes $w_1 w_2 \ldots w_{i-1}$ : $\Pr(w_i | w_1 w_2 \ldots w_{i-1})$. Ainsi, dans notre exemple, le mot *éruption* sera associé à la probabilité $\Pr(\textit{éruption} | \textit{Un volcan entre en})$, et le mot *sud* à la probabilité $\Pr(\textit{éruption} | \textit{Un volcan entre en éruption au})$. Dans un corpus donné, chaque mot $w_i$ sera représenté par autant de probabilités différentes qu'il existe de séquences de mots différentes pouvant le précéder, séquences qui correspondent à autant de contextes dans lesquels $w_i$ peut prendre des sens différents. En utilisant les probabilités conditionnelles d'occurrence des différents termes apparaissant dans la séquence $s = w_1 w_2 \ldots w_k$, il est possible de définir la probabilité d'occurrence de la séquence $s$ elle-même, de la manière suivante :

$$\Pr(s) = \prod_{i=1}^{k} \Pr(w_i | w_1 \ldots w_{i-1}) \tag{4.1}$$

Ainsi, dans notre exemple, la phrase *Un volcan entre en éruption au sud de l'Islande* sera représentée par la probabilité suivante :

$$\begin{aligned} &\Pr(\textit{Un volcan entre en éruption au sud de l'Islande}) = \\ &\quad \Pr(\textit{Un}) \times \Pr(\textit{volcan} | \textit{Un}) \times \Pr(\textit{entre} | \textit{Un volcan}) \times \ldots \times \\ &\quad \Pr(\textit{Islande} | \textit{Un volcan entre en éruption au sud de}) \end{aligned}$$

En se basant sur cette modélisation pour de séquences de termes, un document $d$ peut être défini de deux manières :

- soit en le considérant comme une unique séquence de mots $s$, indépendamment d'une éventuelle ponctuation. C'est par exemple le cas en reconnaissance de la parole, où la ponctuation n'est pas disponible.
- soit comme une succession de séquences $d = s_1 \ldots s_m$ indépendantes, séparées par des signes de ponctuation. Dans ce cas, la probabilité d'appartenance du document $d$ au modèle de langues $\mathcal{L}$ est :

$$\Pr = \prod_{i=1}^{m} \Pr(s_i) \tag{4.2}$$

Cette modélisation n'est cependant pas réaliste d'un point de vue pratique. En effet, le nombre de phrases d'une langue étant potentiellement infini, il n'est pas possible de prendre en compte tous les contextes possibles pour chaque mot, pour au moins deux raisons :

- la mémoire limitée des machines ;
- la taille limitée des corpus utilisés pour estimer les valeurs des probabilités. Cela pose un problème de robustesse du modèle, les probabilités obtenues ayant peu de chances d'être exhaustives par rapport au nombre de cas envisageables, et de fortes chances d'être mal estimées par rapport à la réalité.

Ces obstacles sont contournés en utilisant une approximation consistant à limiter la taille maximale du contexte pris en compte pour chaque $w_i$ à $n-1$ termes. Le modèle de langues ne prend ainsi en considération que des séquences de taille fixe $n$, nommées $n$-grammes, ce qui permet de réduire de manière considérable le nombre total de probabilités à calculer. Dans un modèle de langues $n-gramme$ $\mathcal{L}$, la probabilité d'occurrence d'une séquence $s = w_1 \dots w_k$ devient alors :

$$\Pr(d) \approx \prod_{i=1}^{k} \Pr(w_i | w_{i-n+1} \dots w_{i-1}) \tag{4.3}$$

Si l'on reprend notre exemple initial, la probabilité d'occurrence de la phrase *Un volcan entre en éruption en Islande* modélisée par un modèle bigramme ($n = 2$) sera :

$$\begin{aligned}&\Pr(\textit{Un volcan entre en éruption au sud de l'Islande}) = \\ &\quad \Pr(\textit{Un}) \times \Pr(\textit{volcan}|\textit{Un}) \times \Pr(\textit{entre}|\textit{volcan}) \times \Pr(\textit{en}|\textit{entre}) \times \dots \times \Pr(\textit{Islande}|\textit{de})\end{aligned}$$

Enfin, les probabilités des différents $n$-grammes sont estimées à l'aide d'un corpus $\mathcal{T}$, en effectuant simplement un comptage des occurrences des différents $n$-grammes possibles dans le corpus :

$$\Pr(w_n | w_1 w_2 \dots w_{n-1}) = \frac{C(w_1 w_2 \dots w_{n-1} w_n)}{\sum\limits_{w_i \in \mathcal{T}} C(w_1 w_2 \dots w_{n-1} w_i)} \tag{4.4}$$

où $C(w_1 w_2 \dots w_n)$ désigne le nombre d'occurrences de la séquence de termes $w_1 w_2 \dots w_n$ dans le corpus $\mathcal{T}$.

#### 4.1.1.2 Lissage : principe

Une limite importante de cette méthode d'estimation est qu'un $n$-gramme qui n'apparaît jamais dans le corpus d'apprentissage aura définitivement une probabilité nulle dans le modèle de langues obtenu à partir de ce corpus, alors qu'il est possible que cette probabilité ne soit pas nulle dans le cas réel, car il est impossible de garantir qu'un corpus donné contienne tous les $n$-grammes envisageables. Cette limite est contournée par l'usage de méthodes de lissage (*smoothing*) qui redistribuent une petite part des probabilités affectées $n$-grammes connus aux $n$-grammes inconnus, de sorte à attribuer à ces derniers des probabilités minimales mais non-nulles.

Le transfert d'une partie des masses de probabilités attribuées aux $n$-grammes présents dans le corpus d'apprentissage vers les $n$-grammes inconnus nécessite deux éléments : une méthode de décompte modifié des $n$-grammes, qui n'attribue plus la totalité de la masse de probabilité aux $n$-grammes du corpus d'apprentissage, et une méthode de calcul des probabilités des $n$-grammes inconnus, qui permette d'aller au-delà d'une simple redistribution d'une même masse de probabilité entre les $n$-grammes, hypothèse qui n'est pas forcément réaliste.

**Décompte modifié :** Il s'agit juste de pondérer le décompte initial des probabilités $C$ par un facteur de décompte $D$ :

$$C^* = C.D, \text{ avec } D < 1 \tag{4.5}$$

Le calcul de $D$ peut se faire de différentes manières en fonction de la stratégie de lissage choisie. Les stratégies de lissage que nous avons utilisées et les facteurs de décompte associés sont présentés dans le paragraphe suivant.

**Calcul des probabilités des $n$-grammes inconnus :** plutôt que de répartir de manière égale la masse de probabilités restante entre tous les $n$-grammes inconnus, puisque rien n'indique que l'apparition des différents $n$-grammes inconnus soit équiprobable, leur probabilité d'apparition est calculée en fonction de la probabilité d'apparition dans le corpus d'apprentissage des $n$-grammes d'ordre inférieur : pour tout $n$-gramme inconnu $w_1 w_2 \ldots w_n$, la probabilité d'apparition de ce $n$-gramme est de la forme :

$$\Pr(w_n|w_1, w_2 \ldots w_{n-1}) = f(\Pr(w_n|w_2 w_3 \ldots w_{n-1}), \Pr(w_n|w_3 w_4 \ldots w_{n-1}), \ldots, \Pr(w_n))$$

Cette stratégie s'explique de la manière suivante : supposons qu'un $n$-gramme de probabilité $\Pr(w_n|w_1, w_2 \ldots w_{n-1})$ inconnue soit rencontré, sa probabilité effective d'apparition sera d'autant plus élevée que la probabilité d'apparition du $(n-1)$-gramme $w_2 w_3 \ldots w_n$ sera élevée. Inversement, si $w_2 w_3 \ldots w_n$ a une très faible probabilité d'apparition, celle de $w_1 w_2 \ldots w_n$ sera d'autant plus faible.

Le calcul de la probabilité d'apparition d'un $n$-gramme en fonction des $n$-grammes d'ordre inférieur peut se faire de deux manières distinctes : par interpolation ou par repli (*back-off*). La méthode par interpolation consiste à calculer la probabilité d'un $n$-gramme comme une combinaison linéaire des probabilités des $n$-grammes d'ordre inférieur, de $n-1$ à 1 (voir équation 4.6). La méthode par repli consiste quant à elle à ne calculer la probabilité associée à un $n$-gramme inconnu qu'en fonction de la probabilité du $n-1$-gramme correspondant (voir équation 4.7).

$$\begin{aligned} \Pr_{\mathcal{L}}(w_n|w_1 w_2 \ldots w_{n-1}) = \; & \alpha_1 \Pr_{\mathcal{L}}(w_n|w_2 \ldots w_{n-1}) \\ & + \alpha_2 \Pr_{\mathcal{L}}(w_n|w_3 \ldots w_{n-1}) \\ & + \ldots + \alpha_{n-1} \Pr_{\mathcal{L}}(w_n) \end{aligned} \tag{4.6}$$

$$\Pr_{\mathcal{L}}(w_n|w_1 w_2 \ldots w_{n-1}) = \alpha_{n-1} \Pr_{\mathcal{L}}(w_n|w_2 \ldots w_{n-1}) \tag{4.7}$$

Pour évaluer la probabilité d'apparition d'un $n$-gramme inconnu $w_1 w_2 \ldots w_n$, on utilise donc les $n$-grammes correspondant d'ordres inférieurs ; au pire, si aucun $n$-gramme tel que $n > 1$ n'est connu, on retombe sur la probabilité d'apparition de l'unigramme, ou mot seul, $w_n$. Cependant, il est possible que ce mot $w_n$ n'apparaisse lui-même pas dans le corpus d'apprentissage. On parle alors de mot hors-vocabulaire (OOV, *Out Of Vocabulary*). Le problème du traitement des mots hors-vocabulaire est encore un problème ouvert. Une approche basique pour le résoudre consiste à réserver un mot spécifique "INCONNU" sous lequel on réunit tous les mots n'appartenant pas au vocabulaire initial du modèle de langues [CR97]. Dans le cas spécifique des mots visuels néanmoins, ce problème est moins gênant : le vocabulaire de mots visuels étant fixé lors de l'étape de clustering des descripteurs locaux, il n'est pas possible de rencontrer, dans de nouveaux documents, de mots n'appartenant pas à ce vocabulaire initial. Il est néanmoins possible que tous les mots de ce vocabulaire n'apparaissent pas dans les documents servant à créer un modèle de langues, mais, compte tenu de la relative dispersion des descripteurs parmi les clusters représentant les mots visuels et parmi les documents, ce problème reste marginal.

#### 4.1.1.3 Quelques stratégies de lissage

Nous présentons dans cette section les méthodes de lissage que nous utiliserons par la suite. Il existe d'autres méthodes de lissage, dont Chen et Goodman donnent un aperçu très complet [CG98]. Dans les équations qui suivent, $\nu(x)$ désigne le nombre de $n$-grammes qui apparaissent $x$ fois dans le corpus.

**Lissage linéaire** : il repose sur l'usage du repli et le facteur de décompte suivant :

$$D = 1 - \frac{\nu(1)}{\sum\limits_{W' \in \mathcal{T}} C_{W'}} \tag{4.8}$$

**Lissage absolu** : il repose sur l'usage du repli et le facteur de décompte suivant :

$$D = \frac{C - b}{C} \tag{4.9}$$

La valeur $b = \frac{\nu(1)}{\nu(1)+2\nu(2)}$ est approximativement optimale.

**Lissage de Katz** : il combine un facteur de décompte de Good-Turing et du repli. Le facteur de décompte de Good-Turing se calcule de la manière suivante :

$$D = \begin{cases} \frac{\frac{(C+1)\nu(C+1)}{C\nu(C)} - \frac{(k+1)\nu(k+1)}{\nu(1)}}{1 - \frac{(k+1)\nu(k+1)}{\nu(1)}} & \text{si } C < k \\ 1 & \text{sinon} \end{cases} \tag{4.10}$$

Typiquement, $k = 7$.

**Lissage de Witten-Bell** : il combine l'interpolation linéaire et le facteur de décompte suivant :

$$D = \frac{\sum\limits_{W' \in \mathcal{T}} C_{W'}}{t + \sum\limits_{W' \in \mathcal{T}} C_{W'}}$$

$t$ désigne, étant donné un $n$-gramme $w_1 w_2 \ldots w_n$, le nombre de $n$-grammes distincts du corpus d'apprentissage $\mathcal{T}$ qui commencent par $w_1 w_2 \ldots w_{n-1}$.

### 4.1.2 Applications

Les modèles de langues connaissent des applications variées. Nous décrivons ici en premier lieu l'application de catégorisation de documents que nous allons utiliser, puis nous citons quelques autres applications.

#### 4.1.2.1 Modèles de langues et classification

Les modèles de langues sont utilisés en classification de textes, tâche pour laquelle ils donnent de très bon résultats [BN04, CT94]. L'utilisation des modèles de langues en classification est direct : disposant d'un ensemble $\mathcal{C}$ de $m$ classes, et d'un ensemble de documents (textes ou images) d'apprentissage $\mathcal{T}$ pour la tâche de classification, on constitue un sous-ensembles d'apprentissage $\mathcal{T}_i$ par classe $c_i$, contenant toutes les documents de cette classe :$\mathcal{T}_i = \{d | d \in \mathcal{T} \wedge d \in c_i\}$. Puis, pour chaque sous-ensemble $\mathcal{T}_i$, on construit un modèle de langues $\mathcal{L}_i$ spécifique à la classe $c_i$. Pour tout nouveau document $d$ dont la classe est inconnue, on calcule automatiquement celle-ci de la manière suivante :

$$c(d) = \underset{c_i \in \mathcal{C}}{\operatorname{argmax}}(\Pr_{\mathcal{L}_i}(d)) \tag{4.11}$$

#### 4.1.2.2 Autres applications

**Reconnaissance de la parole :** la reconnaissance de la parole est le domaine qui a vu naître les modèles de langues. Un système de reconnaissance de la parole est, en effet, principalement composé d'un modèle acoustique et d'un modèle de langues. Le modèle acoustique permet, à partir de caractéristiques acoustiques extraites du signal en entrée et d'un dictionnaire des mots possibles, d'établir des hypothèses de mots à reconnaître. Le modèle de langues intervient ensuite pour modéliser les séquences de mots potentielles issues des hypothèses, établir quelle séquence est la plus correcte en fonction des connaissances qu'il possède sur la langue considérée et proposer la séquence de mots qui lui semble être la plus juste pour retranscrire la parole initiale.

**Recherche d'information :** les modèles de langues peuvent également être utilisés en recherche d'information. Ponte et Croft ont ainsi proposé un modèle de recherche d'information basé sur l'interpolation de deux modèles de langues pour la description des documents [PC98] : le premier modèle décrit le contenu des documents, et le second modèle, calculé sur l'ensemble de la collection, permet d'effectuer un lissage du premier pour offrir de meilleures capacités de généralisation. Ce modèle a été amélioré et étendu par Hiemstra [Hie01] et Lafferty *et al* [LZ01], notamment.

## 4.2 Modèles de langues et mots visuels

### 4.2.1 Problématique

La description des images en mots visuels offre un cadre adapté à l'usage des modèles de langues, pour deux raisons :

1. les mots visuels permettent de représenter les images comme des ensembles de symboles ;
2. les relations de proximité géométrique entre ces symboles sont hautement significatives du sens porté par les images.

L'obstacle majeur se situe ici dans la différence de nature entre les proximités entre mots visuels et les proximités entre mots textuels. En effet, en langage naturel, les phrases sont des séquences de mots, constituant un espace à une dimension où la seule relation de proximité immédiate existant entre deux mots $x$ et $y$ est "$x$ se situe à côté de $y$". Les modèles de langues ont été conçus dans le cadre de telles proximités. Dans le cas des images, les mots visuels sont répartis dans un plan, ce qui multiplie les possibilités de proximité entre mots, et rend impossible l'application directe des modèles de langues. Deux approches peuvent être envisagées pour utiliser conjointement modèles de langues et mots visuels :

1. adapter les modèles de langues aux proximités bidimensionelles spécifiques des mots visuels ;
2. adapter la représentation des images au cas des modèles de langues en réduisant les images à des séquences de mots visuels.

La première possibilité pose potentiellement deux problèmes majeurs :

- la notion de proximité entre plusieurs points d'un plan n'est pas définie de manière claire. Si nous considèrons les points proches d'un point $P$ donné comme étant l'ensemble des points se situant dans un rayon fixe $r$, nous nous heurtons, d'une part, à la manière de déterminer ce rayon, en particulier pour avoir un système robuste aux changements d'échelle dans les images, et d'autre part au fait que l'on ne contrôle

pas le nombre potentiel de points se situant dans ce rayon. Si nous considérons les proximités en termes de $n$ plus proches voisins, le fait que la relation "plus proche voisin" n'est pas symétrique peut poser problème ;
- prendre en compte les très nombreuses possibilités de proximités entre mots visuels d'un plan augmente considérablement la complexité du problème, aussi bien en termes de temps de calcul que de calcul des probabilités de $n$-grammes, car si le nombre de configuration de $n$-grammes pertinents augmente trop, il existe un risque de "noyer" les probabilités des ces $n$-grammes pertinents dans la masse de tous les $n$-grammes possibles.

Nous nous sommes donc orientés vers la seconde possibilité, représenter les images comme des séquences de mots visuels, solution qui, commme nous allons le voir, offre l'avantage d'une certaine simplicité et d'une grande efficacité calculatoire.

### 4.2.2 Modéliser les images comme des séquences de mots visuels

Pour modéliser les images comme des séquences de mots visuels, il est impératif que la méthode utilisée produise des séquences de mots similaires pour deux images au contenu similaire, pour que les deux images soient décrites par les mêmes $n$-grammes. En particulier, la transformation adoptée doit être robuste :
- aux rotations ;
- aux changements d'échelle ;
- aux translations.

Nous proposons d'utiliser une projection orthogonale des centres des régions sur un axe, qui possède les propriétés désirées, comme le montre la figure 4.1. Il faut néanmoins remarquer que la robustesse aux rotations n'est assurée qu'à partir du moment où l'axe subit une rotation similaire à celle effectuée par les régions d'intérêt. Le choix de l'axe est discuté plus bas. Il peut aussi être remarqué que l'invariance à ces trois transformations n'est assurée qu'à partir du moment où le détecteur de régions d'intérêt utilisé y est effectivement invariant, ce qui est ici supposé au vu de leur objectif annoncé, même si ça n'est pas tout à fait le cas en pratique. La figure 4.2 illustre notre méthode pour construire des séquences de mots visuels. Par la suite, nous appellerons ces séquences des *phrases visuelles*, par analogie avec le langage naturel.

#### 4.2.2.1 Choix des axes

Comme nous venons de l'évoquer, le choix de l'axe est primordial pour obtenir des phrases visuelles qui soient cohérentes d'une image d'un objet à une autre image d'un même objet. Si la projection assure l'invariance aux translations et changements d'échelle, le choix de l'axe doit quant à lui assurer les deux propriétés suivantes :
- une orientation qui soit caractéristique des objets qui se situent sur l'image, de manière à obtenir des phrases visuelles cohérentes indépendamment des rotations ;
- un sens de lecture des mots visuels de la phrase qui s'adapte au sens dans lequel apparaît l'objet (de gauche à droite ou de droite à gauche).

Pour obtenir un axe qui soit adapté aux objets apparaissant sur l'image, nous nous appuyons sur les régions d'intérêt qui y sont détectées, ou plus précisemment sur les points d'intérêt qui constituent les centres de ces régions. En effet, les points d'intérêts issus des détecteurs classiques possèdent les propriétés de répétabilité et d'invariance nécessaires à obtenir des axes ayant les propriétés que nous souhaitons [MS04]. Nous cherchons donc des axes qui soient caractéristiques du nuage de points d'intérêt dans l'espace à deux dimensions de l'image. L'Analyse en Composantes Principales (ACP) [Jol02] nous permet d'obtenir

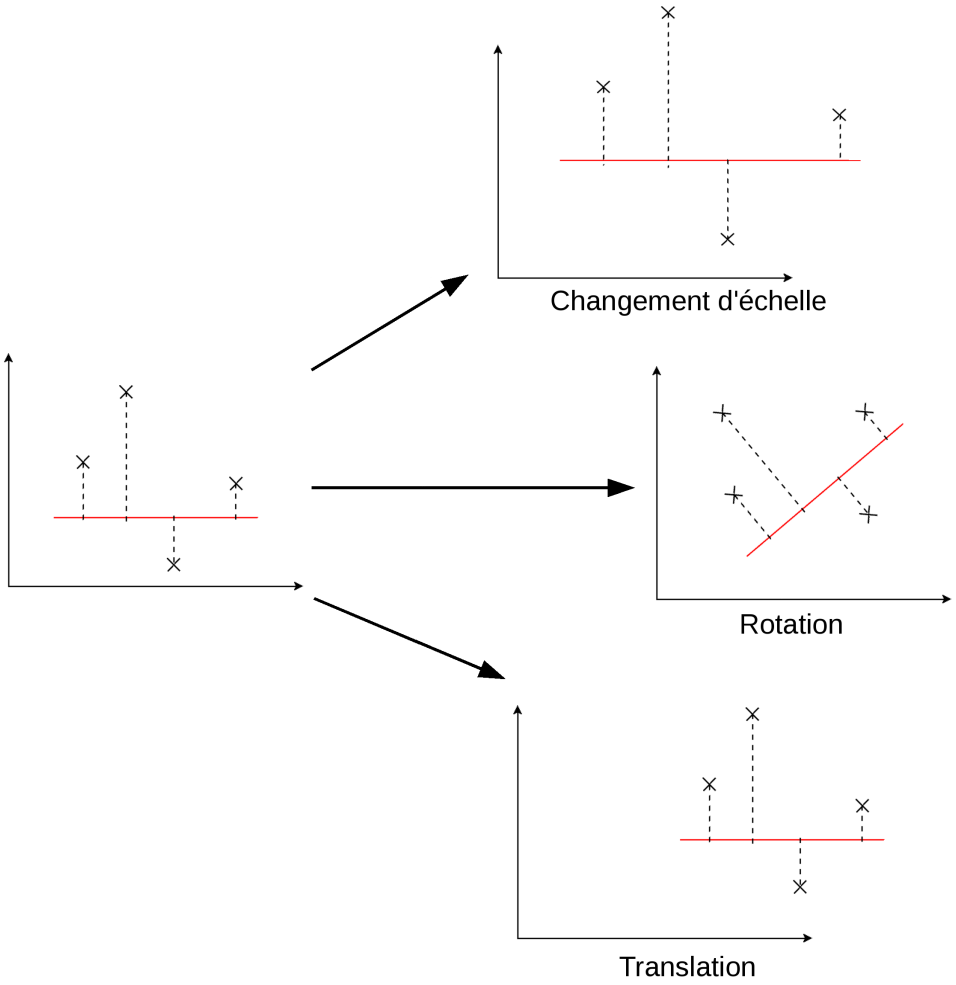



FIG. 4.1 – Invariance de la projection orthogonale aux changements d'échelle, aux translations et aux rotations.

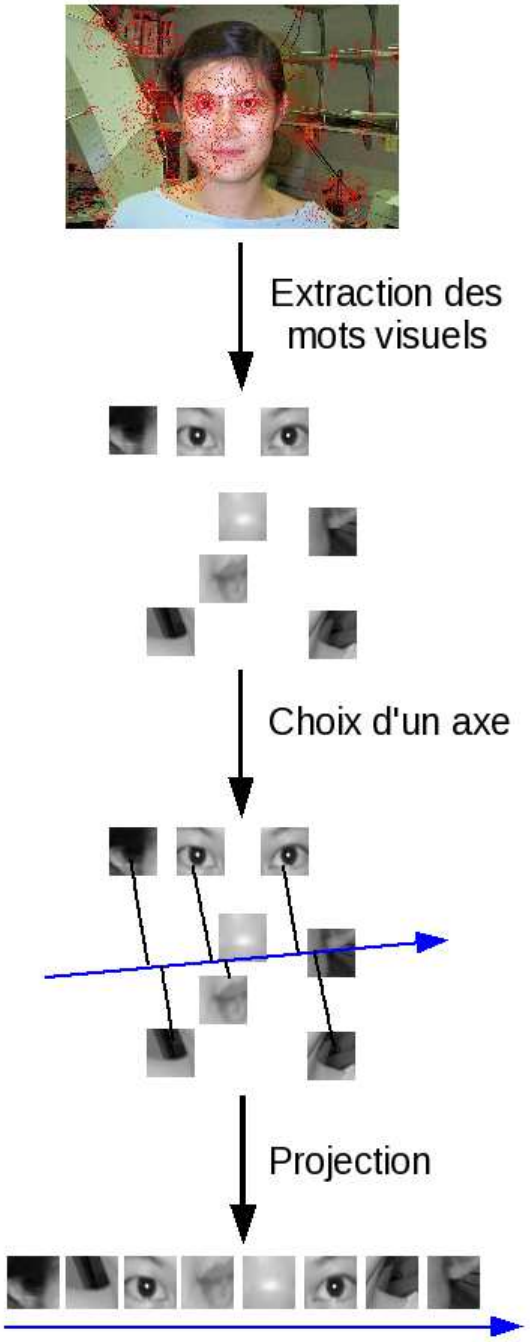



FIG. 4.2 – Méthode de construction des phrases visuelles.

de tels axes : les deux axes obtenus par ACP sont en effet ceux qui, par construction, expliquent le mieux la distribution des points d'intérêt dans l'image. Ainsi, quelles que soient les transformations appliquées au nuage de points (rotation, translation, changement d'échelle), puisque ces transformations respectent les distances entre les points, les axes obtenus auront la même position vis-à-vis du nuage de points. De plus, cette solution est très efficace en termes de temps de calcul : comme elle est effectuée en deux dimensions, l'ACP ne nécessite que la diagonalisation d'une matrice de corrélation $2 \times 2$. La figure 4.3 montre quelques exemples d'axes obtenus grâce à une ACP sur les positions des points d'intérêt.

FIG. 4.3 – Exemples d'axes obtenus par ACP sur les centres des régions d'intérêt. Le premier axe est en rouge, le second en bleu.

Une fois les deux axes de base obtenus, se pose la question du nombre d'axes à considérer. En effet, l'usage d'un axe unique risque de supprimer une trop grande part des informations géométriques entre mots visuels, et de générer des $n$-grammes contenant des mots proches selon la direction de l'axe choisi, mais très éloignés dans la direction orthogonale à cet axe. Il est donc possible d'envisager l'utilisation de plusieurs axes, générant autant de phrases visuelles qui contiennent des informations de proximité géométrique selon les deux dimensions de l'image. Nous avons donc considéré plusieurs cas de figure :

- l'usage de l'axe principal de l'ACP uniquement ;
- l'usage des deux axes de l'ACP, qui sont nécessairement orthogonaux et donnent ainsi des informations complémentaires sur les deux dimensions de l'image ;
- l'usage de dix axes, obtenus par rotations successives de 10 ° à partir de l'axe principal de l'ACP, couvrant ainsi un angle de 90 ° . Considérer un angle plus important (par exemple, 19 axes sur 180 ° ) serait contre-productif, car cela reviendrait à considérer des informations de proximité entre mots visuels similaires entre différents axes, mais dans des sens opposés.

#### 4.2.2.2 Élimination des mots visuels redondants

Comme nous l'avions déjà évoqué au chapitre précédent (voir section 3.1.2 page 64), les détecteurs de régions d'intérêt ont tendance à détecter plusieurs régions aux mêmes coordonées, à des échelles et orientations plus ou moins proches. Si cette redondance des régions peut constituer un avantage dans les approches en sacs de mots visuels, en mettant l'accent sur telle ou telle partie importante de l'image, elles représentent en revanche un obstacle lorsque l'on cherche à dépasser l'hypothèse d'indépendance des mots visuels : si l'on cherche à construire des regroupements de mots visuels proches, les répétitions de mots visuels de sémantique similaire à des positions identiques ou très proches vont créer beaucoup de regroupements ne contenant que des mots représentant une même partie d'un objet. Ces

regroupements n'apporteront pas d'informations intéressantes en termes de relations spatiales entre les mots visuels et risqueront de masquer, par leur nombre, les regroupements vraiment utiles à la description de l'image. Pour éliminer les régions redondantes, nous définissons une relation de redondance $\mathcal{R}$ entre régions. Définissons d'abord trois grandeurs caractéristiques des régions d'intérêt :

- la position $p_r$ : coordonnées du centre de la région $r$ dans l'image, telles qu'elles sont fournies par le détecteur de régions d'intérêt ;
- l'angle $a_r$ : orientation de la région d'intérêt, également donnée par le détecteur ;
- la forme $s_r$ : cette forme est donnée par le ratio entre les longueurs du grand axe et du petit axe de la région d'intérêt, de forme elliptique.

Une fois ces trois grandeurs définies, nous pouvons définir la relation de redondance $\mathcal{R}$ entre régions de la manière suivante :

$$\mathcal{R}(r_1, r_2) \Leftrightarrow \begin{cases} d_{L_2}(p_{r_1}, p_{r_2}) \leqslant \theta_p \\ |a_{r_1} - a_{r_2}| \leqslant \theta_a \\ |s_{r_1} - s_{r_2}| \leqslant \theta_s \end{cases} \tag{4.12}$$

où $d_{L_2}$ désigne la distance euclidienne, et $\theta_p$, $\theta_a$ et $\theta_s$ des seuils qui permettent de paramétrer $\mathcal{R}$ : ils définissent une relation de redondance exacte s'ils sont nuls, approchée sinon. Nous pouvons ensuite définir simplement l'algorithme d'élimination des régions redondantes (algorithme 1). Si l'on note $m_{init}$ le nombre de régions initialement détectées dans l'image, la relation $\mathcal{R}$ étant commutative[1], le nombre de fois où l'on teste $\mathcal{R}$ est au plus de $\frac{m_{init}.(m_{init}-1)}{2}$, dans le cas où il n'existe pas de régions redondantes. La complexité au pire cas de cet algorithme est donc en $O(m_{init}^2)$, mais ceci ne pose pas problème dans la majorité des cas, le nombre de régions détectées par image n'étant pas spécialement élevé (typiquement, $m_{init} < 1000$). La figure 4.4 présente une image avant et après élimination des régions redondantes, avec les seuils de $\mathcal{R}$ définis ainsi : $\theta_p = 5,\ \theta_a = 0.2,\ \theta_s = 0.5$.

$R_I$ : **Régions d'intérêt de l'image I**
**Pour** $r_i \in R_I$ **faire**
  **Pour** $r_j \in R_I$ **faire**
    **Si** $(r_i \neq r_j \wedge r_i \mathcal{R} r_j)$ **Alors**
      $R_I \leftarrow R_I \backslash r_j$
    **Fin Si**
  **Fin Pour**
**Fin Pour**
**Retourner** $R_I$

Algorithme 1: Algorithme d'élimination des régions redondantes.

[1] La distance euclidienne étant elle-même commutative, et sachant que $\forall a, b \in \mathbb{R}, |a - b| = |-(b - a)| = |b - a|$, par définition de la valeur absolue $|\cdot|$.

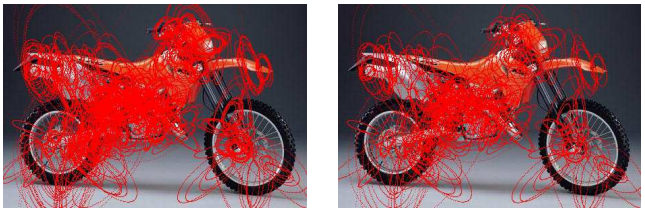


FIG. 4.4 – Une image avant et après filtrage des régions redondantes. La première contient 970 régions, la seconde 318.

## 4.3 Expérimentations

### 4.3.1 Protocole expérimental

#### 4.3.1.1 Données

Nous utilisons, pour ces expériences, les bases d'images adaptées à la catégorisation que nous avions déjà utilisées dans le chapitre précédent, Caltech-6 et Caltech-101. Nous renvoyons le lecteur à la section 3.5.1.2 page 80 pour obtenir des détails sur ces collections d'images.

#### 4.3.1.2 Vocabulaire visuel

Nous utilisons les mêmes paramètres de vocabulaire visuel que dans les expériences du chapitre précédent (voir section 3.5.2.1, page 81).

#### 4.3.1.3 Performance measure

Dans ces expérimentations, nous mesurons les performances des différents systèmes par le taux d'images de test correctement classées. Le score d'un classifieur $c$ est donc calculé ainsi :

$$S_c = \frac{|\text{images correctement classées par } c|}{|\text{images de test}|} \tag{4.13}$$

#### 4.3.1.4 *Baseline*

Nous comparons les performances de notre système à un système classique de l'état de l'art, les machines à vecteurs supports (SVM, *Support Vector Machines*). Nous utilisons l'implémentation *SVM Multiclass* de Joachims dans ces expérimentations [THJA04]. Nous utilisons des vecteurs avec une pondération tf. idf et un noyau linéaire, conformément aux résultats rapportés par Csurka *et al.* [CDF+04], que nous avons confirmés sur quelques expériences.

Nous avons considéré deux cas pour les SVM. Le premier (nommé SVM dans les résultats) repose simplement sur les vecteurs pondérés, le second sur des vecteurs pondérés et normalisés (SVM-N).

#### 4.3.1.5 Implémentation des modèles de langues

Nous utilisons l'implémentation des modèles de langues développée par les universités de Carnegie-Mellon (CMU) et Cambridge [CR97]. Ce logiciel est mis à disposition librement par ses auteurs à des fins de recherche.

### 4.3.2 Résultats

Nous avons d'abord testé les différents paramètres de notre système à base de modèles de langues (choix des axes, choix du lissage, élimination ou non des mots visuels redondants), puis nous avons comparé le meilleur système obtenu à un système à base de SVM.

#### 4.3.2.1 Choix de l'axe

Nous avons évalué les différentes méthodes de sélection d'axes que nous avons proposées dans la section 4.2.2.1 :
- axe principal obtenu par ACP ;
- deux axes orthogonaux obtenus par ACP ;
- dix axes obtenus par rotations de 10 ° à partir de l'axe principal obtenu par ACP.

Nous ajoutons, pour se comparer à des méthodes de choix d'axes *a priori*, deux méthodes supplémentaires :
- utilisation de l'axe des abscisses ;
- utilisation d'un axe choisi aléatoirement, identique pour toutes les images.

| | données d'apprentissage | données de test |
|---|---|---|
| 1 axe ACP | 66.75 | 66.90 |
| 2 axes ACP | 100 | 41.23 |
| 10 axes ACP | 100 | 38.20 |
| axe des abscisses | 83.67 | 68.68 |
| axe aléatoire | 66.75 | 65.67 |

TAB. 4.1 – Performances en classification en fonction des axes considérés (lissage de Katz, $n = 3$).

Pour cette expérimentation, nous utilisons le corpus Caltech-6, divisé en un ensemble d'apprentissage de 1200 images (200 par catégorie) et un ensemble de test composé des 4215 images restantes. Nous testons les différents axes avec un lissage de Katz, un modèle trigramme ($n = 3$), et nous éliminons les mots visuels redondants et les mots vides (voir section suivante). Le tableau 4.1 présente les résultats obtenus pour les différentes méthodes de choix d'axe, sur chacun des ensembles de données. Nous pouvons faire les remarques suivantes :
- l'axe des abscisses donne les meilleurs résultats. Ceci s'explique par la nature des données : dans Caltech-6, les objets sont, pour une très grande majorité, tous alignés avec l'axe des abscisses, ce dernier représente donc un axe optimal sur ces données pour notre système. On peut cependant considérer que, dans un cas général, ce choix constitue un *a priori acceptable*, les photos étant la plupart du temps cadrées correctement. Ce genre d'hypothèse est adoptée dans certains travaux, comme ceux de Jégou *et al.* par exemple [JDS08] ;
- l'axe obtenu par ACP donne des résultats inférieurs à ceux obtenus par l'axe des abscisses. Ceci s'explique par le fait que l'ACP est, dans certaines images, biaisée par la présence de mots visuels dans le fond de l'image, qui ne sont pas significatifs pour extraire un axe pertinent représentant l'objet principal de l'image ;
- l'axe choisi aléatoirement donne des résultats moins bons que l'axe obtenu par ACP, ce qui tend à montrer que l'utilisation de l'ACP, bien que non optimale, permet d'obtenir des résultats majoritairement cohérents. Il faut de plus prendre en compte

que l'axe aléatoire, étant identique pour toutes les images, profite, comme l'axe des abscisses, du fait que les objets sont, dans cette collection, globalement placés dans la même position dans les images ;
- l'utilisation de plusieurs axes aboutit sur un effet de sur-apprentissage (performances parfaites sur les données d'apprentissage, mais mauvaises sur les données de test). Cela s'explique par le fait que l'usage de plusieurs axes nécessite la prise en compte de beaucoup plus de $n$-grammes différents, représentant plus de voisinages possibles par mot visuel, ce qui noie les probabilités des $n$-grammes les plus pertinents parmi l'ensemble des probabilités, dont certaines issues du lissage ou de $n$-grammes bruités.

#### 4.3.2.2 Élimination des mots visuels redondants

Nous testons ici l'efficacité de notre algorithme d'élimination des régions redondantes. Les paramètres employés sont les mêmes que précédemment (1200 images d'apprentissage, 4215 de test, lissage de Katz), mais nous prenons $n \in [1; 4]$. Nous nous comparons également à notre *baseline* pour étudier la pertinence de cet algorithme dans un autre cadre que les modèles de langues. De plus, nous testons dans le contexte de classification d'images les *stop-lists* basées sur pLSA que nous avons présenté en section 3.2.2.

Le tableau 4.2 montre les performances des différents systèmes : pas d'élimination de mots visuels (PE), élimination des régions redondantes (RR), utilisation d'une stop-list (pLSA) et élimination combinée des régions redondantes et des mots vides (RR+pLSA). On remarque que l'élimination des mots visuels redondants est bien nécessaire pour prendre en compte de manière efficace les relations géométriques en mots visuels. En revanche, lorsque l'on ne prend pas ces relations en compte, éliminer les mots visuels redondants n'améliore pas les résultats (cas des SVM et de $n = 1$). Cela s'explique par le fait que ces mots visuels redondants, du fait de la prise de vue des images de cette collection (objets principaux au centre, en gros plan), sont plus présents sur l'objet à décrire que dans le fond des images, et apportent donc une information supplémentaire puisqu'ils ont tendance à donner plus de poids à des mots visuels globalement pertinents. On remarque également que l'utilisation d'une stop-list obtenue selon notre méthode permet ici d'obtenir une légère amélioration des résultats, en plus de limiter le nombre de mots visuels (et donc de réduire les temps de calcul).

| | PE | RR | pLSA | RR+pLSA |
|---|---|---|---|---|
| ML $n = 1$ | 73.59 | 67.35 | 73.90 | 69.06 |
| ML $n = 2$ | 50.23 | 68.09 | 52.19 | 77.72 |
| ML $n = 3$ | 50.44 | 67.85 | 52.34 | 68.68 |
| ML $n = 4$ | 50.15 | 65.29 | 52.00 | 73.17 |
| SVM-N | 56.20 | 55.75 | 56.56 | 55.90 |
| SVM | 54.23 | 25.52 | 56.73 | 42.06 |

TAB. 4.2 – Performances en classification avec ou sans sélection des mots visuels (lissage de Katz).

#### 4.3.2.3 Choix de la longueur des $n$-grammes

Nous testons différentes valeurs de $n$, comprises entre 1 et 10. Nous utilisons un lissage de Katz, et éliminons les mots visuels redondants et les mots vides (*stop-list*). Les résultats obtenus sur l'ensemble d'apprentissage et sur l'ensemble de test sont rapportés dans le

tableau 4.3. Le modèle unigramme ($n = 1$) ne donne pas les meilleurs résultats, ce qui prouve que la prise en compte des relations géométriques permet d'obtenir une amélioration des résultats. Cependant, au-delà de $n = 4$, les résultats se dégradent considérablement. En effet, des $n$-grammes trop longs sont trop spécifiques aux images d'apprentissage et ne permettent pas de généraliser correctement à partir de ces images : ils gèrent mal les petites variations comme l'ajout, la suppression ou la substitution d'un mot visuel. Ces variations sont néanmoins courantes et peuvent être provoquées par différents facteurs : le détecteur de régions d'intérêt, qui ne détecte pas nécessairement les mêmes régions sur des objets pourtant similaires, la phase d'assignation des descripteurs aux mots visuels, qui peut remplacer un mot visuel par un autre, l'imprécision du choix de l'axe qui peut provoquer des variations dans l'ordre des mots visuels, ou encore, simplement, la variabilité des images qui n'ont pas toutes le même fond, et des objets qui varient beaucoup au sein d'une catégorie. Toutes ces raisons justifient la faiblesse des $n$-grammes au-delà de certaines valeurs de $n$ : on ne peut pas s'attendre à ce que deux images d'une même catégorie fournissent des phrases visuelles parfaitement identiques. Prendre en compte uniquement des $n$-grammes courts, donc des relations de proximité locales entre mots visuels, permet d'obtenir une plus grande robustesse.

| $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| données d'apprentissage | 95.58 | 99.92 | 83.67 | 31.42 | 83.33 |
| données de test | 69.06 | 77.72 | 68.68 | 73.17 | 30.49 |
| $n$ | 6 | 7 | 8 | 9 | 10 |
| données d'apprentissage | 66.67 | 83.33 | 100 | 100 | 16.67 |
| données de test | 23.77 | 14.73 | 23.82 | 26.24 | 1.00 |

TAB. 4.3 – Performances en classification en fonction de la longueur $n$ des $n$-grammes (lissage de Katz).

#### 4.3.2.4 Choix du lissage

Nous testons les différents lissages présentés en section 4.1.1.2, avec 1200 images d'apprentissage et 4215 de test, pour $n \in [1; 4]$ et avec élimination des mots visuels redondants et des mots vides (*stop-list*). Les tableaux 4.4 et 4.5 montre les résultats obtenus pour les données d'apprentissage et de test, respectivement. Le lissage linéaire donne les meilleurs résultats, et les lissages absolu et de Witten-Bell donnent des résultats assez proches.

| $n$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| absolu | 95.67 | 99.92 | 99.83 | 32.67 |
| linéaire | 95.42 | 100 | 100 | 50.58 |
| Katz | 95.58 | 99.92 | 83.67 | 31.42 |
| Witten-Bell | 95.42 | 100 | 100 | 100 |

TAB. 4.4 – Performances en classification des différents lissages (données d'apprentissage).

#### 4.3.2.5 Performances globales du système

Nous testons les performances globales de notre système sur les corpus Caltech-6 et Caltech-101 et les comparons aux résultats obtenus avec les SVM. Nous utilisons la meilleure configuration que nous avons observée pour les modèles de langues, à savoir l'axe

| $n$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| absolu | 70.20 | 77.88 | 77.91 | 77.88 |
| linéaire | 72.12 | 80.50 | 80.45 | 80.33 |
| Katz | 69.06 | 77.72 | 68.68 | 73.17 |
| Witten-Bell | 72.12 | 76.63 | 77.84 | 77.79 |

TAB. 4.5 – Performances en classification des différents lissages (données de test).

des abscisses comme axe de projection, un lissage linéaire et l'élimination des modèles de langue. Pour les expériences sur le corpus Caltech-6, nous avons de plus éliminé les mots vides à l'aide de notre méthode basée sur pLSA, pour la classification par modèles de langues comme pour la classification par SVM.

Les figures 4.5 et 4.6 montrent les résultats obtenus en faisant varier le nombre d'images d'apprentissage par catégorie. Les modèles de langues (ML) donnent de meilleures performances globales que les SVM sur les deux corpus utilisés. La figure 4.7 détaille les performances obtenues sur chaque catégorie du corpus Caltech-6. Nous observons une amélioration par rapport aux SVM dans toutes les catégories sauf une pour les SVM (motos) ou deux pour les SVM-N (guitares et voitures). Les performances des SVM-N s'expliquent surtout par l'excellente performance obtenue sur la catégorie des voitures (100% d'images correctement classées), qui tire les résultats vers le haut. Ceci est sans doute dû à une spécificité des images de cette classe, et explique que ce système donne de bien plus mauvais résultats sur 101 catégories. Globalement, les différences les plus remarquables entre SVM et modèles de langues concernent la catégorie des fonds, sur laquelle les SVM échouent dans la quasi-totalité des cas. Ceci s'explique par le fait que ces images contiennent moins de mots visuels, et que ces mots visuels sont répartis de manière aléatoire par rapport à ceux des objets. Ces images contiennent donc moins de $n$-grammes pertinents, et les modèles de langues permettent de bien exclure ces images des catégories contenant des objets qui sont, eux, plutôt décrits par des $n$-grammes.

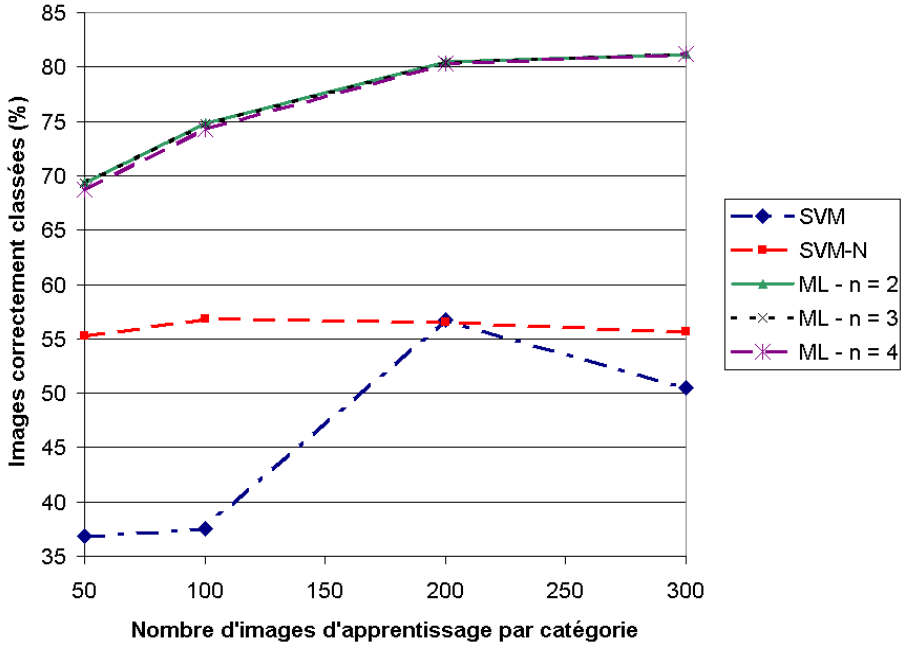



FIG. 4.5 – Performances en classification sur Caltech-6.

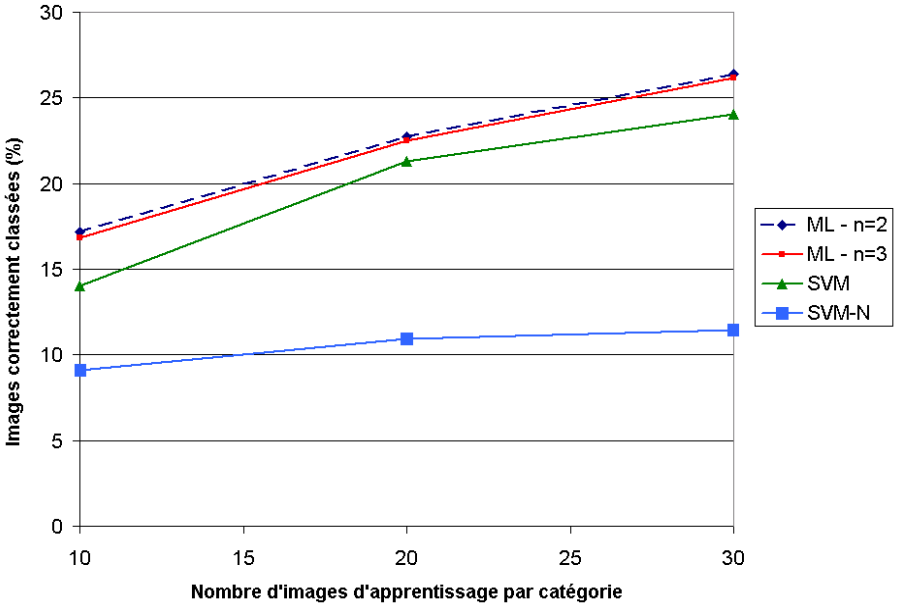



FIG. 4.6 – Performances en classification sur Caltech-101.

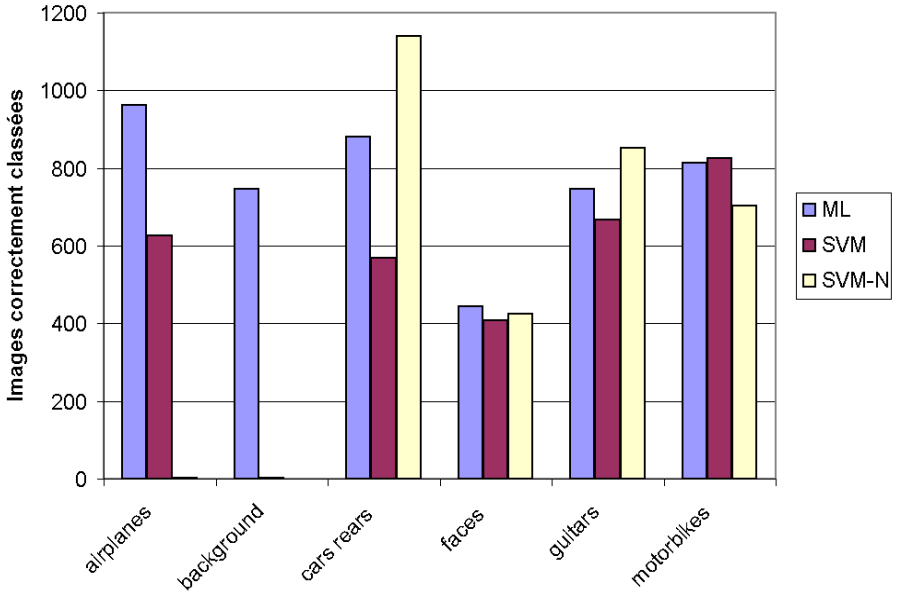



FIG. 4.7 – Performances en classification par catégorie sur Caltech-6 (200 images d'apprentissage par catégorie).

Le tableau 4.6 donne les temps d'exécutions moyennés sur 10 executions de chaque système, sur une machine possédant un processeur Intel Xeon 3GHz et 8 Go de RAM et tournant sous Linux. On voit que les temps d'exécution sont nettement inférieurs pour l'approche utilisant des modèles de langues. Ce résultat s'explique moins par des différences en termes de temps de calcul pur qu'en termes de temps d'entrées/sorties. En effet, alors que les modèles de langues ne prennent en entrée qu'une représentation compacte des documents (phrases visuelles contenant quelques centaines de mots), les SVM reposent elles sur l'utilisation de vecteurs de grande dimension, donc une représentation plus lourde. Ces vecteurs étant creux, il est possible de réduire les temps d'exécution des SVM pour

s'approcher de ceux des modèles de langues en utilisant une description des vecteurs ne faisant pas apparaître les composantes nulles. Il est néanmoins intéressant de remarquer que les modèles de langues permettent d'intégrer des informations de proximité géométrique entre mots visuels sans surcoût calculatoire.

| | Apprentissage | Classification |
|---|---|---|
| ML - $n = 1$ | 0.542 | 5.629 |
| ML - $n = 3$ | 0.973 | 8.019 |
| SVM | 7.293 | 17.94 |
| SVM-N | 5.728 | 18.18 |

TAB. 4.6 – Temps d'exécution moyen en secondes pour Caltech-6 (1200 images d'apprentissage, 4215 de test).

## 4.4 Travaux connexes

### 4.4.1 Modèles de langues et indexation d'images

Quelques auteurs ont utilisé les modèles de langues dans le cas de la classification d'images [ZRZ02, WLL+07] ou de l'annotation d'images [GWL06]. De même que dans nos travaux, les auteurs utilisent une description des images sous forme de symboles, obtenue globalement de la même manière que les mots visuels (*via* des *clusters* de régions d'image). En revanche, leurs travaux diffèrent des nôtres par les descripteurs utilisés (descripteurs de couleurs et texture) et, surtout, par la manière de découper les régions d'images : ils utilisent un découpage en grille régulière. Ce découpage permet d'obtenir des relations de proximité évidentes entre régions (au-dessus, au-dessous, à droite, à gauche), et ainsi de définir aisément un modèle de langues à deux dimensions, mais a le désavantage de prendre en compte toutes les régions de l'image, y compris les moins significatives, là où l'utilisation d'un détecteur permet en principe de ne s'intéresser qu'aux régions de l'image les plus importantes. De plus, parmi ces travaux, à notre connaissance, seuls Wu *et al.* utilisent une forme de lissage des probabilités.

On peut également citer les travaux de Mac Donald *et al.* [MDS05] qui utilisent des modèles de langues dans le cas de l'indexation vidéo. Néanmoins, ils ne cherchent pas à exploiter des relations de proximités entre régions d'image grâce à ces modèles de langues mais uniquement pour fusionner (par une méthode d'interpolation des modèles de langues) différents types d'informations (descripteurs visuels, texte, séquences de plans...).

### 4.4.2 Relations géométriques entre mots visuels

Il existe également quelques travaux sur la prise en compte des relations spatiales entre régions d'intérêt. Carneiro *et al.* [CJ04] ont proposé des relations entre points d'intérêt basées sur l'échelle, la position et l'orientation de ceux-ci, et ont exploité ces relations dans un contexte de reconnaissance d'objets par mise en correspondance de descripteurs locaux. Jamieson *et al.* [JFD+07] ont quant à eux exploité ces mêmes relations pour décrire des logos d'équipes de hockey sur glace à l'aide de graphes de points d'intérêt. Zheng *et al.* [ZWG06] se placent quant à eux, comme nous, dans un contexte de représentation par mots visuels, et constituent des paires de mots visuels (appelées syntagmes visuels, *visual phrases*) à partir de régions d'intérêt se chevauchant. Yuan *et al.* [YWY07] proposent aussi de créer des syntagmes visuels, contenant cette fois une quantité arbitraire de mots visuels.

Les points sont regroupés à l'aide d'un algorithme de $k$-plus proches voisins, puis les motifs les plus généraux obtenus sont retenus comme étant des syntagmes visuels.

Ces approches utilisent des structures de données complexes (comme les graphes) ou nécessitent des mises en correspondance exhaustives des régions d'intérêt, qui peuvent les rendre peu efficaces en termes de temps de calcul, bien que certains auteurs proposent aussi des méthodes d'accélération des calculs [ZWG06]. En comparaison, notre méthode, qui groupe les régions d'intérêt par simple projection sur un axe, est très rapide à mettre en œuvre et relativement robuste grâce au lissage des modèles de langues.

## 4.5 Conclusion

Dans ce chapitre, nous avons proposé une représentation des images sous forme de phrases visuelles, qui constituent une extension des mots visuels et qui permettent de prendre en compte des relations spatiales élémentaires entre les mots visuels. Nous avons associé cette représentation avec des modèles de langues dans un contexte de classification d'images. Nos expériences montrent que :

- la prise en compte de relations spatiales entre les mots visuels permet d'obtenir une amélioration des performances ;
- les modèles de langues permettent d'intégrer ces relations spatiales de manière très efficace dans un classifieur, pour obtenir de meilleures performances que les classifieurs classiques sans surcoût calculatoire.

Ces résultats montrent qu'il est possible d'utiliser des méthodes du TAL pour améliorer les performances des systèmes de recherche d'images par le contenu, ce qui valide notre objectif initial. Néanmoins, notre méthode nécessite, nous l'avons vu, d'éliminer une partie des régions d'intérêt initiales, et donc de supprimer une partie de l'information visuelle, information dont on pourrait sans doute tirer partie plutôt que de l'ignorer.

De plus, il y a un certain nombre d'autres possibilités d'améliorations et de perspectives pour notre système à base de modèles de langues. Premièrement, les modèles de langues peuvent être utilisé en recherche d'information [PC98], et pas seulement en classification comme nous l'avons fait. Utiliser le modèle des phrases visuelles dans un contexte de recherche d'images constitue donc un prolongement naturel des travaux présentés ici. De plus, plusieurs améliorations aux modèles de langues existent, notamment celle proposée par Gao *et al.* [GNWC04] qui permet de prendre en compte des relations entre termes sans que ceux-ci ne soient des voisins directs dans la phrase. Les modèles de ce genre pourraient améliorer notre système en permettant de retrouver certaines relations de proximité perdues lors de la phase de projection des mots visuels sur un axe. Enfin, notre modèle de phrase visuelle ne permet de traiter les images contenant plusieurs objets, pour lesquelles l'utilisation de plusieurs phrases visuelles serait souhaitable. Il faudrait donc pouvoir isoler les objets, ou des parties d'objets, de manière fiable. Ce problème nous ramène à un problème de segmentation, qui constitue encore un problème de vision par ordinateur ouvert à l'heure actuelle.

# Chapitre 5

# Exploitation conjointe des textes et images

Nous abordons dans ce chapitre le second des deux axes directeurs de ces travaux : l'utilisation d'outils du TAL pour l'indexation sémantique d'images. Comme nous l'avons expliqué dans la section 2.3.2 page 60, nous souhaitons ici exploiter les ressources textuelles accompagnant les images pour leur appliquer des techniques du TAL permettant d'en extraire une description de haut-niveau, sémantique, des images. Pour cela, nous nous heurtons à deux problèmes. D'une part, il est nécessaire de disposer de données adaptées à notre cadre d'étude, ce qui n'est pas le cas des corpus bimodaux texte-image existant, comme nous l'avons remarqué à la section 2.3.2.1 page 60. D'autre part, le fossé sémantique (voir section 1.2.3 page 22) limite fortement l'efficacité des systèmes d'annotation reposant sur des descripteurs de bas-niveau classiques (couleurs, textures, formes). Remarquons néanmoins que, bien qu'étant la cause supposée des limites de tels systèmes, l'ampleur de ce fossé sémantique n'a jamais été formellement mise en évidence.

Dans ce chapitre, nous présentons d'abord un corpus présentant des propriétés appropriées à l'objectif que nous nous sommes fixés. Puis nous montrons par une première série d'expériences l'existence du fossé sémantique sur ces données. Enfin, nous proposons une méthode d'annotation exploitant les données textuelles disponibles pour annoter les images. Cette méthode repose sur un outil classique du TAL, la détection d'entités nommées, ainsi que sur des indices visuels de haut-niveau qui nous permettent de contourner le fossé sémantique mis en évidence précédemment.

## 5.1 Données utilisées

### 5.1.1 Description

Nous utilisons des articles de presse parus entre avril et novembre 2006, téléchargés sur le site de TV5[1]. Notre corpus contient 27,041 articles. Chaque article contient :

- un texte principal, qui constitue le corps de l'article ;
- une ou plusieurs images qui illustrent l'article ;
- pour chaque image, une légende en deux parties :
  1. une courte description de l'image en elle-même : elle indique l'essentiel du contenu de l'image ;
  2. une description qui replace l'image dans le contexte de l'article.

[1] `www.TV5.fr`.

La figure 5.1 montre un exemple d'article de presse de notre corpus. Le tableau 5.1 donne quelques données statistiques sur le corpus.

| | Nombre | Longueur moyenne | Écart-type |
|---|---|---|---|
| Textes | 27043 | 480.6 | 175.1 |
| Description des images | 42884 | 15.85 | 6.28 |

TAB. 5.1 – Données statistiques du corpus de presse.

### 5.1.2 Intérêt de ces données

Ces données présentent plusieurs aspects intéressants pour développer et évaluer des méthodes d'indexation sémantique d'images :

- c'est un corpus de grande taille : il contient une quantité de documents significativement supérieure à la plupart des corpus proposant à la fois des images et du texte exprimé en langage naturel, et non sous forme de mots-clefs associés aux images ;
- il fournit deux sources de texte différentes : les textes des articles et les légendes des images. Ces deux sources peuvent être exploitées de manière conjointe pour indexer les images, ou de manière complémentaire à des fins d'évaluation des méthodes d'indexation proposées ;
- il propose des textes complets en langue naturelle, contrairement à de très nombreux corpus existant qui ne contiennent qu'une courte description de l'image (voir chapitre 2) ;
- il est très varié : les articles abordent de nombreux thèmes (politique, culture, sport, économie...) et les images au sein de chaque catégorie ont des aspects visuels différents. Il n'existe pas de biais simplifiant la tâche d'indexation ou d'annotation (un fond similaire pour les images d'objets similaires par exemple) comme il en existe dans beaucoup de corpus artificiels ;
- il correspond à une application réelle : l'indexation d'images de presse.

## 5.2 Caractériser le fossé sémantique

Le fossé sémantique, évoqué au chapitre 1, constitue la principale entrave à l'efficacité des systèmes de recherche d'images. S'il est toujours supposé que les descripteurs visuels ne permettent pas de décrire le contenu sémantique des images, l'écart existant entre description visuelle et contenu sémantique n'a jamais été mis en évidence explicitement. C'est ce que nous proposons de faire dans cette partie.

Ne disposant pas d'une vérité-terrain décrivant le contenu sémantique des images, nous nous basons sur la description fournie par les légendes de celles-ci. Il est en effet couramment admis que le texte (sous forme de mots-clefs ou de textes complets) permet de représenter la sémantique des images. De plus, ici, les légendes des images, bien qu'elles ne constituent pas une description de l'intégralité du contenu des images (par exemple, on ne trouve pas de détails tels que *une voiture rouge au fond à gauche*), correspondent à la description des images telles qu'elle est réalisée par le journaliste qui écrit l'article. Elles représentent donc bien le message essentiel porté par l'image.

Description de l'image

Contexte de l'image

**Pierre Méhaignerie, le 2 septembre 2005 à La Baule, lors de l'ouverture de l'Université d'été de l'UMP**

La grande majorité des élus UMP, déterminés à soutenir un gouvernement avec lequel ils se sentent "sur le même bateau", étaient relativement soulagés jeudi en fin de journée après une manifestation anti-CPE qu'ils jugeaient "moins importante" que prévue. "C'est moins que le 7 février", juge, visiblement soulagé, Pierre Méhaignerie, député d'Ille-et-Vilaine et secrétaire général de l'UMP.

© AFP/Archives - Frank Perry

PARIS (AFP) - 16/03/2006 19h31 - La grande majorité des élus UMP, déterminés à soutenir un gouvernement avec lequel ils se sentent "sur le même bateau", étaient relativement soulagés jeudi en fin de journée après une manifestation anti-CPE qu'ils jugeaient "moins importante" que prévue.Selon la police, 247.500 personnes, principalement des étudiants et des lycéens, ont manifesté jeudi en France contre le contrat première embauche. "C'est moins que le 7 février", juge, visiblement soulagé, Pierre Méhaignerie, député d'Ille-et-Vilaine et secrétaire général de l'UMP. "Je pensais qu'il y aurait davantage de manifestants", renchérit Laurent Hénart, son collègue de Meurthe-et-Moselle et ancien secrétaire d'Etat à la Formation professionnelle des jeunes. Même réaction de la part de Roger Karoutchi, sénateur des Hauts-de-Seine. "Il y a du monde mais ce n'est pas ce rassemblement massif de jeunes que l'on nous avait annoncé. Dans beaucoup d'universités, les étudiants commencent à s'organiser contre le blocage des cours et à dire que ça ne peut pas durer!", affirme cet élu sarkozyste. Entre 218.000 et 400.000 personnes avaient défilé le 7 février, et entre 396.000 selon la police et 1 million selon la CGT avaient manifesté le 7 mars, également contre le CPE. Toutefois, ces journées regroupaient à la fois les jeunes et les salariés. Le véritable test pour le gouvernement aura sans doute lieu samedi, avec les manifestations regroupant jeunes, syndicats et salariés.Ceci d'autant plus que 68% des Français se déclarent désormais favorables au retrait du CPE, soit 13 points de plus que le 8 mars, selon un sondage CSA à paraître vendredi dans Le Parisien-Aujourd'hui en France.Certes, la plupart des élus sont déterminés à défendre le CPE davantage parce qu'ils se sentent "obligés de tenir" que par adhésion à la façon dont le contrat première embauche a été mis en place. "Mais là n'est plus la question aujourd'hui", affirme M. Méhaignerie, qui regrette, comme d'autres avec lui, que le Premier ministre ait agi sans concertation."On ne va pas s'associer à ceux qui n'offrent aucune alternative. Renoncer au CPE, ce serait donner le feu vert à l'immobilisme pour de nombreuses années", affirme-t-il, en plaidant pour "l'ouverture du dialogue" avec les syndicats."Les syndicats vont attendre la décision du Conseil constitutionnel et que la loi s'applique avant de bouger" et de commencer à dialoguer avec le gouvernement, pense M. Hénart, alors que les Sages ont été saisis par la gauche sur "l'inconstitutionnalité", selon elle, du CPE.Marc-Philippe Daubresse, député du Nord, avance quelques solutions: "raccourcir la période d'essai et proposer au bout de six mois un contrat de reclassement personnalisé" en cas de licenciement."Le problème, c'est qu'il n'y a pas eu de négociations avec les partenaires sociaux. On a fait une sorte de procès de défiance aux jeunes, alors qu'il faut établir avec eux un contrat de confiance", dit-il. "Dominique de Villepin a voulu jouer tout seul pour jouer la carte de la +rupture+. Maintenant, c'est compliqué de colmater les fuites. Le bateau est mal en point", remarque un proche de Nicolas Sarkozy, le président de l'UMP.Toutefois, "personne à droite ne gagnerait à ce que le gouvernement soit trop contesté", tempère M. Karoutchi.

Texte de l'article

FIG. 5.1 – Un exemple d'article tiré de notre corpus.

### 5.2.1 Principe

Pour vérifier l'ampleur du fossé sémantique, nous vérifions la corrélation qui existe entre un classement des images réalisé sur des critères purement textuels et un classement des images réalisé selon des critères visuels. S'il existe une corrélation entre les résultats obtenus selon chacune des modalités, cela signifie que les descriptions textuelles et visuelles portent une information similaire, et qu'il est possible d'établir des correspondances directes entre descripteurs de bas-niveau et informations textuelles. Inversement, si les résultats des deux recherches ne sont pas corrélés, cela signifie que chaque modalité apporte une information différente, et met en avant la mesure du fossé sémantique existant entre la description textuelle des images et les descripteurs employés.

### 5.2.2 Protocole expérimental

#### 5.2.2.1 Descripteurs visuels utilisés

Nous nous sommes basés sur des descripteurs globaux de couleur, texture, et forme parmi les plus utilisés en recherche d'images.

**Couleur :** nous utilisons les histogrammes de couleurs pondérés proposés par Boughorbel *et al.* [BBV02]. Nous utilisons une pondération par le Laplacien, qui permet de prendre en compte les contours des objets, et calculons des histogrammes cumulés sur des images découpées en grille de $3 \times 3$. Nos histogrammes comptent 6 bins par canal dans l'espace RGB, ce qui nous donne un descripteur à 216 dimensions. Les histogrammes sont normalisés par la taille de l'image (en pixels) et comparés à l'aide d'une distance $L_1$ (préférable à la distance $L_2$, d'après Tarel *et al.*).

**Texture :** nous nous basons sur des filtres de Gabor pour obtenir une signature des textures des images. Nous utilisons une signature basée sur la moyenne et l'écart-type de l'énergie de la réponse de chaque filtre. Nous calculons des filtres selon 4 échelles et 6 orientations distinctes, ce qui nous donne une signature à 24 dimensions (12 moyennes et 12 écarts-types). Les descripteurs sont comparés à l'aide d'une distance $L_2$.

**Couleurs et texture combinées ;** nous avons aussi employé un descripteur combinant les informations de couleur et de texture pour obtenir une description plus complète des images. Comme les informations sont de nature distincte et que les distances pour les comparer sont différentes, nous ne fusionnons pas directement les descripteurs mais effectuons une combinaison linéaire des distances obtenues selon chaque modalité :

$$d_{globale}(d_i, d_j) = d_{couleur}(d_i, d_j) + d_{texture}(d_i, d_j) \tag{5.1}$$

**Forme :** nous utilisons la description des images en mots visuels que nous avons déjà présentée dans les chapitres précédents. Nous avons constitué notre vocabulaire de la même manière que dans les chapitres précédents :

- détecteur *Hessian-Affine* ;
- descripteur SIFT ;
- algorithme de clustering *k-means* hiérarchique de Nister et Stewenius.

La taille du vocabulaire utilisé est de 19679 mots visuels. Les descripteurs sont comparés à l'aide d'une distance $L_1$.

#### 5.2.2.2 Recherche textuelle

Nous effectuons la recherche textuelle à l'aide du logiciel LEMUR. Nous nous basons sur un modèle vectoriel avec une pondération BM25 des termes, et utilisons une stop-list pour éliminer les mots vides. Ce système de recherche textuelle constitue une *baseline* raisonnable pour effectuer l'indexation et la recherche de documents textuels.

#### 5.2.2.3 Calculs de corrélation

Il existe plusieurs méthodes pour calculer les corrélations entre des listes ordonnées. Les deux principales sont le coefficient de corrélation de Spearman et le $\tau$ de Kendall.

**Coefficient de corrélation de Spearman :** le coefficient de corrélation de Spearman est calculé en fonction de la différence entre les rangs des documents dans chacun des deux classements. Si l'on note le rang d'un document $d$ dans le premier classement $r_d^1$, et $r_d^2$ son rang dans le second classement, le coefficient de Spearman $\rho$ pour un ensemble $\mathcal{D}$ de $n$ documents est :

$$\rho = 1 - \frac{6 \sum_{d \in \mathcal{D}} (r_d^1 - r_d^2)^2}{n(n^2 - 1)} \tag{5.2}$$

Quand $\rho = 0$, il n'existe aucune corrélation entre les classements ; quand $\rho = 1$, les classements sont identiques.

**$\tau$ de Kendall :** le $\tau$ de Kendall mesure quant à lui la corrélation en vérifiant, pour chaque paire de documents $(d_i, d_j) \in \mathcal{D}^2$, si $d_i$ et $d_j$ sont classés dans le même ordre dans chacun des classements. En notant $n = \frac{|\mathcal{D}|(|\mathcal{D}|-1)}{2}$ le nombre total de paires de documents possibles, $n^+$ le nombre de paires classées dans le même ordre dans les deux classements et $n^-$ le nombre de paires classées différemment, le $\tau$ de Kendall est :

$$\tau = \frac{2(n^+ - n^-)}{n(n-1)} \tag{5.3}$$

Si $\tau = 1$, alors les classements sont identiques ; si $\tau = -1$, ils sont inversés ; si $\tau = 0$ les classements sont indépendants.

Néanmoins nous ne pouvons pas utiliser ces mesures de corrélation dans leur formulation classique car elles nécessitent de prendre en compte la totalité des listes. Or, en recherche d'information, la majorité des documents d'un corpus n'ont aucune pertinence vis-à-vis d'un document requête donné. Si l'on peut espérer obtenir, selon deux modalités différentes, des classements similaires parmi les documents plus ou moins pertinents, on ne peut en revanche absolument rien dire sur la manière dont seront classés les documents ne partageant aucun point commun avec la requête. Il convient donc de restreindre le calcul de corrélations aux $k$ premiers documents issus de chaque recherche. Pour cela, nous utilisons le $\tau$ de Kendall modifié proposé par Fagin *et al.* pour établir des corrélations entre des têtes de classements [FKS03].

**$\tau$ de Kendall modifié** : le $\tau$ de Kendall modifié fonctionne sur le même principe que le $\tau$ de Kendall, en observant si l'ordre des paires d'éléments est le même dans chacune des listes. Il ajoute à ce principe certaines hypothèses qui permettent de prendre en compte le fait que certains documents se trouvent parmi les $k$ premiers d'un classement mais pas parmi les $k$ premiers du second. Pour toute paire de documents $(d_i, d_j)$ présente dans un des classements :

- si les deux documents $d_i$ et $d_j$ sont présents dans l'autre classement, on peut déterminer s'ils sont classés dans le même ordre ou non. On leur attribue alors un score $S_{ij}^p = 1$ s'il sont dans un ordre différent (pénalisation), ou $S_{ij}^p = 0$ dans le cas contraire ;
- si seul $d_i$ (resp. $d_j$) est présent dans le second classement, alors $d_j$ (resp. $d_i$) ne se situe pas dans les $k$ premiers éléments, donc le rang de $d_j$ est nécessairement supérieur (resp. inférieur) à celui de $d_i$ : l'ordre de classement de la paire $d_i, d_j$ est donc connu. On peut attribuer à cette paire un score $S_{ij}^p$ de la même manière que dans le cas précédent ;
- si aucun des deux documents de la paire $(d_i, d_j)$ ne fait partie des $k$ premiers documents du second classement, alors on ne peut rien dire sur l'ordre de classement de cette paire. On lui attribue alors un score correspondant à une pénalité $p$ par défaut : $S_{ij}^p = p$.

En notant $\mathcal{P}$ l'ensemble des paires de documents distinctes parmi les $k$ premiers éléments des deux classements, nous obtenons le $\tau$ de Kendall avec pénalité suivant :

$$\tau^p = \frac{1}{|\mathcal{P}|} \sum_{(d_i,d_j)\in\mathcal{P}} S_{ij}^p \tag{5.4}$$

La pénalité par défaut $p \in [0;1]$ est choisie arbitrairement. Une pénalité $p = 0$ correspond au cas "optimiste" où l'on ne souhaite pas pénaliser les cas inconnus, et une pénalité $p = 1$ au cas "pessimiste", où l'on considère par défaut que des paires dont l'ordre dans un des classements est inconnu sont toujours ordonnées différemment. Nous avons ajouté à la formulation de Fagin *et al.* la normalisation par le nombre de paires $|\mathcal{P}|$ qui permet d'obtenir un score situé entre 0 et 1, 0 signifiant une corrélation parfaite et 1 une absence de corrélation.

### 5.2.3 Résultats et discussion

Le tableau 5.2 présente les indices de corrélation ($\tau$ de Kendall avec pénalité $p = 1$) obtenus entre les classements des images issus des recherches selon chaque modalité, textuelle et visuelle. Les indices de corrélation sont tous proches de un, à l'exception de ceux obtenus sur les 10 premiers documents, pour certains descripteurs (couleurs et mots visuels). Néanmoins un examen des requêtes ayant un indice de corrélation fort montre que celui-ci est dû au fait que ces images sont répétées dans le corpus[2]. Ces résultats montrent donc qu'il n'existe globalement pas de corrélation entre un classement obtenu sur des critères visuels et un classement obtenu selon la description textuelle des documents. La figure 5.2 illustre cela en montrant les premiers résultats d'une recherche réalisée grâce à des histogrammes de couleurs : il apparaît clairement que les images partagent des couleurs similaires, mais aussi qu'elles ont toutes un sens très différent des autres.

**Conséquences sur l'annotation automatique d'images :** ce résultat met en avant une limite à laquelle se heurtent inévitablement les systèmes d'annotation basés sur de l'apprentissage statistique lorsqu'il s'agit de de travailler sur des données réelles. Ceux-ci n'obtiennent d'ailleurs que des performances limitées, bien que leur évaluations ne se basent la plupart du temps que sur des jeux de données dont les biais sont connus[3] [MMMP02].

[2]Certaines images sont répétées mais pas les articles, car ce sont des images génériques qui servent d'illustration à des textes différents (la balance de la justice illustre des articles sur des procès, une image de salle de marché des articles sur la bourse...).

[3]En particulier, ils possèdent souvent beaucoup d'images quasi-similaires dans les données d'apprentissage et les données de test.

| Requête | 1 | 2 |
|---|---|---|
| | | |
| Désinfection en raison de la grippe aviaire en mars 2006 dans un village du Niger, pays voisin du Burkina Faso. | Des habitants de Jowhar, ville au nord de Mogadiscio, le 15 juin 2006. | Une rue inondée dans le sud de la Roumanie le 17 avril 2006. |
| 3 | 4 | 5 |
| | | |
| Des policiers irakiens bloquent l'accès à la ville de Moqdadiyah, le 21 mars 2006 à la suite d'un raid rebelle sanglant. | Des enfants palestiniens réunis autour d'un cratère causé par une frappe israélienne, le 1er avril 2006 à Gaza. | Photo non datée de Kinaam, compagnon de Sungaï, varan de Komodo. |

FIG. 5.2 – Les premiers résultats d'une recherche d'images basée sur la couleur.

| $k$ | 10 | 50 | 100 | 500 |
|---|---|---|---|---|
| Couleur | 0.63 (0.21) | 0.88 (0.12) | 0.93 (0.09) | 0.96 (0.06) |
| Texture | 0.99 (0.04) | 0.99 (0.01) | 0.99 (0.01) | 0.97 (0.01) |
| Couleur + texture | 0.61 (0.23) | 0.88 (0.09) | 0.93 (0.06) | 0.96 (0.06) |
| Mots visuels | 0.56 (0.26) | 0.83 (0.19) | 0.88 (0.16) | 0.91 (0.19) |

TAB. 5.2 – $\tau$ de Kendall modifié $\tau^{(1)}$ moyen (et écarts-types) sur les listes tronquées à $k$ éléments, pour chaque descripteur testé.

Ceci appelle à utiliser des méthodes d'annotation différentes des méthodes basées sur un usage basique de techniques d'apprentissage et de descritpeurs de bas-niveau. Nous nous intéresserons à de telles méthodes dans la partie suivante.

**Conséquence sur la fusion texte-image :** l'indépendance entre critères visuels et critères textuels peut constituer une chance pour les systèmes de recherche basés sur une fusion des descripteurs visuels et des descripteurs textuels car elle montre que les deux modalités apportent des informations complémentaires. Ce fait est confirmé par certains travaux, par exemple ceux de Tollari *et al.* [TG07] qui montrent que la fusion des modalités visuelles et textuelles permet effectivement d'améliorer les résultats des recherche d'images.

## 5.3 Utilisation du texte pour annoter les images

Nous souhaitons, dans cette partie, exploiter les textes des articles pour en tirer des mots-clefs qui décrivent leur contenu effectif. En effet, la quasi-totalité des travaux existants (à l'unique exception de [JT09], à notre connaissance) se basent exclusivement, ou essentiellement (dans le cas de [FL08]), sur les légendes des images. Or, l'utilisation des légendes simplifie grandement le problème d'annotation, puisque celles-ci sont déjà des descriptions des images. De plus, si les images sont très souvent utilisées pour illustrer des textes complets, et sont donc accompagnées de tels textes, il n'est pas systématique de leur voir associer des légendes, ce qui diminue la portée de systèmes n'exploitant que ces dernières.

### 5.3.1 Association d'indices visuels et textuels de haut-niveau

Puisqu'il n'existe pas de correspondance directe entre les mots décrivant une image et les descripteurs de bas-niveau que l'on peut extraire de cette image, il est nécessaire d'adopter une stratégie d'annotation qui permette de passer outre cette limite. Pour cela, nous proposons de n'exploiter que des indices textuels de haut-niveau correspondant explicitement à des concepts visuels que l'on peut efficacement détecter dans les images (par exemple, associer des noms de personnes à une image où apparaissent effectivement des personnes). Le système d'annotation ainsi obtenu incorpore donc, contrairement aux systèmes classiques reposant sur des méthodes d'apprentissage de type "boîte noire", des connaissances *a priori* sur les correspondances possibles entre concepts textuels et concepts visuels. Nous ne considérons pas nécéssairement des connaissances très précises sur le contenu des images (par exemple, *une tomate est rouge*), mais des faits génériques que l'humain prend implicitement en compte (comme le fait que toute personne possède un nom).

### 5.3.2 Les entités nommées comme indices textuels

Les entités nommées sont les termes ou syntagmes désignant des entités ou groupes d'entités précis (par exemple, *Alan Turing*, *les Bretons* ou *la tour de Pise*). Elles incluent notamment les noms propres, mais ne s'y limitent pas (exemple : *la nuit du 4 août*). Elles apportent des informations très précises sur le contenu d'un discours (écrit ou parlé), elles constituent donc un élément clef pour les systèmes de recherche d'information (textuelle, audio ou multimédia), ainsi que pour d'autres applications proches (résumé automatique, systèmes de questions-réponses). Dans notre cas, elles permettent de nommer avec précision ce qui apparaît sur les images que l'on souhaite annoter. De plus, elles possèdent des caractéristiques typographiques, morphologiques et syntaxiques communes qui permettent de les détecter de manière efficace. Elles constituent donc des indices textuels de choix pour notre système d'annotation.

#### 5.3.2.1 Systèmes de détection et catégorisation des entités nommées

Les premiers travaux sur la détection et l'annotation des entités nommées sont apparus au début des années 1990. Les systèmes de détection des entités nommées reposent généralement sur des méthodes d'apprentissage supervisé (SVM, arbres de décision...) ou non-supervisé (*clustering*) qui prennent en entrée des attributs spécifiques permettant de déterminer si un terme (ou un syntagme) donné est ou non une entité nommée. Ces attributs peuvent être de différente nature :

- typographique : présence de majuscules (*Darwin*, *eBay*) et de caractères de ponctuation (*C.N.R.S.*) ;
- morphologique : présence de suffixes caractéristiques (*roumains*, *américains*, *mexicains*) ;
- syntaxique : présence d'indices spécifiques avant l'entité nommée (*le peintre Pablo Picasso*), au début (*Université de Rennes 1*), en fin (*The Coca Cola Company*) ; patrons spécifiques (par exemple : *Prénom Nom*) ; position dans la phrase (sujet, complément) ;
- externe : présence d'un e-mail associé par exemple.

Ces systèmes utilisent souvent des connaissances extérieures pour détecter des termes candidats (listes de prénoms, de termes spécifiques : *institut*, *organisation*...) ou pour affiner la détection (comparaison avec un dictionnaire de noms communs par exemple).

Certains systèmes permettent de plus la catégorisation des entités nommées. Les catégories prises en compte dépendent fortement des systèmes eux-mêmes. Ainsi, les noms de lieux (toponymes) sont une catégorie généralement employée, mais qui peut être subdivisée en de nombreuses sous-catégories (villes, pays, régions...). Par exemple, Sekine et Nobata recensent environ 200 catégories possibles d'entités nommées [SN04].

On pourra se reporter à l'état de l'art de Nadeau et Sekine pour plus d'informations sur les divers systèmes de reconnaissance et catégorisation des entités nommées existants [NS07].

#### 5.3.2.2 Système utilisé

Nous utilisons le système Némésis développé par Fourour [Fou02]. Il a l'avantage d'effectuer à la fois la détection des entités nommées et leur catégorisation selon une classe générale et une catégorie spécifique à cette classe. Le tableau 5.3 présente les classes et catégories d'entités nommées gérées par Némésis.

| Classe | Catégorie | Exemples |
|---|---|---|
| Anthroponymes | Patronymes | Gainsbourg, Vian |
| | Prénoms | Serge, Boris |
| | Ethnonymes | Breton, Français |
| | Ensembles artistiques | La Pléiade, La Comédie Française |
| | Organisations | Parti Communiste Français, UNICEF |
| Toponymes | Pays | France, Belgique |
| | Villes | Rennes, Quimper |
| | Grands toponymes | Europe, Asie |
| | Toponymes moyens | Bretagne, Bourgogne |
| | Microtoponymes | Beaulieu, Villejean |
| | Hydronymes | La Loire, La Manche |
| | Oronymes | Les Alpes, Les Monts d'Arrée |
| | Rues | Avenue des Champs-Élysées |
| | Déserts | Sahara |
| | Édifices | Tour Eiffel, Phare d'Eckmühl |
| Ergonymes | Marques ou produits | |
| | Entreprises | Google Inc. |
| | Établissements | Université de Rennes 1 |
| | Œuvres | La symphonie pastorale |
| Praxonymes | Faits historiques | Prise de la Bastille |
| | Périodes Historiques | La Révolution Française |
| | Événements | Festival d'Avignon |
| Phénonymes | Catastrophes | L'ouragan Katrina |
| | Astres ou comètes | Saturne, Jupiter |

TAB. 5.3 – Classes et catégories d'entités nommées prises en charge par NEMESIS.

### 5.3.3 Indices visuels associés aux entités nommées

Il n'est prudent d'annoter une image par une entité nommée particulière qu'après avoir détecté dans l'image un ou plusieurs indices visuels qu'une entité de la catégorie correspondante soit effectivement présente dans cette image. Chaque indice visuel considéré doit être à la fois suffisamment générique pour pouvoir correspondre à une ou plusieurs catégories complètes d'entités nommées, et avoir un aspect visuel suffisamment spécifique pour être détectable de manière fiable avec les outils de vision par ordinateur existant. Nous nous sommes intéressés à deux indices visuels particuliers :

- **les visages** : un visage est spécifique à la présence d'une personne dans l'image. C'est donc un indice visuel que l'on peut facilement associer à une entité nommée de type patronyme. De plus, les caractéristiques visuelles régulières des visages (yeux, nez, bouche...) ainsi que leur grand intérêt pour de nombreuses applications de vision par ordinateur (indexation ou vidéosurveillance, par exemple) font que l'on dispose aujourd'hui de nombreux travaux traitant de la détection de visages ;
- **les logos (et panneaux)** : les logos ayant pour objectif de représenter une entité particulière, il est possible de les associer à des entités nommées précises appartenant aux catégories suivantes : ensembles artistiques, organisations, événements, ainsi que toutes les catégories d'ergonymes (marques et produits, entreprises, établissements, œuvres). Ils présentent de plus une certaine unité visuelle : devant être clairement

identifiables par le public, ils contiennent peu de couleurs distinctes et peu de détails. Cet aspect visuel tranche avec le contenu souvent riche des images naturelles et est un atout pour les détecter automatiquement.

### 5.3.4 Annotation des images par les entités nommées

L'annotation effective des images par les entités nommées s'effectue en 2 étapes :

1. sélection d'entités nommées candidates en fonction de leur importance dans le texte ;
2. annotation par les entités nommées candidates les plus significatives en fonction du nombre de concepts visuels détectés.

#### 5.3.4.1 Sélection des entités nommées candidates à l'annotation

Nous attribuons aux entités nommées détectées dans le texte un score reflétant leur importance comme terme d'annotation, puis nous retenons comme candidates à l'annotation les entités nommées dont le score dépasse un seuil fixé manuellement.

Nos scores se basent sur les quatres grandeurs statistiques suivantes, inspirées des pondérations classsiques utilisées en recherche d'information textuelle :

- la fréquence f : nombre d'occurrences de l'entité nommées dans le texte ;
- la fréquence documentaire df[4] : nombre de documents dans lesquels apparaissent l'entité nommée ;
- la fréquence d'annotation af[5] : nombre de documents annotés par l'entité nommée au cours d'une première itération du processus d'annotation. Les valeurs de af dépendent de la valeur du seuil utilisé pour réaliser la première série d'annotation, et se base sur des annotations initiales qui peuvent être justes ou erronées. On notera par la suite ce score af-$x$, où $x$ indique le seuil initialement utilisé pour annoter les images, sur la base de la fréquence seule ;
- la fréquence d'annotation apprise laf[6] : nombre d'images annotées par cette entité nommée dans la vérité-terrain. Comme dans tout processus d'apprentissage, cette valeur doit être apprise sur un ensemble d'images d'apprentissage distinct de celui d'images de test. En revanche, contrairement à la fréquence d'annotation décrite précédemment, elle ne se base que sur des annotations initiales correctes. Elle fournit donc un indicateur des meilleures performances qu'il serait possible d'obtenir avec un critère d'annotation basé sur la fréquence d'annotation des images.

La fréquence f nous informe sur l'importance d'un terme au sein d'un document : c'est une pondération locale. df, af et laf sont quant à elles basées sur la collection de document : ce sont donc des pondérations globales. De manière évidente, plus f est élevée, plus l'entité nommée est significative dans le texte. En revanche, le problème reste ouvert pour les pondérations globales : les images ont-elles tendance à illustrer des entités rares ou des entités fréquentes de la collection ? Nous proposons donc deux types de scores qui combinent la fréquence f et une pondération globale $\mathrm{g} \in \{\mathrm{df}, \mathrm{af}, \mathrm{laf}\}$ calculée sur une collection de $N$ documents :

- un score direct $\mathrm{f} - \mathrm{g}$ proportionnel à la fréquence f et à g :

$$\mathrm{f} - \mathrm{g} = \mathrm{f}.(1 + \frac{\mathrm{g}}{N}) \tag{5.5}$$

[4]Document frequency.
[5]Annotation frequency.
[6]Learned annotation frequency.

– un score inverse f − ig proportionnel à f et inversement proportionnel à g :

$$\mathrm{f-ig} = \mathrm{f}.\left(\log(\frac{N}{\mathrm{g}})\right) \tag{5.6}$$

#### 5.3.4.2 Nombre d'entités nommées candidates retenues

Le nombre d'entités retenues pour l'annotation finale de l'image dépend évidemment du nombre de concepts visuels détectés dans l'image. Cependant, ce nombre dépend également de la nature d'un concept visuel que l'on a retenu :

1. pour les visages, on a la garantie, sauf cas particulier (montage ou miroir), et hors détection de faux-positifs, que chaque visage détecté sur une image correspond à une personne distincte, donc à une entité nommée distincte. Une image contenant $n$ visages sera donc annotée par les $n$ entités nommées candidates ayant le score le plus élevé ;
2. pour les logos, cette garantie n'existe pas : un logo peut être répété un nombre quelconque de fois dans une image. Nous choisissons donc de ne retenir que l'entité nommée candidate ayant le meilleur score en cas de détection de plus d'un logo.

#### 5.3.4.3 Cas d'ambiguité

Lorsque plusieurs entités nommées candidates ont un score identique et qu'il faut choisir certaines d'entre elles et pas d'autres pour réaliser l'annotation, nous ne disposons pas d'indices suffisants pour faire un choix. Dans ce cas, nous n'utilisons aucune des entités nommées en position d'ambiguité pour l'annoter, quitte à annoter l'image par moins d'entités nommées que le nombre de concepts visuels détectés ne l'aurait voulu.

### 5.3.5 Expérimentations

#### 5.3.5.1 Détection des concepts visuels

**Visages** : nous détectons les visages à l'aide du détecteur de visages proposé dans la librairie OpenCV d'Intel, développé par Lienhart *et al.* [LKP]. D'après ses auteurs, ce détecteur atteint une précision de 80% pour un rappel de 90%. Néanmoins, leur corpus de test ne contient que des images de faces, ce qui facilite une détection efficace. Dans notre corpus, les visages contenus sur les images se présentent sous tous les angles (face, profil, trois-quart et positions intermédiaires) et sous des échelles et luminosités variées, ce qui réduit l'efficacité de ce détecteur de visages. En effectuant une évaluation rapide de ce détecteur sur 200 images tirées aléatoirement de notre corpus, nous obtenons une précision comparable (78,4%), mais un rappel plus faible : 45,7% si nous considérons tous les visages, et 57,9% si nous ne comptons pas certains visages très difficiles à détecter compte tenu de leur taille réduite[7]. De même, certaines erreurs n'ont pas d'impact sur les résultats de notre algorithme : un visage manqué peut être compensé par un faux-positif sur la même image ; sur les 200 images que nous avons testé, ce cas s'est produit sur 4 images.

[7] Cette approximation est acceptable dans notre application car ces visages sont des visages apparaissant dans des foules ou dans le fond de l'image. Les entités nommées correspondant au nom de ces personnes n'apparaissent jamais dans les textes.

**Logos et panneaux** : nous avons développé un détecteur rapide de logos et panneaux basé sur la représentation des images par mots visuels. Ce détecteur permet de détecter les logos dans les images naturelles (extraites du corpus d'images de presse utilisé ici) avec une précision de 95% pour une rappel de 60%. Les détails de la méthode de détection que nous avons employée n'entrant pas à proprement parler dans le cadre de ce chapitre, nous renvoyons le lecteur à l'annexe B qui présente en détail l'algorithme utilisé ainsi que l'évaluation des ses performances.

#### 5.3.5.2 Vérité-terrain et mesure des performances

Nous utilisons dans ces expériences les légendes des images comme vérité terrain. La précision moyenne des annotations des images d'un corpus $\mathcal{C}$ est donc calculée de la manière suivante :

$$P_{\mathcal{C}} = \frac{1}{|\mathcal{C}|} \sum_{I \in \mathcal{C}} \frac{\text{Nombre d'annotations de } I \text{ apparaissant dans la légende}}{\text{Nombre total d'annotations de } I} \tag{5.7}$$

En faisant varier le seuil de sélection des entités nommées candidates, nous pouvons obtenir plusieurs points (nombre d'images annotées, précision des annotations) qui nous permettent de tracer des courbes dans l'esprit des classiques courbes de rappel-précision.

#### 5.3.5.3 Résultats et discussion

Les figures 5.3, 5.4 et 5.5 donnent les performances des différents scores en annotation de visages, et les figures 5.6, 5.7 et 5.8 les performances en annotation de logos. La figure 5.9 donne des exemples d'annotations obtenues avec notre système.

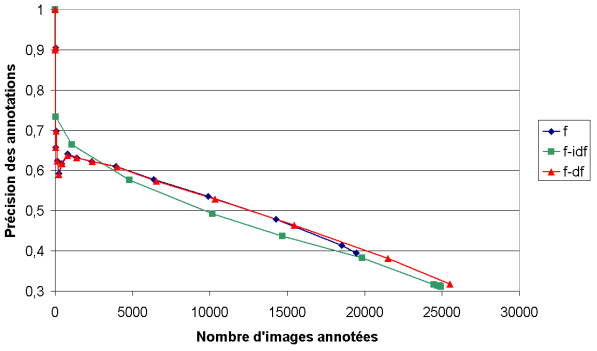



FIG. 5.3 – Performances des critères d'annotation de type df pour les visages.

D'une manière générale, les performances sont du même ordre pour les deux types d'associations entre entités nommées et concepts visuels que l'on prend en compte. Les effets des différents scores sur les performances sont également similaires. Plus spécifiquement, nous pouvons faire les remarques suivantes :

- les résultats des scores basés sur la fréquence d'annotation dépendent du seuil utilisé pour l'annotation initiale. Plus précisemment, plus le seuil initial est élevé, plus les résultats sont proches des performances de l'annotation initiale. En effet, l'utilisation d'un seuil élevé ayant pour conséquence de limiter le nombre d'images annotées, on dispose alors de moins d'informations pour calculer les fréquences d'annotation, qui donnent alors des scores plus proches des scores initiaux ;

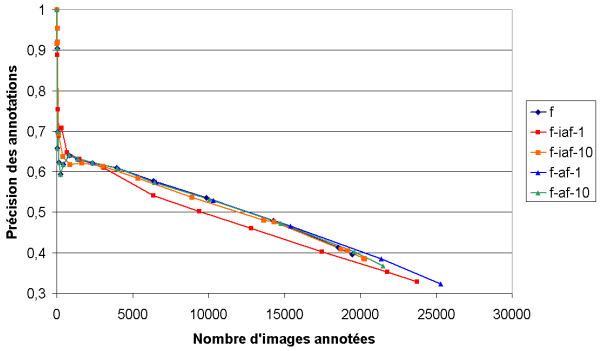



FIG. 5.4 – Performances des critères d'annotation de type af pour les visages.

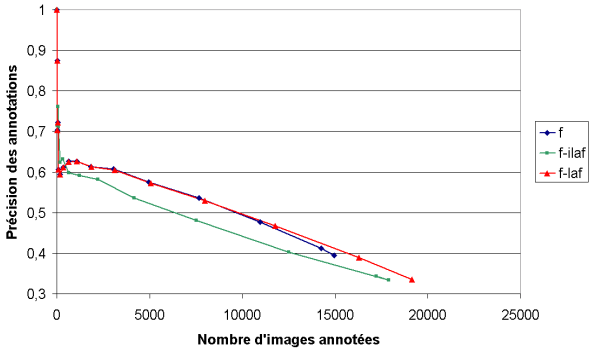



FIG. 5.5 – Performances des critères d'annotation de type laf pour les visages.

- le score reposant sur la fréquence des entités nommées uniquement permet d'annoter moins d'images que les autres scores. Ceci est dû au fait que ce score provoque beaucoup de cas d'ambiguïté (la probabilité que deux entités nommées apparaissent le même nombre de fois dans un texte étant assez élevée), contrairement aux autres scores qui combinent deux sources d'information, ce qui diminue la probabilité de trouver deux scores identiques ;
- les scores de type direct permettent d'améliorer légèrement les performances alors que les scores de type inverse font nettement diminuer la précision. Ceci est indicateur d'une tendance à illustrer les articles par des images contenant des entités communes plutôt que représentant des entités apparaissant dans peu d'articles (et dont on pourrait penser qu'elles soient plus spécifiques aux articles où elles apparaissent). Cette tendance peut s'expliquer de plusieurs manières, par exemple par des volontés de réutilisation des ressources photographiques disponibles, ou d'accrocher le lecteur par des concepts connus. La tendance observée peut donc se retrouver dans d'autres corpus de presse. Néanmoins, la faiblesse des améliorations obtenues par rapport à l'utilisation de la fréquence seule montre que cette dernière reste le facteur prépondérant pour sélectionner correctement les entités nommées.

## 5.4 Bilan

Nous nous sommes intéressés dans ce chapitre à l'utilisation du texte comme description du contenu sémantique des images. Dans un premier temps, nous vérifions expérimentalement l'existance du fossé sémantique, montrant ainsi la difficulté qui existe à trouver des

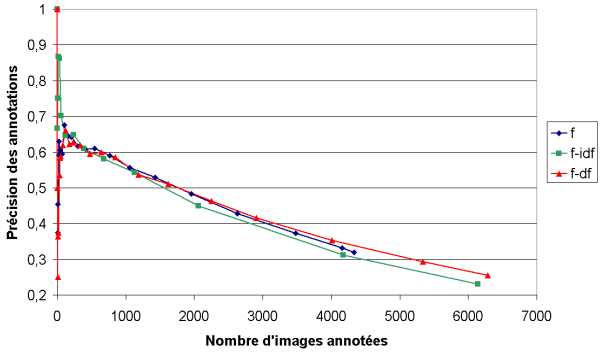



FIG. 5.6 – Performances des critères d'annotation de type df pour les logos.

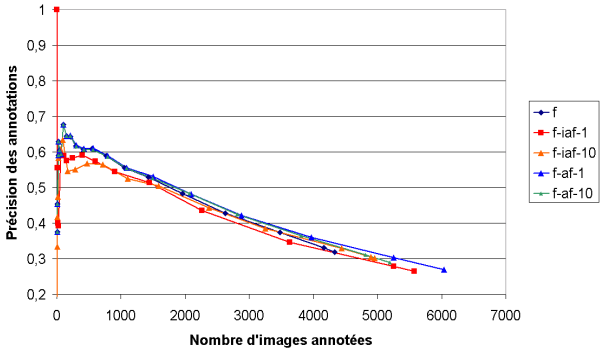



FIG. 5.7 – Performances des critères d'annotation de type af pour les logos.

relations entre les descripteurs de bas-niveau et les mots décrivant les images. Nous proposons ensuite une méthode d'annotation d'images qui contourne cet obstacle en employant uniquement des informations, visuelles ou textuelles, de haut-niveau. Nous nous appuyons en particulier sur la détection d'entités nommées, outil issu du TAL. Les résultats expérimentaux montrent que notre technique, malgré les erreurs dûes aux outils sur lesquels elle repose (détecteurs d'entités nommées, de visages, de logos) et la difficulté de travailler sur un corpus réel, donne des résultats satisfaisants. En comparaison des systèmes d'annotation classiques, y compris ceux qui travaillent sur des données artificielles plus faciles à traiter, ces résultats sont plutôt encourageants compte-tenu de la simplicité de notre approche. Ils montrent qu'il est possible d'obtenir des bons résultats d'annotation en exploitant de manière adaptée des informations textuelles et visuelles correctement choisies. Utiliser des techniques de TAL pour annoter des images à partir des textes les accompagnant semble donc être une voie prometteuse pour obtenir des systèmes d'annotation performants.

Enfin, une perspective immédiate pour notre système d'annotation serait d'améliorer le critère de sélection des entités nommées candidates. Les différents scores que nous proposons ne tiennent pas compte de la longueur des documents, or celle-ci influe sur la fréquence des entités nommées. De plus, le seuil est fixé de manière arbitraire pour l'ensemble des documents. Il serait sans doute bénéfique de sélectionner plutôt les entités nommées en fonction de leur importance relative dans le documents, en regardant celles qui ont une fréquence significativement supérieure aux autres entités nommées du même texte.

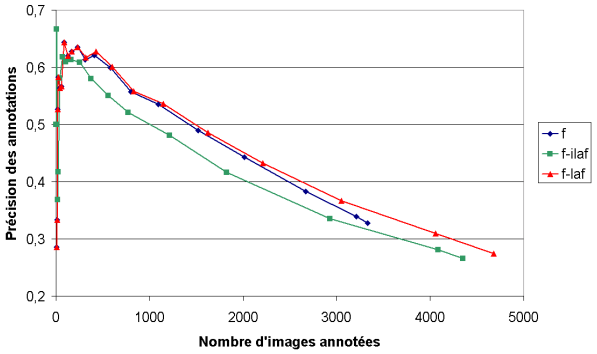
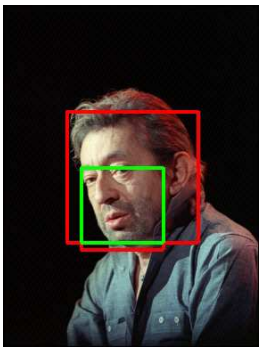



FIG. 5.8 – Performances des critères d'annotation de type laf pour les logos.

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



**Gainsbourg** 17
Nelson 2
Melody 2
Birkin 2
Hardy 2

**Arcelor** 17
Mittal 5
ADAM 2
Institutional Shareholders
Services 1
ISS 1
Association française 1

FIG. 5.9 – Des exemples d'annotations d'images obtenues avec notre système. Les entités nommées candidates et leurs fréquences sont indiquées, l'entité retenue pour annoter l'image est celle en gras.

# Conclusion

## Bilan

L'objectif de nos travaux était d'étudier les apports possibles des techniques de TAL aux systèmes d'indexation et de recherche d'images. Nous avons identifié deux niveaux auxquels ces apports pouvaient se faire :

- au niveau de la description du contenu visuel des images, en utilisant un formalisme de description des images inspiré de la description des textes utilisée en recherche d'information textuelle. Le formalisme que nous avons adopté pour cela est celui des mots visuels proposé par Sivic et Zisserman, qui permet de représenter les images comme des sacs de mots visuels, par analogie avec la représentation classique des textes en sacs de mots ;
- au niveau de la description textuelle des images. Nous nous sommes pour cela basés sur un corpus d'articles de presse que nous avons constitué et qui nous fournissait des images accompagnées de textes qui ne soient pas de simples descriptions des images. Il fallait donc pouvoir identifier dans ces textes les termes susceptibles de décrire le contenu sémantique des images.

Dans le cadre de la représentation des images par sacs de mots visuels, nous nous sommes intéressés à deux problématiques majeures du domaine du texte, problématiques qui se posent également dans le cadre des mots visuels.

La première de ces problématiques était : comment déterminer automatiquement les mots visuels les plus significatifs pour décrire les images ? Nous avons étudié les deux méthodes traditionnellement utilisées dans la communauté du texte pour répondre à cette question : les stop-lists et les pondérations. Nous avons proposé une méthode de constitution de stop-lists basée sur pLSA. Nous avons montré que cette nouvelle méthode permet d'éliminer une partie des mots visuels tout en n'accusant qu'une perte de performance limitée du système de recherche d'images, mais aussi d'améliorer les performances dans les tâches de classification d'images. Nous avons ensuite proposé une étude en profondeur des techniques de pondération, ainsi que des pondérations qui nous semblaient adaptées au cas de la recherche d'images. Nous avons complété cette étude par une étude des distances de Minkowski utilisées pour comparer les images. Les résultats obtenus ne permettent pas de préconiser l'usage d'un schéma de pondération précis pour l'ensemble des images, mais ils mettent en avant certaines propriétés des sacs de mots visuels :

- le schéma de pondération optimal dépend de la nature de l'objet requête. Nous avons en effet observé que, pour une majorité de requêtes, il est préférable de limiter le poids attribué aux mots visuels fréquents localement, mais aussi que pour certaines requêtes spécifiques, dont le contenu visuel est caractérisé par de nombreuses parties répétées (comme les bâtiments), ce sont les mots visuels fréquents qui doivent avoir un poids important ;

- il n'est pas souhaitable d'utiliser des pondérations quand le corpus contient des images catégorisées, dont le contenu varie fortement. D'une manière générale, il est apparu que la phase de quantification des descripteurs ne permet pas d'obtenir un système plus robuste aux variations visuelles, mais introduit plutôt du bruit dans la mise en correspondance des descripteurs. Cette conclusion est confirmée par la tendance à utiliser des vocabulaires visuels de très grande taille [NS06, PCI+07], ainsi que par l'efficacité d'approches comme le *hamming embedding* [JDS08] qui repose sur un découpage très fin de l'espace des descripteurs, pour s'approcher d'une mise en correspondance directe des descripteurs locaux. L'intérêt des mots visuels se situerait donc uniquement dans le gain de rapidité obtenu par rapport à une mise en correspondance directe des descripteurs locaux initiaux. Ceci a également été soulevé dans le cadre de la classification d'images [BSI08].

De plus, la comparaison entre l'effet des différentes distances de Minkowski et l'effet des pondérations a mis au jour un lien entre le paramètre $k$ des distances $L_k$ et l'usage de certaines pondérations, car les deux agissent sur l'amplitude des distances mesurées localement, au niveau d'un dimension donnée des vecteurs.

La seconde problématique étudiée dans le cadre des représentations par sacs de mots visuels était : comment prendre en compte les relations de proximité géométrique entre mots visuels au sein d'une image, et donc dépasser l'hypothèse d'indépendance des termes ? Pour cela, nous avons prolongé l'analogie entre texte et image en proposant une représentation des images sous forme de phrases visuelles qui permettent de conserver une partie de ces relations de proximité. Nous avons utilisé cette représentation avec des modèles de langues, un outil très populaire du TAL, et obtenu des améliorations de performances dans le cadre de la classification d'images. Cette notion de phrase visuelle se retrouve depuis dans d'autres travaux [AMC10, ESMUD10].

Dans le cadre de la représentation textuelle des images, nous avons d'abord cherché à caractériser le fossé sémantique souvent évoqué dans les travaux d'annotation d'images. Nous avons montré que, dans un corpus réel de taille convenable, il n'existe pas de corrélation entre les résultats d'une recherche d'image basée sur le texte et une recherche d'image basée sur des descripteurs de bas-niveau classiques, ce qui rend difficile la mise en place de méthodes d'annotation automatique basée sur des descripteurs de bas-niveau. Nous avons donc proposé une méthode d'annotation qui consiste à mettre en relation des concepts visuels de haut-niveau détectés dans les images avec des concepts textuels de heut niveau extraits des textes les accompagnant. Nous avons appliqué ce principe avec la mise en correspondance d'entités nommées avec des logos et des visages. Les performances obtenues, malgré la simplicité du système proposé, sont très acceptables compte-tenu de la difficulté des données employées et des limites des outils sur lesquels notre système repose.

## Perpectives

Nous avons déjà présenté quelques perspectives immédiates aux travaux que nous avons réalisés dans les conclusions des chapitres correspondants. Nous allons adopter ici un point de vue plus global sur le couplage du TAL et de la recherche d'images, mais aussi de la recherche d'images en général.

Le couplage entre TAL et recherche d'image par le contenu tel que nous l'avons étudié s'appuie essentiellement sur la représentation des images à base de mots visuels. Or, comme nous l'avons dit plus haut, une partie des travaux présentés ici met en avant certaines

limites de la représentation par mots visuels. De nombreux travaux [NS06, PCI+07, JDS08] convergent en effet implicitement vers l'idée selon laquelle la quantification des descripteurs n'est qu'un moyen d'approcher les performances de la mise en correspondance directe de descripteurs locaux avec un coût calculatoire acceptable pour des applications de recherche d'information. Ceci explique notamment la grande réussite des approches par mots visuels pour la reconnaissance de scènes identiques, où les variations entre images similaires sont essentiellement prises en charge par les propriétés d'invariance des détecteurs de régions d'intérêt et des descripteurs locaux. Cette tendance se confirme également dans le domaine de la classification d'images, où il apparaît que les facultés de généralisation des systèmes seraient plutôt dûes aux classifieurs employés qu'aux descripteurs quantifiés [BSI08]. De ce point de vue, l'utilisation du TAL en recherche d'images par le contenu trouve ses limites dans celles de la représentation par mots visuels.

D'un point de vue plus global sur la recherche d'images par le contenu, ces propriétés des mots visuels expliquent à la fois l'intérêt porté à la recherche de scènes identiques et le succès rencontré dans cette tâche ces dernières années. Ainsi, il est aujourd'hui possible d'effectuer des recherches de ce type sur de grandes quantités d'images (de l'ordre de plusieurs millions) avec une précision appréciable [JDS08, JDS09]. En revanche, la tâche de recherche d'images catégorisées, plus difficile, mais ayant un spectre d'utilisation beaucoup plus large du point de vue de l'utilisateur, n'est que très rarement abordée dans le cadre des mots visuels, à de rares exceptions près [ZWG06, Pha09]. Cette tâche, étudiée par les premiers travaux de recherche d'images (à l'aide de descripteurs globaux notamment), n'est particulièrement plus traitée dans les travaux les plus récents, qui se concentrent sur la recherche de scènes identiques ou l'annotation d'images. Elle présente clairement un défi de taille, en raison de la forte disparité visuelle qu'il peut exister entre différents éléments d'une même catégorie, mais aussi les points commun que peuvent présenter certaines catégories d'objets. Nous pensons que de nombreuses opportunités de recherche existent encore dans cette direction, pour laquelle les systèmes actuels présentent encore des performances très limitées, comme l'ont montré nos expériences du chapitre 3 sur le corpus Caltech-101. Un piste intéressante serait notamment d'étudier les moyens d'ajouter de la supervision à cette tâche, pour pouvoir capter les variations intra-classes des objets, comme nous l'avons évoqué en conclusion du chapitre 3 par exemple.

Nos travaux sur la nature du fossé sémantique posent aussi des questions intéressantes. Nous avons montré son existence de manière uniquement globale, c'est-à-dire entre des descripteurs globaux et l'ensemble des mots-clefs décrivant les images. Cette existence explique clairement les limites des nombreux modèles d'annotation d'images qui fonctionnent par apprentissage de mots-clefs à partir de descripteurs de bas-niveau, de manière globale. Une telle étude devrait également se faire localement, c'est-à-dire en cherchant des corrélations entre certains descripteurs et certaines catégories de mots-clefs. Il semble en effet évident que certains descripteurs sont très adaptés à certains types d'objets (la couleur pour le ciel, la forme pour les voitures...) et d'autres non. Cette propriété des descripteurs avait été mise en évidence dans quelques travaux de Tollari *et al.* [Tol05]. Elle explique aussi le récent succès des techniques d'annotation basées sur la propagation de mots-clefs et l'apprentissage de distances [GMVS09], ou encore des méthodes de sélection de descripteurs à l'issue d'une première recherche basée sur le texte, pour en reclasser les résultats [KAVJ10]. Mieux saisir les relations entre mots-clefs, concepts et descripteurs permettrait de mieux comprendre et mieux sélectionner ces derniers, pour des tâches d'annotation comme de recherche par le contenu.

D'une manière plus générale, une meilleure compréhension des descripteurs permettrait

de mieux formaliser les tâches de recherche d'images. Il serait alors possible d'exploiter certains travaux de recherche d'information textuelle d'une autre manière que celle que nous avons employée ici : pas en appliquant directement des techniques de TAL à des descripteurs d'images, mais en empruntant la méthodologie utilisée pour formaliser le fonctionnement et les propriétés souhaitées des systèmes de recherche d'information, comme l'ont fait Fang *et al.* [FTZ04], puis Clinchant et Gaussier [CG10], et de mieux exploiter les modèles de recherche d'information existant dans le cadre de la recherche d'images.

Enfin, nos travaux montrent qu'il y a fort à gagner à exploiter les informations textuelles accompagnant les images quand il en existe, et que les outils du TAL permettent de le faire efficacement. Il est intéressant de se demander dans quelle mesure la méthode que nous avons proposée peut s'étendre à d'autres associations que celle entre entités nommées et visages ou logos, et en particulier à des types de concepts plus génériques. Ici aussi, mieux connaître les descripteurs et les ensembles de concepts auxquels ils peuvent correspondent, comme nous le proposions plus haut, permettrait d'étendre plus largement notre approche d'annotation. Cette dernière ouvre de plus la voie à l'utilisation de nombreuses autres techniques du TAL, pour renforcer la présence des entités ou concepts détectés dans les textes. En effet, la seule fréquence d'un terme est un indice assez faible de la présence du concept attenant dans un texte, et l'usage de méthodes comme l'extraction de cooccurrences, l'annotation d'entités nommées ou la résolution d'anaphores permettrait d'avoir une idée plus précise de cette présence, et donc de mieux saisir l'intérêt du texte. Les possibilités d'application du TAL pour la recherche d'images sont donc nombreuses en recherche sémantique. Des méthodes similaires pourraient également être exploitées non pas dans un cadre d'annotation, mais de recherche combinée entre descripteurs visuels et descripteurs textuels, en améliorant la qualité de ces derniers.

Enfin, nous nous sommes limités à un type de données, des articles de presse d'une taille limitée, pour notre étude, mais les supports mélangant image et texte sont nombreux. Un premier exemple est celui de la vidéo, qui mélange des séquences d'images avec le texte issu des bandes-son. Ce média fournit un cadre très particulier de couplage entre texte et images, pour au moins deux raisons. Premièrement, le texte obtenu par transcription à partir de la bande-son est contient de nombreuses erreurs et n'a pas de structure (ponctuation, paragraphes...), ce qui rend difficile l'application directe d'outils du TAL et nécessite de les adapter à ce contenu particulier. Deuxièmement, il ne s'agit pas d'utiliser des textes illustrant ou illustrés par des images, mais des textes vraiment complémentaires aux images, avec une dimension de synchronisation temporelle qui n'existait pas dans les données que nous avons étudiées. L'indexation de vidéos constitue donc un cadre de couplage de méthodes du TAL et de méthodes d'indexation d'images très particulier, différent de celui dans lequel nous nous sommes placé ici et pour le quel de nombreuses perspectives de recherche restent ouvertes.

Les données issues des blogs et réseaux sociaux constituent un autre exemple de données mêlant textes et images. la production de données de ce type a récemment explosé, comme nous l'avions évoqué en début d'introduction, et offre un nouveau cadre de recherche intéressant. Ce nouveau type de support offre en effet la possibilité d'exploiter des méthodes de TAL pour indexer des images, mais également des vidéos qui sont maintenant très souvent embarquées dans des blogs, souvent par différents utilisateurs, ce qui multiplie les sources de textes décrivant une même image ou vidéo, et donne la possibilité d'affiner sa description. Enfin, ces réseaux sociaux offrent de nouvelles perspectives en termes d'indexation en général, et d'images ou de vidéos en particulier, de par leur aspect communautaire : les relations entre les utilisateurs permettent de mieux cerner les documents qu'ils s'échangent

ou partagent, et donc de mieux les décrire, par exemple en se basant sur ces relations pour propager des annotations, comme dans les travaux de Sawant *et al.* [SDLW10], mais aussi en exploitant des données textuelles plus riches (commentaires, articles échangés...). Une dernière perspective ouverte par ces communautés d'utilisateurs est d'utiliser les liens entre utilisateurs non pas pour obtenir une meilleure description "objective" des images, mais pour personnaliser la description des images en fonction des utilisateurs, par exemple en s'adaptant au vocabulaire qu'ils utilisent lors du choix des annotations.

# Annexe A

# Expériences sur les pondérations : détail des résultats

Dans cette annexe, nous présentons les résultats détaillés des expériences sur les pondérations que nous avons effectuées dans le chapitre 3. Avant de présenter les tableaux contenant les résultats complets de chaque expérimentation, nous rappelons les schémas de pondération, distances, corpus et mesures de performances utilisées.

## A.1 Corpus utilisés

Nous avons réalisé ces expériences sur les quatres corpus présentés dans la section 3.5.1 page 79. Rappelons que ces différents corpus correspondent à deux tâches distinctes de recherche d'images. Les corpus Caltech-6 et Caltech-101 correspondent à la tâche de recherche d'images catégorisées, dans laquelle nous disposons d'images organisées en catégories (visages, avions...). Nous considérons dans cette tâche que chaque image appartenant à une catégorie similaire à celle de la requête est pertinente. Les variations visuelles au sein d'une même classe peuvent être importantes, ces qui rend cette tâche particulièrement difficile. Les corpus Kentucky et Oxford, quant à eux, correspondent à une tâche de recherche de scènes similaires. Dans cette tâche, une image est considérée comme pertinente si elle représente la même scène ou le même objet que la requête, avec des variations d'échelle, de point de vue ou d'illumination uniquement, ainsi que d'éventuelles occultations. Il n'y a pas de variations dans les formes ou les couleurs de objets, contrairement au cas de la recherche d'images catégorisées, ce qui en fait une tâche moins difficile.

## A.2 Schémas de pondération

Les schémas de pondération utilisés sont soit construits comme une combinaison d'un pondération locale et d'une pondération globale, soit des poids issus des modèles DFR de recherche d'information. Ils sont présentés de la manière suivante :

- schéma de pondération $l_i g_j$ : il est la combinaison (produit) du poids local $l_i$ et du poids global $g_j$. Les poids locaux utilisés sont résumés dans le tableau 3.7 page 85 et les poids globaux dans le tableau 3.8 page 85 ;
- schéma de pondération DFR $XYZ$ : il est la combinaison d'un modèle aléatoire $X$, d'un modèle de divergence $Y$ et d'une normalisation $Z$. Les différents modèles aléatoires utilisés sont résumés dans le tableau 3.9 page 86, les modèles de divergence dans le tableau 3.5.5.1 page 86, et les normalisations dans le tableau 3.3 page 78.

## A.3 Distances employées

Nous avons employé deux distances de Minkowski classiques pour comparer les schémas de pondération, $L_1$ et $L_2$. Rappelons que la distance $L_2$, utilisée ici sur des vecteurs normalisés, est équivalente (au sens de l'ordre induit) à la distance du cosinus, distance classiquement employée en recherche d'information textuelle (voir section 1.3.2.4 page 28). Les résultats présentés pour chaque schéma de pondération et chaque distance doivent être lus à la lumière des observations sur les effets similaires et potentiellement cumulés des distances de Minkowski et des pondérations que nous avons formulées en section 3.5.7.2 page 94.

## A.4 Mesures de performance

Nous utilisons deux mesures de performances classiques de la recherche d'information : la précision et la MAP. La précision nous donne la proportion de documents pertinents parmi les premiers documents retournés, qui sont les plus intéressants à considérer car les seuls consultés par les utilisateurs. Les précisions ont été calculées à différentes DCV, en fonction des propriétés des corpus employés. Nous employons des DCV à 10, 20, 50 et 100 documents pour les corpus Caltech-6 et Caltech-101 car ceux-ci contiennent un grand nombre d'images pertinentes pour certaines requêtes (plus de 1000 pour certaines catégories de Caltech-6). Nous n'employons qu'un DCV à 4 documents pour le corpus Kentucky car il ne possède que des groupes de 4 images pertinentes, requête comprise. Enfin, nous employons des DCV à 5, 10, 20 et 50 documents pour le corpus Oxford qui contient un nombre variable de documents pertinents pou chaque requête, mais en quantité moindre que dans les corpus Caltech-6 et Caltech-101. La MAP nous fournit quant à elle une mesure plus globale des performances, et reflète en particulier le fait qu'il existe ou non des documents pertinents particulièrement mal classés.

Pour chaque mesure de performance, distance et corpus considérés, le meilleur résultat obtenu est souligné. Nous indiquons de plus en gras ce résultat, ainsi que tous les résultats dont la différence ce dernier n'est pas statistiquement significative. Nous avons pour cela utilisé le test de Wilcoxon, avec un seuil de *p-value* fixé à 0.05.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | 0.7685 | 0.7102 | 0.6318 | 0.5712 | 0.3827 |
| $l_1g_1$ | 0.7797 | 0.7228 | 0.6445 | 0.5833 | 0.3893 |
| $l_1g_2$ | 0.7822 | 0.7261 | 0.6458 | 0.5855 | 0.3903 |
| $l_1g_3$ | 0.7848 | 0.7284 | 0.6538 | 0.5919 | 0.3932 |
| $l_1g_4$ | 0.7827 | 0.7244 | 0.6481 | 0.5867 | 0.3908 |
| $l_1g_5$ | **0.7964** | 0.7332 | 0.6455 | 0.5882 | 0.3938 |
| $l_2g_0$ | 0.7777 | 0.7211 | 0.6426 | 0.5823 | 0.3884 |
| $l_2g_1$ | 0.7853 | 0.7315 | 0.6556 | 0.5945 | 0.3943 |
| $l_2g_2$ | 0.7838 | 0.7345 | 0.6576 | 0.5973 | 0.3951 |
| $l_2g_3$ | 0.7822 | 0.7353 | 0.6617 | 0.6005 | 0.3972 |
| $l_2g_4$ | 0.7873 | 0.7419 | 0.6605 | 0.6004 | 0.3973 |
| $l_2g_5$ | **0.8005** | 0.7434 | 0.6692 | 0.6104 | 0.4030 |
| $l_3g_0$ | 0.7883 | 0.7332 | 0.6562 | 0.5911 | 0.3900 |
| $l_3g_1$ | **0.8005** | 0.7431 | 0.6681 | 0.5994 | 0.3948 |
| $l_3g_2$ | **0.8000** | 0.7434 | 0.6677 | 0.5995 | 0.3954 |
| $l_3g_3$ | **0.7954** | 0.7442 | 0.6718 | 0.6022 | 0.3965 |
| $l_3g_4$ | **0.8036** | 0.7444 | 0.6740 | 0.6065 | 0.3986 |
| $l_3g_5$ | **0.8020** | **0.7548** | **0.6825** | **0.6193** | **0.4049** |
| $l_4g_0$ | 0.7848 | 0.7282 | 0.6522 | 0.5896 | 0.3880 |
| $l_4g_1$ | **0.7949** | 0.7363 | 0.6642 | 0.5957 | 0.3925 |
| $l_4g_2$ | **0.7944** | 0.7373 | 0.6659 | 0.5975 | 0.3931 |
| $l_4g_3$ | **0.7934** | 0.7409 | 0.6658 | 0.5995 | 0.3940 |
| $l_4g_4$ | 0.7929 | 0.7416 | 0.6695 | 0.6051 | 0.3966 |
| $l_4g_5$ | **0.7970** | 0.7482 | **0.6818** | **0.6174** | 0.4032 |
| $l_5g_0$ | 0.7685 | 0.7102 | 0.6318 | 0.5712 | 0.3827 |
| $l_5g_1$ | 0.7797 | 0.7228 | 0.6445 | 0.5833 | 0.3893 |
| $l_5g_2$ | 0.7822 | 0.7261 | 0.6458 | 0.5855 | 0.3903 |
| $l_5g_3$ | 0.7848 | 0.7284 | 0.6538 | 0.5919 | 0.3932 |
| $l_5g_4$ | 0.7827 | 0.7244 | 0.6481 | 0.5867 | 0.3908 |
| $l_5g_5$ | **0.7964** | 0.7332 | 0.6455 | 0.5882 | 0.3938 |
| $l_6g_0$ | 0.7193 | 0.6503 | 0.5735 | 0.5128 | 0.3569 |
| $l_6g_1$ | 0.7279 | 0.6589 | 0.5866 | 0.5262 | 0.3636 |
| $l_6g_2$ | 0.7259 | 0.6657 | 0.5897 | 0.5284 | 0.3648 |
| $l_6g_3$ | 0.7355 | 0.6739 | 0.5948 | 0.5338 | 0.3685 |
| $l_6g_4$ | 0.7269 | 0.6622 | 0.5807 | 0.5199 | 0.3603 |
| $l_6g_5$ | 0.7218 | 0.6596 | 0.5791 | 0.5219 | 0.3601 |
| $l_7g_0$ | 0.7853 | 0.7254 | 0.6519 | 0.5898 | 0.3917 |
| $l_7g_1$ | 0.7914 | 0.7406 | 0.6641 | 0.6003 | 0.3971 |
| $l_7g_2$ | 0.7883 | 0.7406 | 0.6660 | 0.6008 | 0.3978 |
| $l_7g_3$ | 0.7888 | 0.7393 | 0.6688 | 0.6038 | 0.3993 |
| $l_7g_4$ | **0.7949** | 0.7444 | 0.6712 | 0.6065 | 0.4007 |
| $l_7g_5$ | **0.8015** | **0.7536** | 0.6791 | **0.6198** | **0.4068** |

TAB. A.1 – Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Caltech6.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | 0.6208 | 0.5518 | 0.4678 | 0.4150 | **0.3287** |
| $l_1g_1$ | 0.6503 | 0.5680 | 0.4728 | 0.4212 | **0.3368** |
| $l_1g_2$ | **0.6533** | 0.5695 | 0.4739 | 0.4224 | **0.3378** |
| $l_1g_3$ | **0.6665** | **0.5832** | 0.4823 | 0.4259 | **0.3406** |
| $l_1g_4$ | **0.6574** | **0.5797** | **0.4877** | **0.4316** | **0.3342** |
| $l_1g_5$ | 0.6315 | **0.5685** | **0.4931** | **0.4387** | **0.3268** |
| $l_2g_0$ | 0.5878 | 0.4959 | 0.4066 | 0.3644 | 0.3195 |
| $l_2g_1$ | 0.5985 | 0.5094 | 0.4189 | 0.3709 | 0.3280 |
| $l_2g_2$ | 0.5995 | 0.5124 | 0.4214 | 0.3722 | 0.3291 |
| $l_2g_3$ | 0.6198 | 0.5332 | 0.4380 | 0.3853 | 0.3335 |
| $l_2g_4$ | 0.6411 | 0.5574 | 0.4535 | 0.4041 | 0.3343 |
| $l_2g_5$ | **0.6741** | **0.5906** | **0.4902** | 0.4276 | **0.3301** |
| $l_3g_0$ | 0.5406 | 0.4614 | 0.3678 | 0.3275 | 0.3101 |
| $l_3g_1$ | 0.5624 | 0.4751 | 0.3755 | 0.3339 | 0.3176 |
| $l_3g_2$ | 0.5655 | 0.4723 | 0.3787 | 0.3349 | 0.3185 |
| $l_3g_3$ | 0.5766 | 0.4848 | 0.3974 | 0.3514 | 0.3228 |
| $l_3g_4$ | 0.5985 | 0.5140 | 0.4209 | 0.3708 | 0.3275 |
| $l_3g_5$ | 0.6386 | 0.5627 | 0.4743 | 0.4066 | 0.3253 |
| $l_4g_0$ | 0.5254 | 0.4447 | 0.3540 | 0.3137 | 0.3064 |
| $l_4g_1$ | 0.5487 | 0.4538 | 0.3616 | 0.3213 | 0.3135 |
| $l_4g_2$ | 0.5503 | 0.4548 | 0.3608 | 0.3233 | 0.3143 |
| $l_4g_3$ | 0.5604 | 0.4690 | 0.3828 | 0.3380 | 0.3185 |
| $l_4g_4$ | 0.5777 | 0.4952 | 0.4028 | 0.3552 | 0.3233 |
| $l_4g_5$ | 0.6208 | 0.5541 | 0.4640 | 0.4037 | 0.3241 |
| $l_5g_0$ | 0.6208 | 0.5518 | 0.4678 | 0.4150 | **0.3287** |
| $l_5g_1$ | 0.6503 | 0.5680 | 0.4728 | 0.4212 | **0.3368** |
| $l_5g_2$ | **0.6533** | 0.5695 | 0.4739 | 0.4224 | **0.3378** |
| $l_5g_3$ | **0.6665** | **0.5832** | 0.4823 | 0.4259 | **0.3406** |
| $l_5g_4$ | **0.6574** | **0.5797** | **0.4877** | **0.4316** | **0.3342** |
| $l_5g_5$ | 0.6315 | **0.5685** | **0.4931** | **0.4387** | **0.3268** |
| $l_6g_0$ | 0.5645 | 0.5175 | 0.4636 | 0.4171 | 0.2952 |
| $l_6g_1$ | 0.5838 | 0.5302 | 0.4776 | **0.4358** | **0.3062** |
| $l_6g_2$ | 0.5802 | 0.5289 | 0.4770 | **0.4371** | **0.3077** |
| $l_6g_3$ | 0.5766 | 0.5312 | 0.4755 | **0.4392** | 0.3102 |
| $l_6g_4$ | 0.5761 | 0.5140 | 0.4645 | **0.4233** | **0.2985** |
| $l_6g_5$ | 0.5579 | 0.4997 | 0.4496 | 0.4133 | **0.2967** |
| $l_7g_0$ | 0.5543 | 0.4744 | 0.3870 | 0.3463 | 0.3161 |
| $l_7g_1$ | 0.5812 | 0.4952 | 0.4038 | 0.3568 | 0.3251 |
| $l_7g_2$ | 0.5858 | 0.4982 | 0.4063 | 0.3594 | 0.3263 |
| $l_7g_3$ | 0.6010 | 0.5114 | 0.4242 | 0.3747 | 0.3309 |
| $l_7g_4$ | 0.6137 | 0.5299 | 0.4348 | 0.3878 | 0.3331 |
| $l_7g_5$ | 0.6431 | 0.5718 | 0.4759 | 0.4129 | **0.3287** |

TAB. A.2 – Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Caltech-6.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | **0.3146** | **0.2596** | **0.2030** | 0.1656 | 0.1071 |
| $l_1g_1$ | **0.3101** | 0.2553 | **0.2003** | 0.1597 | **0.1060** |
| $l_1g_2$ | **0.3106** | 0.2553 | **0.2003** | 0.1597 | **0.1060** |
| $l_1g_3$ | **0.3071** | 0.2457 | 0.1910 | 0.1541 | 0.1034 |
| $l_1g_4$ | **0.3086** | 0.2535 | 0.1982 | **0.1644** | **0.1072** |
| $l_1g_5$ | 0.2869 | 0.2396 | 0.1864 | 0.1577 | 0.1042 |
| $l_2g_0$ | **0.3167** | 0.2604 | **0.2048** | **0.1667** | **0.1077** |
| $l_2g_1$ | **0.3116** | 0.2548 | **0.2008** | 0.1609 | **0.1063** |
| $l_2g_2$ | **0.3116** | 0.2548 | **0.2005** | 0.1607 | **0.1062** |
| $l_2g_3$ | **0.3076** | 0.2465 | 0.1917 | 0.1547 | 0.1034 |
| $l_2g_4$ | **0.3091** | **0.2571** | 0.2006 | **0.1651** | **0.1078** |
| $l_2g_5$ | 0.2848 | 0.2409 | 0.1889 | 0.1584 | 0.1048 |
| $l_3g_0$ | **0.3152** | **0.2624** | **0.2040** | **0.1663** | **0.1081** |
| $l_3g_1$ | **0.3141** | 0.2545 | 0.1982 | 0.1604 | 0.1063 |
| $l_3g_2$ | **0.3136** | 0.2538 | 0.1983 | 0.1603 | 0.1062 |
| $l_3g_3$ | **0.3066** | 0.2439 | 0.1919 | 0.1539 | 0.1033 |
| $l_3g_4$ | **0.3101** | **0.2561** | 0.1997 | **0.1640** | **0.1080** |
| $l_3g_5$ | 0.2884 | 0.2422 | 0.1888 | 0.1580 | 0.1051 |
| $l_4g_0$ | **0.3192** | **0.2609** | 0.2009 | **0.1660** | **0.1074** |
| $l_4g_1$ | **0.3111** | 0.2528 | 0.1976 | 0.1587 | 0.1059 |
| $l_4g_2$ | **0.3111** | 0.2525 | 0.1975 | 0.1586 | 0.1058 |
| $l_4g_3$ | 0.3056 | 0.2427 | 0.1911 | 0.1529 | 0.1027 |
| $l_4g_4$ | 0.3081 | 0.2558 | 0.1978 | **0.1627** | **0.1077** |
| $l_4g_5$ | 0.2879 | 0.2417 | 0.1881 | 0.1569 | 0.1049 |
| $l_5g_0$ | **0.3146** | **0.2596** | **0.2030** | 0.1656 | 0.1071 |
| $l_5g_1$ | **0.3101** | 0.2553 | **0.2003** | 0.1597 | **0.1060** |
| $l_5g_2$ | **0.3106** | 0.2553 | **0.2003** | 0.1597 | 0.1060 |
| $l_5g_3$ | **0.3071** | 0.2457 | 0.1910 | 0.1541 | 0.1034 |
| $l_5g_4$ | **0.3086** | 0.2535 | 0.1982 | **0.1644** | **0.1072** |
| $l_5g_5$ | 0.2869 | 0.2396 | 0.1864 | 0.1577 | 0.1042 |
| $l_6g_0$ | 0.2975 | 0.2371 | 0.1887 | 0.1558 | 0.1030 |
| $l_6g_1$ | 0.2975 | 0.2369 | 0.1869 | 0.1541 | 0.1022 |
| $l_6g_2$ | 0.2970 | 0.2369 | 0.1873 | 0.1540 | 0.1022 |
| $l_6g_3$ | 0.2939 | 0.2348 | 0.1812 | 0.1501 | 0.1006 |
| $l_6g_4$ | 0.2949 | 0.2356 | 0.1862 | 0.1556 | 0.1028 |
| $l_6g_5$ | 0.2778 | 0.2237 | 0.1754 | 0.1513 | 0.1002 |
| $l_7g_0$ | 0.3157 | 0.2614 | 0.2037 | 0.1661 | 0.1078 |
| $l_7g_1$ | **0.3126** | 0.2538 | 0.1990 | **0.1604** | 0.1062 |
| $l_7g_2$ | **0.3121** | 0.2540 | 0.1989 | **0.1603** | 0.1061 |
| $l_7g_3$ | **0.3071** | 0.2434 | 0.1908 | 0.1542 | 0.1033 |
| $l_7g_4$ | **0.3086** | **0.2578** | 0.1989 | **0.1643** | **0.1078** |
| $l_7g_5$ | 0.2864 | 0.2427 | 0.1886 | 0.1581 | 0.1048 |

TAB. A.3 – Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Caltech-101.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | 0.2727 | 0.2237 | **0.1823** | **0.1524** | 0.0985 |
| $l_1g_1$ | 0.2747 | 0.2222 | 0.1713 | 0.1426 | 0.0961 |
| $l_1g_2$ | 0.2753 | 0.2227 | 0.1709 | 0.1422 | 0.0960 |
| $l_1g_3$ | 0.2490 | 0.1985 | 0.1483 | 0.1239 | 0.0895 |
| $l_1g_4$ | 0.2611 | 0.2091 | 0.1697 | 0.1435 | 0.0959 |
| $l_1g_5$ | 0.2020 | 0.1636 | 0.1374 | 0.1228 | 0.0879 |
| $l_2g_0$ | 0.2838 | 0.2336 | **0.1848** | **0.1542** | **0.1005** |
| $l_2g_1$ | 0.2884 | 0.2280 | 0.1729 | 0.1428 | 0.0971 |
| $l_2g_2$ | 0.2889 | 0.2278 | **0.1731** | 0.1425 | 0.0971 |
| $l_2g_3$ | 0.2561 | 0.1972 | 0.1488 | 0.1204 | 0.0894 |
| $l_2g_4$ | 0.2727 | 0.2210 | 0.1710 | 0.1459 | 0.0977 |
| $l_2g_5$ | 0.2025 | 0.1636 | 0.1369 | 0.1216 | 0.0885 |
| $l_3g_0$ | 0.2843 | 0.2333 | **0.1837** | **0.1533** | **0.1006** |
| $l_3g_1$ | 0.2843 | 0.2283 | **0.1708** | 0.1407 | 0.0964 |
| $l_3g_2$ | 0.2843 | 0.2278 | **0.1705** | 0.1404 | 0.0964 |
| $l_3g_3$ | 0.2551 | 0.1912 | 0.1435 | 0.1155 | 0.0883 |
| $l_3g_4$ | 0.2732 | 0.2210 | 0.1717 | 0.1444 | 0.0981 |
| $l_3g_5$ | 0.2020 | 0.1654 | 0.1346 | 0.1194 | 0.0882 |
| $l_4g_0$ | 0.2823 | 0.2273 | **0.1819** | 0.1501 | 0.0993 |
| $l_4g_1$ | 0.2747 | 0.2182 | 0.1674 | 0.1366 | 0.0946 |
| $l_4g_2$ | 0.2747 | 0.2179 | 0.1672 | 0.1363 | 0.0945 |
| $l_4g_3$ | 0.2465 | 0.1866 | 0.1406 | 0.1110 | 0.0866 |
| $l_4g_4$ | 0.2631 | 0.2136 | 0.1683 | 0.1415 | 0.0970 |
| $l_4g_5$ | 0.2010 | 0.1626 | 0.1312 | 0.1165 | 0.0872 |
| $l_5g_0$ | 0.2727 | 0.2237 | **0.1823** | **0.1524** | 0.0985 |
| $l_5g_1$ | 0.2747 | 0.2222 | 0.1713 | 0.1426 | 0.0961 |
| $l_5g_2$ | 0.2753 | 0.2227 | 0.1709 | 0.1422 | 0.0960 |
| $l_5g_3$ | 0.2490 | 0.1985 | 0.1483 | 0.1239 | 0.0895 |
| $l_5g_4$ | 0.2611 | 0.2091 | 0.1697 | 0.1435 | 0.0959 |
| $l_5g_5$ | 0.2020 | 0.1636 | 0.1374 | 0.1228 | 0.0879 |
| $l_6g_0$ | 0.2303 | 0.1871 | 0.1476 | 0.1294 | 0.0891 |
| $l_6g_1$ | 0.2222 | 0.1780 | 0.1443 | 0.1261 | 0.0879 |
| $l_6g_2$ | 0.2222 | 0.1780 | 0.1446 | 0.1262 | 0.0879 |
| $l_6g_3$ | 0.2141 | 0.1679 | 0.1333 | 0.1156 | 0.0848 |
| $l_6g_4$ | 0.2278 | 0.1790 | 0.1445 | 0.1262 | 0.0884 |
| $l_6g_5$ | 0.2015 | 0.1578 | 0.1320 | 0.1173 | 0.0848 |
| $l_7g_0$ | **0.2899** | **0.2356** | 0.1835 | 0.1526 | **0.1005** |
| $l_7g_1$ | 0.2884 | 0.2273 | **0.1709** | 0.1406 | 0.0966 |
| $l_7g_2$ | 0.2884 | 0.2275 | 0.1710 | 0.1403 | 0.0965 |
| $l_7g_3$ | 0.2586 | 0.1929 | 0.1442 | 0.1162 | 0.0885 |
| $l_7g_4$ | 0.2737 | 0.2227 | 0.1730 | 0.1449 | 0.0980 |
| $l_7g_5$ | 0.2035 | 0.1636 | 0.1352 | 0.1206 | 0.0882 |

TAB. A.4 – Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Caltech-101.

| | P@4 | MAP |
|---|---|---|
| $l_1g_0$ | 0.6970 | 0.5229 |
| $l_1g_1$ | 0.7104 | 0.5316 |
| $l_1g_2$ | 0.7113 | 0.5318 |
| $l_1g_3$ | 0.7121 | **0.5349** |
| $l_1g_4$ | 0.6970 | 0.5232 |
| $l_1g_5$ | 0.6540 | 0.5024 |
| $l_2g_0$ | 0.7029 | 0.5254 |
| $l_2g_1$ | **0.7180** | **0.5341** |
| $l_2g_2$ | **0.7180** | **0.5347** |
| $l_2g_3$ | **0.7197** | **0.5371** |
| $l_2g_4$ | 0.7045 | 0.5277 |
| $l_2g_5$ | 0.6658 | 0.5083 |
| $l_3g_0$ | 0.6886 | 0.5228 |
| $l_3g_1$ | 0.7088 | **0.5324** |
| $l_3g_2$ | 0.7096 | **0.5332** |
| $l_3g_3$ | **0.7172** | **0.5340** |
| $l_3g_4$ | 0.6995 | 0.5277 |
| $l_3g_5$ | 0.6582 | 0.5060 |
| $l_4g_0$ | 0.6852 | 0.5209 |
| $l_4g_1$ | 0.6995 | **0.5294** |
| $l_4g_2$ | 0.6995 | **0.5303** |
| $l_4g_3$ | 0.7104 | **0.5320** |
| $l_4g_4$ | 0.6886 | 0.5249 |
| $l_4g_5$ | 0.6591 | 0.5044 |
| $l_5g_0$ | 0.6970 | 0.5229 |
| $l_5g_1$ | 0.7104 | 0.5316 |
| $l_5g_2$ | 0.7113 | 0.5318 |
| $l_5g_3$ | 0.7121 | **0.5349** |
| $l_5g_4$ | 0.6970 | 0.5232 |
| $l_5g_5$ | 0.6540 | 0.5024 |
| $l_6g_0$ | 0.6279 | 0.4891 |
| $l_6g_1$ | 0.6557 | 0.4996 |
| $l_6g_2$ | 0.6566 | 0.5003 |
| $l_6g_3$ | 0.6675 | 0.5075 |
| $l_6g_4$ | 0.6355 | 0.4875 |
| $l_6g_5$ | 0.6103 | 0.4705 |
| $l_7g_0$ | 0.6995 | 0.5243 |
| $l_7g_1$ | **0.7146** | **0.5340** |
| $l_7g_2$ | **0.7138** | **0.5347** |
| $l_7g_3$ | **0.7180** | **0.5361** |
| $l_7g_4$ | 0.7029 | 0.5280 |
| $l_7g_5$ | 0.6675 | 0.5079 |

TAB. A.5 – Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Kentucky.

| | P@4 | MAP |
|---|---|---|
| $l_1g_0$ | 0.5985 | 0.4611 |
| $l_1g_1$ | 0.6254 | 0.4792 |
| $l_1g_2$ | 0.6305 | **0.4811** |
| $l_1g_3$ | **0.6414** | **0.4847** |
| $l_1g_4$ | 0.5766 | 0.4502 |
| $l_1g_5$ | 0.4579 | 0.3682 |
| $l_2g_0$ | 0.6271 | 0.4753 |
| $l_2g_1$ | **0.6507** | 0.4918 |
| $l_2g_2$ | **0.6524** | **0.4920** |
| $l_2g_3$ | **0.6549** | **0.4917** |
| $l_2g_4$ | 0.6019 | 0.4641 |
| $l_2g_5$ | 0.4731 | 0.3715 |
| $l_3g_0$ | 0.6313 | 0.4744 |
| $l_3g_1$ | **0.6557** | **0.4884** |
| $l_3g_2$ | **0.6557** | 0.4884 |
| $l_3g_3$ | **0.6524** | 0.4861 |
| $l_3g_4$ | 0.6103 | 0.4644 |
| $l_3g_5$ | 0.4638 | 0.3626 |
| $l_4g_0$ | 0.6296 | 0.4714 |
| $l_4g_1$ | 0.6481 | 0.4829 |
| $l_4g_2$ | 0.6490 | 0.4833 |
| $l_4g_3$ | 0.6397 | 0.4798 |
| $l_4g_4$ | 0.6086 | 0.4632 |
| $l_4g_5$ | 0.4714 | 0.3625 |
| $l_5g_0$ | 0.5985 | 0.4611 |
| $l_5g_1$ | 0.6254 | 0.4792 |
| $l_5g_2$ | 0.6305 | **0.4811** |
| $l_5g_3$ | **0.6414** | **0.4847** |
| $l_5g_4$ | 0.5766 | 0.4502 |
| $l_5g_5$ | 0.4579 | 0.3682 |
| $l_6g_0$ | 0.4184 | 0.3460 |
| $l_6g_1$ | 0.4428 | 0.3678 |
| $l_6g_2$ | 0.4461 | 0.3705 |
| $l_6g_3$ | 0.4604 | 0.3801 |
| $l_6g_4$ | 0.4125 | 0.3436 |
| $l_6g_5$ | 0.3763 | 0.3169 |
| $l_7g_0$ | 0.6288 | 0.4746 |
| $l_7g_1$ | **0.6498** | **0.4899** |
| $l_7g_2$ | **0.6549** | **0.4910** |
| $l_7g_3$ | **0.6498** | **0.4881** |
| $l_7g_4$ | 0.6103 | 0.4665 |
| $l_7g_5$ | 0.4739 | 0.3704 |

TAB. A.6 – Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Kentucky.

| | P@5 | P@10 | P@20 | P@50 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | 0.7200 | 0.5600 | 0.4218 | 0.2764 | 0.2698 |
| $l_1g_1$ | 0.7200 | 0.5800 | 0.4400 | 0.2844 | 0.2792 |
| $l_1g_2$ | 0.7200 | 0.5818 | 0.4400 | 0.2855 | 0.2795 |
| $l_1g_3$ | **0.7236** | **0.5945** | **0.4518** | 0.2920 | 0.2853 |
| $l_1g_4$ | 0.7164 | 0.5764 | 0.4327 | 0.2855 | 0.2798 |
| $l_1g_5$ | **0.7273** | 0.5855 | **0.4464** | 0.2887 | 0.2874 |
| $l_2g_0$ | 0.7091 | 0.5618 | 0.4255 | 0.2691 | 0.2606 |
| $l_2g_1$ | 0.7127 | 0.5709 | 0.4327 | 0.2804 | 0.2697 |
| $l_2g_2$ | 0.7127 | 0.5709 | 0.4327 | 0.2804 | 0.2701 |
| $l_2g_3$ | **0.7236** | 0.5782 | 0.4418 | 0.2836 | 0.2769 |
| $l_2g_4$ | 0.7127 | 0.5691 | 0.4345 | 0.2756 | 0.2705 |
| $l_2g_5$ | 0.7055 | 0.5800 | **0.4464** | 0.2855 | 0.2803 |
| $l_3g_0$ | 0.6945 | 0.5455 | 0.4082 | 0.2604 | 0.2455 |
| $l_3g_1$ | 0.7018 | 0.5618 | 0.4182 | 0.2695 | 0.2549 |
| $l_3g_2$ | 0.7018 | 0.5618 | 0.4191 | 0.2698 | 0.2552 |
| $l_3g_3$ | 0.7055 | 0.5709 | 0.4264 | 0.2756 | 0.2615 |
| $l_3g_4$ | 0.7018 | 0.5636 | 0.4236 | 0.2691 | 0.2570 |
| $l_3g_5$ | **0.7055** | 0.5782 | 0.4300 | 0.2760 | 0.2665 |
| $l_4g_0$ | 0.6800 | 0.5382 | 0.4064 | 0.2585 | 0.2412 |
| $l_4g_1$ | 0.6873 | 0.5527 | 0.4136 | 0.2662 | 0.2497 |
| $l_4g_2$ | 0.6873 | 0.5545 | 0.4136 | 0.2669 | 0.2500 |
| $l_4g_3$ | 0.6982 | 0.5673 | 0.4236 | 0.2724 | 0.2563 |
| $l_4g_4$ | 0.6909 | 0.5564 | 0.4155 | 0.2665 | 0.2513 |
| $l_4g_5$ | 0.7018 | 0.5709 | 0.4273 | 0.2727 | 0.2615 |
| $l_5g_0$ | 0.7200 | 0.5600 | 0.4218 | 0.2764 | 0.2698 |
| $l_5g_1$ | 0.7200 | 0.5800 | 0.4400 | 0.2844 | 0.2792 |
| $l_5g_2$ | 0.7200 | 0.5818 | 0.4400 | 0.2855 | 0.2795 |
| $l_5g_3$ | **0.7236** | **0.5945** | **0.4518** | 0.2920 | 0.2853 |
| $l_5g_4$ | 0.7164 | 0.5764 | 0.4327 | 0.2855 | 0.2798 |
| $l_5g_5$ | **0.7273** | 0.5855 | **0.4464** | 0.2887 | 0.2874 |
| $l_6g_0$ | 0.7309 | 0.5891 | 0.4455 | 0.2865 | 0.2886 |
| $l_6g_1$ | **0.7418** | **0.5964** | **0.4527** | 0.2949 | 0.2981 |
| $l_6g_2$ | **0.7382** | **0.5964** | **0.4536** | 0.2956 | 0.2983 |
| $l_6g_3$ | **0.7382** | **0.6018** | **0.4609** | **0.3040** | **0.3030** |
| $l_6g_4$ | **0.7345** | 0.5891 | **0.4527** | **0.2949** | 0.2973 |
| $l_6g_5$ | **0.7236** | **0.6018** | **0.4609** | **0.2964** | **0.3003** |
| $l_7g_0$ | 0.7018 | 0.5545 | 0.4182 | 0.2640 | 0.2550 |
| $l_7g_1$ | 0.7127 | 0.5655 | 0.4282 | 0.2738 | 0.2636 |
| $l_7g_2$ | **0.7164** | 0.5655 | 0.4300 | 0.2745 | 0.2643 |
| $l_7g_3$ | **0.7236** | 0.5709 | 0.4345 | 0.2782 | 0.2707 |
| $l_7g_4$ | **0.7200** | 0.5673 | 0.4291 | 0.2735 | 0.2651 |
| $l_7g_5$ | 0.7055 | 0.5818 | 0.4382 | 0.2804 | 0.2743 |

TAB. A.7 – Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Oxford.

| | P@5 | P@10 | P@20 | P@50 | MAP |
|---|---|---|---|---|---|
| $l_1g_0$ | 0.6800 | **0.5655** | 0.4209 | 0.2782 | 0.2730 |
| $l_1g_1$ | **0.7055** | **0.5800** | 0.4482 | **0.2931** | 0.2911 |
| $l_1g_2$ | **0.7055** | **0.5800** | 0.4482 | **0.2938** | 0.2921 |
| $l_1g_3$ | **0.7018** | **0.5836** | **0.4600** | **0.2982** | **0.2997** |
| $l_1g_4$ | 0.6836 | **0.5782** | **0.4491** | 0.2905 | 0.2868 |
| $l_1g_5$ | 0.5855 | 0.5109 | 0.4027 | 0.2644 | 0.2539 |
| $l_2g_0$ | 0.6655 | 0.5418 | 0.4091 | 0.2622 | 0.2555 |
| $l_2g_1$ | 0.6909 | 0.5582 | 0.4300 | 0.2778 | 0.2760 |
| $l_2g_2$ | 0.6909 | 0.5582 | 0.4300 | 0.2782 | 0.2764 |
| $l_2g_3$ | **0.6982** | **0.5764** | 0.4464 | 0.2825 | 0.2888 |
| $l_2g_4$ | 0.6764 | **0.5709** | 0.4373 | 0.2822 | 0.2811 |
| $l_2g_5$ | 0.6109 | 0.4945 | 0.4018 | 0.2625 | 0.2562 |
| $l_3g_0$ | 0.6291 | 0.5091 | 0.3945 | 0.2531 | 0.2343 |
| $l_3g_1$ | 0.6509 | 0.5364 | 0.4127 | 0.2640 | 0.2553 |
| $l_3g_2$ | 0.6509 | 0.5382 | 0.4136 | 0.2640 | 0.2558 |
| $l_3g_3$ | 0.6691 | 0.5564 | 0.4300 | 0.2702 | 0.2706 |
| $l_3g_4$ | 0.6618 | 0.5455 | 0.4227 | 0.2716 | 0.2634 |
| $l_3g_5$ | 0.5855 | 0.4855 | 0.3927 | 0.2549 | 0.2477 |
| $l_4g_0$ | 0.6109 | 0.4891 | 0.3809 | 0.2455 | 0.2248 |
| $l_4g_1$ | 0.6400 | 0.5164 | 0.4045 | 0.2589 | 0.2451 |
| $l_4g_2$ | 0.6400 | 0.5164 | 0.4045 | 0.2596 | 0.2455 |
| $l_4g_3$ | 0.6618 | 0.5309 | 0.4236 | 0.2662 | 0.2614 |
| $l_4g_4$ | 0.6364 | 0.5364 | 0.4191 | 0.2669 | 0.2555 |
| $l_4g_5$ | 0.5818 | 0.4727 | 0.3845 | 0.2484 | 0.2405 |
| $l_5g_0$ | 0.6800 | **0.5655** | 0.4209 | 0.2782 | 0.2730 |
| $l_5g_1$ | **0.7055** | **0.5800** | 0.4482 | **0.2931** | 0.2911 |
| $l_5g_2$ | **0.7055** | **0.5800** | 0.4482 | **0.2938** | 0.2921 |
| $l_5g_3$ | **0.7018** | **0.5836** | **0.4600** | **0.2982** | **0.2997** |
| $l_5g_4$ | 0.6836 | **0.5782** | **0.4491** | 0.2905 | 0.2868 |
| $l_5g_5$ | 0.5855 | 0.5109 | 0.4027 | 0.2644 | 0.2539 |
| $l_6g_0$ | 0.6145 | 0.5091 | 0.3791 | 0.2480 | 0.2348 |
| $l_6g_1$ | 0.6291 | 0.5345 | 0.4073 | 0.2640 | 0.2551 |
| $l_6g_2$ | 0.6291 | 0.5382 | 0.4100 | 0.2647 | 0.2559 |
| $l_6g_3$ | 0.6291 | 0.5436 | 0.4282 | 0.2738 | 0.2636 |
| $l_6g_4$ | 0.6073 | 0.5200 | 0.3955 | 0.2531 | 0.2397 |
| $l_6g_5$ | 0.5782 | 0.4945 | 0.3745 | 0.2436 | 0.2261 |
| $l_7g_0$ | 0.6473 | 0.5291 | 0.4000 | 0.2593 | 0.2466 |
| $l_7g_1$ | 0.6618 | 0.5545 | 0.4236 | 0.2691 | 0.2664 |
| $l_7g_2$ | 0.6618 | 0.5545 | 0.4236 | 0.2687 | 0.2670 |
| $l_7g_3$ | 0.6836 | **0.5691** | 0.4455 | 0.2756 | 0.2818 |
| $l_7g_4$ | 0.6764 | 0.5600 | 0.4327 | 0.2796 | 0.2742 |
| $l_7g_5$ | 0.6073 | 0.4855 | 0.3982 | 0.2593 | 0.2521 |

TAB. A.8 – Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Oxford.

| | P@5 | P@10 | P@20 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.4020 | 0.3376 | 0.2845 | 0.2264 | 0.2832 |
| PLH1 | 0.6640 | 0.5812 | 0.5008 | 0.3852 | 0.3282 |
| PLH2 | 0.6081 | 0.5061 | 0.4350 | 0.3256 | 0.3136 |
| PBH0 | 0.4051 | 0.3396 | 0.2914 | 0.2351 | 0.2849 |
| PBH1 | 0.6792 | 0.5934 | 0.5155 | 0.3985 | 0.3291 |
| PBH2 | 0.6193 | 0.5183 | 0.4462 | 0.3360 | 0.3151 |
| DLH0 | 0.4020 | 0.3365 | 0.2848 | 0.2265 | 0.2832 |
| DLH1 | **0.8112** | **0.7538** | **0.6805** | **0.5409** | **0.3583** |
| DLH2 | 0.7249 | 0.6218 | 0.5576 | 0.4217 | 0.3353 |
| DBH0 | 0.4051 | 0.3396 | 0.2911 | 0.2354 | 0.2849 |
| DBH1 | **0.8010** | **0.7431** | **0.6744** | **0.5386** | **0.3547** |
| DBH2 | 0.7228 | 0.6289 | 0.5599 | 0.4301 | 0.3347 |
| GLH0 | 0.4122 | 0.3299 | 0.2843 | 0.2258 | 0.2836 |
| GLH1 | 0.5980 | 0.5056 | 0.4330 | 0.3175 | 0.3119 |
| GLH2 | 0.5553 | 0.4563 | 0.3924 | 0.2860 | 0.3032 |
| GBH0 | 0.4162 | 0.3401 | 0.2942 | 0.2320 | 0.2851 |
| GBH1 | 0.6000 | 0.5091 | 0.4393 | 0.3256 | 0.3128 |
| GBH2 | 0.5543 | 0.4629 | 0.4003 | 0.2938 | 0.3046 |
| BLH0 | 0.4122 | 0.3299 | 0.2843 | 0.2257 | 0.2836 |
| BLH1 | 0.5980 | 0.5056 | 0.4330 | 0.3177 | 0.3119 |
| BLH2 | 0.5553 | 0.4569 | 0.3924 | 0.2860 | 0.3032 |
| BBH0 | 0.4162 | 0.3401 | 0.2942 | 0.2320 | 0.2851 |
| BBH1 | 0.6010 | 0.5091 | 0.4393 | 0.3257 | 0.3128 |
| BBH2 | 0.5543 | 0.4629 | 0.4005 | 0.2938 | 0.3047 |
| InLH0 | 0.4112 | 0.3365 | 0.2878 | 0.2280 | 0.2844 |
| InLH1 | 0.6782 | 0.5782 | 0.5048 | 0.3819 | 0.3268 |
| InLH2 | 0.5929 | 0.4970 | 0.4325 | 0.3182 | 0.3119 |
| InBH0 | 0.4162 | 0.3416 | 0.2947 | 0.2352 | 0.2860 |
| InBH1 | 0.6853 | 0.5827 | 0.5183 | 0.3916 | 0.3270 |
| InBH2 | 0.5959 | 0.5010 | 0.4378 | 0.3288 | 0.3129 |
| IneLH0 | 0.4122 | 0.3305 | 0.2845 | 0.2265 | 0.2838 |
| IneLH1 | 0.6772 | 0.5782 | 0.5008 | 0.3784 | 0.3265 |
| IneLH2 | 0.5888 | 0.4964 | 0.4302 | 0.3148 | 0.3113 |
| IneBH0 | 0.4162 | 0.3406 | 0.2954 | 0.2325 | 0.2855 |
| IneBH1 | 0.6772 | 0.5848 | 0.5109 | 0.3881 | 0.3279 |
| IneBH2 | 0.5929 | 0.4985 | 0.4388 | 0.3251 | 0.3133 |
| HGLH0 | 0.6162 | 0.5117 | 0.4353 | 0.3196 | 0.3130 |
| HGLH1 | 0.5025 | 0.4234 | 0.3977 | 0.3932 | 0.3097 |
| HGLH2 | 0.4934 | 0.4127 | 0.3863 | 0.3769 | 0.3256 |
| HGBH0 | 0.6294 | 0.5320 | 0.4490 | 0.3366 | 0.3155 |
| HGBH1 | 0.5005 | 0.4254 | 0.4038 | 0.3928 | 0.3065 |
| HGBH2 | 0.5015 | 0.4173 | 0.3893 | 0.3808 | 0.3225 |

TAB. A.9 – Performance des mesures de similarité DFR sur le corpus Caltech-6.

| | P@4 | MAP |
|---|---|---|
| PLH0 | 0.3409 | 0.2819 |
| PLH1 | 0.6675 | 0.4981 |
| PLH2 | 0.6094 | 0.4642 |
| PBH0 | 0.3434 | 0.2808 |
| PBH1 | 0.6338 | 0.4860 |
| PBH2 | 0.5892 | 0.4542 |
| DLH0 | 0.3392 | 0.2815 |
| DLH1 | **0.6953** | **0.5237** |
| DLH2 | 0.6465 | 0.4905 |
| DBH0 | 0.3426 | 0.2799 |
| DBH1 | 0.6625 | 0.5084 |
| DBH2 | 0.6212 | 0.4788 |
| GLH0 | 0.3577 | 0.2906 |
| GLH1 | 0.6431 | 0.4823 |
| GLH2 | 0.5951 | 0.4482 |
| GBH0 | 0.3653 | 0.2962 |
| GBH1 | 0.6263 | 0.4751 |
| GBH2 | 0.5816 | 0.4438 |
| BLH0 | 0.3577 | 0.2905 |
| BLH1 | 0.6439 | 0.4830 |
| BLH2 | 0.5960 | 0.4483 |
| BBH0 | 0.3653 | 0.2961 |
| BBH1 | 0.6263 | 0.4753 |
| BBH2 | 0.5816 | 0.4439 |
| InLH0 | 0.3603 | 0.2921 |
| InLH1 | 0.6582 | 0.4900 |
| InLH2 | 0.6052 | 0.4549 |
| InBH0 | 0.3653 | 0.2964 |
| InBH1 | 0.6397 | 0.4825 |
| InBH2 | 0.5825 | 0.4476 |
| IneLH0 | 0.3586 | 0.2907 |
| IneLH1 | 0.6591 | 0.4918 |
| IneLH2 | 0.6069 | 0.4566 |
| IneBH0 | 0.3662 | 0.2960 |
| IneBH1 | 0.6507 | 0.4872 |
| IneBH2 | 0.5985 | 0.4525 |
| HGLH0 | 0.5859 | 0.4449 |
| HGLH1 | 0.6254 | 0.4825 |
| HGLH2 | 0.6498 | 0.4989 |
| HGBH0 | 0.5707 | 0.4391 |
| HGBH1 | 0.5707 | 0.4484 |
| HGBH2 | 0.6237 | 0.4799 |

TAB. A.10 – Performance des mesures de similarité DFR sur le corpus Kentucky.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.1631 | 0.1381 | 0.1252 | 0.1126 | 0.0853 |
| PLH1 | 0.2586 | 0.2242 | 0.1800 | 0.1489 | 0.0953 |
| PLH2 | 0.2677 | 0.2253 | 0.1804 | 0.1499 | 0.0967 |
| PBH0 | 0.1687 | 0.1394 | 0.1283 | 0.1171 | 0.0869 |
| PBH1 | 0.2551 | 0.2250 | 0.1819 | 0.1514 | 0.0965 |
| PBH2 | 0.2662 | 0.2263 | 0.1813 | 0.1519 | 0.0978 |
| DLH0 | 0.1636 | 0.1376 | 0.1253 | 0.1123 | 0.0853 |
| DLH1 | 0.2884 | **0.2389** | 0.1826 | **0.1498** | 0.0999 |
| DLH2 | **0.3005** | **0.2399** | **0.1834** | **0.1510** | **0.1011** |
| DBH0 | 0.1682 | 0.1396 | 0.1285 | 0.1170 | 0.0869 |
| DBH1 | 0.2894 | **0.2381** | **0.1852** | **0.1519** | 0.1012 |
| DBH2 | **0.2995** | **0.2391** | **0.1868** | **0.1531** | **0.1022** |
| GLH0 | 0.1662 | 0.1399 | 0.1254 | 0.1117 | 0.0852 |
| GLH1 | **0.3015** | **0.2412** | **0.1839** | **0.1513** | **0.1012** |
| GLH2 | **0.3045** | **0.2379** | **0.1829** | **0.1516** | **0.1008** |
| GBH0 | 0.1682 | 0.1399 | 0.1290 | 0.1164 | 0.0869 |
| GBH1 | **0.2995** | **0.2437** | **0.1872** | **0.1542** | **0.1023** |
| GBH2 | **0.3015** | **0.2386** | **0.1847** | **0.1528** | **0.1019** |
| BLH0 | 0.1662 | 0.1399 | 0.1254 | 0.1117 | 0.0852 |
| BLH1 | **0.3005** | **0.2409** | **0.1841** | **0.1513** | **0.1012** |
| BLH2 | **0.3051** | **0.2374** | **0.1828** | **0.1516** | **0.1008** |
| BBH0 | 0.1682 | 0.1399 | 0.1291 | 0.1163 | 0.0869 |
| BBH1 | **0.3010** | **0.2434** | **0.1869** | **0.1543** | **0.1023** |
| BBH2 | **0.3020** | **0.2386** | **0.1846** | **0.1528** | **0.1019** |
| InLH0 | 0.1677 | 0.1391 | 0.1269 | 0.1129 | 0.0857 |
| InLH1 | **0.3030** | **0.2427** | 0.1848 | **0.1525** | **0.1016** |
| InLH2 | **0.3035** | **0.2384** | **0.1838** | **0.1525** | **0.1012** |
| InBH0 | 0.1682 | 0.1417 | 0.1296 | 0.1170 | 0.0873 |
| InBH1 | **0.2985** | 0.2417 | **0.1873** | **0.1544** | **0.1024** |
| InBH2 | **0.2990** | 0.2374 | 0.1853 | 0.1529 | 0.1020 |
| IneLH0 | 0.1662 | 0.1402 | 0.1253 | 0.1118 | 0.0852 |
| IneLH1 | **0.3015** | **0.2412** | **0.1838** | **0.1513** | **0.1012** |
| IneLH2 | **0.3051** | **0.2376** | **0.1828** | **0.1513** | **0.1007** |
| IneBH0 | 0.1682 | 0.1402 | 0.1290 | 0.1163 | 0.0869 |
| IneBH1 | **0.2995** | **0.2432** | **0.1869** | **0.1541** | **0.1023** |
| IneBH2 | **0.3020** | **0.2389** | **0.1843** | **0.1527** | **0.1019** |
| HGLH0 | 0.2414 | 0.2038 | 0.1600 | 0.1349 | 0.0936 |
| HGLH1 | 0.2581 | 0.2172 | 0.1705 | 0.1405 | 0.0954 |
| HGLH2 | 0.2737 | 0.2313 | 0.1800 | 0.1488 | 0.0980 |
| HGBH0 | 0.2404 | 0.2053 | 0.1621 | 0.1388 | 0.0949 |
| HGBH1 | 0.2576 | 0.2167 | 0.1729 | 0.1442 | 0.0968 |
| HGBH2 | 0.2763 | 0.2301 | 0.1828 | 0.1515 | 0.0992 |

TAB. A.11 – Performance des mesures de similarité DFR sur le corpus Caltech-101.

| | P@5 | P@10 | P@20 | P@50 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.0633 | 0.0889 | 0.1163 | 0.1720 | 0.1450 |
| PLH1 | 0.1292 | 0.1801 | 0.2368 | 0.3073 | 0.2992 |
| PLH2 | 0.1231 | 0.1689 | 0.2112 | 0.2825 | 0.2718 |
| PBH0 | 0.0658 | 0.0905 | 0.1172 | 0.1749 | 0.1501 |
| PBH1 | 0.1282 | 0.1747 | 0.2359 | 0.3066 | 0.2966 |
| PBH2 | 0.1205 | 0.1659 | 0.2136 | 0.2785 | 0.2726 |
| DLH0 | 0.0633 | 0.0889 | 0.1157 | 0.1720 | 0.1451 |
| DLH1 | **0.1343** | **0.1905** | **0.2511** | **0.3178** | **0.3165** |
| DLH2 | 0.1286 | 0.1787 | 0.2243 | 0.3028 | 0.2936 |
| DBH0 | 0.0658 | 0.0908 | 0.1163 | 0.1746 | 0.1507 |
| DBH1 | **0.1299** | **0.1886** | 0.2437 | 0.3139 | 0.3075 |
| DBH2 | 0.1279 | 0.1739 | 0.2242 | 0.2983 | 0.2892 |
| GLH0 | 0.0655 | 0.0872 | 0.1127 | 0.1712 | 0.1446 |
| GLH1 | 0.1303 | 0.1774 | 0.2223 | 0.2935 | 0.2896 |
| GLH2 | 0.1201 | 0.1587 | 0.2034 | 0.2623 | 0.2567 |
| GBH0 | 0.0658 | 0.0906 | 0.1188 | 0.1759 | 0.1499 |
| GBH1 | 0.1296 | 0.1728 | 0.2204 | 0.2952 | 0.2894 |
| GBH2 | 0.1189 | 0.1633 | 0.2030 | 0.2670 | 0.2603 |
| BLH0 | 0.0655 | 0.0872 | 0.1127 | 0.1712 | 0.1446 |
| BLH1 | 0.1303 | 0.1774 | 0.2223 | 0.2935 | 0.2898 |
| BLH2 | 0.1201 | 0.1587 | 0.2040 | 0.2632 | 0.2570 |
| BBH0 | 0.0658 | 0.0908 | 0.1188 | 0.1759 | 0.1499 |
| BBH1 | 0.1296 | 0.1744 | 0.2221 | 0.2952 | 0.2897 |
| BBH2 | 0.1189 | 0.1633 | 0.2030 | 0.2670 | 0.2605 |
| InLH0 | 0.0655 | 0.0868 | 0.1159 | 0.1721 | 0.1460 |
| InLH1 | **0.1326** | 0.1790 | 0.2231 | 0.3001 | 0.2931 |
| InLH2 | 0.1214 | 0.1638 | 0.2070 | 0.2689 | 0.2618 |
| InBH0 | 0.0658 | 0.0909 | 0.1182 | 0.1763 | 0.1512 |
| InBH1 | 0.1310 | 0.1742 | 0.2252 | 0.2952 | 0.2910 |
| InBH2 | 0.1203 | 0.1655 | 0.2079 | 0.2692 | 0.2648 |
| IneLH0 | 0.0655 | 0.0873 | 0.1127 | 0.1712 | 0.1450 |
| IneLH1 | **0.1326** | 0.1793 | 0.2249 | 0.2987 | 0.2940 |
| IneLH2 | 0.1220 | 0.1626 | 0.2050 | 0.2673 | 0.2615 |
| IneBH0 | 0.0658 | 0.0908 | 0.1185 | 0.1767 | 0.1503 |
| IneBH1 | 0.1310 | 0.1758 | 0.2247 | 0.3011 | 0.2930 |
| IneBH2 | 0.1217 | 0.1638 | 0.2056 | 0.2702 | 0.2650 |
| HGLH0 | 0.1003 | 0.1391 | 0.1857 | 0.2402 | 0.2249 |
| HGLH1 | **0.1369** | **0.1916** | **0.2498** | **0.3110** | **0.3178** |
| HGLH2 | **0.1422** | **0.1955** | **0.2490** | **0.3165** | **0.3241** |
| HGBH0 | 0.1026 | 0.1412 | 0.1856 | 0.2442 | 0.2311 |
| HGBH1 | 0.1295 | 0.1856 | **0.2372** | **0.3069** | 0.3060 |
| HGBH2 | 0.1348 | **0.1946** | **0.2437** | **0.3154** | 0.3146 |

TAB. A.12 – Performance des mesures de similarité DFR sur le corpus Oxford.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | **0.7888** | 0.7338 | 0.6583 | 0.5952 | 0.3939 |
| PLH1 | **0.7853** | 0.7307 | 0.6546 | 0.5919 | 0.3934 |
| PLH2 | **0.7883** | 0.7325 | 0.6557 | 0.5925 | 0.3937 |
| PBH0 | 0.7883 | **0.7449** | 0.6612 | 0.6008 | 0.3974 |
| PBH1 | 0.7843 | 0.7360 | **0.6622** | 0.5983 | **0.3968** |
| PBH2 | 0.7878 | 0.7391 | **0.6622** | 0.5988 | 0.3971 |
| DLH0 | **0.7878** | 0.7338 | 0.6569 | 0.5944 | 0.3936 |
| DLH1 | **0.7939** | **0.7424** | **0.6668** | **0.6072** | 0.3991 |
| DLH2 | **0.7954** | 0.7345 | 0.6638 | 0.6013 | 0.3968 |
| DBH0 | **0.7893** | **0.7419** | 0.6608 | 0.6003 | 0.3971 |
| DBH1 | **0.7975** | **0.7470** | **0.6718** | **0.6116** | **0.4020** |
| DBH2 | **0.7985** | **0.7467** | **0.6680** | 0.6076 | 0.4001 |
| GLH0 | **0.7893** | **0.7442** | **0.6643** | 0.5934 | 0.3937 |
| GLH1 | 0.7822 | 0.7371 | 0.6572 | 0.5909 | 0.3911 |
| GLH2 | 0.7843 | 0.7368 | 0.6592 | 0.5920 | 0.3920 |
| GBH0 | **0.7909** | **0.7449** | **0.6689** | 0.6018 | **0.3974** |
| GBH1 | **0.7888** | 0.7371 | **0.6622** | 0.5975 | 0.3947 |
| GBH2 | **0.7919** | 0.7406 | **0.6632** | 0.5984 | 0.3956 |
| BLH0 | **0.7893** | **0.7442** | **0.6641** | 0.5933 | 0.3937 |
| BLH1 | 0.7817 | 0.7365 | 0.6570 | 0.5908 | 0.3911 |
| BLH2 | 0.7843 | 0.7371 | 0.6591 | 0.5920 | 0.3920 |
| BBH0 | **0.7909** | **0.7452** | **0.6688** | 0.6017 | **0.3974** |
| BBH1 | **0.7888** | 0.7365 | **0.6620** | 0.5975 | 0.3947 |
| BBH2 | **0.7924** | 0.7409 | **0.6632** | 0.5984 | 0.3956 |
| InLH0 | **0.7898** | **0.7442** | **0.6646** | 0.5993 | 0.3960 |
| InLH1 | **0.7898** | 0.7409 | **0.6641** | 0.5986 | 0.3969 |
| InLH2 | **0.7893** | **0.7424** | **0.6640** | 0.5994 | 0.3968 |
| InBH0 | **0.7924** | **0.7475** | **0.6710** | **0.6051** | 0.3997 |
| InBH1 | **0.7959** | **0.7447** | **0.6696** | **0.6070** | **0.4006** |
| InBH2 | **0.7944** | **0.7452** | **0.6712** | **0.6073** | **0.4005** |
| IneLH0 | **0.7919** | 0.7411 | **0.6632** | 0.5942 | 0.3943 |
| IneLH1 | 0.7873 | 0.7376 | **0.6619** | 0.5965 | 0.3952 |
| IneLH2 | **0.7873** | **0.7401** | **0.6639** | 0.5962 | 0.3951 |
| IneBH0 | **0.7924** | **0.7472** | **0.6690** | **0.6050** | **0.3986** |
| IneBH1 | **0.7924** | **0.7434** | **0.6684** | 0.6041 | **0.3995** |
| IneBH2 | **0.7970** | **0.7457** | **0.6691** | **0.6054** | **0.3994** |
| HGLH0 | **0.7873** | **0.7396** | **0.6611** | 0.5976 | **0.3972** |
| HGLH1 | 0.7827 | 0.7274 | 0.6533 | 0.5842 | 0.3871 |
| HGLH2 | 0.7812 | 0.7338 | 0.6537 | 0.5878 | 0.3896 |
| HGBH0 | **0.7878** | **0.7452** | **0.6659** | **0.6032** | 0.4006 |
| HGBH1 | 0.7812 | 0.7340 | 0.6571 | 0.5922 | 0.3900 |
| HGBH2 | 0.7817 | 0.7371 | **0.6605** | 0.5946 | 0.3927 |

TAB. A.13 – Performance des pondérations DFR avec la distance $L_1$ sur le corpus Caltech-6.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.5751 | 0.4881 | 0.3944 | 0.3494 | 0.3216 |
| PLH1 | 0.5604 | 0.4711 | 0.3750 | 0.3329 | 0.3191 |
| PLH2 | 0.5650 | 0.4827 | 0.3859 | 0.3421 | 0.3209 |
| PBH0 | 0.6051 | 0.5218 | 0.4275 | 0.3809 | 0.3302 |
| PBH1 | 0.5878 | 0.4997 | 0.4085 | 0.3629 | 0.3270 |
| PBH2 | 0.5970 | 0.5089 | 0.4169 | 0.3737 | 0.3290 |
| DLH0 | 0.5777 | 0.4924 | 0.3976 | 0.3518 | 0.3223 |
| DLH1 | 0.6340 | 0.5515 | 0.4656 | 0.4138 | 0.3393 |
| DLH2 | 0.6132 | 0.5287 | 0.4411 | 0.3922 | 0.3329 |
| DBH0 | 0.6081 | 0.5231 | 0.4305 | 0.3845 | 0.3309 |
| DBH1 | **0.6513** | **0.5711** | **0.4840** | **0.4338** | **0.3441** |
| DBH2 | 0.6340 | 0.5536 | 0.4660 | 0.4159 | 0.3395 |
| GLH0 | 0.5584 | 0.4726 | 0.3749 | 0.3317 | 0.3164 |
| GLH1 | 0.5421 | 0.4594 | 0.3620 | 0.3207 | 0.3137 |
| GLH2 | 0.5462 | 0.4622 | 0.3656 | 0.3238 | 0.3146 |
| GBH0 | 0.5863 | 0.5056 | 0.4083 | 0.3625 | 0.3253 |
| GBH1 | 0.5604 | 0.4835 | 0.3901 | 0.3431 | 0.3213 |
| GBH2 | 0.5690 | 0.4883 | 0.3958 | 0.3475 | 0.3225 |
| BLH0 | 0.5579 | 0.4726 | 0.3751 | 0.3318 | 0.3164 |
| BLH1 | 0.5421 | 0.4594 | 0.3618 | 0.3206 | 0.3137 |
| BLH2 | 0.5462 | 0.4622 | 0.3656 | 0.3239 | 0.3146 |
| BBH0 | 0.5863 | 0.5058 | 0.4084 | 0.3626 | 0.3253 |
| BBH1 | 0.5599 | 0.4835 | 0.3901 | 0.3431 | 0.3213 |
| BBH2 | 0.5690 | 0.4883 | 0.3958 | 0.3477 | 0.3225 |
| InLH0 | 0.5680 | 0.4810 | 0.3891 | 0.3436 | 0.3207 |
| InLH1 | 0.5777 | 0.4898 | 0.4012 | 0.3540 | 0.3242 |
| InLH2 | 0.5761 | 0.4886 | 0.3986 | 0.3525 | 0.3236 |
| InBH0 | 0.6036 | 0.5269 | 0.4296 | 0.3790 | 0.3302 |
| InBH1 | 0.6066 | 0.5234 | 0.4313 | 0.3848 | 0.3324 |
| InBH2 | 0.6051 | 0.5239 | 0.4308 | 0.3845 | 0.3321 |
| IneLH0 | 0.5624 | 0.4706 | 0.3765 | 0.3336 | 0.3175 |
| IneLH1 | 0.5645 | 0.4820 | 0.3894 | 0.3452 | 0.3211 |
| IneLH2 | 0.5680 | 0.4817 | 0.3873 | 0.3433 | 0.3204 |
| IneBH0 | 0.5827 | 0.5020 | 0.4085 | 0.3625 | 0.3269 |
| IneBH1 | 0.5934 | 0.5099 | 0.4183 | 0.3708 | 0.3304 |
| IneBH2 | 0.5949 | 0.5107 | 0.4177 | 0.3682 | 0.3299 |
| HGLH0 | 0.5888 | 0.5033 | 0.4080 | 0.3634 | 0.3260 |
| HGLH1 | 0.5127 | 0.4312 | 0.3408 | 0.2983 | 0.3101 |
| HGLH2 | 0.5239 | 0.4401 | 0.3468 | 0.3033 | 0.3125 |
| HGBH0 | **0.6310** | 0.5371 | 0.4454 | 0.3958 | 0.3349 |
| HGBH1 | 0.5411 | 0.4591 | 0.3668 | 0.3239 | 0.3167 |
| HGBH2 | 0.5452 | 0.4693 | 0.3759 | 0.3308 | 0.3195 |

TAB. A.14 – Performance des pondérations DFR avec la distance $L_2$ sur le corpus Caltech-6.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | **0.3106** | **0.2548** | **0.1990** | 0.1595 | 0.1057 |
| PLH1 | **0.3096** | **0.2540** | **0.1984** | **0.1600** | 0.1056 |
| PLH2 | **0.3106** | **0.2543** | **0.1984** | **0.1599** | 0.1056 |
| PBH0 | **0.3106** | **0.2551** | **0.1998** | **0.1635** | **0.1074** |
| PBH1 | **0.3121** | **0.2556** | 0.1993 | **0.1631** | **0.1074** |
| PBH2 | **0.3111** | **0.2553** | **0.1997** | **0.1632** | **0.1073** |
| DLH0 | **0.3111** | **0.2545** | **0.1991** | 0.1594 | 0.1056 |
| DLH1 | 0.3086 | **0.2528** | **0.1970** | 0.1586 | 0.1053 |
| DLH2 | **0.3106** | **0.2533** | **0.1975** | **0.1594** | 0.1054 |
| DBH0 | **0.3111** | **0.2563** | **0.2000** | **0.1632** | **0.1074** |
| DBH1 | **0.3111** | **0.2540** | **0.1979** | **0.1624** | **0.1069** |
| DBH2 | **0.3106** | **0.2553** | **0.1987** | **0.1631** | **0.1071** |
| GLH0 | **0.3101** | **0.2545** | **0.1985** | 0.1594 | 0.1057 |
| GLH1 | 0.3096 | 0.2538 | **0.1981** | **0.1596** | 0.1056 |
| GLH2 | 0.3096 | 0.2540 | **0.1982** | 0.1595 | 0.1056 |
| GBH0 | **0.3116** | **0.2556** | 0.1995 | 0.1633 | **0.1076** |
| GBH1 | **0.3106** | 0.2551 | **0.1993** | 0.1631 | 0.1074 |
| GBH2 | **0.3111** | 0.2556 | **0.1992** | 0.1632 | **0.1074** |
| BLH0 | **0.3096** | **0.2548** | **0.1985** | 0.1595 | 0.1057 |
| BLH1 | 0.3101 | 0.2538 | **0.1980** | **0.1596** | 0.1056 |
| BLH2 | 0.3101 | 0.2540 | 0.1979 | 0.1594 | 0.1056 |
| BBH0 | **0.3121** | **0.2558** | **0.1993** | **0.1634** | **0.1076** |
| BBH1 | **0.3116** | 0.2553 | 0.1988 | 0.1630 | 0.1073 |
| BBH2 | **0.3116** | 0.2556 | 0.1987 | **0.1633** | 0.1074 |
| InLH0 | **0.3126** | **0.2548** | **0.1988** | **0.1607** | 0.1063 |
| InLH1 | **0.3131** | **0.2540** | **0.1987** | **0.1602** | 0.1062 |
| InLH2 | **0.3126** | **0.2548** | **0.1990** | **0.1603** | 0.1062 |
| InBH0 | **0.3111** | **0.2566** | **0.1997** | **0.1641** | **0.1078** |
| InBH1 | **0.3091** | **0.2571** | 0.1992 | **0.1639** | 0.1076 |
| InBH2 | **0.3086** | **0.2571** | 0.1989 | **0.1638** | 0.1076 |
| IneLH0 | **0.3101** | **0.2543** | **0.1983** | 0.1593 | 0.1057 |
| IneLH1 | 0.3096 | **0.2540** | **0.1978** | **0.1595** | 0.1055 |
| IneLH2 | 0.3096 | 0.2538 | **0.1981** | **0.1595** | 0.1056 |
| IneBH0 | **0.3111** | **0.2561** | **0.1995** | **0.1631** | **0.1075** |
| IneBH1 | **0.3111** | 0.2553 | **0.1993** | 0.1628 | 0.1073 |
| IneBH2 | **0.3111** | 0.2553 | **0.1991** | 0.1628 | **0.1074** |
| HGLH0 | **0.3106** | **0.2543** | **0.1992** | **0.1599** | 0.1058 |
| HGLH1 | 0.3086 | **0.2553** | **0.1987** | **0.1601** | 0.1057 |
| HGLH2 | 0.3091 | **0.2543** | **0.1989** | 0.1598 | 0.1058 |
| HGBH0 | **0.3121** | **0.2551** | **0.2003** | **0.1635** | **0.1075** |
| HGBH1 | **0.3111** | **0.2571** | **0.1998** | **0.1634** | **0.1075** |
| HGBH2 | **0.3111** | **0.2566** | **0.2001** | **0.1633** | **0.1075** |

TAB. A.15 – Performance des pondérations DFR avec la distance $L_1$ sur le corpus Caltech-101.

| | P@10 | P@20 | P@50 | P@100 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.2828 | **0.2258** | **0.1699** | **0.1396** | **0.0957** |
| PLH1 | 0.2848 | **0.2258** | **0.1695** | **0.1390** | **0.0956** |
| PLH2 | **0.2874** | **0.2268** | **0.1698** | **0.1390** | **0.0957** |
| PBH0 | 0.2783 | **0.2255** | **0.1710** | **0.1439** | 0.0979 |
| PBH1 | 0.2773 | **0.2268** | 0.1708 | **0.1435** | 0.0977 |
| PBH2 | 0.2768 | **0.2265** | **0.1713** | **0.1439** | 0.0977 |
| DLH0 | 0.2838 | **0.2247** | **0.1692** | **0.1386** | **0.0956** |
| DLH1 | **0.2904** | **0.2258** | **0.1685** | **0.1384** | **0.0958** |
| DLH2 | **0.2894** | **0.2270** | **0.1698** | **0.1394** | **0.0959** |
| DBH0 | 0.2778 | **0.2242** | 0.1706 | 0.1428 | 0.0977 |
| DBH1 | 0.2788 | **0.2268** | **0.1707** | **0.1440** | 0.0978 |
| DBH2 | 0.2778 | **0.2273** | **0.1722** | **0.1440** | **0.0978** |
| GLH0 | 0.2808 | 0.2242 | **0.1690** | **0.1387** | **0.0953** |
| GLH1 | **0.2869** | 0.2237 | **0.1677** | **0.1386** | **0.0951** |
| GLH2 | **0.2864** | **0.2245** | **0.1681** | **0.1386** | **0.0952** |
| GBH0 | 0.2798 | **0.2245** | **0.1709** | **0.1435** | 0.0978 |
| GBH1 | 0.2763 | **0.2237** | **0.1725** | **0.1433** | 0.0975 |
| GBH2 | 0.2788 | **0.2245** | **0.1720** | **0.1433** | **0.0977** |
| BLH0 | 0.2813 | 0.2242 | **0.1685** | **0.1384** | **0.0953** |
| BLH1 | **0.2869** | 0.2240 | 0.1673 | **0.1380** | **0.0950** |
| BLH2 | **0.2869** | **0.2240** | **0.1679** | **0.1383** | **0.0951** |
| BBH0 | 0.2783 | **0.2250** | 0.1710 | **0.1433** | 0.0977 |
| BBH1 | 0.2783 | **0.2235** | **0.1720** | **0.1428** | 0.0975 |
| BBH2 | 0.2793 | **0.2247** | **0.1717** | **0.1431** | 0.0977 |
| InLH0 | 0.2828 | **0.2280** | **0.1720** | **0.1413** | **0.0966** |
| InLH1 | **0.2894** | **0.2270** | **0.1707** | **0.1406** | **0.0963** |
| InLH2 | **0.2879** | **0.2273** | **0.1708** | **0.1404** | **0.0965** |
| InBH0 | 0.2722 | **0.2202** | 0.1693 | **0.1434** | 0.0976 |
| InBH1 | 0.2702 | **0.2205** | 0.1702 | **0.1433** | 0.0974 |
| InBH2 | 0.2712 | **0.2205** | 0.1702 | **0.1436** | 0.0975 |
| IneLH0 | 0.2803 | 0.2242 | **0.1690** | **0.1383** | **0.0952** |
| IneLH1 | **0.2869** | **0.2245** | **0.1680** | **0.1382** | **0.0950** |
| IneLH2 | **0.2864** | **0.2240** | **0.1679** | **0.1382** | **0.0951** |
| IneBH0 | 0.2793 | **0.2250** | **0.1710** | **0.1432** | 0.0977 |
| IneBH1 | 0.2763 | **0.2232** | **0.1725** | **0.1429** | **0.0975** |
| IneBH2 | 0.2773 | **0.2250** | **0.1724** | **0.1431** | **0.0976** |
| HGLH0 | **0.2869** | **0.2280** | **0.1711** | **0.1405** | **0.0961** |
| HGLH1 | 0.2758 | 0.2217 | **0.1674** | **0.1387** | **0.0951** |
| HGLH2 | 0.2773 | 0.2237 | **0.1679** | **0.1390** | **0.0953** |
| HGBH0 | 0.2798 | **0.2270** | **0.1720** | **0.1443** | **0.0983** |
| HGBH1 | 0.2692 | **0.2235** | 0.1693 | **0.1429** | 0.0973 |
| HGBH2 | 0.2682 | **0.2235** | 0.1699 | 0.1428 | 0.0974 |

TAB. A.16 – Performance des pondérations DFR avec la distance $L_2$ sur le corpus Caltech-101.

| | P@4 | MAP |
|---|---|---|
| PLH0 | **0.7197** | **0.5356** |
| PLH1 | **0.7205** | **0.5349** |
| PLH2 | **0.7197** | **0.5354** |
| PBH0 | 0.7146 | 0.5316 |
| PBH1 | 0.7113 | 0.5292 |
| PBH2 | 0.7130 | 0.5306 |
| DLH0 | **0.7197** | **0.5357** |
| DLH1 | **0.7214** | **0.5371** |
| DLH2 | **0.7197** | **0.5362** |
| DBH0 | **0.7146** | 0.5319 |
| DBH1 | 0.7130 | 0.5318 |
| DBH2 | **0.7163** | 0.5321 |
| GLH0 | 0.7113 | **0.5344** |
| GLH1 | 0.7113 | 0.5333 |
| GLH2 | 0.7104 | 0.5336 |
| GBH0 | 0.7037 | 0.5293 |
| GBH1 | 0.7045 | 0.5288 |
| GBH2 | 0.7029 | 0.5283 |
| BLH0 | 0.7121 | **0.5345** |
| BLH1 | 0.7113 | 0.5333 |
| BLH2 | 0.7104 | 0.5336 |
| BBH0 | 0.7037 | 0.5293 |
| BBH1 | 0.7045 | 0.5289 |
| BBH2 | 0.7029 | 0.5284 |
| InLH0 | 0.7130 | 0.5341 |
| InLH1 | 0.7096 | 0.5329 |
| InLH2 | 0.7113 | 0.5332 |
| InBH0 | 0.7003 | 0.5282 |
| InBH1 | 0.7012 | 0.5275 |
| InBH2 | 0.6995 | 0.5278 |
| IneLH0 | 0.7130 | **0.5355** |
| IneLH1 | 0.7121 | **0.5348** |
| IneLH2 | 0.7138 | **0.5351** |
| IneBH0 | 0.7113 | 0.5315 |
| IneBH1 | 0.7113 | 0.5309 |
| IneBH2 | 0.7121 | 0.5306 |
| HGLH0 | **0.7180** | **0.5347** |
| HGLH1 | 0.7138 | 0.5325 |
| HGLH2 | **0.7180** | 0.5336 |
| HGBH0 | 0.7121 | 0.5299 |
| HGBH1 | 0.7029 | 0.5271 |
| HGBH2 | 0.7054 | 0.5277 |

TAB. A.17 – Performance des pondérations DFR avec la distance $L_1$ sur le corpus Kentucky.

| | P@4 | MAP |
|---|---|---|
| PLH0 | 0.6532 | 0.4889 |
| PLH1 | 0.6473 | 0.4846 |
| PLH2 | 0.6498 | 0.4877 |
| PBH0 | 0.6187 | 0.4746 |
| PBH1 | 0.6170 | 0.4708 |
| PBH2 | 0.6195 | 0.4733 |
| DLH0 | 0.6524 | 0.4894 |
| DLH1 | **0.6751** | **0.5061** |
| DLH2 | **0.6726** | 0.5031 |
| DBH0 | 0.6187 | 0.4751 |
| DBH1 | 0.6431 | 0.4895 |
| DBH2 | 0.6347 | 0.4840 |
| GLH0 | 0.6515 | 0.4879 |
| GLH1 | 0.6465 | 0.4834 |
| GLH2 | 0.6465 | 0.4845 |
| GBH0 | 0.6279 | 0.4787 |
| GBH1 | 0.6263 | 0.4759 |
| GBH2 | 0.6279 | 0.4768 |
| BLH0 | 0.6515 | 0.4879 |
| BLH1 | 0.6465 | 0.4834 |
| BLH2 | 0.6465 | 0.4846 |
| BBH0 | 0.6271 | 0.4787 |
| BBH1 | 0.6263 | 0.4761 |
| BBH2 | 0.6279 | 0.4768 |
| InLH0 | 0.6540 | 0.4894 |
| InLH1 | 0.6490 | 0.4886 |
| InLH2 | 0.6481 | 0.4886 |
| InBH0 | 0.6044 | 0.4642 |
| InBH1 | 0.6103 | 0.4659 |
| InBH2 | 0.6120 | 0.4667 |
| IneLH0 | 0.6524 | 0.4881 |
| IneLH1 | 0.6490 | 0.4873 |
| IneLH2 | 0.6507 | 0.4877 |
| IneBH0 | 0.6347 | 0.4808 |
| IneBH1 | 0.6347 | 0.4817 |
| IneBH2 | 0.6330 | 0.4816 |
| HGLH0 | 0.6574 | 0.4930 |
| HGLH1 | 0.6355 | 0.4768 |
| HGLH2 | 0.6397 | 0.4798 |
| HGBH0 | 0.6237 | 0.4770 |
| HGBH1 | 0.6086 | 0.4640 |
| HGBH2 | 0.6136 | 0.4657 |

TAB. A.18 – Performance des pondérations DFR avec la distance $L_2$ sur le corpus Kentucky.

| | P@5 | P@10 | P@20 | P@50 | MAP |
|---|---|---|---|---|---|
| PLH0 | **0.7091** | **0.5655** | **0.4309** | **0.2778** | 0.2646 |
| PLH1 | **0.7200** | **0.5655** | 0.4300 | 0.2760 | 0.2660 |
| PLH2 | **0.7200** | **0.5655** | **0.4309** | 0.2764 | 0.2660 |
| PBH0 | **0.7127** | **0.5691** | **0.4309** | 0.2753 | 0.2661 |
| PBH1 | **0.7164** | **0.5691** | **0.4327** | **0.2756** | 0.2675 |
| PBH2 | **0.7164** | **0.5691** | **0.4327** | **0.2760** | 0.2675 |
| DLH0 | 0.7127 | **0.5673** | **0.4309** | **0.2782** | 0.2657 |
| DLH1 | 0.7200 | **0.5691** | **0.4318** | **0.2767** | 0.2684 |
| DLH2 | **0.7236** | **0.5673** | **0.4318** | **0.2767** | 0.2679 |
| DBH0 | **0.7127** | **0.5709** | **0.4318** | **0.2767** | 0.2672 |
| DBH1 | **0.7200** | **0.5727** | **0.4355** | **0.2764** | **0.2699** |
| DBH2 | **0.7164** | **0.5709** | **0.4327** | **0.2771** | 0.2693 |
| GLH0 | 0.7018 | 0.5618 | 0.4218 | 0.2731 | 0.2592 |
| GLH1 | **0.7127** | **0.5655** | 0.4282 | 0.2727 | 0.2624 |
| GLH2 | **0.7091** | 0.5636 | 0.4264 | 0.2724 | 0.2613 |
| GBH0 | **0.7091** | **0.5655** | 0.4236 | 0.2716 | 0.2612 |
| GBH1 | **0.7127** | **0.5673** | 0.4273 | 0.2724 | 0.2640 |
| GBH2 | **0.7164** | 0.5655 | 0.4264 | 0.2716 | 0.2634 |
| BLH0 | 0.7018 | 0.5618 | 0.4218 | 0.2735 | 0.2594 |
| BLH1 | **0.7127** | **0.5655** | 0.4282 | 0.2735 | 0.2624 |
| BLH2 | **0.7091** | 0.5636 | 0.4264 | 0.2727 | 0.2614 |
| BBH0 | **0.7091** | **0.5655** | 0.4255 | 0.2716 | 0.2613 |
| BBH1 | **0.7127** | **0.5673** | 0.4282 | 0.2724 | 0.2641 |
| BBH2 | **0.7164** | 0.5655 | 0.4273 | 0.2716 | 0.2635 |
| InLH0 | **0.7091** | 0.5618 | 0.4227 | 0.2724 | 0.2601 |
| InLH1 | **0.7127** | **0.5655** | 0.4273 | 0.2742 | 0.2629 |
| InLH2 | **0.7127** | **0.5655** | 0.4264 | 0.2727 | 0.2623 |
| InBH0 | **0.7018** | **0.5673** | 0.4255 | 0.2705 | 0.2612 |
| InBH1 | **0.7200** | **0.5673** | 0.4273 | 0.2724 | 0.2645 |
| InBH2 | **0.7164** | **0.5673** | 0.4273 | 0.2731 | 0.2637 |
| IneLH0 | 0.7055 | 0.5618 | 0.4227 | 0.2742 | 0.2603 |
| IneLH1 | **0.7127** | **0.5655** | 0.4291 | 0.2745 | 0.2632 |
| IneLH2 | **0.7091** | **0.5655** | 0.4273 | 0.2735 | 0.2624 |
| IneBH0 | **0.7127** | **0.5673** | 0.4245 | 0.2724 | 0.2626 |
| IneBH1 | **0.7236** | **0.5673** | **0.4300** | **0.2749** | 0.2655 |
| IneBH2 | **0.7164** | **0.5673** | 0.4273 | 0.2738 | 0.2644 |
| HGLH0 | **0.7091** | 0.5636 | 0.4300 | 0.2742 | 0.2647 |
| HGLH1 | **0.7236** | 0.5655 | 0.4282 | 0.2753 | 0.2658 |
| HGLH2 | **0.7200** | 0.5636 | 0.4291 | 0.2738 | 0.2658 |
| HGBH0 | **0.7164** | **0.5691** | **0.4318** | **0.2764** | 0.2664 |
| HGBH1 | **0.7164** | **0.5727** | **0.4336** | **0.2745** | 0.2671 |
| HGBH2 | **0.7164** | **0.5709** | **0.4336** | **0.2753** | 0.2671 |

TAB. A.19 – Performance des pondérations DFR avec la distance $L_1$ sur le corpus Oxford.

| | P@5 | P@10 | P@20 | P@50 | MAP |
|---|---|---|---|---|---|
| PLH0 | 0.6691 | 0.5527 | 0.4209 | 0.2680 | 0.2639 |
| PLH1 | 0.6691 | 0.5527 | 0.4200 | 0.2698 | 0.2646 |
| PLH2 | 0.6727 | 0.5509 | 0.4218 | 0.2698 | 0.2659 |
| PBH0 | 0.6727 | **0.5600** | 0.4264 | 0.2775 | 0.2722 |
| PBH1 | 0.6836 | 0.5600 | 0.4291 | 0.2785 | 0.2723 |
| PBH2 | 0.6800 | 0.5600 | 0.4309 | 0.2793 | 0.2741 |
| DLH0 | **0.6691** | 0.5545 | 0.4218 | 0.2691 | 0.2655 |
| DLH1 | **0.6982** | **0.5727** | **0.4409** | 0.2800 | 0.2834 |
| DLH2 | **0.6945** | **0.5655** | **0.4336** | 0.2771 | 0.2774 |
| DBH0 | **0.6764** | **0.5618** | 0.4300 | 0.2785 | 0.2743 |
| DBH1 | **0.6982** | **0.5727** | **0.4436** | **0.2920** | **0.2887** |
| DBH2 | **0.6909** | **0.5691** | **0.4409** | 0.2862 | 0.2844 |
| GLH0 | 0.6473 | 0.5436 | 0.4145 | 0.2658 | 0.2570 |
| GLH1 | 0.6618 | 0.5455 | 0.4173 | 0.2651 | 0.2603 |
| GLH2 | 0.6618 | 0.5473 | 0.4173 | 0.2647 | 0.2593 |
| GBH0 | 0.6655 | 0.5491 | 0.4218 | 0.2738 | 0.2665 |
| GBH1 | 0.6691 | 0.5527 | 0.4227 | 0.2749 | 0.2701 |
| GBH2 | 0.6691 | 0.5527 | 0.4245 | 0.2742 | 0.2695 |
| BLH0 | 0.6509 | 0.5436 | 0.4155 | 0.2658 | 0.2578 |
| BLH1 | 0.6618 | 0.5473 | 0.4173 | 0.2651 | 0.2604 |
| BLH2 | 0.6618 | 0.5491 | 0.4173 | 0.2647 | 0.2596 |
| BBH0 | 0.6655 | 0.5491 | 0.4218 | 0.2735 | 0.2669 |
| BBH1 | 0.6691 | 0.5527 | 0.4227 | 0.2749 | 0.2703 |
| BBH2 | 0.6691 | 0.5527 | 0.4245 | 0.2742 | 0.2696 |
| InLH0 | 0.6545 | 0.5436 | 0.4173 | 0.2673 | 0.2603 |
| InLH1 | 0.6618 | 0.5545 | 0.4200 | 0.2684 | 0.2654 |
| InLH2 | 0.6618 | 0.5527 | 0.4209 | 0.2680 | 0.2649 |
| InBH0 | 0.6582 | 0.5491 | 0.4282 | 0.2756 | 0.2689 |
| InBH1 | 0.6764 | 0.5600 | 0.4300 | 0.2800 | 0.2732 |
| InBH2 | 0.6727 | 0.5600 | 0.4291 | 0.2778 | 0.2728 |
| IneLH0 | 0.6509 | 0.5418 | 0.4136 | 0.2647 | 0.2584 |
| IneLH1 | 0.6655 | 0.5509 | 0.4182 | 0.2662 | 0.2630 |
| IneLH2 | 0.6655 | 0.5473 | 0.4182 | 0.2665 | 0.2625 |
| IneBH0 | 0.6655 | 0.5491 | 0.4218 | 0.2753 | 0.2678 |
| IneBH1 | 0.6764 | 0.5600 | 0.4273 | 0.2767 | 0.2737 |
| IneBH2 | **0.6764** | 0.5582 | 0.4264 | 0.2764 | 0.2726 |
| HGLH0 | **0.6873** | 0.5564 | 0.4282 | 0.2720 | 0.2725 |
| HGLH1 | 0.6691 | 0.5491 | 0.4200 | 0.2698 | 0.2634 |
| HGLH2 | 0.6691 | 0.5545 | 0.4209 | 0.2713 | 0.2658 |
| HGBH0 | **0.6873** | **0.5655** | 0.4364 | 0.2833 | 0.2786 |
| HGBH1 | 0.6727 | 0.5582 | 0.4291 | 0.2793 | 0.2714 |
| HGBH2 | 0.6727 | **0.5600** | 0.4309 | 0.2793 | 0.2732 |

TAB. A.20 – Performance des pondérations DFR avec la distance $L_2$ sur le corpus Oxford.

# Annexe B

# Fonctionnement et évaluation du détecteur de logos

Nous présentons dans cette annexe le détecteur de logos que nous avons mis au point pour effectuer nos expériences d'annotation d'images du chapitre 5. Ce sont les besoins de cette application qui ont motivé les propriétés de ce détecteur, ainsi que la manière de l'évaluer. Ce détecteur devait posséder une bonne précision, pour réduire le nombre d'erreurs provoquées par celui-ci dans notre processus d'annotation final, et être suffisamment rapide pour pouvoir être utilisé sur un corpus de données conséquent, ce qui nous pousse à effectuer quelques approximations dans la définition des fenêtres de détection, pour en réduire la nombre. Ces approximations restent néanmoins acceptables car, pour notre application, le fait de détecter un logo est plus important que la localisation exacte de celui-ci. Nous avons donc employé une mesure de performance adaptée à cet état de fait, qui se base sur l'intersection entre les fenêtres détectées comme contenant un logo et les fenêtres de référence (vérité-terrain) plutôt que sur la mesure de l'aire de la zone d'intersection, qui n'a pour nous qu'un intérêt secondaire.

Pour ce détecteur de logo, nous utilisons le formalisme des mots visuels présenté en section 2.3.1.1 page 58. Cette représentation des images a plusieurs avantages pour la détection de logos. D'abord, elle repose sur des régions d'image, ce qui permet d'avoir des informations locales sur les images, utiles pour un détecteur. Mais surtout, les descripteurs employés (descripteurs locaux SIFT) permettent de très bien saisir la nature des logos, car ceux-ci doivent être facilement identifiables, et contiennent donc plutôt des régions uniformes dont les frontières sont bien marquées, contrairement aux régions d'images naturelles qui contiennent plus de détails et sont donc plus bruitées. Les descripteurs locaux basés sur les directions des gradients (comme le sont les SIFT), permettent de bien rendre compte de cette différence.

Cette annexe s'organise de la manière suivante : d'abord nous présentons notre algorithme de détection de logos, puis quelques expériences qui nous ont permis de d'évaluer les valeurs optimales des différents paramètres et d'avoir un aperçu de l'efficacité de notre détecteur. Enfin, nous nous situons par rapport aux travaux existant les plus proches de ceux que nous exposons ici.

## B.1 Algorithme de détection

L'algorithme 2 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 [LKP], 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.

$M_I$ : **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**
  $l_{max} \leftarrow \text{argmax}_{l \in L}(\texttt{score}(l))$
  **Si** $(\texttt{score}(l_{max}) > \text{MIN_SCORE})$ **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 2: Algorithme de détection de logos.

### B.1.1 Calcul du score de détection

Pour chaque candidat logo, un score est calculé (fonction `score` de l'algorithme 2) puis comparé à un seuil pour savoir si ce candidat est effectivement un logo ou non. Notre calcul de score s'inspire de la théorie bayésienne. Soit $w$ un mot visuel et $L$ l'événement "est un logo", alors la quantité

$$q(w) = \frac{\Pr(w|L)}{\Pr(w)} \tag{B.1}$$

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 :

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

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 B.2.

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 :

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

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 :

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

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

$$n_1 = \frac{N_R}{A_R} \tag{B.5}$$

$$n_2 = \frac{N_R^2}{A_R} \tag{B.6}$$

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 B.2), en utilisant un lissage Laplacien pour éviter les probabilités nulles :

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

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 :

$$\mathrm{Pr}_{dist}(w) = \frac{1 + \sum_{k=1}^{Occ(w)} \frac{d_{min}(w)}{d(w^k)}}{|V| + \sum_{v \in V} \sum_{j=1}^{Occ(v)} \frac{d_{min}(v)}{d(v^j)}} \tag{B.8}$$

### B.1.2 Complexité

#### B.1.2.1 Phase d'apprentissage

Contrairement à de nombreux détecteurs basés sur de l'apprentissage supervisé (celui d'openCV [LKP] par exemple), notre détecteur nécessite une phase d'apprentissage très légère puisqu'il suffit d'estimer les probabilités $\mathrm{Pr}(w)$ et $\mathrm{Pr}(w|L)$ sur des ensembles d'apprentissage adaptés. Ces ensembles d'apprentissage peuvent eux-mêmes êtres obtenus à moindre coût (voir Section B.2).

#### B.1.2.2 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.

#### B.1.2.3 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.

#### B.1.2.4 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 visuels 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.

## B.2 Évaluation

### B.2.1 Conditions expérimentales

#### B.2.1.1 Données d'apprentissage

Notre détecteur nécessite de déterminer par apprentissage les probabilités d'occurrence des mots visuels. Les probabilités *a priori* $\Pr(w)$ d'occurrence des mots visuels sont calculées sur l'ensemble des images de presse du corpus présenté au chapitre 6. 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).

#### B.2.1.2 Conditions d'évaluation

Nous utilisons 413 images extraites de notre corpus d'articles de presse. 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 finale d'annotation (voir chapitre 6), 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.

#### B.2.1.3 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 [MS04]. Nous utilisons le descripteur local SIFT qui est le plus utilisé et offre d'excellentes propriétés [MS04]. Le vocabulaire est constitué grâce à une implémentation de l'algorithme de clustering *k-means* pour GPU.

### B.2.2 Résultats

#### B.2.2.1 Taille du vocabulaire

La figure B.1 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.

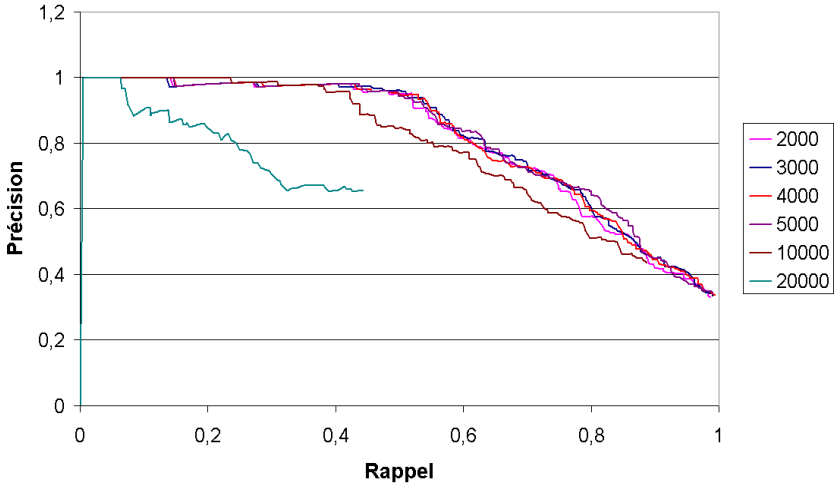



FIG. B.1 – Performances du détecteur pour différentes tailles de vocabulaire.

#### B.2.2.2 Taille minimale des logos

La figure B.2 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 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.

#### B.2.2.3 Élimination des mots superflus

La figure B.3 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

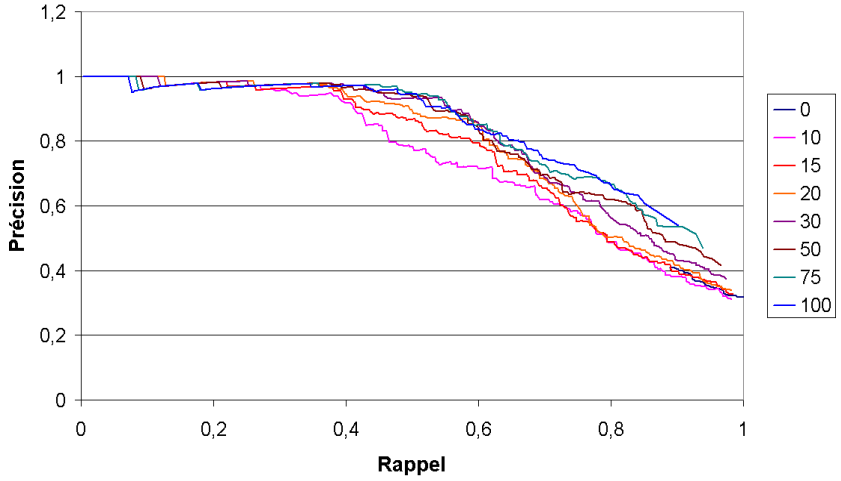



FIG. B.2 – Performances du détecteur en fonction de la taille minimale des régions détectées (la taille indiquée est celle d'un coté du rectangle).

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.

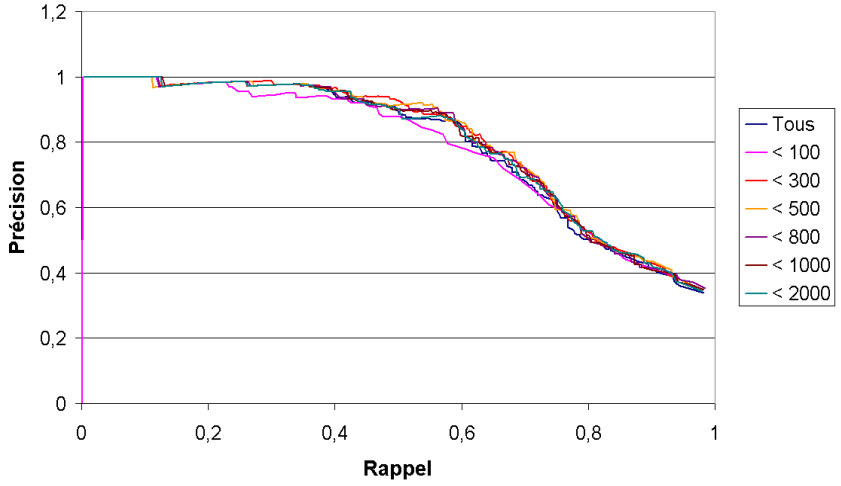



FIG. B.3 – Performances du détecteur en fonction de la taille des mots visuels retenus.

#### B.2.2.4 Scores de détection

La figure B.4 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 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.

La figure B.5 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

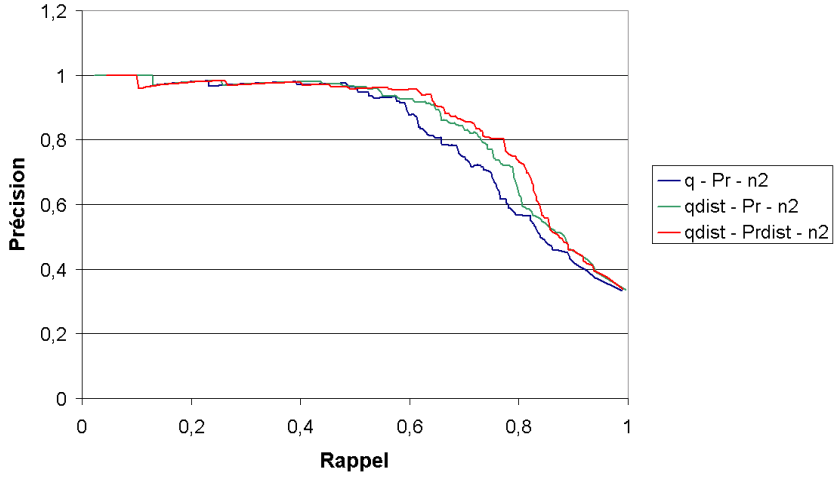



FIG. B.4 – Performances du détecteur en fonction du type de score utilisé.

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.

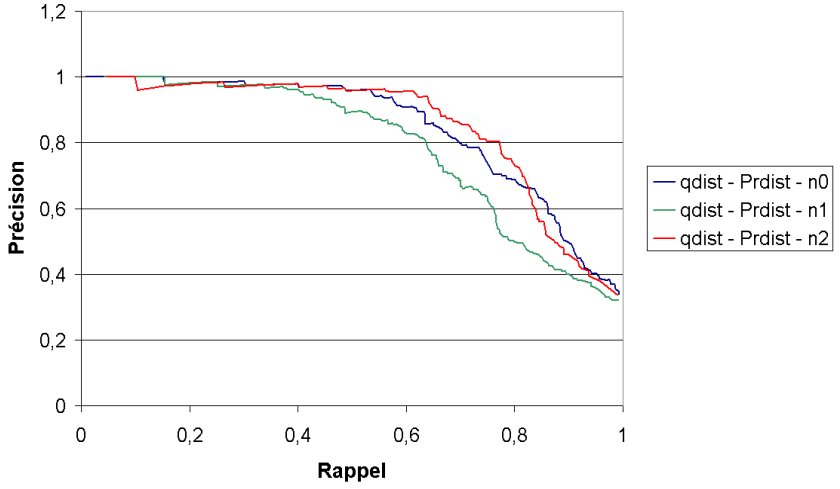



FIG. B.5 – Performances du détecteur en fonction de la normalisation de score utilisée.

#### B.2.2.5 Temps d'exécution

Le tableau B.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 B.1.2 (taille minimale des régions candidates : 50 pixels de côté, aire maximale des mots visuels : 500). On voit 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. B.1 – Temps d'exécution moyen par image (en secondes) des prétraitements et du détecteur.

## B.3 Travaux connexes

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 [SDI+97, Pha03]. 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 [PLS02, MTZK05]. 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 [SS07]. 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 [GY01, GEF04, Wu05] 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.

# Bibliographie


[AG01] Laurent Amsaleg et Patrick Gros : Content-based retrieval using local descriptors : Problems and issues from a database perspective. *Pattern Analysis and Applications*, 4(2-3):108–124, 2001.

[AHK01] Charu C. Aggarwal, Alexander Hinneburg et Daniel A. Keim : On the surprising behaviour of distance metrics in high dimensional spaces. *In Proceedings of the 8th International Conference on Database Theory*, Lecture Notes in Computer Science, pages 420–434, Londres, Royaume-Uni, 2001.

[AMC10] Rami Albatal, Philippe Mulhem et Yves Chiaramella : Phrases visuelles pour l'annotation automatique d'images. *In Actes de la ConfÉrence Francophone en Recherche d'Information et Applications (CORIA)*, Sousse, Tunisie, mars 2010.

[APBC+09] Julien Ah-Pine, Marco Bressan, Stéphane Clinchant, Gabriela Csurka, Yves Hoppenot et Jean-Michel Renders : Crossing textual and visual content in different application scenarios. *Multimedia Tools and Applications*, 42(1):31–56, mars 2009.

[AVR02] Gianni Amati et Cornelis Joost Van Rijsbergen : Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems*, 20(4):357–389, octobre 2002.

[BBE+04] Tamara L. Berg, Alexander C. Berg, Jaety Edwards, Michael Maire, Ryan White, Yee-Whye Teh, Erik Learned-Miller et David Forsyth : Names and faces in the news. *In Proceedings of the international conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 848–854, Washington, DC, États-Unis, juin-juillet 2004.

[BBV02] Sabri Boughorbel, Nozha Boujemaa et Constantin Vertan : Histogram-based color signatures for image indexing. *In Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, pages 977–984, Annecy, France, juillet 2002.

[BCD08] Nicolas Bonnel, Max Chevalier et Bernard Dousset : Métaphores de visualisation des résultats de recherche d'information sur le web. *In* Mohand Boughanem et Jacques Savoy, éditeurs : *Recherche d'information : état des lieux et perspectives*, pages 295–339. Hermès-Lavoisier, 2008.

[BDF03a] Kobus Barnard, Pinar Duygulu et David Forsyth : Recognition as translating images into text. *In Proceedings of Internet Imaging IX, Electronic Imaging*, pages 168–178, Santa Clara, CA, États-Unis, janvier 2003.

[BDF+03b] Kobus Barnard, Pinar Duygulu, David Forsyth, Nando de Freitas, David M. Blei et Michael I. Jordan : Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, mars 2003.

[BDG+03] Kobus Barnard, Pinar Duygulu, Raghavendra Guru, Prasad Gabbur et David Forsyth : The effects of segmentation and feature choice in a translation model of object recognition. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 675–682, Madison, WI, États-Unis, juin 2003.

[BF01] Kobus Barnard et David Forsyth : Learning the semantics of words and pictures. *In Proceedings of the International Conference of Computer Vision (ICCV)*, volume 2, pages 408–415, Vancouver, Canada, juillet 2001.

[BFM+96] Julio Barros, James French, Worthy Martin, Patrick Kelly et Mike Cannon : Using the triangle inequality to reduce the number of comparisons required for similarity-based retrieval. *In Proc. of SPIE/IS&T Conf. on Storage and Retrieval for Image and Video Databases IV*, pages 392–403, 1996.

[BJ03a] Kobus Barnard et Matthew Johnson : Word sense disambiguation with pictures. *In Proceedings of the Human Language Technology Conference, workshop on learning word meaning from non-linguistic data*, pages 1–5, Edmonton, Canada, mai-juin 2003.

[BJ03b] David M. Blei et Michael I. Jordan : Modeling annotated data. *In Proceedings of the ACM SIGIR conference*, pages 127–134, Toronto, Canada, juillet-août 2003.

[BN04] Jing Bai et Jian-Yun Nie : Using language models for text classification. *In Proceedings of the Asia Information Retrieval Symposium (AIRS)*, Pékin, Chine, octobre 2004.

[Bro66] Phil Brodatz : *Textures - A photographic Album for artists and designers.* Dover, New York, USA, 1966.

[BSA92] C. Buckley, G. Salton et J. Allan : Automatic retrieval with locality information using SMART. *In Proceedings of the first Text Retrieval Conference*, pages 59–72, Gaithersburg, USA, 1992.

[BSI08] Oren Boiman, Eli Shechtman et Michal Irani : In defense of nearest-neighbor based image classification. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 1–8, Anchorage, AK, États-Unis, juin 2008.

[BTVG08] Herbert Bay, Tinne Tuytelaars et Luc Van Gool : Surf : speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[BV00] Chris Buckley et Ellen M. Voorhees : Evaluating evaluation measure stability. *In Proceedings of the ACM SIGIR conference*, pages 33–40, Athènes, Grèce, juillet 2000.

[BZM06] Anna Bosch, Andrew Zisserman et Xavier Muñoz : Scene classification via pLSA. *In Proceedings of the European Conference on Computer Vision (ECCV)*, pages 517–530, Graz, Autriche, mai 2006.

[CC05] Chaur-Chin Chen et Hsueh-Ting Chu : Similarity measurement between images. *In Proceedings of the International Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 41–42, Édimbourg, Royaume-Uni, juillet 2005.

[CCS04] Claudio CUSANO, Gianluigi CIOCCA et Raimondo SCHETTINI : Image annotation using SVM. *In Proceedings of SPIE Internet Imaging V*, pages 330–338, San José, CA, États-Unis, janvier 2004.

[CDF+04] Gabriela CSURKA, Christopher R. DANCE, Lixin FAN, Jutta WILLAMOWSKI et Cédric BRAY : Visual categorization with bags of keypoints. *In Proceedings of the European Conference on Computer Vision (ECCV), workshop on statistical learning and computer vision*, pages 1–22, Prague, République Tchèque, mai 2004.

[CELM07] Yann CHEVALEYRE, Ulle ENDRISS, Jérome LANG et Nicolas MAUDET : A short introduction to computational social choices. *In Proceedings of the Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, pages 51–69, Harrachov, République Tchèque, janvier 2007.

[CG98] Stanley F. CHEN et Joshua GOODMAN : An empirical study of smoothing techniques for language modeling. Rapport technique TR-10-98, Harvard University, juillet 1998.

[CG10] Stéphane CLICHANT et Éric GAUSSIER : Modèles de ri fondés sur l'information. *In Actes de la Conférence Francophone en Recherche d'Information et Applications (CORIA)*, Sousse, Tunisie, mars 2010.

[CGSW03] Edward CHANG, Kingshy GOH, Gerard SYCHAY et Gang WU : CBSA : content-based soft annotation for multimodal image retrieval using bayes point machines. *IEEE Transactions on circuits and systems for video technology*, 13(1):26–38, 2003.

[CHS09] Xin CHEN, Xiaohua HU et Xiajiong SHEN : Spatial weighting for bag-of-visual-words and its application in content-based image retrieval. *In Proc. of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 867–874, Bangkok, Tha 2009.

[CJ04] Gustavo CARNEIRO et Allan D. JEPSON : Flexible spatial models for grouping local image features. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, volume 2, pages 747–754, Washington, DC, États-Unis, juin-juillet 2004.

[Cle67] C. W. CLEVERDON : The cranfield tests on index language devices. *Aslib Proceedings*, 19(6):173–194, 1967.

[CLZ04] Lianping CHEN, Guojun LU et Dengsheng ZHANG : Effects of different gabor filter parameters on image retrieval by texture. *In Proceedings of the international Multimedia Modelling Conference (MMM)*, pages 273–278, Brisbane, Australie, janvier 2004.

[CM02] D. COMANICU et P. MEER : Mean shift : a robust approach towards feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, mai 2002.

[CMK02] Antoine CORNUÉJOLS, Laurent MICLET et Yves KODRATOFF : *Apprentissage artificiel : concepts et algorithmes.* Eyrolles, 2002.

[CMS07] Vincent CLAVEAU, Fabienne MOREAU et Pascale SÉBILLOT : Description des textes. *In* Patrick GROS, éditeur : *L'indexation multimédia : description et recherche automatiques*, pages 163–190. Hermès-Lavoisier, 2007.

[CPS+07] Ondrej CHUM, James PHILBIN, Josef SIVIC, Michael ISARD et Andrew ZISSERMAN : Total recall : automatic query expansion with a generative feature

model for object retrieval. *In Proceedings of the International Conference of Computer Vision (ICCV)*, pages 1–8, Rio de Janeiro, Brésil, octobre 2007.

[CR97] Philip Clarckson et Ronald Rosenfeld : Statistical language modeling using the cmu-cambridge toolkit. *In Proceedings of Eurospeech*, volume 5, pages 2707–2710, Rhodes, Grèce, septembre 1997.

[CSFB03] Vincent Claveau, Pascale Sébillot, Cécile Fabre et Pierrette Bouillon : Learning semantic lexicons from a part-of-speech and semantically tagged corpus using inductive logic programming. *Journal of Machine Learning Research : Special issue on inductive logic programming*, 4:493–525, décembre 2003.

[CT94] William B. Cavnar et John M. Trenkle : N-gram-based text categorization. *In Proceedings of the Symposium on Document Analysis and Information Retrieval (SDAIR)*, pages 161–175, Las Vegas, NV, États-Unis, 1994.

[CTB+99] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein et Jitendra Malik : Blobworld : a system for region-based image indexing and retrieval. *In Proceedings of the international conference on visual information systems (VISUAL)*, pages 509–516, Amsterdam, Pays-Bas, juin 1999.

[CV02] Abdurrahman Carkacioglu et Fatos-Yarman Vural : Learning similarity space. *In Proceedings of the International Conference on Image Processing (ICIP)*, pages 405–408, Rochester, NY, USA, septembre 2002.

[Dai93] B. Daille : Conceptual structuring through term variations. *In Proceedings of the ACL Workshop on Multiword Expressions : Analysis, Acquisition and Treatment*, pages 9–16, Colombus, OH, États-Unis, juin 1993.

[dB81] Jean-Charles de Borda : Mémoire sur les élections au scrutin. *In Histoire de l'Académie Royale des Sciences*. 1781.

[DDH90] Scott Deerwester, Susan T. Dumais et Richard Harshman : Indexing by latent semantic analysis. *Journal of the American Society for Information Science and Technology*, 41(6):391–407, septembre 1990.

[DI07] Divna Djordjevic et Ebroul Izquierdo : An object and user-driven system for semantic-based image annotation and retrieval. *IEEE Transactions on circuits and systems for video technology*, 17(3):313–323, 2007.

[DJS+09] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg et Cordelia Schmid : Evaluation of gist descriptors for web-scale image search. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, Santorini, Fira, Grèce, juillet 2009.

[DLR77] A. P. Dempster, N. M. Laird et D. B. Rubin : Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[DM07] Koen Deschacht et Marie-Francine Moens : Text analysis for automatic image annotation. *In Proceedings of the annual meeting of the Association of Computational Linguistics (ACL)*, pages 1000–1007, Prague, République Tchèque, juin 2007.

[ESMUD10] Ismail El Sayad, Jean Martinet, Thierry Urruty et Chabane Djeraba : Visual sentence-phrase-based document representation for effective and efficient content-based image retrieval. *In Actes de la conférence Extraction et Gestion de Connaissances (EGC)*, pages 157–162, Hammamet, Tunisie, janvier 2010.

[FBC08] Marin FERECATU, Nozha BOUJEMAA et Michel CRUCIANU : Semantic interactive image retrieval combining visual and conceptual content description. *ACM Multimedia Systems Journal*, 13(5-6):309–322, 2008.

[Fel98] Christiane FELLBAUM, éditeur. *WordNet : an electronic lexical database*. The MIT Press, 1998.

[FFFP07] L. FEI-FEI, R. FERGUS et P. PERONA : Learning generative visual models from few training examples : an incremental bayesian approach tested on 101 object categories. volume 106, pages 59–70, avril 2007.

[FKS03] Ronald FAGIN, Ravi KUMAR et D. SIVAKUMAR : Comparing top-k lists. *SIAM Journal of Discrete Mathematics*, 17(1):134–160, 2003.

[FL08] Yansong FENG et Mirella LAPATA : Automatic image annotation using auxiliary text information. *In Proceedings of the annual meeting of the Association of Computational Linguistics (ACL)*, pages 272–280, Colombus, OH, États-Unis, juin 2008.

[FMnR+02] J. FREIXENET, X. MUÑOZ, D. RABA, J. MARTÍ et X. CUFÍ : Yet another survey on image segmentation : region and boundary information integration. *In Proceedings of the European Conference on Computer Vision (ECCV)*, pages 408–422, Copenhague, Danemark, mai 2002.

[Fou02] Nordine FOUROUR : Nemesis, un système de reconnaissance incrémentielle des entités nommées pour le français. *In Actes de la conférence Traitement Automatique des Langues Naturelles (TALN)*, pages 265–274, Nancy, France, juin 2002.

[FSC04] Huamin FENG, Rui SHI et Tat-Seng CHUAN : A bootstrapping framework for annotating and retrieving www images. *In Proceedings of ACM Multimedia*, pages 960–967, Juan-Les-Pins, France, décembre 2004.

[FTATG08] Ali FAKERI-TABRIZI, Massih-Reza AMINI, Sabrina TOLLARI et Patrick GALLINARI : UPMC/LIP6 at ImageCLEF's WikipediaMM : an image annotation model for an image search-engine. *In Working notes for the CLEF 2008 workshop*, Aarhus, Danemark, septembre 2008.

[FTZ04] Hui FANG, Tao TAO et ChengXiang ZHAI : A formal study of information retrieval heuristics. *In Proceedings of the ACM SIGIR conference*, pages 49–56, Sheffield, Royaume-Uni, juillet 2004.

[GCL05] King-Shy GOH, Edward Y. CHANG et Beitao LI : Using one-class and two-class svms for multiclass image annotation. *IEEE Transactions on Knowledge and Data Engineering*, 17(10):1333–1346, 2005.

[GCMD06] Michael GRUBINGER, Paul D. CLOUGH, Henning MÜLLER et Thomas DESELAERS : The iapr benchmark : A new evaluation resource for visual information systems. *In Proceedings of the Language Ressources Evaluation Conference (LREC), Workshop OntoImage : Language Resources for Content-Based Image Retrieval*, pages 3–23, Gênes, Italie, mai 2006.

[GEF04] Julinda GLLAVATA, Ralph EWERTH et Bernd FREISLEBEN : Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. *In Proceedings of the International Conference on Pattern Recognition (ICPR)*, volume 1, pages 425–428, Cambridge, Royaume-Uni, août 2004.

[GHP07] G. GRIFFIN, A. HOLUB et P. PERONA : Caltech-256 object category dataset. Rapport technique 7694, California Institute of Technology, 2007.

[GIK05] Arnab GHOSHAL, Pavel IRCING et Sanjeev KHUDANPUR : Hidden markov models for automatic image annotation and content-based retrieval of images and video. *In Proceedings of the ACM SIGIR conference*, pages 544–551, Salvador, Brésil, août 2005.

[GMVS09] Matthieu GUILLAUMIN, Thomas MENSINK, Jakob VERBEEK et Cordelia SCHMID : Tagprop : Discriminative metric learning in nearest neighbor models for image auto-annotation. *In Proceedings of the International Conference of Computer Vision (ICCV)*, Kyoto, Japon, septembre-octobre 2009.

[GNWC04] Jianfeng GAO, Jian-Yun NIE, Guangyuan WU et Guihong CAO : Dependance language model for information retrieval. *In Proceedings of the ACM SIGIR conference*, pages 170–177, Sheffield, Royaume-Uni, juillet 2004.

[GPK02] Simona E. GRIGORESCU, Nicolai PETKOV et Peter KRUIZINGA : Comparison of texture features based on gabor filters. *IEEE Transactions on Image Processing*, 11(10):1160–1167, 2002.

[GWL06] Sheng GAO, De-Hong WANG et Chin-Hui LEE : Automatic image annotation through multi-topic text categorization. *In Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*, volume 2, Toulouse, France, mai 2006.

[GY01] Jiang GAO et Jie YANG : An adaptative algorithm for text detection from natural scenes. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 84–89, Hawa 2001.

[Har75] Stephen P. HARTER : A probabilistic approach to automatic keyword indexing. *Journal of the american society for information science*, 26(4):197–206, juillet-août 1975.

[Har79] Robert M. HARALICK : Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[Hea92] M. A. HEARST : Automatic acquisition of hyponyms from large text corpora. volume 2, pages 539–545, Nantes, France, août 1992.

[HGR+89] Z. HARRIS, M. GOTTFRIED, T RYCKMAN, P. MATTICK(JR), A. DALADIER, T. N. HARRIS et S. HARRIS : *The Form of Information in Science : analysis of an immunology sublanguage*. Kluwer, 1989.

[Hie01] Djoerd HIEMSTRA : *Using language models for information retrieval*. Thèse de doctorat, University of Twente, Twente, Pays-Bas, 2001.

[HKM+97] Jing HUANG, S. Ravi KUMAR, Mandar MITRA, Wei-Jing ZHU et Ramin ZABIH : Image indexing using color correlograms. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 762–768, Porto-Rico, juin 1997.

[HL05] Jonathon S. HARE et Paul H. LEWIS : Saliency-based models of image content and their application to auto-annotation by semantic propagation. *In Proceedings of the European Semantic Web Conference*, Héraklion, Crète, mai 2005.

[Hof98] Thomas HOFMANN : Learning and representing topic. a hierarchical mixture model for word occurrences in document databases. *In Proceedings of the Conference for Automated Learning and Discovery (CONALD)*, Pittsburgh, PA, USA, juin 1998.

[Hof99] Thomas HOFMANN : Probabilistic latent semantic indexing. *In Proceedings of the ACM SIGIR conference*, pages 50–57, Berkeley, CA, États-Unis, août 1999.

[HR04] Peter HOWARTH et Stefan RÜGER : Evaluation of texture features for content-based image retrieval. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, pages 326–334, Dublin, Irlande, juillet 2004.

[HR05] Peter HOWARTH et Stefan RÜGER : Fractional distance measures for content-based image retrieval. *In Proceedings of the European Conference on Information Retrieval (ECIR)*, pages 447–456, Saint-Jacques-de-Compostelle, Espagne, mars 2005.

[HSL+06] Jonathon S. HARE, Patrick A. S. SINCLAIR, Paul H. LEWIS, Kirk MARTINEZ, Peter G. B. ENSER et Christine J. SANDOM : Bridging the semantic gap in multimedia information retrieval : top-down and bottom-up approaches. *In Proceedings of the international semantic web conference*, Budva, Monténégro, june 2006.

[HSWW03] Laura HOLLINK, Guus SCHREIBER, Jan WIELEMAKER et Bob WIELINGA : Semantic annotation of image collections. *In Proceedings of the workshop on Knowledge Markup and Semantic Annotation (KCAP)*, Floride, États-Unis, octobre 2003.

[JDS08] Hervé JÉGOU, Matthijs DOUZE et Cordelia SCHMID : Hamming embedding and weak geometric consistency for large scale image search. *In Proceedings of the European Conference on Computer Vision (ECCV)*, volume 1, pages 304–317, Marseille, France, octobre 2008.

[JDS09] Hervé JÉGOU, Matthijs DOUZE et Cordelia SCHMID : Packing bag-of-features. *In Proceedings of the International Conference of Computer Vision (ICCV)*, pages 2–9, Kyoto, Japon, septembre 2009.

[JFD+07] Michael JAMIESON, Afsaneh FAZLY, Sven DICKINSON, Suzanne STEVENSON et Sven WACHSMUTH : Learning structure appearance models from captionned images of cluttered scenes. *In Proceedings of the International Conference of Computer Vision (ICCV)*, pages 1–8, Rio de Janeiro, Brésil, octobre 2007.

[JHS07] Hervé JÉGOU, Hedi HARZALLAH et Cordelia SCHMID : A contextual dissimilarity measure for accurate and efficient image search. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 1–8, Minneapolis, MN, États-Unis, june 2007.

[JLM03] J. JEON, Victor LAVRENKO et R. MANMATHA : Automatic image annotation and retrieval using cross-media relevance models. *In Proceedings of the ACM SIGIR conference*, pages 119–126, Toronto, Canada, juillet-août 2003.

[JLZZ04] Feng JING, Mingjing LI, Hong-Jiang ZHANG et Bo ZHANG : Keyword propagation for image retrieval. *In Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 53–56, Vancouver, Canada, mai 2004.

[JM04] J. JEON et R. MANMATHA : Automatic image annotation of news images with large vocabularies and low quality training data. *In Proceedings of ACM Multimedia*, New-York, NY, États-Unis, octobre 2004.

[JN09] Yu-Gang JIANG et Chong-Wah NGO : Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval. *Computer Vision and Image Understanding*, 113(3):405–414, mars 2009.

[JNY07] Yu-Gang JIANG, Chong-Wah NGO et Jun YANG : Towards optimal bag-of-features for object categorization and semantic video retrieval. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, pages 494–501, Amsterdam, Pays-Bas, juillet 2007.

[Jol02] I.T. JOLLIFFE : *Principal Component Analysis*. Springer, 2002.

[JT05] Frédéric JURIE et Bill TRIGGS : Creating efficient codebooks for visual recognition. *In Proceedings of the International Conference of Computer Vision (ICCV)*, volume 1, pages 604–610, Pékin, Chine, octobre 2005.

[JT09] Tao JIANG et Ah-Hwee TAN : Learning image-text associations. *IEEE Transactions on Knowledge and Data Engineering*, 21(2):161–177, février 2009.

[KAVJ10] Josip KRAPAC, Moray ALLAN, Jakob VERBEEK et Frédéric JURIE : Improving web-image search results using query-relative classifiers. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, San Francisco, CA, États-Unis, june 2010.

[KCB03] M. KOKARE, B. N. CHATTERJI et P. K. BISWAS : Comparison of similarity metrics for texture image retrieval. *In Proceedings of the IEEE Conference on Convergent Technologies for Asia-Pacific Region*, pages 571–575, Bangalore, Inde, octobre 2003.

[KNA+04] Andrea KUTICS, Akihiko NAKAGAWA, Shoji ARAI, Hiroyuki TANAKA et Sakuichi OHTSUKA : Relating words and image segments on multiple layers for effective browsing and retrieval. *In Proceedings of the International Conference on Image Processing (ICIP)*, pages 2203–2206, Singapour, octobre 2004.

[KS04] Yan KE et Rahul SUKTHANKAR : PCA-SIFT : A more distinctive representation for local image descriptors. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, volume 2, pages 506–513, Washington, DC, États-Unis, juin-juillet 2004.

[KSA+08] Judith L. KLAVANS, Carolyn SHEFFIELD, Eileen ABELS, Jimmy LIN, Rebecca PASSONNEAU, Tandeep SIDHU et Dagobert SOERGEL : Computational linguistics for metadata building (CLiMB) : using text mining for the automatic identification, categorization, and disambiguation of subject terms for image metadata. *Multimedia Tools and Applications*, 42(1):115–138, mars 2008.

[LAJA08] Herwig LEJSEK, Friðrik Heiðar ÁSMUNDSSON, Björn Þór JÓNSSON et Laurent AMSALEG : Nv-tree : An efficient disk-based index for approximate search in very large high-dimensional collections. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):869–883, mai 2008.

[LCC04] Wen-Cheng LIN, Yih-Chen CHANG et Hsin-Hsi CHEN : From text to image : Generating visual query for image retrieval. *In Proceedings of the Cross-Language Evaluation Forum, Worshop on Multilingual Information Access for Text, Speech and Audio*, pages 664–675, Bath, Royaume-Uni, septembre 2004.

[LCC+08] Xiaoyan LI, Lidan CHOU, Gang CHEN, Tianlei HU et Jinxiang DONG : Modeling image data for effective indexing and retrieval in large general image databases. *IEEE Transactions on Knowledge and Data Engineering*, 20(11):1566–1580, novembre 2008.

[LCLG08] Joo-Hwee LIM, Jean-Pierre CHEVALLET, Diem Thi Hoang LE et Hanlin GOH : Bi-modal conceptual indexing for medical image retrieval. *In Proceedings of the conference on Multimedia Modeling (MMM)*, pages 456–465, Kyoto, Japan, janvier 2008.

[LCW03] Beitao Li, Edward Chang et Yi Wu : Discovery of a perceptual distance function for measuring image similarity. *Multimedia Systems, Special Issue on Content-Based Image Retrieval*, 8(6):512–522, avril 2003.

[LDXE08] Fei Li, Qionghai Dai, Wenli Xu et Guihua Er : Multilabel neighborhood propagation for region-based image retrieval. *IEEE Transactions on Multimedia*, 10(8):1592–1604, 2008.

[LGC03] Beitao Li, Kingshy Goh et Edward Y. Chang : Confidence-based dynamic ensemble for image annotation and semantics discovery. *In Proceedings of ACM Multimedia*, pages 195–206, Berkeley, CA, États-Unis, novembre 2003.

[LJ09] Diane Larlus et Frédéric Jurie : Latent mixture vocabularies for object categorization. *Image and Vision Computing*, 27(5):523–534, avril 2009.

[LKP] Rainer Lienhart, Alexander Kuranov et Vadim Pisarevsky : Empirical analysis of detection cascades of boosted classifiers for rapid object detection. *In Proceedings of the annual Pattern Recognition Symposium of the German Association for Pattern Recognition (DAGM)*, pages 297–304, Magdeburg, Allemagne, septembre.

[LL03] Suryani Lim et Guojun Lu : Effectiveness and efficiency of six colour spaces for content based image retrieval. *In Proceedings of the conference on Content-Based Multimedia Indexing (CBMI)*, pages 215–219, Rennes, France, septembre 2003.

[LLCL07] Caroline Lacoste, Joo-Hwee Lim, Jean-Pierre Chevallet et Thi Hoang Diem Le : Medical image retrieval based on knowledge-assisted text and image indexing. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(7):889–900, juillet 2007.

[LM01] L. Lucchese et S.K. Mitra : Colour image segmentation : a state-of-the-art survey. *Proceedings of the Indian National Science Academy*, 67(2):207–221, 2001.

[LMJ03] Victor Lavrenko, R. Manmatha et J. Jeon : A model for learning the semantics of pictures. *Advances in Neural Information Processing Systems*, 16:553–560, 2003.

[LMS+09] Stefanie Lindstaedt, Roland Mörzinger, Robert Sorschag, Victoria Pammer et Georg Thallinger : Automatic image annotation using visual content and folksonomies. *Multimedia Tools and Applications*, 42(1):97–113, mars 2009.

[Low99] David G. Lowe : Object recognition from local scale-invariant features. *In Proceedings of the International Conference of Computer Vision (ICCV)*, volume 2, pages 1150–1157, Kerkyra, Corfou, Grèce, août 1999.

[Low04] David G. Lowe : Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, novembre 2004.

[LQL+05] Ying Liu, Tao Qin, Tie-Yan Liu, Lei Zhang et Wei-Ying Ma : Similarity space projection for web image search and annotation. *In Proceedings of ACM Multimedia, Workshop on Multimedia Information Retrieval (MIR)*, pages 49–56, novembre 2005.

[LRB09] Marie-Jeanne Lesot, Maria Rifqi et H. Benhadda : Similarity measures for binary and numerical data : a survey. *International Journal on Knowledge Engineering and Soft Data Paradigms*, 1(1):63–84, décembre 2009.

[LRD09] Marie-Jeanne LESOT, Maria RIFQI et Marcin DETYNIECKI : Comparaison de distances et noyaux classiques par degré dâĂŹéquivalence des ordres induits. *In Actes de la conférence Extraction et Gestion de Connaissances (EGC)*, pages 51–61, Strasbourg, France, janvier 2009.

[LSR+08] Haiming LIU, Dawei SONG, Stefan RÜGER, Rui HU et Victoria UREN : Comparing dissimilarity measures for content-based image retrieval. *In Proceedings of the Asia Information Retrieval Symposium (AIRS)*, pages 44–50, Harbin, Chine, janvier 2008.

[Luh58] H. P. LUHN : The automatic creation of literature abstracts. *IBM Journal on Research and Development*, 2(2):159–165, 1958.

[LZ01] John LAFFERTY et ChengXiang ZHAI : Document language models, query models, and risk minimization for information retrieval. *In Proceedings of the ACM SIGIR conference*, pages 111–119, Nouvelle-Orléans, LA, États-Unis, septembre 2001.

[MDS05] Kieran MC DONALD et Alan F. SMEATON : A comparison of score, rank and probability-based fusion methods for video shot retrieval. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, pages 61–70, Singapour, juillet 2005.

[MGP04] Florent MONAY et Daniel GATICA-PEREZ : pLSA-based image auto-annotation : constraining the latent space. *In Proceedings of ACM Multimedia*, pages 348–351, New-York, NY, États-Unis, octobre 2004.

[MMMP02] Henning MÜLLER, Stéphane MARCHAND-MAILLET et Thierry PUN : The truth about corel-evaluation in image retrieval. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, pages 38–49, Londres, Royaume-Uni, juillet 2002.

[Mor06] Fabienne MOREAU : *Revisiter le Couplage Traitement Automatique des Langues et Recherche d'Information.* Thèse de doctorat, Université de Rennes 1, Rennes, France, 2006.

[MOVY01] B. S. MANJUNATH, Jens-Rainer OHM, Vinod D. VASUDEVAN et Akio YAMADA : Color and texture descriptors. *IEEE Transactions on circuits and systems for video technology*, 11(6):703–715, 2001.

[MPK08] Ameesh MAKADIA, Vladimir PAVLOVIC et Sanjiv KUMAR : A new baseline for image annotation. *In Proceedings of the European Conference on Computer Vision (ECCV)*, pages 316–329, Marseille, France, mai 2008.

[MS04] Krystian MIKOLAJCZYK et Cordelia SCHMID : Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, octobre 2004.

[MS05] Krystian MIKOLAJCZYK et Cordelia SCHMID : A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, octobre 2005.

[MTO99] Yasuhide MORI, Hironobu TAKAHASHI et Ryuichi OKA : Image-to-word transformation based on dividing and vector quantizing images with words. *In Proceedings of the International Workshop on Multimedia Intelligent Storage and Retrieval Management (MISRM)*, Orlando, FL, États-Unis, décembre 1999.

[MTS+05] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir et Luc Van Gool : A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[MTZK05] Katrin Meisinger, Tobias Troeger, Marcus Zeller et André Kaup : Automatic tv logo removal using statistical based logo detection and frequency selective inpainting. *In Proceedings of the European Signal Processing Conference (EUSIPCO)*, Antalya, Turquie, septembre 2005.

[MZ98] Wei-Ying Ma et H. J. Zhang : Benchmarking of image features for content-based image retrieval. *In Proceedings of the Asilomar Conference on Signals, Systems and Computers*, pages 253–257, Pacific Grove, CA, États-Unis, novembre 1998.

[NS06] David Nistér et Henrik Stewénius : Scalable recognition with a vocabulary tree. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, volume 2, pages 2161–2168, juin 2006.

[NS07] David Nadeau et Satoshi Sekine : A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.

[OT01] Aude Oliva et Antonio Torralba : Modeling the shape of the scene : a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[PC98] Jay M. Ponte et W. Bruce Croft : A language modeling approach to information retrieval. *In Proceedings of the ACM SIGIR conference*, pages 275–281, Melbourne, Australie, août 1998.

[PCI+07] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic et Andrew Zisserman : Object retrieval with large vocabularies and fast spatial matching. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 1–8, Minneapolis, MN, États-Unis, juin 2007.

[PCI+08] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic et Andrew Zisserman : Lost in quantization : Improving particular object retrieval in large scale image databases. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 1–8, Anchorage, AK, États-Unis, juin 2008.

[PCL08] Trong Ton Pham, Jean-Pierre Chevallet et Joo Hwee Lim : Fusion de multimodalités et réduction par sémantique latente. *In Actes de la Conférence Francophone en Recherche d'Information et Applications (CORIA)*, pages 34–59, Tregastel, France, mars 2008.

[PG08] Adrian Popescu et Gregory Grefenstette : A conceptual approach to web image retrieval. *In Proceedings of the Language Ressources Evaluation Conference (LREC)*, Marrakech, Maroc, mai 2008.

[Pha03] Tuan D. Pham : Unconstrained logo detection in document images. *Pattern Recognition*, 36(12):3023–3025, 2003.

[Pha09] Nguyen-Khang Pham : *Analyse factorielle des correspondances pour l'indexation et la recherche d'information dans une grande base de données d'images.* Thèse de doctorat, Université de Rennes 1, Rennes, France, novembre 2009.

[PLS02] Hao Pan, Baoxin Li et M. Ibrahim Sezan : Automatic detection of replay segments in broadcast sports programs by detection of logos in scene transitions. *In Proceedings of the International Conference on Audio, Speech and*

*Signal Processing (ICASSP)*, pages 3385–3388, Orlando, FL, États-Unis, mai 2002.

[PSW03] Katerina PASTRA, Horacio SAGGION et Yorick WILKS : Intelligent indexing of crime scene photographs. *IEEE Intelligent Systems*, 18(1):55–61, janvier 2003.

[QH07] Xiaojun QI et Yutao HAN : Incorporating multiple SVMs for automatic image annotation. *Pattern Recognition*, 40(2):728–741, fevrier 2007.

[Rob77] S.E. ROBERTSON : The probability ranking principle in information retrieval. *Journal of documentation*, 33:294 – 304, 1977.

[RPTB01] Yossi RUBNER, Jan PUZICHA, Carlo TOMASI et Joachim M. BUHMANN : Empirical evaluation of dissimilarity measures for color and texture. *Computer Vision and Image Understanding*, 84(1):24–43, octobre 2001.

[RWB+96] Stephen E. ROBERTSON, Stephen G. WALKER, M. M. BEAULIEU, M. GATFORD et A. PAYNE : Okapi at trec 4. *In Proceedings of the Text Retrieval Conference (TREC)*, pages 73–96, Gaithersburg, MD, États-Unis, novembre 1996.

[RYWL07] Xiaoguang RUI, Nenghai YU, Taifeng WANG et Mingjing LI : A search-based web image annotation method. *In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 655–658, Pékin, Chine, juillet 2007.

[Sal71] G. SALTON : *The SMART Retrieval System – Experiments in Automatic Document Processing.* Prentice-Hall, Inc., 1971.

[Sav08] Jacques SAVOY : Dépister de l'information sur le web. *In* Mohand BOUGHANEM et Jacques SAVOY, éditeurs : *Recherche d'information : état des lieux et perspectives*, pages 45–74. Hermès-Lavoisier, 2008.

[SB88] Gerard SALTON et Chris BUCKLEY : Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[SB94] Rohini K. SRIHARI et Debra T. BURHANS : Visual semantics : Extracting visual information from text accompanying pictures. *In Proceedings of the AAAI National Conference on Artificial Intelligence*, volume 1, pages 793–798, Seattle, WA, États-Unis, juillet-août 1994.

[SC96] J.R. SMITH et Shih-Fu CHANG : Automated binary texture feature sets for image retrieval. *In Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*, volume 4, pages 2239–2242, Atlanta, GA, États-Unis, mai 1996.

[SDI+97] Steve SEIDEN, Michael DILLENCOURT, Sandy IRANI, Roland BORREY et Timothy MURPHY : Logo detection in document images. *In Proceedings of the International Conference on Imaging Science, Systems, and Technology*, pages 446–449, Las Vegas, NV, États-Unis, juin 1997.

[SDLW10] Neela SAWANT, Ritendra DATTA, Jia LI et James Z. WANG : Quest for relevant tags using local interaction networks and visual content. *In Proceedings of the international conference on Multimedia Information Retrieval (MIR)*, pages 231–240, mars 2010.

[Sha76] Glenn SHAFER : *A Mathematical Theory of Evidence.* Princeton University Press, 1976.

[SJ72] Karen Spärck Jones : A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):132–142, 1972.

[SJWR00a] Karen Spärck Jones, Stephen G. Walker et Stephen E. Robertson : A probabilistic model of information retrieval : Development and comparative experiments (part 2). *Information Processing and Management*, 36(6):809–840, novembre 2000.

[SJWR00b] Karen Spärck Jones, Stephen G. Walker et Stephen E. Robertson : A probabilistic model of information retrieval : Development and comparative experiments (part 1). *Information Processing and Management*, 36(6):779–808, novembre 2000.

[SM97] Cordelia Schmid et Roger Mohr : Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, mai 1997.

[SM00] Jianbo Shi et Jitendra Malik : Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[Sma93] Frank A. Smadja : Retrieving collocations from text : Xtract. *Computational Linguistics*, 19(1):143–177, 1993.

[SN04] Satoshi Sekine et Chikashi Nobata : Definition, dictionaries and tagger for extended named entities hierarchy. *In Proceedings of the Language Ressources Evaluation Conference (LREC)*, Lisbonne, Portugal, mai 2004.

[SNK99] Shin'ichi Satoh, Yuichi Nakamura et Takeo Kanade : Name-it : Naming and detecting faces in news video. *IEEE Transactions on Multimedia*, 6(1):22–35, 1999.

[SS07] Subhajit Sanyal et S. H. Srinivasan : Logoseeker : a system for detecting and matching logos in natural images. *In Proceedings of ACM Multimedia*, pages 166–167, Augsburg, Allemagne, septembre 2007.

[SWY75] G. Salton, A. Wong et C. S. Yang : A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[SZ03] J. Sivic et A. Zisserman : Video Google : A text retrieval approach to object matching in videos. *In Proceedings of the International Conference of Computer Vision (ICCV)*, volume 2, pages 1470–1477, Nice, France, octobre 2003.

[TDFT+08] Sabrina Tollari, Marcin Detyniecki, Ali Fakeri-Tabrizi, Massih-Reza Amini et Patrick Gallinari : UPMC/LIP6 at ImageCLEFphoto 2008 : on the exploitation of visual concepts (VCDT). *In Working notes for the ImageCLEF workshop*, Aarhus, Danemark, septembre 2008.

[TDFT+09] Sabrina Tollari, Marcin Detyniecki, Ali Fakeri-Tabrizi, Christophe Marsala, Massih-Reza Amini et Patrick Gallinari : Using visual concepts and fast visual diversity to improve image retrieval. *In Evaluating Systems for Multilingual and Multimodal Information Access – Workshop of the Cross-Language Evaluation Forum, Articles sélectionnés*, pages 577–584. Springer, 2009.

[TG07] Sabrina Tollari et Hervé Glotin : Web image retrieval on ImagEVAL : Evidences on visualness and textualness concept dependency in fusion model. *In Proceedings of the Conference on Image and Video Retrieval (CIVR)*, pages 65–72, Amsterdam, Pays-Bas, juillet 2007.

[THJA04] I. Tsochantaridis, T. Hofmann, T. Joachims et Y. Altun : Support vector machine learning for interdependent and structured output spaces. *In Proceedings of the International Conference on Machine Learning (ICML)*, pages 823–830, Banff, Canada, juillet 2004.

[THL+06] Hanghang Tong, Jingrui He, Mingjing Li, Wei-Ying Ma, Hong-Jiang Zhang et Changshui Zhang : Manifold-ranking-based keyword propagation for image retrieval. *EURASIP Journal on Applied Signal Processing*, 2006(2):1–11, janvier 2006.

[TL06] Jiayu Tang et Paul H. Lewis : Image auto-annotation using 'easy' and 'more challenging' training sets. *In Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*, pages 121–124, Incheon, Corée du Sud, avril 2006.

[TMY78] H. Tamura, S. Mori et T. Yamawaki : Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8(6):460–473, 1978.

[Tol05] Sabrina Tollari : Filtrage de l'indexation textuelle d'une image au moyen du contenu visuel pour un moteur de recherche d'images sur le web. *In Actes de la Conférence Francophone en Recherche d'Information et Applications (CORIA)*, pages 261–275, Grenoble, France, mars 2005.

[Tol06] Sabrina Tollari : *Indexation et recherche d'images par fusion d'informations textuelles et visuelles.* Thèse de doctorat, Université du Sud Toulon-Var, Toulon, France, octobre 2006.

[TS00] C. P. Town et D. Sinclair : Content based image retrieval using semantic visual categories. Rapport technique, AT&T Labs Cambridge, 2000.

[Voo02] Ellen M. Voorhees : The philosophy of information retrieval evaluation. *In Evaluation of cross-language information retrieval systems – Workshop of the Cross-Language Evaluation Forum, Articles sélectionnés*, pages 355–370. Springer, 2002.

[VR77] C. J. Van Rijsbergen : *Information retrieval.* Butterworth, 1977.

[VS07] Julia Vogel et Bernt Schiele : Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision*, 72(2):133–157, avril 2007.

[VZ05] Manik Varma et Andrew Zisserman : A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1/2):61–81, 2005.

[WHY09] Lei Wu, Steven C.H. Hoi et Nenghai Yu : Semantics-preserving bag-of-words models for efficient image annotation. *In Proceedings of ACM Multimedia, Workshop on Large-Scale Multimedia Retrieval and Mining (LS-MMRM)*, pages 19–26, Pékin, Chine, octobre 2009.

[WJY08] Wang Wei, Hong Jun et Tang Yiping : Image matching for geomorphic measurement based on SIFT and RANSAC methods. *In Proceedings of the International Conference on Computer Science and Software Engineering (CSSE)*, volume 2, pages 317–320, Wuhan, Chine, décembre 2008.

[WL08] R. C. F. Wong et C. H. C. Leung : Automatic semantic annotation of real-world web images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1933–1944, novembre 2008.

[WLC08] Huan WANG, Song LIU et Liang Tien CHIA : Image retrieval with a multi-modality ontology. *Multimedia Systems*, 13(5-6):379–390, 2008.

[WLL+07] Lei WU, Mingjing LI, Zhiwei LI, Wei-Ying MA et Nenghai YU : Visual language modeling for image classification. *In Proceedings of ACM Multimedia, Workshop on Multimedia Information Retrieval (MIR)*, pages 115–124, Augsburg, Allemagne, juillet 2007.

[Wu05] Ching-Tung WU : Embedded-text detection and its application to anti-spam filtering. Mémoire de D.E.A., University of California, Santa Barbara, 2005.

[WZW85] S. K. M. WONG, Wojciech ZIARKO et Patrick C. N. WONG : Generalized vector spaces model in information retrieval. *In Proceedings of the ACM SIGIR conference*, pages 18–25, Montréal, Canada, 1985.

[YDH06] Changbo YANG, Ming DONG et Jing HUA : Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 2057–2063, New-York, NY, États-Unis, juin 2006.

[YJHN07] Jun YANG, Yu-Gang JIANG, Alex G. HAUPTMANN et Chong-Wah NGO : Evaluating bag-of-visual-words representations in scene classification. *In Proceedings of ACM Multimedia, Workshop on Multimedia Information Retrieval (MIR)*, pages 197–206, Augsburg, Allemagne, juillet 2007.

[YWY07] Junsong YUAN, Ying WU et Ming YANG : Discovery of collocation patterns : from visualwords to visual phrases. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, pages 1–8, Minneapolis, MN, États-Unis, juin 2007.

[ZG02] Rong ZHAO et William I. GROSKY : Narrowing the semantic gap – improved text-based web document retrieval using visual features. *IEEE Transactions on Multimedia*, 4(2):189–200, 2002.

[ZH02] Xiang Sean ZHOU et Thomas S. HUANG : Unifying keywords and visual content in image retrieval. *IEEE Transactions on Multimedia*, 9(2):22–33, avril 2002.

[Zha03] DongQing ZHANG : A bayesian framework for fusing multiple word knowledge models in videotext recognition. *In Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, volume 2, pages 528–533, Madison, WI, États-Unis, juin 2003.

[ZLX09] Shile ZHANG, Bin LI et Xiangyang XUE : Semi-automatic dynamicauxiliary-tag-aidedimageannotation. *Pattern Recognition*, 43(2):470–477, fevrier 2009.

[Zob98] Justin ZOBEL : How reliable are the results of large-scale information retrieval expreiments ? *In Proceedings of the ACM SIGIR conference*, pages 307–314, Melbourne, Australie, août 1998.

[ZRZ02] Lei ZHU, Aibing RAO et Aidong ZHANG : Theory of keyblock-based image retrieval. *ACM Transactions on Information Systems*, 20(2):224–257, 2002.

[ZS01] H. J. ZHANG et Z. SU : Improving CBIR by semantic propagation and cross modality query expansion. *Joho Shori Gakkai Shinpojiumu Ronbunshu*, 2001(17):113–116, 2001.

[ZWG06] Qing-Fang ZHENG, Wei-Qiang WANG et Wen GAO : Effective and efficient object-based image retrieval using visual phrases. *In Proceedings of ACM Multimedia*, pages 77–80, Santa Barbara, CA, États-Unis, octobre 2006.

[ZYC06] Qiang ZHU, Mei-Chen YEH et Kwang-Ting CHENG : Multimodal fusion using learned text concepts for image categorization. *In Proceedings of ACM Multimedia*, pages 211–220, Santa Barbara, CA, États-Unis, octobre 2006.

# Table des figures


1.1 Un système de recherche d'images classique. . . . . 14
1.2 Exemple de courbe rappel-précision. . . . . 18
1.3 Une image en niveau de gris et la matrice de pixels lui correspondant. . . . 21
1.4 Deux images de croix ayant des matrices de pixels totalement différentes. . . 21
1.5 Images similaires au sens de la recherche de scènes identiques (corpus Kentucky). . . . . 23
1.6 Images similaires au sens de la recherche d'images catégorisées (corpus Caltech-101). . . . . 24
1.7 Deux façons de découper une image en blocs. . . . . 32
1.8 Une image et une segmentation possible de celle-ci. . . . . 32
1.9 Détection de régions d'intérêt sur une image. . . . . 33
1.10 Une image annotée globalement et une image annotée localement. . . . . 39

2.1 Processus de construction d'un vocabulaire visuel et de description d'une image comme un ensemble de mots visuels. . . . . 58

3.1 Visage représenté par un ensemble de mots visuels. . . . . 65
3.2 Nombre de régions d'intérêt détectées en fonction de la taille des images. . . 66
3.3 Détection d'un nombre variable de régions d'intérêt sur des images de taille comparable. . . . . 66
3.4 Exemple d'une région détectée plusieurs fois sur une image de moto. . . . . 67
3.5 Importance des termes dans un document selon l'hypothèse de Luhn. . . . . 69
3.6 Sélection de mots visuels basée sur pLSA. . . . . 70
3.7 Influence de $k$ sur les performances des distances $L_k$. . . . . 84
3.8 Gain de performances d'un système utilisant les pondérations des Tableaux 3.7 and 3.8 et la distance $L_1$, par rapport à la pondération de base $l_1g_0$. . . . . 87
3.9 Gain de performances d'un système utilisant les pondérations des Tableaux 3.7 and 3.8 et la distance $L_2$, par rapport à la pondération de base $l_1g_0$. . . . . 88
3.10 Performances d'un système utilisant les pondérations DFR et la distance $L_1$, par rapport à la pondération de base $l_1g_0$. . . . . 89
3.11 Performances d'un système utilisant les pondérations DFR et la distance $L_2$, par rapport à la pondération de base $l_1g_0$. . . . . 90
3.12 Importance accordée à la distance locale en fonction de la valeur de $k$. . . . 91
3.13 Fréquence moyenne et écart-type de fréquence des mots visuels. . . . . 97
3.14 Images requêtes de la base d'images Oxford. . . . . 98

4.1 Invariance de la projection orthogonale aux changements d'échelle, aux translations et aux rotations. . . . . 106
4.2 Méthode de construction des phrases visuelles. . . . . 107
4.3 Exemples d'axes obtenus par ACP sur les centres des régions d'intérêt . . . 108
4.4 Une image avant et après filtrage des régions redondantes. . . . . 110
4.5 Performances en classification sur Caltech-6. . . . . 114
4.6 Performances en classification sur Caltech-101. . . . . 115
4.7 Performances en classification par catégorie sur Caltech-6. . . . . 115

5.1 Un exemple d'article tiré de notre corpus. . . . . 121
5.2 Les premiers résultats d'une recherche d'images basée sur la couleur. . . . . 125
5.3 Performances des critères d'annotation de type df pour les visages. . . . . 131
5.4 Performances des critères d'annotation de type af pour les visages. . . . . 132
5.5 Performances des critères d'annotation de type laf pour les visages. . . . . 132
5.6 Performances des critères d'annotation de type df pour les logos. . . . . 133
5.7 Performances des critères d'annotation de type af pour les logos. . . . . 133
5.8 Performances des critères d'annotation de type laf pour les logos. . . . . 134
5.9 Des exemples d'annotations d'images obtenues avec notre système. Les entités nommées candidates et leurs fréquences sont indiquées, l'entité retenue pour annoter l'image est celle en gras. . . . . 134

B.1 Performances du détecteur pour différentes tailles de vocabulaire. . . . . 168
B.2 Performances du détecteur en fonction de la taille minimale des régions détectées (la taille indiquée est celle d'un coté du rectangle). . . . . 169
B.3 Performances du détecteur en fonction de la taille des mots visuels retenus. 169
B.4 Performances du détecteur en fonction du type de score utilisé. . . . . 170
B.5 Performances du détecteur en fonction de la normalisation de score utilisée. 170

# Liste des tableaux


3.1 Nombre de mots visuels en fonction du nombre de catégories où ils apparaissent (données Caltech6). . . . . 65
3.2 Notations utilisées dans ce chapitre. . . . . 71
3.3 Normalisations pour les poids DFR. . . . . 78
3.4 Taille du vocabulaire employé pour chaque ensemble de données. . . . . 81
3.5 Nombre de requêtes utilisées pour chaque ensemble de données. . . . . 82
3.6 Résultats des expériences sur les *stop-lists*. . . . . 82
3.7 Pondérations locales d'un terme $t_i$ dans le document $d_j$. . . . . 85
3.8 Pondérations globales pour un terme $t_i$. . . . . 85
3.9 Modèles aléatoires testés et leurs approximations. . . . . 86
3.10 Modèles de divergence . . . . . 86
3.11 Amélioration maximale des performances par rapport au poids de référence $l_1g_0$, en utilisant la distance $L_1$. . . . . 91
3.12 Comparaison entre hapax et mots visuels tels que $\overline{\mathrm{tf}_i} = 1$. . . . . 92

4.1 Performances en classification en fonction des axes considérés (lissage de Katz, $n = 3$). . . . . 111
4.2 Performances en classification avec ou sans sélection des mots visuels (lissage de Katz). . . . . 112
4.3 Performances en classification en fonction de la longueur $n$ des $n$-grammes (lissage de Katz). . . . . 113
4.4 Performances en classification des différents lissages (données d'apprentissage). 113
4.5 Performances en classification des différents lissages (données de test). . . . 114
4.6 Temps d'exécution moyen en secondes pour Caltech-6. . . . . 116

5.1 Données statistiques du corpus de presse. . . . . 120
5.2 $\tau$ de Kendall modifié $\tau^{(1)}$ moyen (et écarts-types) sur les listes tronquées à $k$ éléments, pour chaque descripteur testé. . . . . 126
5.3 Classes et catégories d'entités nommées prises en charge par NEMESIS. . . . 128

A.1 Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Caltech6. . . . . 143
A.2 Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Caltech-6. . . . . 144
A.3 Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Caltech-101. . . . . 145
A.4 Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Caltech-101. . . . . 146

A.5 Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Kentucky. . . . . 147
A.6 Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Kentucky. . . . . 148
A.7 Performances des pondérations locales et globales avec la distance $L_1$ sur le corpus Oxford. . . . . 149
A.8 Performances des pondérations locales et globales avec la distance $L_2$ sur le corpus Oxford. . . . . 150
A.9 Performance des mesures de similarité DFR sur le corpus Caltech-6. . . . . 151
A.10 Performance des mesures de similarité DFR sur le corpus Kentucky. . . . . 152
A.11 Performance des mesures de similarité DFR sur le corpus Caltech-101. . . . 153
A.12 Performance des mesures de similarité DFR sur le corpus Oxford. . . . . 154
A.13 Performance des pondérations DFR avec la distance $L_1$ sur le corpus Caltech-6. . . . . 155
A.14 Performance des pondérations DFR avec la distance $L_2$ sur le corpus Caltech-6. . . . . 156
A.15 Performance des pondérations DFR avec la distance $L_1$ sur le corpus Caltech-101. . . . . 157
A.16 Performance des pondérations DFR avec la distance $L_2$ sur le corpus Caltech-101. . . . . 158
A.17 Performance des pondérations DFR avec la distance $L_1$ sur le corpus Kentucky. 159
A.18 Performance des pondérations DFR avec la distance $L_2$ sur le corpus Kentucky. 160
A.19 Performance des pondérations DFR avec la distance $L_1$ sur le corpus Oxford. 161
A.20 Performance des pondérations DFR avec la distance $L_2$ sur le corpus Oxford. 162

B.1 Temps d'exécution moyen par image (en secondes) des prétraitements et du détecteur. . . . . 171

# Liste des Algorithmes

1 Algorithme d'élimination des régions redondantes. . . . . 109
2 Algorithme de détection de logos. . . . . 164

## Résumé

Bien que s'inscrivant dans un cadre global de recherche d'information (RI) classique, l'indexation d'image ne tire que peu parti des nombreux travaux existants en RI textuelle et en traitement automatique des langues (TAL). Nous identifions deux niveaux auxquels de tels travaux peuvent s'intégrer aux systèmes d'indexation d'images.

Le premier niveau est celui de la description du contenu visuel des images. Pour y intégrer des techniques de TAL, nous adoptons la description des images par mots visuels proposée par Sivic et Zisserman. Cette représentation soulève deux problématiques similaires aux problématiques classiques de la RI textuelle : le choix des termes d'indexation les plus pertinents pour décrire les documents et la prise en compte des relations entre ces termes. Pour répondre à la première de ces problématiques nous proposons une étude des stop-lists et des pondérations dans le cadre de l'indexation d'images. Cette étude montre que, contrairement au cas des textes, il n'existe pas de pondération optimale pour tous types de requêtes, et que la pondération doit être choisie en fonction de la requête. Pour la seconde, nous utilisons des modèles de langues, outil classique du TAL que nous adaptons au cas des images, pour dépasser l'hypothèse d'indépendance des termes dans un cadre de classification d'images. Nos expérimentations montrent que prendre en compte des relations géométriques entre mots visuels permet d'améliorer les performances des systèmes.

Le second niveau étudié est l'indexation sémantique des images : il est possible d'utiliser des méthodes de TAL sur des textes accompagnant les images pour obtenir des descriptions textuelles de celles-ci. Dans un premier temps, nous montrons que les descripteurs classiques d'images ne permettent pas d'obtenir des systèmes d'annotation d'images efficaces. Puis nous proposons une méthode d'annotation qui contourne cet écueil en se basant sur des descripteurs textuels et visuels de haut-niveau : nous extrayons des textes des entités nommées, que nous mettons en relation avec des concepts visuels détectés dans les images afin d'annoter celles-ci. Nous validons notre approche sur un corpus réel et de grande taille composé d'articles de presse.



## Abstract

Although it is globally in line with traditional information retrieval (IR), image indexing makes poor use of the existing work about textual IR and natural language processing (NLP). We identify two levels where such work could become integrated to image indexing systems.

The first level is the description of the visual content of images. To integrate NLP at this level, we adopt a visual word-based representation of images, as proposed by Sivic and Zisserman. This representation raises two issues that are classical in textual IR: choosing relevant index terms and taking into account the relations between index terms. We address the first issue by studying stop-lists and weighting schemes in the context of image indexing. Our experiments show that there is no optimal weighting scheme in the general case, and that it should be chosen in keeping with the query. Then, we address the second issue by adapting language models to images, to go beyond the term independence hypothesis. Our experiments show that, in the context of image classification, taking account of spatial relations between visual words can improve the systems' performances.

The second level where we integrate NLP to image indexing is semantic image indexing: we can use NLP techniques on texts coming with images to extract a textual description of these images. We first show that standard image descriptors are not suited to image annotation, then we propose an image annotation scheme that avoid this problem by using high-level textual and visual concepts: we extract named entities from texts and associate them with visual concepts that we detect in the images. We validate our approach on a real-world and large-scale news corpus.