

UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté des Sciences

Département d'Informatique

MEMOIRE DE MAGISTER

Option : Systèmes d'Informations et de Connaissances

INDEXATION TEXTUELLE SEMANTIQUE POUR UN MOTEUR DE RECHERCHE D'IMAGES SUR LE WEB

Par

Mlle BAKALEM Mahdia

Devant le jury composé de :

A. GUESSOUM	Professeur, USDB	Président
W. HIDOUCI	Maitre de conférences A, ESI (ex. INI)	Examineur
F. SOUAMI	Maitre de conférences A, USTHB	Examineur
S. OUKID	Maitre de conférences A, USDB	Promoteur
N. BENBLIDIA	Maitre de conférences A, USDB	Co-Promoteur

**La soutenance aura lieu le Mercredi 29 Septembre 2010 à 9^h, à la
bibliothèque de la faculté de l'ingénieur (PV10)**

Blida, Sep 2010

RESUME

La recherche d'images sur le web s'effectue à travers des moteurs de recherche d'images qui permettent de retrouver rapidement et efficacement des images parmi une grande masse d'informations disponible sur le Web.

Le problème qui se pose dans la recherche d'images est la difficulté d'associer une sémantique à une image, car les images sont nommées d'une manière générique (img001.jpg) et, généralement, les moteurs de recherche indexent le texte associé à l'image dans les pages Web. Ce texte ne correspond pas à l'image.

Afin d'améliorer la recherche des images sur le Web, nous effectuons l'indexation textuelle sémantique des images appelée aussi annotation qui permet d'attribuer des mots clés à l'image correspondant au sens véhiculé. Afin d'obtenir une annotation efficace, une stratégie consiste à corréliser les informations textuelles et visuelles.

Dans ce travail, nous proposons un système d'indexation textuelle sémantique (ou auto-annotation) des images qui utilise un corpus pour l'apprentissage à travers un nouvel algorithme, **AnnotB-LSA**. Cet algorithme permet d'annoter les blobs de segments visuels d'images similaires afin d'annoter les images elles-mêmes. L'algorithme intègre le modèle d'analyse sémantique latente LSA afin d'extraire les relations sémantiques latentes entre les mots clé et minimiser l'ambiguïté entre les annotations de corpus. Une nouvelle image pourra être annotée par affectation aux blobs annotés par l'algorithme.

Notre système est validé par un benchmark fourni par Corel. En comparant les annotations d'images obtenues par le système avec celles de Corel, nous avons obtenu des résultats encourageants.

Mots clés : annotation d'images, AnnotB-LSA, LSA, recherche d'images sur le web, segmentation.

ABSTRACT

Web image retrieval is made through images search engines that can find quickly and effectively images among a lot of information available on the Web.

Semantics of an image is a big problem in the Web image retrieval, because the images are named in a generic way (img001.jpg) and, generally, the search engines index the text associated to the image of the Web pages. This text doesn't correspond to them.

The image semantic textual indexing called Image annotation is an effective technology for improving the Web image retrieval. Indeed, it she allows assigning semantics to an image, by attributing to the images of keywords corresponding to the senses conveyed by these images. To improve the automatic image annotation (AIA), a strategy consists in correlating the textual and visual information of the images.

In this optics, we propose an image semantic textual indexing system (image auto-annotation system), that use a corpus, based on a new algorithm, AnnotB-LSA, that correlates between the blobs of visual segments of images similar and the annotations of images corpus. In the goal to extract the latent semantic relations in the textual space of key words and to minimize the ambiguousness between the annotations of images corpus, the algorithm integrates the model of latent semantic analysis LSA. A new image can, then, be annotated by affecting it to the blobs annotated by the algorithm.

Our system is validated by a Corel benchmark. While comparing the annotations gotten by the system with those of the Corel corpus, the results are encouraging.

Keywords: AnnotB-LSA, LSA, Web image retrieval, image annotation.

الملخص

يتم البحث عن الصور على شبكة الإنترنت عن طريق محركات بحث الصور التي تسمح بإيجاد الصور من مجموعة كبيرة من المعلومات المتوفرة على الشبكة بسرعة وكفاءة.

المشكلة التي تطرح نفسها في البحث هي صعوبة ربط دلالة بالصورة، وذلك لأن تسمية الصور تتم بشكل عام (img001.jpg)، وعموماً، فإن محركات البحث تفهرس النص المرتبط بالصورة في صفحات الويب. هذا النص لا يتطابق مع الصورة.

لتحسين البحث عن الصور على شبكة الإنترنت، نقوم بالفهرسة النصية الدلالية للصور و تسمى أيضاً الحاشية (تعليق على نص) الذي يسمح لك بتعيين كلمات رئيسة للصور مرتبطة بالمعنى المنقول. للحصول على شرح فعال هناك استراتيجيات لربط المعلومات النصية والبصرية.

في هذا العمل، نقترح وضع نظام الفهرسة النصية الدلالية (أو الحاشية الذاتية) للصور باستخدام مجموعة كاملة Corpus من أجل التعلم من خلال خوارزمية جديدة، AnnotB-LSA (الشرح ب LSA). هذه الخوارزمية تسمح بتعليم نقط المقاطع البصرية من الصور المماثلة لتعليم هذه الصور نفسها. الخوارزمية تدمج نموذج التحليل الدلالي الكامن LSA لاستخراج العلاقات الدلالية الكامنة بين الكلمات الرئيسية والحد من الغموض بين حواشي الملف. صورة جديدة يمكن تعليمها عن طريق تعيين النقط المشروحة من قبل الخوارزمية.

تحقق من صحة نظامنا بواسطة معيار قدمه كوريل (COREL). بمقارنة شروح للصور التي حصل عليها النظام مع تلك لكوريل، لقد تحصلنا على نتائج مشجعة.

كلمات المفتاح:

التعليق على الصور، AnnotB_LSA، التحليل الدلالي الكامن (ت د ك LSA)، بحث الصور على شبكة الإنترنت، تجزئة.

REMERCIEMENTS

Tout d'abord, je remercie mon bon DIEU tout puissant d'avoir guidé mes pas vers les portes du savoir tout en illuminant mon chemin, et de m'avoir accordé la volonté, la foi et la force, secret de l'achèvement de mon travail dans de bonnes conditions.

Je tiens à remercier vivement ma gratitude à Madame N. BENBLIDIA pour avoir encadré et dirigé mes recherches. Je la remercie pour toute la confiance qu'elle a su me porter, et pour la patience et la gentillesse dont elle a fait preuve à mon égard.

Je tiens à remercier très sincèrement Madame S. OUKID m'avoir toujours fait confiance, et avoir guidé la conduite de la thèse. Je la remercie pour sa patience et sa confiance.

Je tiens à exprimer ma reconnaissance aux membres de jury pour avoir accepté de juger mon travail.

Je tiens à remercier de tout cœur Monsieur D. E. MENACER qui m'a toujours redonné le moral et la confiance si indispensables dans les périodes difficiles. Je le remercie plus particulièrement pour ses conseils et remarques constructives qui m'ont permis d'améliorer grandement la qualité de mon travail et pour l'attention qu'il a portée à la lecture de ce mémoire.

Mes remerciements du fond du cœur s'adressent également à mes très chers parents, qui n'ont eu de cesse de me soutenir et de croire en moi pendant mes études.

J'exprime toute ma profonde gratitude à mon unique sœur Djawida et à mes frères Ismail, Abdou, Nasro et Mohamed sans oublier mes belles sœurs et Je tiens également à remercier mes neveux et nièces surtout Bibicha qui ont

supporté avec moi les bons et les mauvais moments et qui m'ont donné la force d'aller au bout de cette aventure.

Je tiens à adresser mes remerciements toute ma famille, à mes collègues doctorants, à mes amies Amina, Nadjiba, Samira et Assia et un grand merci à toutes les personnes qui m'ont aidé de près ou de loin.

DEDICACE

*A Mes chers **Parents** que Dieu les garde,*

A Mes frères et sœurs,

A Mes neveux et nièces,

*Et spécialement à **H***

Je dédie ce mémoire.

TABLE DES MATIERES

RESUME	1
REMERCIEMENTS	4
DEDICACE	6
TABLE DES MATIERES	7
LISTE DES FIGURES	13
LISTE DES TABLEAUX	16
LISTE DES EQUATIONS	17
INTRODUCTION	18
1. Introduction	18
2. Problématique générale	18
3. Objectif du mémoire	19
4. Organisation du mémoire	20
CHAPITRE 1 : LA RECHERCHE D'INFORMATION	21
1.1. Introduction	21
1.2. Historique sur la recherche d'informations (RI)	21
1.4. Système de Recherche d'Information (SRI)	22
1.4.1. Documents	23
1.4.2. Représentation des informations ou Indexation	23
1.4.2.1. Indexation manuelle	23
1.4.2.2. Indexation automatique	24
1.4.2.3. Indexation semi-automatique	26

1.4.3. Représentation physique des informations	26
1.4.4. Appariement document-requête	28
1.4.5. Restitution des résultats de recherche	28
1.5. Processus de RI	30
1.6. Modèles de Recherche d'Informations	31
1.6.1. Modèle booléen	32
1.6.2. Modèles vectoriel	33
1.6.3. Modèles probabiliste	34
1.7. Évaluation des SRI	35
1.8. Recherche d'information sur le Web	36
1.8.1. Outils de recherche sur le Web	37
1.8.2. Fonctionnement d'un moteur de recherche	37
1.8.3. Architecture du moteur de recherche	38
1.8.3.1. Architecture centralisée	38
1.8.3.2. Architecture décentralisée	38
1.8.3.3. Architecture moderne	39
1.9. Conclusion	40
CHAPITRE 2 : LA RECHERCHE D'IMAGES SUR LE WEB	42
2.1. Introduction	42
2.2. Définitions d'Image	42
2.3. Descripteurs visuels d'image	44
2.3.1 La couleur	44
2.3.1.1. L'espace RGB	45
2.3.1.2. L'espace XYZ	46
2.3.1.3. L'espace $L^*u^*v^*$	46
2.3.1.4. L'espace $L^*a^*b^*$	46
2.3.1.5. L'espace $L^*C^*h^*$	47

2.3.1.6. L'espace HSV (Hue-Saturation-Value)	47
2.3.2. La texture	48
2.3.2.1. La matrice de cooccurrence	48
2.3.2.2. Transformée de Fourier	49
2.3.2.3. Les ondelettes	49
2.3.3. La forme	50
2.3.3.1. Les descripteurs basés région	50
2.3.3.2. Les descripteurs basés frontière	50
2.4. Segmentation	51
2.5. Classification	51
2.5.1. Les méthodes monodimensionnelles	51
2.5.2. Les méthodes multidimensionnelles	52
2.5.2.1 Classification supervisée	52
2.5.2.2. Classification non supervisée (clustering)	52
2.6. Indexation d'images	53
2.6.1. Indexation textuelle d'image	53
2.6.1.1. Indexation textuelle manuelle	54
2.6.1.2. Indexation textuelle automatique	54
2.6.2. Indexation d'image par le contenu visuel	55
2.7. Recherche d'images	56
2.7.1. Recherche basée sur le contenu visuel	56
2.7.1.1. Modèle de requêtes	56
2.7.1.2. Modèle de correspondance	57
2.7.2. Recherche textuelle basée sur les mots clés	58
2.7.3. Recherche textuelle visuelle	58
2.8. Systèmes de recherche d'images	58
2.8.1. Systèmes basés sur le contenu visuel : Approche signal	59

2.8.2. Systèmes textuels : Approches symboliques basées sur les mots clés -	61
2.8.3. Systèmes textuels basés sur le contenu visuel : Approches hybrides ---	61
2.9. Conclusion -----	63
CHAPITRE 3 : ANNOTATION D'IMAGES -----	64
3.1. Introduction -----	64
3.2. Annotation d'images-----	65
3.2.1. Annotation manuelle -----	65
3.2.2. Annotation automatique-----	65
3.2.3. Annotation semi-automatique -----	65
3.3. Modèles d'annotation -----	65
3.3.1. Indexation multimédia par analyse de la sémantique latente-----	66
3.3.1.1. Latente Semantic Analysis : LSA-----	66
3.3.1.2. Probabiliste LSA : PLSA-----	69
3.4. Survol de quelques travaux sur l'annotation d'image-----	70
3.4.1. Annotation dans l'espace latent -----	70
3.4.1.1. Espace latent linéaire-----	70
3.4.1.2. Espace latent non linéaire-----	71
3.4.2. Annotation avec ontologie et web sémantique -----	72
3.4.3. Annotation avec les techniques de datamining et de fouilles d'images (Image mining)-----	72
3.4.4. Annotation avec les algorithmes génétiques-----	73
3.4.5. Annotation avec les réseaux bayésiens -----	74
3.4.6. Annotation avec SVM (séparateurs à vaste marges)-----	75
3.4.7. Annotation avec le modèle probabiliste-----	76
3.4.8. Annotation avec un réseau de neurones-----	76
3.4.9. Annotation avec une segmentation multiple-----	77
3.4.10. Annotation avec la fusion de région et d'image-----	77
3.5. Conclusion -----	78

CHAPITRE 4 : SYSTEME D'AUTO ANNOTATION DES IMAGES SUR LE WEB	79
4.1. Introduction	79
4.2. Système d'annotation automatique proposé	80
4.2.1. Apprentissage	81
4.2.1.1. Constructions des blobs	82
4.2.1.2. Auto-annotation des blobs : Algorithme AnnotB-LSA	85
4.2.1.3. Base de données d'apprentissage	91
4.2.2. Auto annotation d'une nouvelle image	94
4.3. Intégration du système d'auto-annotation d'images dans un moteur de recherche	98
4.3.1. Système de collecte des images	99
4.3.1.1. Acquisition des images depuis le web	99
4.3.1.2. Annotation des images collectées	100
4.3.2. Système de recherche des images annotées	101
4.3.2.1. Phase d'interrogation	101
4.3.2.2. Phase de recherche	102
4.3.2.3. Phase de restitution	102
4.3.3. Aperçu du moteur de recherche images avec le système d'auto-annotation	103
4.4. Conclusion	104
CHAPITRE 5 : MISE EN ŒUVRE DU SYSEME ET APPLICATION AU CORPUS COREL	105
5.1. Introduction	105
5.2. Description du corpus Corel	105
5.3. Environnement du système d'auto-annotation d'images	106
5.4. Architecture logicielle du système d'auto-annotation d'images	108
5.4.1. Phase d'apprentissage	109
5.4.1.1. Construction des blobs	109

5.4.1.2. Auto-annotation de blobs	114
5.4.1.3. Base de données d'apprentissage	115
5.4.2. Phase de traitement d'une nouvelle image	115
5.4.2.1. Base de données d'images annotées	119
5.5. Tests et validation	120
5.6. Discussion	125
5.7. Conclusion	126
CONCLUSION	127
REFERENCES	129
LEXIQUE	142

LISTE DES FIGURES

Figure 1 : Présentation du système de recherche d'information.....	22
Figure 2 : Fichier inverse	27
Figure 3: Processus de la recherche d'informations	31
Figure 4: Modèle de la recherche d'information	32
Figure 5: Rappel et précision [67]	36
Figure 6: Architecture du moteur de recherche Altavista [53]	38
Figure 7: Architecture du système Harvest [53].....	39
Figure 8: Architecture du moteur de recherche Google [53].....	39
Figure 9: L'espace RGB	45
Figure 10: Diagramme de chromaticité xy	46
Figure 11: Représentation de l'espace HSV.....	47
Figure 12: Le système de recherche d'images par le contenu [32].....	60
Figure 13: Le système de recherche d'images combinant l'aspect visuel et textuel [32]	62
Figure 14 : Présentation des phases du système d'Auto-annotation proposé	81
Figure 15 : Diagramme d'activité «Apprentissage»	82
Figure 16 : Schéma d'apprentissage : Construction de blobs.....	83
Figure 17 : Segmentation d'une image	84
Figure 188: Extraction de caractéristiques visuelles.....	84
Figure 19: Diagramme d'activité de la catégorisation des régions d'images	85
Figure 20: Diagramme d'activité de l'algorithme AnnotB-LSA	87
Figure 21: Schéma de déroulement de l'algorithme AnnotB-LSA	91

Figure 22 : Diagramme de classes du système d'auto annotation d'images.....	92
Figure 23: Diagramme d'activité global d'annotation automatique d'images.....	95
Figure 24: Segmentation d'une nouvelle image	95
Figure 25 : Diagramme d'activité d'affectation.....	96
Figure 26: Diagramme d'activité d'annotation.....	97
Figure 27: Auto-annotation d'une nouvelle image.....	98
Figure 28: Moteur de recherche d'images classique	99
Figure 29 : Présentation du système de Collecte d'images.....	100
Figure 30 : Présentation du système d'auto-annotation d'images.....	100
Figure 31 : Diagramme de classe de la base d'images annotées	101
Figure 32 : Présentation du système de recherche d'images	101
Figure 33: Aperçu d'une restitution des résultats	103
Figure 34: Présentation du processus complet du moteur de recherche images	103
Figure 36 : Diagramme de collaboration du processus complet du moteur de recherche images	104
Figure 36 : Environnement du système	107
Figure 37 : Architecture logicielle du système d'auto-annotation d'images	109
Figure 38 : Module de segmentation	110
Figure 39 : Segmentation d'image	111
Figure 40 : Module extraction	111
Figure 41 : Valeurs des paramètres texturaux pour les régions considérées.....	112
Figure 42 : Module catégorisation.....	113
Figure 43 : Représentation des régions du blob 222	113
Figure 44 : Module auto-annotation blobs	114
Figure 45 : Résultat de l'annotation de 16 blobs	114
Figure 46 : Schéma de la base de données apprentissage	115
Figure 48 : Segmentation de la nouvelle image	116
Figure 49 : Résultats de la caractérisation texturale des différentes régions	117

Figure 50 : Module affectation	117
Figure 51 : Affectation des régions aux blobs	118
Figure 52 : Module auto-annotation d'une nouvelle image	118
Figure 53 : Annotation de la nouvelle image	119
Figure 54 : Base d'images annotées	119
Figure 55 : Images à annoter	120
Figure 56 : Résultats de notre Système	123
Figure 57 : Exemple de comparaison des résultats	125
Figure 58 : Graphe de comparaison des annotations	126

LISTE DES TABLEAUX

Tableau 1 : Mesures de similarité utilisées dans le modèle vectoriel	34
Tableau 2: Différentes représentations d'image	44
Tableau 3: Formules des paramètres de texture utilisés	49
Tableau 4: Description des classes	93
Tableau 5 : Données de comparaison des annotations	125

LISTE DES EQUATIONS

Equation 1 : Similarité modèle booléen.....	34
Equation 2 : Similarité modèle probabiliste.....	33
Equation 3 : Précision	36
Equation 4 : Rappel.....	36
Equation 5 : Calcul SVD.....	67
Equation 6 : Calculs PLSA.....	69
Equation 7 : Règle bayes.....	69
Equation 8 : Probabilité jointe.....	69
Equation 9 : Probabilité occurrence d'un mot.....	70
Equation 10 : Fréquence d'occurrence	88
Equation 11 : Similarité de cosinus.....	89

INTRODUCTION

1. Introduction

Le développement des nouvelles technologies de l'information et de la communication et l'essor du web ont entraîné un accroissement très important de l'information en termes de diversité et d'hétérogénéité. La recherche de cette information est devenue ainsi un domaine d'activité très important.

La recherche d'informations s'effectue généralement grâce aux systèmes de recherche d'information (SRI). Ces derniers ont pour but de retrouver rapidement et efficacement l'information dont l'utilisateur a besoin parmi la grande masse d'informations disponible. Retrouver l'information pertinente (qui répond aux besoins de l'utilisateur) demeure de plus en plus difficile à trouver.

Sur le web, les moteurs de recherches (par ex. Google) sont les principaux outils de recherche d'information. Ils scrutent le web et maintiennent une base d'index sur des documents de nature variée. Les documents image sont parmi les documents les plus recherchés sur le web. En effet, le développement de domaines tels que la médecine, les images satellite et la cartographie ...etc a provoqué l'accroissement d'images numérisées sur le Web. Par leur nature particulière, les images ont besoin d'outils de recherche plus élaborés. Ces outils sont regroupés dans les systèmes de recherche d'images (SRIM).

2. Problématique générale

Le problème majeur qui se pose dans la recherche d'images sur le web est comment rechercher rapidement et efficacement des images pertinentes parmi une grande masse d'images disponibles sur le web?

Lors de la recherche des images sur le web, le problème de la sémantique d'une image se pose car les images sont nommées de manière générique (par exemple img001.jpg), elles ne comportent aucun sens.

La recherche d'images sur le web est généralement basée sur le texte des pages web contenant les images. Mais souvent ce texte ne correspond pas au mieux aux images de la page ce qui donne de mauvais résultats de recherche.

Une approche prometteuse consiste à associer des mots clés à l'image, reflétant le sens véhiculé par celle-ci, cette méthode s'appelle l'annotation.

L'annotation peut se faire d'une manière manuelle ou bien automatique. L'annotation manuelle est facile et subjective mais elle est difficile pour un grand ensemble d'images. Par contre l'annotation automatique peut annoter un grand ensemble d'images mais le passage de l'aspect visuel vers l'aspect sémantique reste difficile.

Une approche pour annoter automatiquement les images consiste à trouver une corrélation entre l'aspect visuel et l'aspect sémantique. Plusieurs travaux ont été effectués dans ce sens mais cela reste à l'état expérimental.

Le problème que nous abordons dans ce mémoire est le suivant : Comment annoter automatiquement les images du web en faisant la corrélation entre l'aspect visuel et sémantique?

3. Objectif du mémoire

Ce travail vise à faciliter la recherche des images en faisant l'indexation textuelle sémantique des images pour un moteur de recherche sur le web.

Notre objectif principal est de proposer un algorithme, que nous avons appelé AnnotB-LSA, en utilisant la corrélation entre l'aspect visuel et l'aspect sémantique, qui permet d'annoter les blobs (un blob regroupe les régions visuellement similaires) afin d'annoter les images.

Nos objectifs secondaires sont :

- La réalisation d'un système d'auto annotation d'images basé sur l'algorithme AnnotB-LSA.
- L'intégration du système d'auto-annotation d'images dans un moteur de recherche d'images sur le web.
- L'évaluation du système d'auto-annotation d'images.

4. Organisation du mémoire

Le présent mémoire est organisé en deux parties :

- I. La première présente un état de l'art sur le domaine de recherche : la recherche d'informations, la recherche d'images sur le web et l'annotation des images qui sont présentées dans trois chapitres :
 1. Le chapitre 1 définit des notions de base de la recherche d'informations, le processus de la recherche, les mesures d'évaluation des systèmes de recherche d'informations. Une attention particulière est donnée à la recherche d'informations sur le web.
 2. Le chapitre 2 présente l'image et ses caractéristiques de base, la segmentation et la classification. Le processus de la recherche d'images est également présenté.
 3. Le chapitre 3 concerne l'annotation d'images avec ses différents types, ainsi que les différents travaux de recherche sur le domaine.
- II. La deuxième partie consiste à présenter notre contribution, elle se compose de deux chapitres :
 1. Le chapitre 4 présente notre contribution sous forme d'un système d'auto-annotation d'images, pour un moteur de recherche sur le web. Ce système est basé sur l'annotation des blobs.
 2. Le chapitre 5 montre la réalisation d'un prototype afin de tester les performances du système. Une évaluation expérimentale du prototype sera réalisée afin de valider le système.

CHAPITRE 1

LA RECHERCHE D'INFORMATION

1.1. Introduction

Aujourd'hui, le problème du manque d'informations n'existe plus, bien au contraire, l'information se caractérise par sa diversité, son hétérogénéité et sa croissance grâce au développement des nouvelles technologies de l'information et de la communication ainsi qu'à l'essor du Web (plusieurs centaines de téraoctets d'informations actuellement).

Le problème qui persiste est la difficulté de trouver l'information pertinente qui répond aux besoins de l'utilisateur. En d'autres termes, comment permettre à l'utilisateur de trouver la bonne information, celle qui le satisfait.

Dans ce chapitre, nous introduisons la notion de recherche d'informations, en mettant l'accent sur le processus de recherche. Nous présentons également les mesures d'évaluation des systèmes de recherche d'informations. Finalement, une attention particulière est donnée à la recherche d'informations sur le Web.

1.2. Historique sur la recherche d'informations (RI)

Le nom « recherche d'information » (information retrieval) fut donné, pour la première fois, en 1948 par Calvin N. Mooers quand il travaillait sur son mémoire de maîtrise [1].

Dans les années 1950, de petites expérimentations furent réalisées, utilisant de petites collections de documents. Plus tard, d'autres expérimentations plus larges furent menées dont :

1 Projet **Cranfield** (dirigé par Cyril Cleverdon, 1957-1967) [2],

- 2 Projet **MEDLARS** – MEDical Literature Analysis and Retrieval System (F. Wilfrid Lancaster, complete en 1968) [3],
- 3 **SMART** (Gerard Salton, 1^{ière} version 1961-1965) [4],
- 4 Projet **STAIRS** - SStorage And Information Retrieval System (Blair et Maron) [5],
- 5 **TREC** - Text REtrieval Conference [6].

Les années 1980 ont été influencées par le développement de l'intelligence artificielle que les chercheurs ont tenté alors d'intégrer à la RI.

Les années 1990, (surtout à partir de 1995), marquent le début de l'Internet. Cela a eu pour effet de propulser la RI en avant scène de beaucoup d'applications. On traite maintenant plus souvent des documents multimédia. Cependant, les techniques de base utilisées dans les moteurs de recherche sur le Web restent identiques.

1.4. Système de Recherche d'Information (SRI)

La Recherche d'Information [49, 38] est une branche de l'informatique qui s'intéresse à l'**acquisition**, l'**organisation**, le **stockage**, la **recherche** et la **sélection** d'information

Un Système de Recherche d'Information (SRI) est un programme (ou ensemble de programmes) informatique qui a pour but de sélectionner des **informations pertinentes** répondant à des **besoins utilisateurs**.

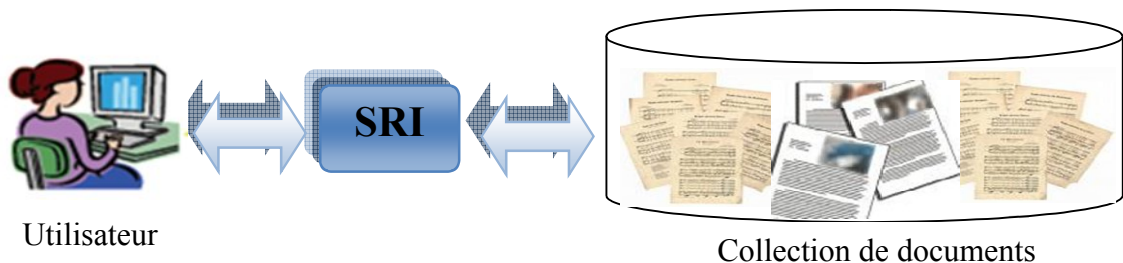


Figure 1 : Présentation du système de recherche d'information

La plupart des SRI sont en réalité des *systèmes de recherche de documents* (SRD) : le système de recherche d'information ne fournit pas directement l'information cherchée mais des documents devant la contenir.

1.4.1. Documents

Le document, élément essentiel dans les systèmes de recherche d'informations, existe selon différentes caractéristiques :

- Formes : le document peut être texte, images, sons, vidéo, graphiques, etc...
- Propriétés : le document peut être structuré, non structuré ou semi structuré (XML) (HTML),
- Hétérogénéité : le document peut être de différents langages (multilingues), medias (multimédia) et structures.

1.4.2. Représentation des informations ou Indexation

L'indexation, processus très important dans la recherche d'information, consiste à analyser chaque document de la collection afin de créer un ensemble d'éléments «clés» qui caractérisent le contenu d'un document.

L'indexation est basée sur :

- Un langage contrôlé : lexique (liste de mots), thesaurus (liste de mots clés avec relation sémantiques entre eux), ontologie (liste de concepts avec relations entre eux) ou réseau sémantique.
- Un langage libre : éléments pris directement des documents.

L'indexation peut être manuelle, automatique et semi-automatique.

1.4.2.1. Indexation manuelle

Elle s'effectue par un spécialiste du domaine ou par un documentaliste (indexeur), qui fait le choix des mots basés sur un vocabulaire contrôlé. Cette indexation dépend du savoir-faire de l'indexeur, deux indexeurs différents pouvant donner deux termes différents pour caractériser un document.

L'indexation manuelle assure une bonne satisfaction de l'utilisateur; une meilleure pertinence dans les réponses apportées par le SRI. Elle permet aussi une recherche par concept (thème, sujet), et assure la classification des documents. Cependant, elle est difficile à maintenir et prend beaucoup de temps pour sa réalisation.

1.4.2.2. Indexation automatique

L'indexation automatique a pour but de transformer automatiquement des documents en substituts capables de représenter le contenu de ces documents [25] [50].

Cette indexation est faite à travers plusieurs étapes : l'extraction automatique des mots des documents, l'élimination des mots vides, la lemmatisation (radicalisation ou normalisation), le repérage de groupes de mots, la pondération des mots.

- **L'analyse lexicale**

Dans cette étape, le document est vu comme un ensemble de termes. Un terme est une unité lexicale ou un radical [15], une suite de caractères séparés par un blanc ou un signe de ponctuation, les caractères spéciaux,... L'analyse lexicale est un processus qui permet d'extraire les termes du document. Ce processus dépend de la langue.

- **L'élimination des mots vides**

L'étape précédente produit un ensemble de termes. Il faut maintenant éliminer les mots vides (par exemple les pronoms personnels, les prépositions, les mots athématiques tels que contenir, appartenir, etc...) et de garder que les termes significatifs.

On distingue deux techniques pour éliminer les mots vides :

- l'utilisation d'une liste de mots vides (ou anti-dictionnaire),
- la suppression des mots trop fréquents et inutiles mais pas les abréviations (USA, A (vitamine)...).

- **La lemmatisation**

La lemmatisation est un processus morphologique permettant de regrouper les variantes d'un mot. Dans cette étape, une substitution des termes par leur racine (lemme) est utilisée pour résoudre le problème d'indexer tous les variantes d'un mot (par exemple le mot économie regroupe les mots économiquement, économétrie, économétrique, économiste...etc).

Frakes et Baeza-yates distinguent cinq types stratégiques de lemmatisation [16] : la table de consultation (dictionnaire), l'élimination des affixes (algorithme de Porter [31]), troncature, variétés de successeurs, méthode des n-gramme [7].

Il est clair que ce processus permet de réduire le nombre de termes indexés, mais il supprime la sémantique originale du mot. Ce problème est résolu par C.J. Crouch [14], qui a proposé une méthode qui se déroule en deux étapes :

- la lemmatisation des mots, dans une première recherche effectuée.
- un ré ordonnancement des documents est effectué, en se basant sur les vrais termes (termes non lemmatisés) de la requête.

- **La pondération des termes**

On désigne la mesure d'importance d'un terme dans un document par pondération des termes. Cette dernière a pour objectif de trouver les termes qui représentent au mieux le contenu d'un document.

Il existe différentes approches de pondération des termes:

- Une approche basée sur la fréquence d'occurrences : La fréquence d'apparition d'un terme dans une collection est un bon indicateur de l'importance de ce terme. Ainsi, la loi de Zipf [48] permet de classer les mots dans l'ordre décroissant de leur fréquence, et de leur donner un numéro de rang (1, 2, ...), tel que: Rang * Fréquence \approx constante.
- Une autre approche est basée sur la valeur discriminatoire d'un terme, proposée par Salton en 1975. Cette approche consiste à comparer la

similarité entre documents avec et sans le terme, l'objectif étant de mesurer le degré de présence d'un terme dans plusieurs documents.

- L'approche la plus répandue est basée sur $tf \cdot idf$: elle combine les pondérations locales et globales d'un terme, tels que :
 - Tf (Term Frequency) est la fréquence d'un terme dans un document (ou pondération locale) et $\log (tf)$ mesure la présence ou l'absence du terme.
 - Idf (Inverse of Document Frequency) est la fréquence d'un terme dans une collection de documents (ou pondération globale). Elle est formulée comme suit : $idf = \log (N/n_i)$, où N est le nombre total de documents dans la collection et n_i est le nombre de documents contenant le terme.

La mesure $tf \cdot idf$ donne une bonne approximation de l'importance du terme dans le document.

Pour la normalisation, Robertson [35] et Singhal et al. [40] ont introduit la taille des documents à la formule de pondération.

1.4.2.3. Indexation semi-automatique

L'indexation semi-automatique [24, 9], est la combinaison des deux autres types d'indexation, où le choix final revient aux indexeurs qui interviennent souvent pour choisir d'autres termes significatifs basés sur un thésaurus ou une base terminologique.

1.4.3. Représentation physique des informations

Afin de mémoriser les informations sélectionnées lors du processus d'indexation, des structures de stockage sont nécessaires. Les moyens de stockage les plus utilisées sont : les fichiers inverses ("inverted files"), les tableaux de suffixes ("suffix- arrays") et les fichiers de signatures ("signature files").

Les fichiers inverses sont composés de deux éléments principaux [67]:

- Le vocabulaire, qui est l'ensemble de tous les mots significatifs du texte.

- Liste d'occurrences ou posting, qui est la liste de toutes les positions dans le texte pour lesquelles un mot apparaît.

Les fichiers inverses utilisent des blocs d'adressage [8] pour résoudre le problème d'espace disque. Un récapitulatif sur les fichiers inverses est illustré dans la figure 2.

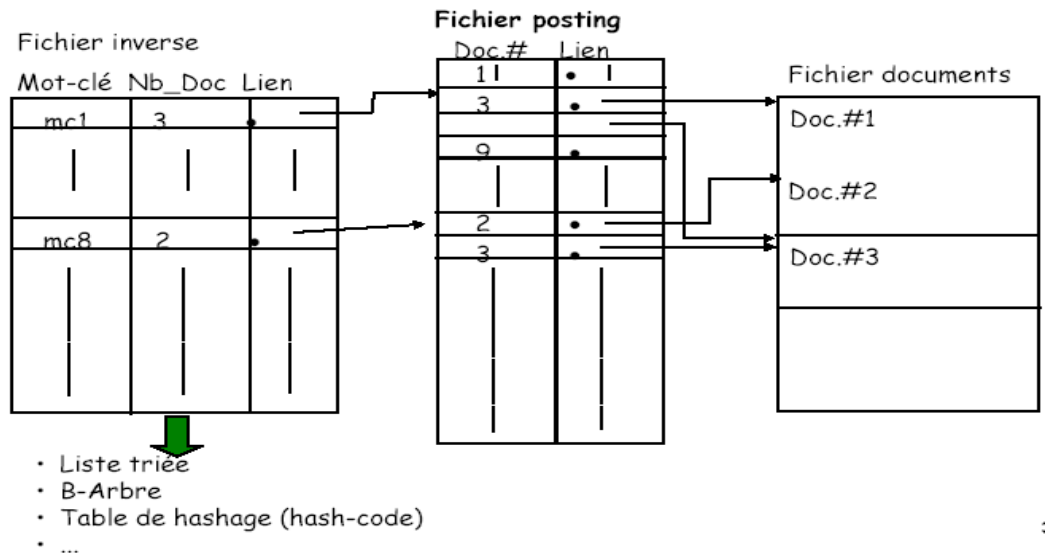


Figure 2 : Fichier inverse

Les tableaux de suffixes sont plus rapides pour des recherches de phrases et des requêtes moins communes. Ils sont cependant plus difficiles à construire et à maintenir [51].

Enfin, les fichiers de signatures sont basés sur le concept de hashing. Ils étaient très populaires dans les années 1980, mais sont aujourd'hui beaucoup moins utilisés [51].

Les structures couramment utilisées pour l'indexation sont [8] :

- Les arbres de recherche digitaux (ou "trees")
- Les arbres de recherche binaires
- Les B-arbres
- Les tableaux triés ("sorted arrays")
- Les tables de hashage

1.4.4. Appariement document-requête

La pertinence d'un document vis-à-vis de la requête est en fin de compte une valeur de comparaison entre le document et la requête. Cette valeur est calculée par une fonction ou une probabilité de similarité notée **RSV** (Q,d) (Retrieval Status Value) où Q est la requête et d le document.

La fonction d'appariement document-requête permet d'ordonner les documents renvoyés à l'utilisateur. Elle est très étroitement liée aux opérations d'indexation et de pondération des termes de la requête et des documents de la collection. L'appariement peut être exact ou approché :

- **Appariement exact** : la requête spécifie de manière précise les critères recherchés et les documents sélectionnés (non ordonnés) respectent exactement la requête.
- **Appariement approché** : la requête décrit les critères recherchés dans un document; la liste des documents sélectionnés est ordonnés selon un degré de pertinence (similarité/ probabilité) vis-à-vis de la requête.

1.4.5. Restitution des résultats de recherche

Afin d'exploiter efficacement les résultats, la restitution permet d'organiser et de visualiser les résultats de la recherche.

- **Organisation du résultat** : regrouper les documents similaires et les placer dans des espaces de résultat selon leurs degrés de similarité. Google utilise des algorithmes (PageRank, VisuelRank...) afin d'organiser les résultats. D'autres systèmes de recherche d'informations (moteur de recherche) organisent les résultats par clusters, par exemple Vivisimo et Grokkeri [119] [122].
- **Visualisation des résultats** : définir une interface dans laquelle les résultats de recherche sont présentés d'une manière adaptée à l'organisation.

Il existe plusieurs types de visualisation [53] :

- A. Présentation par liste de réponses et classification : Utilisée dans Google [120]. Dans ce type, il existe plusieurs présentations :
- i. Présentation par liste triée selon la fréquence d'apparition des mots de la requête,
 - ii. Présentation par regroupement des résultats en catégories,
 - iii. Présentation des résultats sous forme de texte.
- B. Visualisation graphique des résultats : ce type de visualisation peut présenter les résultats en deux, trois ou quatre dimensions.
- i. Représentation en deux dimensions : elle permet de présenter les résultats multidimensionnels obtenus en deux dimensions afin de faciliter la lecture des résultats. Exemple : représentation des catégories de document en deux dimensions par carte Sammon [54] et par visualisation radiale.
 - ii. Représentation hiérarchique : les résultats de recherche peuvent être présentés sous forme hiérarchique tree-map. Le système Carey [55] est un exemple de cette présentation.
 - iii. Représentation en trois dimensions : les résultats de recherche sont présentés en trois dimensions afin de capitaliser le nombre d'informations affichées. Ces représentations sont classées en deux approches [56] :
 1. Approches cartographiques :
 - a) géographique (terrains, villes...) : VxInsight [57] City CF News [58, 59],
 - b) non géographique : NIVE [60].
 2. Approches non cartographiques :
 - a) placement selon 3 axes : Easy-DOR [61], VR-VIBE [62], DocCube [63],
 - b) hiérarchiques : InformationCube [65], Cat-a-Cone [64],
 - c) graphes,
 - d) autres : WebBook et WebForager [66].

1.5. Processus de RI

Le processus de recherche d'information répond aux intentions des humains à travers un système de recherche d'information, en présence d'une masse d'information. Ces intentions sont devenues des besoins, qui sont devenus des requêtes après leur formulation par l'utilisateur.

Ce processus doit retourner à l'utilisateur des réponses pertinentes par rapport à son besoin (requête formulée). La pertinence est subjective, c'est à dire dépendante de l'utilisateur, donc difficile à automatiser. Les systèmes de recherche d'informations font correspondre la pertinence système avec la pertinence utilisateur. Dans ces réponses, les documents sont ordonnés selon leurs pertinences.

Le but du processus de recherche d'information est de chercher une réponse appropriée au besoin de l'être humain dans la masse de données disponible.

Ce processus de recherche est couramment appelé Processus en U de Recherche d'Information [38, 10], il est composé de :

- L'indexation des documents et des requêtes,
- L'appariement requête-document, qui permet de comparer la requête et le document,
- La reformulation de la requête, qui permet de faire correspondre au mieux la pertinence utilisateur et la pertinence du système. Elle se fait en deux étapes principales : trouver des termes d'extension à la requête initiale, et re pondérer les termes dans la nouvelle requête.

La figure 3 schématise le processus de la recherche d'informations

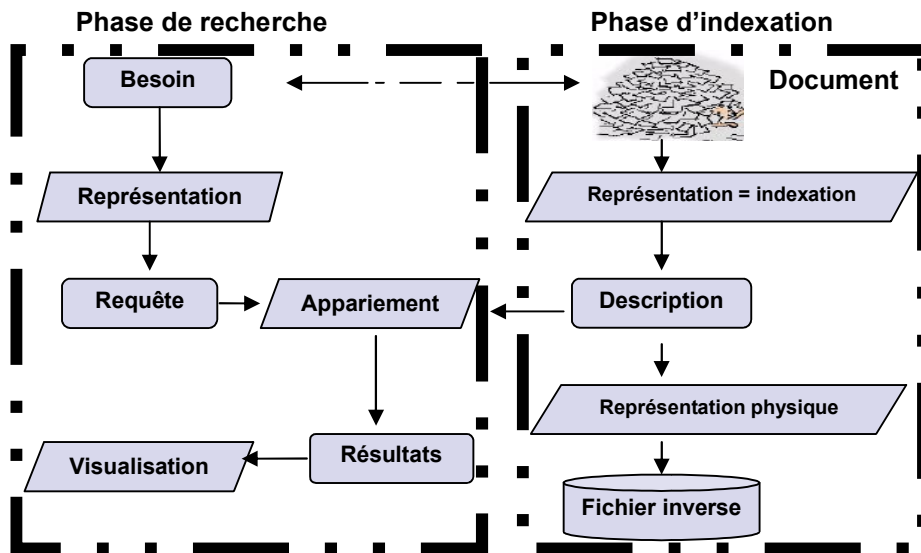


Figure 3: Processus de la recherche d'informations

1.6. Modèles de Recherche d'Informations

Le rôle d'un modèle est de fournir une formalisation du processus de recherche d'information. Les modèles de RI se distinguent par le principe d'appariement (*matching*); l'appariement peut être exact ou approché (*Exact /Best matching*).

Un modèle de RI doit contenir, au moins, les éléments suivants :

- Un langage de description de documents (Indexation),
- Un langage de requêtes,
- Un algorithme d'appariement, qui permet de construire une relation entre requête et indexation,
- Une fonction de pertinence (ou de « ranking »), qui est la mesure de similarité.

La figure 4 présente l'ensemble des modèles de la recherche d'information.

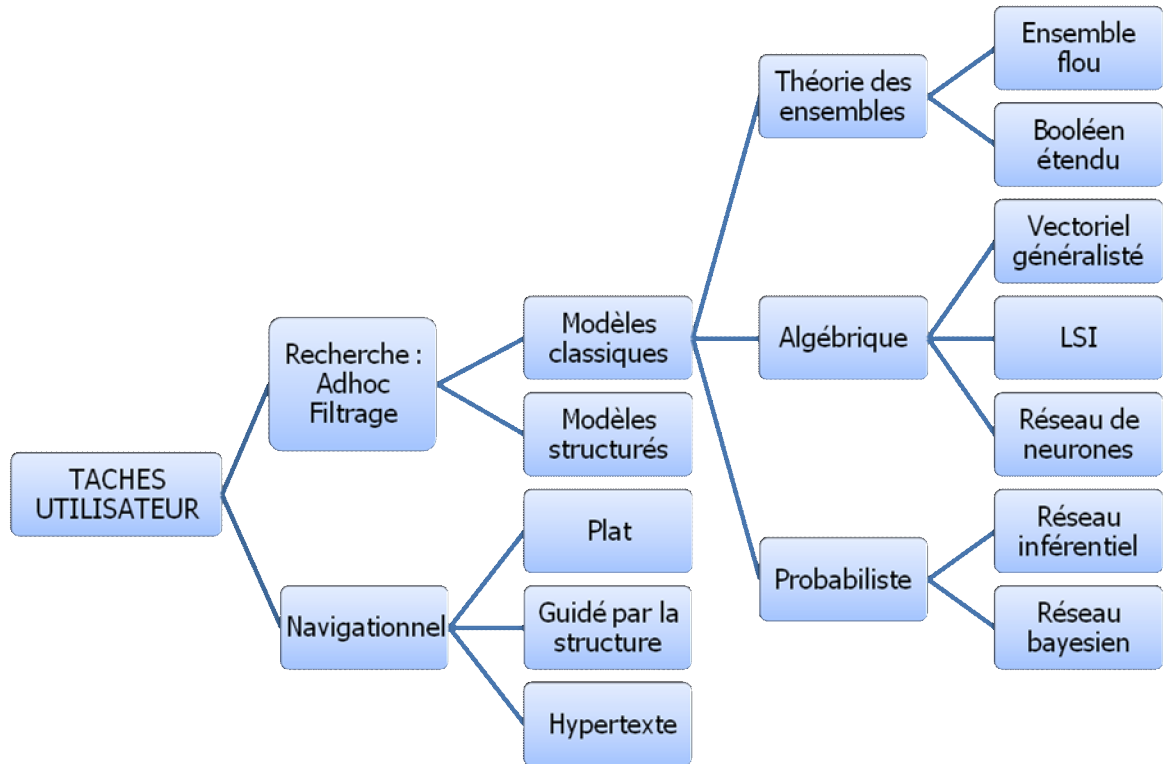


Figure 4: Modèle de la recherche d'information

Il existe trois grands modèles classiques : les modèles booléens reposant sur la théorie des ensembles, les modèles vectoriels reposant sur l'algèbre, et les modèles probabilistes reposant sur les probabilités. Ces modèles ont été décrits dans de nombreux ouvrages sur la Recherche d'Information [38, 8].

Dans ce qui suit, nous désignons par :

- $D = \{d_1, d_2, \dots, d_n\}$ l'ensemble des documents de la collection,
- $T = \{t_1, t_2, \dots, t_n\}$ l'ensemble de termes dans la collection,
- q est la requête définie.

1.6.1. Modèle booléen

Il est basé sur la théorie des ensembles [36], et constitue le premier modèle simple qui s'est imposé dans le domaine de la recherche d'information.

Dans ce modèle, le document est représenté par un ensemble de termes, la requête est une combinaison de termes (ensemble de mots) et des opérateurs

booléens (AND (^), OR (v), NOT (¬)). L'appariement exact est basé sur la présence ou l'absence des termes de la requête dans les documents.

La similarité entre un document et une requête est définie par :

$$RSV(q, d) = \begin{cases} 1 & \text{si } d \in \text{à l'ensemble décrit par la requête} \\ 0 & \text{sinon} \end{cases} \quad (1)$$

Le modèle booléen affirme que le document est soit pertinent, soit non pertinent; il répond donc exactement à la requête qui a été formulée.

L'inconvénient de ce modèle est que la sélection d'un document est basée sur une décision binaire, donc il n'y a pas d'ordre pour les documents sélectionnés. Le nombre de ces documents peut être considérable. Notons aussi la difficulté de formulation de la requête pour les utilisateurs.

Il existe d'autres modèles basés sur la théorie des ensembles tels que : le modèle booléen étendu [37] et le modèle flou [67].

1.6.2. Modèles vectoriel

Le modèle vectoriel repose sur l'algèbre. Les documents et les requêtes sont exprimés comme des vecteurs de poids dans un espace vectoriel de termes tel que $d_j = \{w_{1j}, w_{2j}, \dots, w_{ij}\}$, où w_{ij} est le poids des termes dans le document (0 ou 1), et $q = \{w_{1q}, w_{2q}, \dots, w_{iq}\}$, où w_{iq} est le poids des termes dans la requête.

Dans ce modèle, la pertinence d'un document vis-à-vis d'une requête est définie par des mesures de distance dans un espace vectoriel.

Le tableau 1 présente les mesures de similarité utilisées dans le modèle vectoriel.

Mesures	Similarité
Le produit scalaire	$RSV(\vec{q}, \vec{d}) = \sum (q_i \times d_i)$
Coefficient de Dice	$RSV(\vec{q}, \vec{d}) = \frac{2 \times \sum (q_i \times d_i)}{\sum q_i^2 + \sum d_i^2}$

Mesure de Cosinus	$RSV(\vec{q}, \vec{d}) = \frac{\sum(q_i \times d_i)}{\sqrt{\sum q_i^2 \times \sum d_i^2}}$
Mesure de Jaccard	$RSV(\vec{q}, \vec{d}) = \frac{\sum(q_i \times d_i)}{\sum q_i^2 + \sum d_i^2 - \sum(q_i \times d_i)}$

Tableau 1 : Mesures de similarité utilisées dans le modèle vectoriel

Ce modèle permet de renvoyer des documents qui répondent approximativement à la requête. Grâce à la mesure de similarité, les documents sont ordonnés selon leur degré de pertinence vis à vis de la requête. La pondération améliore les résultats de recherche et augmente les performances du système, mais la représentation vectorielle suppose l'indépendance entre les termes.

Le modèle vectoriel généralisé [46], le modèle d'Indexation Sémantique Latent (LSI) [17] et le modèle connexionniste sont également des modèles algébriques.

1.6.3. Modèles probabiliste

Le modèle probabiliste repose sur les probabilités. Au début des années 1960, Maron et Kuhns [25] ont proposé le premier modèle probabiliste qui est basé sur la probabilité de pertinence d'un document vis-à-vis d'une requête pour le classement des documents. Nous désignons par :

- $P(\text{Pert}/d, q)$: la probabilité de pertinence de document vis à vis de la requête.
- $P(\text{Non-Pert}/d, q)$: la probabilité de non pertinence de document vis à vis de la requête.

Le calcul de la probabilité de pertinence d'un document D [44], notée $P(\text{Pert}/D)$ est donné par la formule suivante :

$$p(\text{Pert}/D) = \sum_{i=1}^t \log \frac{P(w_{ij}/\text{Pert})}{P(w_{ij}/\text{NonPert})} \quad (2)$$

Dans le modèle probabiliste, il existe plusieurs applications, dont le modèle 2-Poisson développé par Robertson et Walker [35] et le moteur de recherche Okapi [34, 45].

Il existe d'autres modèles probabilistes tels que : les réseaux bayésiens [30] et le modèle de langage [68].

1.7. Évaluation des SRI

L'évaluation des performances des systèmes est basée sur deux critères : l'efficacité et l'efficience (par rapport au temps d'exécution et l'espace mémoire utilisé). Ces dernières sont utilisées seulement pour les systèmes qui assurent parfaitement une fonction précise. Dans les systèmes de recherche d'information, d'autres mesures de performance ont été introduites pour évaluer l'efficacité des systèmes, parmi lesquelles on trouve la facilité d'utilisation du système ou la présentation des résultats [13]. Mais la mesure la plus significative est l'évaluation de la satisfaction des utilisateurs relativement à leurs demandes d'information. Cette mesure montre la capacité d'un système à sélectionner des documents pertinents.

La qualité du SRI est mesurée par l'écart entre les réponses fournies par le système et les documents jugés pertinents par des spécialistes du domaine. Pour pouvoir comparer les Systèmes de Recherche d'Informations, on utilise, sur les mêmes bases documentaires, des mesures qui sont la **précision** et le **rappel** (voir figure 5). Désignons par :

- $|A|$ le nombre de documents renvoyés par un système pour une requête donnée.
- $|R|$ le nombre de documents pertinents dans la collection pour cette requête.
- $|Ra|$ le nombre de documents pertinents renvoyés par le système

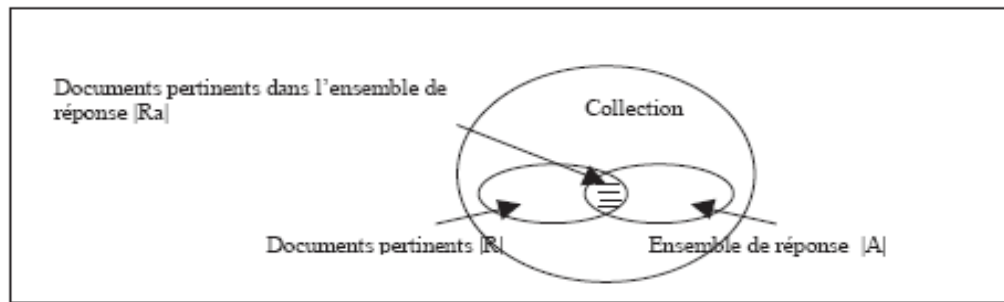


Figure 5: Rappel et précision [67]

La précision [67] mesure la proportion de documents pertinents relativement à l'ensemble des documents restitués par le système. Elle est exprimée par :

$$precision = \frac{|Ra|}{|A|} \quad (3)$$

Le rappel [67] mesure la proportion de documents pertinents restitués par le système relativement à l'ensemble des documents pertinents contenus dans la base documentaire. Il est exprimé par :

$$rappel = \frac{|Ra|}{|R|} \quad (4)$$

Pour pouvoir examiner les résultats efficacement, il faut calculer les deux mesures (taux de rappel, taux de précision) pour chaque document restitué [67].

La précision est une fonction décroissante du rappel. Notons que les valeurs de précision et de rappel sont dans l'intervalle [0..1].

Nous pouvons calculer aussi la précision moyenne et exacte. Il existe d'autres mesures d'évaluation des systèmes de recherche d'informations : mesures alternatives (*Moyenne harmonique* [39] et *Mesure E* [44]), mesures orientées utilisateur [21].

1.8. Recherche d'information sur le Web

L'Internet est un réseau informatique à l'échelle mondiale constitué de multiples réseaux nationaux, universitaires, régionaux et privés. Il permet l'accès à une grande masse d'informations, de différents types et formats.

Le Web (extension du World Wide Web, WWW), créé en 1989, est le service le plus important d'Internet. Le Web désigne le réseau maillé formé par Internet. Ce réseau relie les serveurs Web (ou serveurs http) à travers un protocole de transmission de données appelé HTTP (HyperText Transfer Protocol).

1.8.1. Outils de recherche sur le Web

Il existe deux grandes catégories d'outils de recherche sur Internet: les annuaires et les moteurs de recherche

1. **Annuaire** : Un annuaire cherche à qualifier l'information en s'appuyant sur une sélection manuelle. Les métas moteurs sont une sous-classe des annuaires qui transmettent la requête d'un internaute simultanément à plusieurs moteurs de recherche.
2. **Moteurs de recherche** : Un moteur de recherche est un logiciel qui parcourt le Web à l'aide de robots logiciels (Spiders, crawlers ou agents), à intervalles réguliers et de façon automatique, pour découvrir de nouvelles adresses URL (Uniform Resource Locator), en suivant les liens hypertextes des pages afin de les indexer dans une base de données.

1.8.2. Fonctionnement d'un moteur de recherche

Le fonctionnement d'un moteur de recherche se décompose en trois phases principales [52] : la collecte automatique de données, l'indexation et la recherche.

1. **Phase de collecte de données** : Le robot logiciel explore le Web, en parcourant de lien en lien les milliards des pages du Web et recense les adresses URL des sites visités.
2. **Phase d'indexation** : L'analyse des pages précédemment collectées consiste à en extraire les mots considérés comme significatifs. Le contenu des pages (ensemble de mots clés) et leur adresse URL sont stockés dans un index (base de données).
3. **Phase de recherche** : Le système interroge l'index et présente les résultats par ordre de pertinence calculée par un algorithme [53] (Hits, PageRank, SALSA, PHITS, HubRank, VisuelRank...).

1.8.3. Architecture du moteur de recherche

Il existe trois types d'architecture : centralisée, décentralisée et moderne.

1.8.3.1. Architecture centralisée

L'architecture centralisée est l'architecture de première génération des systèmes de recherches comme Altavista [121]. La figure 6 montre le fonctionnement du système.

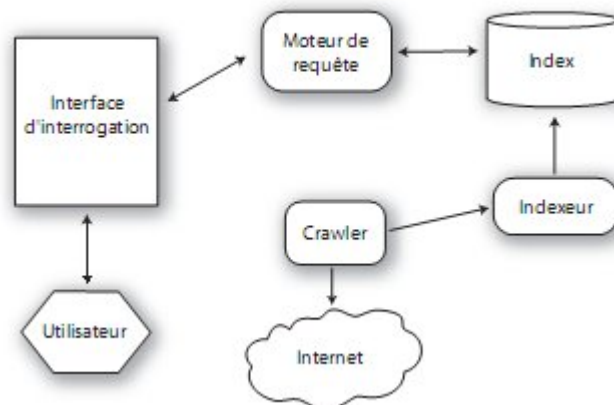


Figure 6: Architecture du moteur de recherche Altavista [53]

Il s'agit d'une architecture très simple qui repose sur le mécanisme indexeur-crawler. L'utilisateur interroge l'index par sa demande afin de retrouver les documents qui répondent à la requête.

1.8.3.2. Architecture décentralisée

La NASA, l'Académie des Sciences Nationales des Etats-Unis d'Amérique utilise l'architecture décentralisée dans son système Harverst, qui se compose d'un récolteur et d'un broker. Le récolteur collecte et extrait les informations d'indexation (textes, images) d'une manière périodique depuis le Web. Le broker, quant à lui, fournit le mécanisme d'indexation et l'interface d'interrogation sur les données amassées par le récolteur. La figure 7 montre l'architecture décentralisée du système.

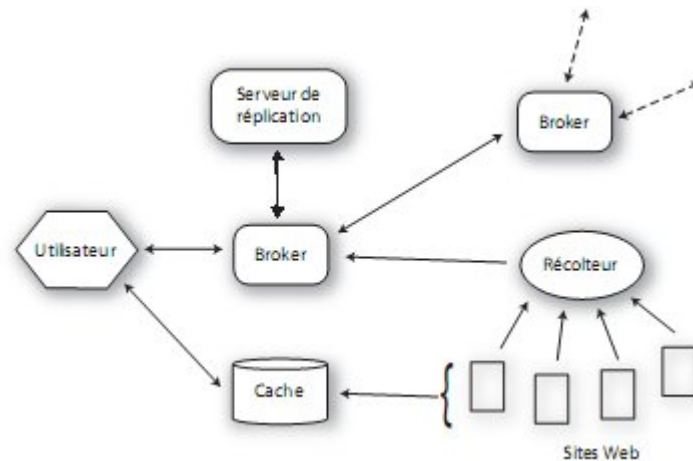


Figure 7: Architecture du système Harvest [53]

L'Architecture décentralisée a le même mécanisme indexeur-crawler que l'architecture centralisée. Cependant, les brokers et les récolteurs sont spécialisés dans un domaine précis

Les brokers et les récolteurs peuvent être configurés de manière à répartir le besoin en ressources sur un ou plusieurs domaines particuliers.

1.8.3.3. Architecture moderne

Le moteur de recherche Google [120] repose sur l'architecture moderne, qui se compose de plusieurs éléments comme le montre la figure 8 :

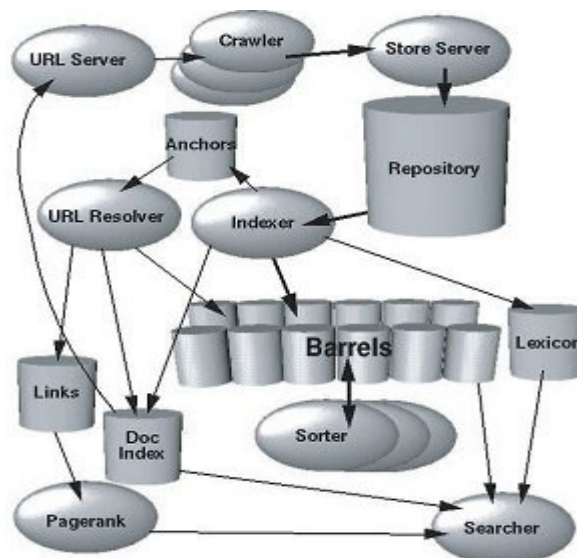


Figure 8: Architecture du moteur de recherche Google [53]

Cette architecture est distribuée sur un grand nombre de machines standard et permet d'obtenir des temps de réponse inférieurs à la seconde pour une requête donnée. C'est actuellement le moteur de recherche le plus utilisé; il a montré, depuis son lancement en 1998 une très grande adaptabilité à l'évolution de la taille d'Internet jusqu'à indexer en juin 2004 près de 4,3 milliards de pages Web.

1.9. Conclusion

Nous avons vu dans ce chapitre le processus de la recherche d'information et les mesures d'évaluation des systèmes de recherche d'informations ainsi que la recherche d'informations sur le web.

La Recherche d'Information est un domaine en pleine expansion et les demandes d'information (utilisateurs) n'arrêtent pas d'augmenter. Les problèmes de la recherche d'information se résument dans quelques points :

1. La formulation de la requête : le besoin d'information de l'utilisateur est parfois vague et toujours subjectif, de plus une formulation peut être comprise de plusieurs façons par des personnes différentes. Cela est dû à la perte d'information entre la réalité du besoin d'information et son expression,
2. L'insuffisance des outils de recherche d'information comme les moteurs de recherche qui sont déterministes, et qui ne sont pas désignés à s'adapter aux préférences personnelles.

Les résultats obtenus par l'utilisateur posent aussi des problèmes :

1. Les résultats retournés à l'utilisateur ne sont pas pertinents car la pertinence d'un document pour une requête est une notion variable et très complexe à définir (elle dépend du besoin de l'utilisateur, besoin difficile à définir).
2. La restitution des résultats par les moteurs de recherche n'offre pas toujours à l'utilisateur la possibilité d'exploiter et de visualiser efficacement les informations retournées,

3. La surcharge d'information (volume d'information retourné) éloigne l'utilisateur des informations pertinentes,
4. L'ambiguïté : recevoir des résultats dans des contextes totalement différents à ses besoins. Par exemple les résultats de recherche du mot avocat peuvent représenter l'homme de droit ou bien le fruit.

Quelques propositions ont été faites afin de résoudre ces problèmes comme la reformulation de la requête, le filtrage d'information, l'intégration du profil utilisateur dans le processus de recherche d'information...

Il n'existe pas de système de recherche d'information parfait. L'évaluation d'un système de recherche dépasse les aspects habituels de performance informatique. Elle nécessite la mise en place de métriques particulières. L'intervention de juges humains reste nécessaire, mais ceux-ci sont subjectifs et soumis à des variations entre des individus différents d'une part, et pour un même individu d'autre part.

Dans le chapitre suivant, nous présenterons un type particulier de recherche d'information : la recherche d'images.

CHAPITRE 2

LA RECHERCHE D'IMAGES SUR LE WEB

2.1. Introduction

L'image, qui existe depuis plus de trente mille ans, constituait un moyen de communication pour l'être humain avant l'invention de l'écriture.

Avec l'essor des technologies de l'électronique, de l'informatique et de l'Internet, le nombre d'images n'arrête pas d'augmenter. D'après l'étude de Lawrence et Giles [69], il y a 180 millions d'images indexées sur le Web occupant environ 3To, et plus d'un million d'images numériques sont produites chaque jour [70].

L'interprétation d'image se fait selon deux aspects : visuel (syntaxique) et textuel (sémantique). L'aspect visuel concerne la couleur, la texture, la forme...; le textuel véhicule, quant à lui, le sens de l'image.

Les systèmes de recherche d'images (SRIm) sont des systèmes de recherche qui permettent de retrouver les images dont l'utilisateur a besoin. On distingue les systèmes qui se basent sur l'aspect visuel ou l'aspect textuel ou les deux à la fois.

Dans la suite, nous définissons l'image et ses caractéristiques de base ainsi que la segmentation et la classification. Puis nous détaillons les différentes étapes du processus de la recherche d'images et nous présentons quelques systèmes de recherche d'images existants.

2.2. Définitions d'Image





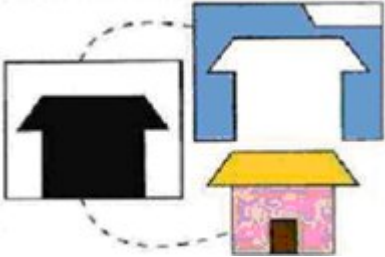
La définition du dictionnaire Larousse donne : «**image** : n.f. (lat. imago) [. . .] Représentation imprimée d'un sujet quelconque [. . .]».

L'image est un objet très riche en informations. On distingue deux niveaux d'informations :

- 1 Le niveau **syntaxique** (le contenu), dans lequel on trouve des informations de base sur l'image (couleur, texture, forme...).
- 2 Le niveau **sémantique**, qui désigne son interprétation (donne un sens à l'image). Cette interprétation est subjective et dépend du contexte social et sociétal de l'observateur.

Il existe différents formats d'image : JPEG, BMP, GIF, PNG...

Selon les niveaux d'abstraction auxquels une image peut être vue, on peut distinguer différentes représentations d'image (tableau 2).

Représentation d'une image	Exemple
Ensemble de pixels colorés	
Ensemble de régions, de contours, de lignes, de courbes, de coins...	
Ensemble d'objets avec des caractéristiques	
Ensemble d'objets en relation les uns avec les autres	
Ensemble d'objets composés en relation	

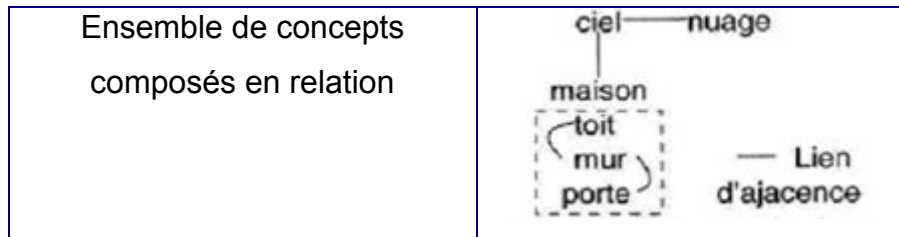


Tableau 2: Différentes représentations d'image

2.3. Descripteurs visuels d'image

Les attributs d'une image sont classés dans deux familles :

- 1 Les attributs globaux qui concernent l'image entière.
- 2 Les attributs locaux qui concernent une région (ou partie) d'une image.

Le choix des attributs dépend de l'image et du contexte. Il existe trois familles d'attributs: la couleur, la texture et la forme. Dans ce travail, nous ne présentons que les attributs jugés nécessaires pour notre application.

2.3.1 La couleur

La couleur est le descripteur visuel le plus employé. Elle est définie par sa longueur d'onde, ou par un mélange de longueurs d'ondes. La science qui s'occupe de l'aspect psychophysique de la couleur est la colorimétrie. Différentes définitions ont été données par différents auteurs, parmi lesquels :

- 1 Lozano définit la colorimétrie comme une science qui tente de qualifier une source lumineuse de manière absolue [71].
- 2 Gevers considère la colorimétrie comme la standardisation du triplet lumière objet-observateur [72].

Le physicien Maxwell a prouvé que toute couleur peut être obtenue par la synthèse de trois couleurs seulement : le rouge, le vert et le bleu.

La compagnie internationale de l'éclairage CIE a défini plusieurs espaces pour la représentation de la couleur. Ces espaces sont différenciés et classés selon l'uniformité de la perception.

L'uniformité de la perception dans un espace de représentation est vérifiée par les deux critères suivants :

- 1 La distance $d(c_1, c_2)$ entre les deux couleurs c_1 et c_2 est correcte, si et seulement si, la valeur issue de cette distance se rapproche de la différence perçue par l'œil humain,
- 2 La distance $d(c_i, c_1) = n \cdot d(c_i, c_2)$ est correcte, si et seulement si, l'œil humain perçoit la couleur c_1 n -fois plus éloignée de la couleur c_i que la couleur c_2 .

La distance utilisée est la distance euclidienne. Les différents espaces de représentation de la couleur sont :

2.3.1.1. L'espace RGB

En 1931, la CIE a défini l'espace RGB [73] comme suit: le R= rouge, le G= vert et le B=bleu. Cet espace est donc représenté par les trois primaires monochromatiques de couleurs dont les longueurs d'ondes sont respectivement 700.0 nm, 546.1 nm et 435.8 nm, et par le blanc de référence qui n'est pas unique. La représentation dans cet espace est faite par la synthèse soustractive.

Dans la figure 9 l'espace RGB est représenté par un cube dont les sommets sont les couleurs de base.

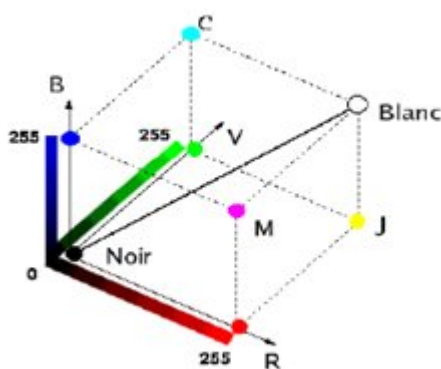


Figure 9: L'espace RGB

L'inconvénient majeur de cet espace est l'impossibilité de représenter toutes les couleurs par la superposition des trois spectres, à cause de l'existence d'une partie négative, qui peut poser des problèmes si on travaille en synthèse additive.

Cet espace est sensible aux changements d'illumination, et ne prend pas en compte la sensibilité de l'œil.

2.3.1.2. L'espace XYZ

Afin de tenir compte de la sensibilité de l'œil dans l'espace RGB, un espace est défini par trois primaires X, Y et Z dites virtuelles. Le passage à l'espace XYZ est effectué par une transformation linéaire sur l'espace RGB.

La figure 10 présente le diagramme de chromaticité xy qui permet de tenir compte de la sensibilité de l'œil aux différents stimuli.

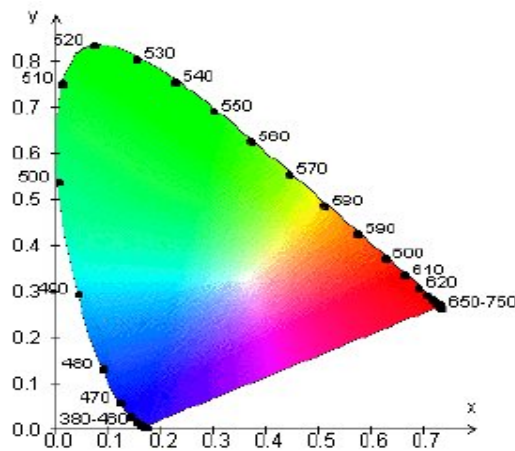


Figure 10: Diagramme de chromaticité xy

Cet espace n'est pas perceptuellement uniforme [74], car la distance entre deux couleurs du diagramme de chromaticité xy n'est pas perçue de la même façon que le système visuel humain (SVH).

2.3.1.3. L'espace L*u*v*

En 1976, L'espace L*u*v* devient un standard de la CIE [75], où la luminosité L représente la clarté, u l'opposition de couleurs vert-rouge et v l'opposition de couleurs bleu-jaune. L'espace L*u*v* est un espace perceptuellement uniforme.

2.3.1.4. L'espace L*a*b*

L'espace L*a*b* est un espace perceptuellement uniforme introduit par la CIE, en 1976, où la luminosité représente la clarté, et a*, b* représentent

respectivement l'opposition de couleurs vert-rouge et l'opposition de couleurs bleu-jaune. Les axes a^* et b^* sont orientés (vert pour $a^* < 0$ et $b^* = 0$, rouge pour $a^* > 0$ et $b^* = 0$, etc.).

Cet espace est obtenu par des relations non-linéaires à partir de l'espace XYZ. A cause de ces relations, il n'est pas possible de définir un diagramme de chromaticité.

2.3.1.5. L'espace $L^*C^*h^*$

L'espace $L^*C^*h^*$ désigne la luminance L^* , le chroma C^* et la teinte h^* . Cet espace est le résultat du passage des espaces $L^*a^*b^*$ et $L^*u^*v^*$ en coordonnées semi-polaires.

2.3.1.6. L'espace HSV (Hue-Saturation-Value)

Cet espace s'appelle aussi le cône hexagonal. Il représente la couleur selon trois entités : la teinte (H), la saturation (S) et l'intensité ou luminosité (V). La figure 11 présente le cône hexagonal de l'espace HSV

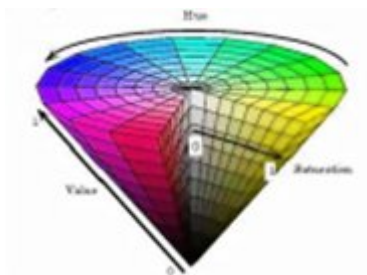


Figure 11: Représentation de l'espace HSV.

L'espace HSV simule le comportement visuel humain. Cet espace est perceptuel mais pas uniforme.

Il existe d'autres espaces de représentation de couleur tels que : l'espace CMY qui est dédié à l'impression couleur, l'espace d'Ohta...etc.

2.3.2. La texture

Il n'existe pas de définition précise de la texture. On peut dire que la texture est la répétition d'éléments de base construits à partir de pixels qui respectent un certain ordre. On distingue deux types de texture [76]:

- La première est **déterministe (ou régulière)** et fait référence à une répétition spatiale d'un motif de base dans différentes directions. Cette approche structurelle correspond à une vision **macroscopique** des textures. La peau des reptiles est un exemple de texture macroscopique dans lequel la primitive est l'écaille.
- La seconde est **probabiliste (ou aléatoire)** et cherche à caractériser l'aspect chaotique qui ne comprend ni motif localisable, ni fréquence de répétition principale. Cette approche est **microscopique** et l'herbe en est un excellent exemple.

L'analyse de texture, qui se fait par la matrice de cooccurrence ou la matrice de longueur de plage [76], permet d'obtenir des informations sur la texture macroscopique (calculs statistiques au niveau du gris). Le calcul des attributs de texture mixte s'effectue par l'analyse de Fourier et l'analyse en ondelettes.

La texture peut nous aider dans l'aspect sémantique, car elle différencie les parties de l'image dont les descripteurs de couleurs sont identiques.

2.3.2.1. La matrice de cooccurrence

La texture d'une image peut être interprétée comme la régularité d'apparition de couples de niveaux de gris selon une distance donnée dans l'image. La matrice de cooccurrence [77] contient les fréquences spatiales relatives d'apparition des niveaux de gris selon quatre directions : $\theta = 0$, $\theta = \pi/2$, $\theta = \pi/4$ et $\theta = 3\pi/4$. La matrice de cooccurrence est une matrice carrée $n * n$ où n est le nombre de niveaux de gris de l'image.

Une fois la matrice de cooccurrence calculée, différents attributs de texture peuvent être extraits comme : la variance, l'homogénéité locale, le contraste et l'entropie qui sont utilisés dans le cadre de ce travail (Tableau 3).

Paramètres	Formules
La variance	$VAR = \frac{1}{N} \sum_i \sum_j (p(i,j) - Moy)^2$ $Moy = \frac{1}{N} \sum_i \sum_j p(i,j)$
L'homogénéité locale	$HOML = \frac{1}{N} \sum_{i,j} \frac{1}{1 + (i-j)^2} p(i,j)$
Le contraste	$CONT = \frac{1}{N * (L-1)^2} \sum_{k=0}^{L-1} k^2 \sum_{ a-b =k} p(i,j)$
L'entropie	$ENT = -\frac{1}{N} \sum_i \sum_j p(i,j) \ln p(i,j)$

Tableau 3: Formules des paramètres de texture utilisés

2.3.2.2. Transformée de Fourier

La transformée de Fourier permet de transformer les coordonnées de l'image du domaine spatial en coordonnées du domaine fréquentiel de l'image.

Le module de la transformée de Fourier ou spectre de Fourier sont souvent utilisés pour éviter l'interprétation de la phase quand on travaille dans des plans complexes.

2.3.2.3. Les ondelettes

La transformée en ondelettes [78, 79] est un outil mathématique qui décompose un signal en fréquences en conservant une localisation spatiale. Le signal de départ est projeté sur un ensemble de fonctions de base qui varient en fréquence et en espace. Ces fonctions de base s'adaptent aux fréquences du signal à analyser. Cette transformation permet donc d'avoir une localisation en temps et en fréquence du signal analysé.

Par conséquent, on peut extraire différents types d'attributs et à différents niveaux de résolution, à travers la transformée en ondelettes.

2.3.3. La forme

Les attributs de la forme sont utilisés pour caractériser les objets dans les images. Ils ne sont utilisés qu'avec une image segmentée. La forme contient deux catégories : les descripteurs basés région et les descripteurs basés frontière.

2.3.3.1. Les descripteurs basés région

Le but de cette catégorie est de caractériser l'intégralité de la forme d'une région à travers les moments invariants [80] [81].

- **Les attributs géométriques de région**

Il existe différents types de forme que peuvent prendre les objets d'une scène. Les attributs géométriques de région permettent de les distinguer ces objets. À travers la segmentation de l'image en différentes régions, des calculs peuvent se faire sur les régions comme : la surface, le centre de masse, la longueur de contour, la compacité...

- **Les moments géométriques**

Le but des moments géométriques est de décrire une forme à l'aide de propriétés statistiques [82]. Hu a introduit sept invariants aux translations, rotations et changement d'échelle, appelés moments de Hu [80].

Les moments géométriques consomment du temps de calcul, mais ils sont simples à manipuler.

2.3.3.2. Les descripteurs basés frontière

Dans cette catégorie, on utilise les descripteurs de Fourier afin de caractériser les contours d'une forme [83] [84].

2.4. Segmentation

La segmentation d'une image est un processus qui découpe l'image en parties homogènes selon un critère déterminé: couleur, texture, niveau de gris, indice,...

Il existe différentes méthodes de segmentation classées en trois grandes approches :

- Globales : la segmentation est basée sur les pixels. Le seuillage et l'histogramme (segmentation par classification monodimensionnelle) sont des exemples de telles approches.
- Locales : la segmentation est basée soit sur les régions, soit sur les contours :
 - La segmentation basée sur les régions groupe les pixels semblables. Les méthodes de seuillage/classification, de division et fusion, d'agrégation, accroissement de régions sont des exemples de telles approches.
 - La segmentation basée sur les contours permet de rechercher les pixels dissemblables (contours entre zones hétérogènes). L'espace-échelle, modèles dérivatifs sont des exemples de telles approches.
- Hybrides : la segmentation combine les approches globales et locales. La méthode split and merge est un exemple de ces approches.

2.5. Classification

La classification permet de grouper (classifier) les individus en exploitant leur similarité. Les individus peuvent être : les pixels, une sous image de taille fixe ou une région [76].

Selon le nombre d'attributs considérés, les méthodes de classification peuvent être monodimensionnelles ou multidimensionnelles [76].

2.5.1. Les méthodes monodimensionnelles

Elles consistent à classifier les individus en fonction d'un seul attribut (niveau de gris). Basées sur l'analyse de l'histogramme, ces méthodes déterminent des seuils qui constitueront les limites des différentes classes. Il existe trois catégories

de méthodes de seuillage : seuillage global, seuillage local et seuillage dynamique [76].

2.5.2. Les méthodes multidimensionnelles

Elles exploitent plusieurs attributs afin de classer les individus. Nous présentons deux types de méthodes de classification multidimensionnelle : la classification supervisée et la classification non supervisée [76].

2.5.2.1 Classification supervisée

La classification supervisée (appelée classement ou classification inductive) a pour objectif « d'apprendre » par l'exemple. Elle cherche à expliquer et à prédire l'appartenance des individus à des classes connues a priori [86].

À partir d'une classification connue à priori d'un certain nombre d'individus, la classification supervisée permet de construire une fonction d'identification ou de discrimination pour les autres individus afin de découper l'espace de représentation en zones. A chaque zone est affectée une classe de la classification à priori. Les autres individus sont ensuite classés en fonction de leur position dans l'espace de représentation [76].

Parmi les méthodes classiques de classification supervisée, nous citons les K plus proche voisins, la classification bayésienne, les séparateurs à vaste marges...etc.

2.5.2.2. Classification non supervisée (clustering)

Elle consiste à découper l'espace de représentation en zones homogènes et distinctes selon un critère de ressemblance entre les individus, c'est à dire selon un critère de proximité dans l'espace des attributs [76]. Ces zones sont des classes (clusters).

Parmi les méthodes classiques de classification non supervisée, nous citons la classification ascendante hiérarchique, les cartes d'auto-organisatrices, K-moyennes...etc.

Dans le cadre de ce travail, nous utilisons la méthode classique K-moyennes.

La classification K-moyennes

Elle s'appelle aussi nuées dynamiques ou méthode de partitionnement autour des centres mobiles (en anglais K-means). Cette méthode est un outil de partitionnement des données non-hiérarchiques qui permet de répartir les données en K clusters homogènes.

Résumé de l'algorithme de classification k-moyennes :

1. *Initialiser k centres de clusters (aléatoirement),*
2. *Affecter chaque point d au cluster qui minimise l'inertie locale $\|d - \mu_{c_k}\|^2$,*
3. *Recalculer le centre de chaque cluster*
4. *Réitérer les étapes (2) et (3) jusqu'à ce que l'algorithme converge vers un minimum local.*

Le nombre de clusters K est déterminé par l'utilisateur à l'initialisation, plus le nombre de données est grand plus l'utilisateur aura des problèmes lors de la détermination du nombre de clusters.

2.6. Indexation d'images

L'étape la plus délicate de la recherche d'images est l'indexation des images à partir d'une collection d'images (qui peut être spécifique, hétérogène..). L'indexation est un processus qui sert à extraire des informations à partir du document image et à représenter ces informations dans une structure appelée index.

2.6.1. Indexation textuelle d'image

Dans cette approche, l'indexation des images est basée sur les mots clés, qui peuvent être des mots clés globaux ou locaux.

- Les mots clés globaux peuvent être associés aux images entières.
- Les mots clés locaux peuvent être associés aux régions des images.

2.6.1.1. Indexation textuelle manuelle

L'indexation textuelle manuelle d'images est un processus manuel, qui s'effectue par un iconographe. Ce dernier utilise un thésaurus afin d'extraire les catégories et les groupes de mots pour qu'il puisse les associer aux images dans le but de les classer et les indexer.

Dans ce processus, le problème majeur est le choix des termes par l'iconographe. Ce choix restant toujours subjectif, il dépend du savoir-faire de l'iconographe. En effet, une image peut être indexée de différentes manières selon le sens perçu par l'iconographe (polysémie de l'image).

Il existe deux types d'indexation d'image d'après Krauze [89] :

- Hard indexing (ofness) qui correspond à ce que voit l'indexeur dans l'image.
- Soft indexing (aboutness) qui correspond à ce que signifie l'image.

L'indexation textuelle manuelle d'images est une méthode robuste, qui permet d'associer le sens aux images. Mais devant un grand volume d'images, l'indexation devient difficile, presque impossible.

2.6.1.2. Indexation textuelle automatique

L'indexation textuelle automatique s'effectue sans intervention humaine. Elle consiste à associer des mots clés à une image. Deux approches existent : l'Indexation textuelle automatique à partir du texte associé à l'image et **l'Indexation textuelle automatique à partir du contenu visuel de l'image**, qu'on appelle aussi **annotation automatique** ou **auto-annotation**

2.6.1.2.1. Indexation textuelle automatique à partir du texte associé à l'image

Dans cette approche, l'indexation n'est pas possible sauf si les images sont associées à du texte. C'est le cas des images du Web, où l'indexation textuelle automatique peut s'effectuer à partir ([90][91][92]):

- des mots présents dans le titre de la page,

- des mots les plus fréquents dans la page,
- des mots pertinents de cette page,
- des URL avec le nom d'image. Cependant, les images sont généralement nommées d'une manière générique (img001.jpg) qui ne comporte aucun sens,
- des mots associés à l'attribut ALT de la balise IMG d'une image,
- du texte proche de l'image,
- ou de la fusion de toutes ces informations.

2.6.1.2.2. Indexation textuelle automatique à partir du contenu visuel de l'image

Cette approche, souvent appelée l'**auto-annotation** par le contenu, permet de combiner et de trouver des liens entre les informations textuelles et visuelles. On utilise des méthodes de classification où les images sont associées à des classes de mots ou d'apprentissage pour prédire des mots sur de nouvelles images.

2.6.2. Indexation d'image par le contenu visuel

Afin d'exploiter les informations contenues dans une image, une extraction des descripteurs visuels (attributs ou caractéristiques de bas niveau c'est-à-dire la couleur, la texture et la forme) est nécessaire. Ces descripteurs visuels sont extraites automatiquement [93], où:

- La couleur est représentée par des histogrammes dans des espaces de représentation de couleurs (RGB, HSV, Luv...).
- La texture est extraite par de nombreuses méthodes, parmi lesquelles les méthodes statistiques (la matrice de cooccurrence) [94].

L'ensemble des informations extraites d'une image est représenté par un vecteur descripteur (signature) dont les descriptions peuvent être globales (image entière) ou locales (région). Les régions sont des parties cohérentes de l'image selon une ou plusieurs caractéristiques de bas niveau (niveau signal) [95].

L'index décrit les caractéristiques globales de l'image entière. Les descripteurs de chaque région sont intégrés dans l'index de l'image.

La pondération dans les SRIm, comme dans les SRI, permet de montrer l'importance des attributs par rapport aux autres pour décrire l'image ou la région.

Notons que les valeurs qui se trouvent dans un index représentent seulement la quantité de ces attributs dans l'image, mais pas l'importance sémantique dans celle-ci.

2.7. Recherche d'images

Il existe trois types de recherche d'images.

2.7.1. Recherche basée sur le contenu visuel

Dans cette recherche, il y a deux étapes : la formulation d'un besoin (requête) et la mise en correspondance entre les représentations (vecteurs descripteurs) de la requête et d'images.

2.7.1.1. Modèle de requêtes

La formulation d'un besoin par une requête déclenche le processus de recherche d'images. A travers l'interface d'interrogation, l'utilisateur doit définir précisément ses besoins. Afin de faciliter la comparaison. Une analyse se fait sur la requête en extrayant les mêmes types de traits que ceux extraits lors de l'indexation.

Quelle que ce soit la manière de formuler la requête, les requêtes sont représentées de la même façon que les documents images.

1. Requête par spécification directe de trait

Le besoin utilisateur se formule par la spécification directe de caractéristiques visuelles des images recherchées ou par la combinaison pondérée des caractéristiques visuelles. Pour cela, il faut que l'interface soit compatible à ce type de requête.

L'inconvénient majeur de ce mode d'interrogation est que les caractéristiques spécifiées portent sur l'image entière.

2. Requête par l'exemple

Dans ce mode d'interrogation, l'utilisateur définit sa requête de différentes manières :

- Soit en donnant un exemple de ce qu'il recherche. L'exemple peut être une ou des images, une région, une composition logique de catégories de régions, une composition spatiale de régions,
- Soit en dessinant un croquis,
- Soit en précisant des rectangles, des ovales de couleurs et textures différentes,
- Soit en indiquant leur disposition et leur taille sur l'image.

L'interface d'interrogation doit disposer d'outils simples de dessin. Quand il s'agit d'images complexes, l'utilisateur doit avoir des compétences artistiques pour le dessin d'un croquis.

Ce mode d'interrogation pose trois problèmes :

- L'image par l'exemple n'est pas toujours pertinente, car l'utilisateur ne peut pas spécifier ce qui est pertinent dans l'image de la recherche.
- Si l'utilisateur fournit au système une image exemple satisfaisante, cela veut dire qu'il a déjà fait une première interrogation à l'aide d'une autre modalité.
- Le problème majeur dans ce mode d'interrogation, est que les requêtes ne symbolisent pas la sémantique. Elles sont exprimées par des caractéristiques de bas niveau (ou visuelles) de l'image.

3. Requête par navigation

Dans ce type de requête, l'utilisateur navigue dans les images.

2.7.1.2. Modèle de correspondance

Ce modèle permet de faire la correspondance (calcul de similarité) entre requêtes et documents images. Ces derniers sont représentés par des caractéristiques de bas niveau. Une similarité est calculée entre chacune des

caractéristiques de l'image se trouvant dans l'index, et les caractéristiques de la requête image. Le résultat est un ensemble d'images ordonnées selon leur degré de pertinence.

Les mesures de similarité sont nombreuses. Nous citons quelques mesures de distances dont certains spécifiques aux histogrammes et aux distributions : la distance euclidienne, les distances entre histogrammes, les distances quadratiques, les distances entre distributions...

2.7.2. Recherche textuelle basée sur les mots clés

Après l'indexation textuelle des images, la recherche des images est déclenchée par une requête textuelle. Le processus de recherche utilise un des modèles de recherche d'informations (modèle booléen, modèle vectoriel, modèle probabiliste...) (cf. chapitre1).

2.7.3. Recherche textuelle visuelle

La recherche dans ce mode dépend de la requête.

- 1 Si la requête est textuelle, la recherche est textuelle,
- 2 Si la requête est visuelle, la recherche est basée sur le contenu visuel,
- 3 Mais si la requête est composée de mots clés et d'une requête visuelle, la recherche s'effectue dans les deux espaces textuel et visuel.

2.8. Systèmes de recherche d'images

Le système de recherche d'image est un cas particulier du système de recherche d'information. Dans la littérature, il y a deux types de base d'images :

- Des bases d'images spécialisées où les images sont de même type. Par exemple, images médicales, images géographiques ...
- Des bases d'images généralistes où les images sont de différents types

D'une manière générale, les systèmes de recherche d'images appartiennent à deux types : Les systèmes textuels de recherche d'images où l'indexation est textuelle et la recherche se fait à travers des mots clés, et les systèmes de

recherche d'images par le contenu visuel où l'indexation et la recherche se font à partir des descripteurs visuels.

D'après [95], il existe deux approches de recherche par le signal ou par le sens de l'image, respectivement désignées par approches signal et approches symboliques basées sur les mots clés.

D'autres auteurs estiment que les systèmes de recherche d'images sont classés d'après le niveau d'abstraction : Les systèmes à faible niveau d'abstraction, les systèmes à moyen niveau d'abstraction et les systèmes à haut niveau d'abstraction [96].

Nous présentons les trois catégories de systèmes de recherche d'images.

2.8.1. Systèmes basés sur le contenu visuel : Approche signal

Les SRIm **CBIR** (Content-Based Image Retrieval) de type signal sont les premiers systèmes de recherche d'images basés sur le contenu visuel. Ils sont apparus au début des années 90, et sont dédiés à la recherche en photographie couleur.

La figure 12 qui représente les différents composants du système de recherche d'images par le contenu visuel.

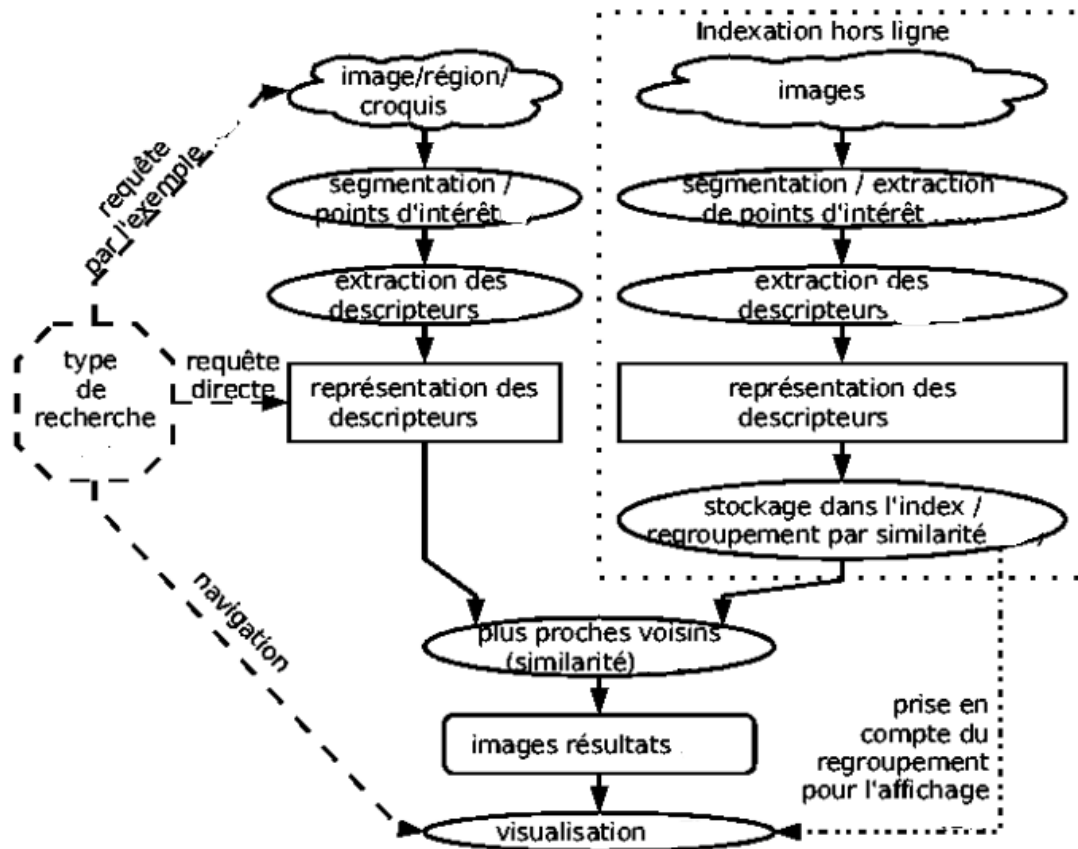


Figure 12: Le système de recherche d'images par le contenu [32]

Dans ce type de système, une image est représentée par un vecteur de caractéristiques visuelles (couleur, forme, texture...). Une requête de type image ou image-requête est effectuée. Après avoir extrait le vecteur de caractéristiques de l'image-requête, le système le compare avec ceux des images de la base. Le résultat est un ensemble d'images de la base qui possèdent un vecteur de caractéristiques le plus proche de celui de l'image-requête.

Dans cette approche, il existe trois catégories de systèmes, selon l'objectif visé :

- Catégorie 1 : On cherche à reconnaître un objet particulier (reconnaitances de formes ou d'objets). Exemples : images médicales, photos satellites.
- Catégorie 2 : On cherche à classer les images en fonction du type de scènes qu'elles représentent (reconnaitances / classification de scènes).

- Catégorie 3 : On cherche à reconnaître les images similaires à une image-requête (on traite les images généralistes).

Parmi les premiers systèmes de recherche d'image basés sur le contenu visuel, citons le système QBIC [97] [98].

Dans les systèmes de recherche basés sur le signal, la représentation des documents images et des requêtes ne tient pas compte de la sémantique.

Cette approche pose aussi le problème de polysémie visuelle où deux représentations visuelles peuvent être similaires au niveau signal, tout en ayant des sémantiques très différentes. Par exemple deux images similaires du point de vue signal, peuvent véhiculer deux sens différents (image de la mer avec image de ciel bleu).

2.8.2. Systèmes textuels : Approches symboliques basées sur les mots clés

L'approche symbolique a été introduite pour palier aux insuffisances de l'approche signal. Elle consiste à intégrer la sémantique dans les représentations des images.

Généralement, ces approches sont basées sur les mots clés. Ces derniers permettent d'avoir une représentation plus riche et une recherche plus fine des images.

Dans ce type de système, les images et les requêtes sont représentées par un ensemble de mots clés. Le système compare les mots clés de la requête avec ceux des images de la base. Le résultat est un ensemble d'images de la base qui correspondent à la requête.

Parmi les systèmes de recherche d'images textuels citons: Google Image.

2.8.3. Systèmes textuels basés sur le contenu visuel : Approches hybrides

Les systèmes hybrides permettent d'exploiter les avantages des deux systèmes en combinant le texte et le signal. La figure 13 décrit le système textuel basé sur le contenu visuel.

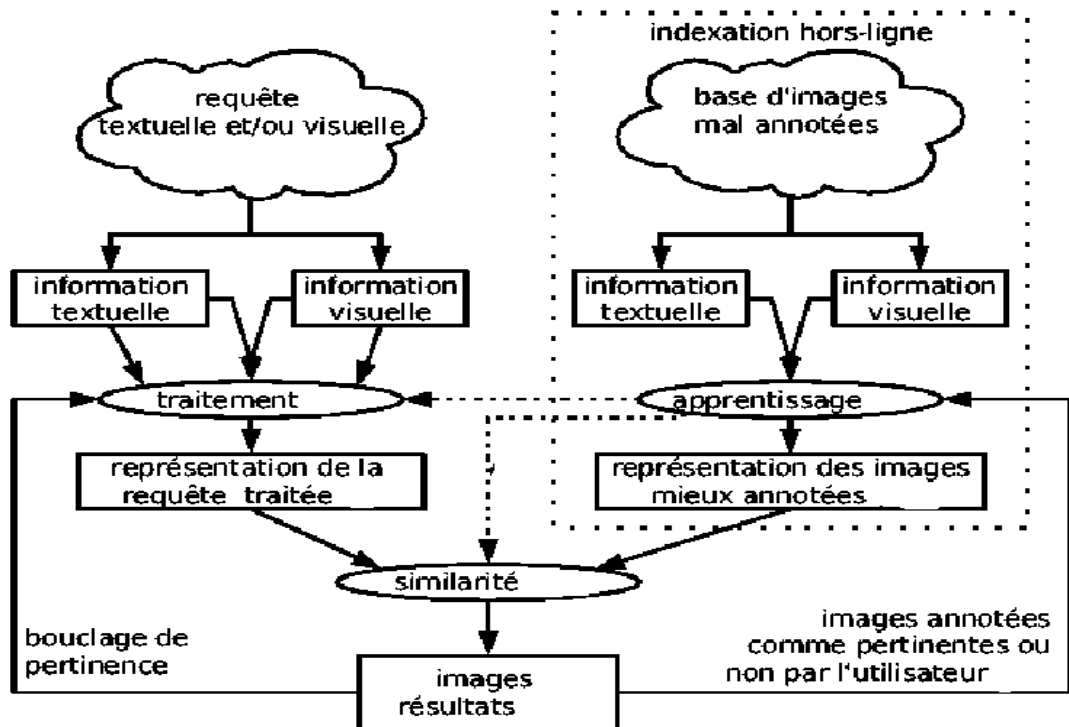


Figure 13: Le système de recherche d'images combinant l'aspect visuel et textuel [32]

Dans cette approche, le système de recherche d'images peut rechercher :

- Les images visuellement similaires à la requête, si la requête est visuelle, comme dans un système classique CBIR.
- Les images sémantiquement similaires vis-à-vis de la requête textuelle, comme dans un système textuel classique de la recherche d'information.
- Les images qui ont les mêmes caractéristiques visuelles et les mêmes mots clés que la requête de l'utilisateur (visuellement et sémantiquement similaires à la requête textuelle visuelle).

Le premier système qui a réellement corrélé l'information textuelle et visuelle est le système PICTION [100]. Le texte est extrait de la légende associée aux images d'articles de presse. La corrélation est résumée dans une hiérarchie visuelle qui combine à la fois des informations de structures textuelles extraites de WordNet [42] et des informations de couleur, de texture et de forme.

D'autres systèmes de recherche textuelle d'images basés sur le contenu visuel sont : Système A-LIP [99], Système PHOTOBOK [101], Système IMAGEROVER [102] et Système PICHUNTER [103]...

2.9. Conclusion

Dans ce chapitre, nous avons présenté l'image et sa description visuelle ainsi que le processus de recherche d'images. Ensuite, nous avons présenté les différents types de système de recherche d'images.

L'indexation des images dans les SRIm peut se faire selon le contenu d'image (textuel, visuel) : indexation textuelle et indexation visuelle.

L'indexation textuelle peut être manuelle subjective ou automatique. Cette indexation ne décrit pas le contenu visuel de l'image mais apporte des informations sémantiques.

L'indexation visuelle décrit le contenu visuel de l'image (couleurs, formes, textures..), mais l'extraction de la sémantique reste difficile.

La solution peut se résumer comme suit : **combiner l'aspect visuel et l'aspect sémantique**, c'est-à-dire trouver une **corrélation** entre les caractéristiques visuelles et les sens véhiculés par une image pour une recherche plus fine.

Dans le chapitre qui suit, nous aborderons une approche textuelle prometteuse : l'annotation d'images.

CHAPITRE 3

ANNOTATION D'IMAGES

3.1. Introduction

Les systèmes de recherche d'images ont pour but de trouver les images dont l'utilisateur a besoin. Les systèmes de recherche d'images textuels indexent les images à partir du texte associé. Généralement, ce texte ne reflète pas l'image. Les systèmes de recherche retournent des images qui ne répondent pas forcément aux attentes des utilisateurs.

L'annotation d'images permet d'associer des mots aux images en décrivant leurs aspects sémantiques. Cette technique permet d'améliorer la recherche d'images.

L'annotation automatique d'images est un domaine récent, dont les premiers travaux datent de 1999 et l'accroissement en nombre de publications date de 2002.

Dans le but d'améliorer les résultats d'annotation d'images, les travaux existants intègrent dans le processus d'annotation différents domaines : Ontologie, Web sémantique, réseau bayésien, réseau de neurones, datamining, image mining, techniques d'apprentissage, probabilité et espace latent linéaire et non linéaire...

Nous allons introduire le domaine de l'annotation d'images et présenter quelques travaux existants.

3.2. Annotation d'images

L'annotation est un moyen d'exprimer la sémantique d'une image. Elle permet d'associer à l'image un mot ou un ensemble de mots, qui dénote son contenu sémantique. Elle peut être annotation manuelle, annotation automatique et annotation semi-automatique.

3.2.1. Annotation manuelle

L'annotation manuelle est un processus qui consiste à attribuer manuellement à chaque image un ensemble de mots clé. Ce processus est généralement réalisé par un expert humain.

Dans ce type d'annotation, la précision augmente (expertise humaine) et la productivité baisse (lorsque le volume d'images devient important).

3.2.2. Annotation automatique

L'annotation automatique, ou auto-annotation, s'effectue à travers un système informatique (machine) sans intervention humaine. C'est une indexation textuelle automatique à partir du contenu visuel de l'image.

La précision est faible dans ce type d'annotation, mais la productivité augmente (possibilité d'annoter un volume important d'images).

3.2.3. Annotation semi-automatique

L'annotation semi-automatique est la combinaison des deux types d'annotations précédents. Généralement un processus d'annotation automatique est initié, suivi par une validation d'un expert humain.

3.3. Modèles d'annotation

Dans notre travail, nous nous intéressons à l'auto-annotation. Voici quelques modèles d'auto-annotation existants.

3.3.1. Indexation multimédia par analyse de la sémantique latente

3.3.1.1. Latente Semantic Analysis : LSA

LSA est un modèle par analyse sémantique latente [104]. Cette technique statistique automatique est appliquée sur un grand corpus de texte. Le but de LSA est d'extraire et d'inférer les relations entre les mots à partir de leur contexte. Elle est vue comme un modèle d'acquisition et de représentation des connaissances.

Principe

Le sens d'un mot peut être défini statistiquement à partir de l'ensemble des contextes (phrases, paragraphes, textes) dans lesquels ce mot apparaît. Par exemple, le mot voiture est co-occurent souvent avec les mots : démarrer, route... mais rarement avec mer, soleil... Cependant, le mot automobile est co-occurent avec le mot voiture, donc le mot voiture et automobile sont proches et ne peuvent jamais être ensemble.

Le contexte ne suffit pas pour définir le sens d'un mot, car il n'y a aucune information sur les relations avec les mots qui n'apparaissent jamais ensemble.

Si une requête contient l'un des deux mots (voiture ou automobile), le système ne retourne que les documents qui contiennent les mêmes mots que la requête. Or il existe des documents pertinents qui contiennent des mots proches de ceux de la requête et qui ne sont pas sélectionnés par le système.

L'analyse sémantique latente résout ce problème, et définit correctement le sens d'un mot à partir de l'ensemble des contextes dans lequel ce mot apparaît ou n'apparaît pas, à travers un mécanisme d'enchaînement de lien de cooccurrence à plusieurs niveaux.

LSA résout les problèmes de la synonymie et de la polysémie des mots en capturant les liens sémantiques entre les documents et les mots :

- **Synonymie** : deux mots ayant le même contexte, sont proches dans l'espace latent.

- Ambiguïté : plusieurs documents possédant le même mot, mais dans des sens différents, n'auront pas le même contexte et seront donc éloignés dans l'espace latent,
- Absence : même si un mot est absent d'un document sur le même thème, les deux seront proches dans l'espace latent car ils ont le même contexte.

Dans la recherche d'image, ces problèmes se posent :

- Une même couleur peut décrire deux objets différents et deux couleurs différentes peuvent décrire le même objet.
- Deux images peuvent être sémantiquement proches, même si les mots qui les annotent ou leurs caractéristiques visuelles sont différents.

Processus de LSA

La technique LSA utilise le modèle LSI (Latent Semantic Indexing) [17].

Soit d documents et t termes. Chaque document est représenté comme un vecteur de termes.

\mathbf{A} est une matrice dont les colonnes sont les vecteurs de termes (les termes sont les lignes et les documents sont les colonnes) et k son rang.

Les étapes du processus LSA sont les suivantes :

1. Construire une matrice de cooccurrence terme-document \mathbf{A} , où chaque cellule de \mathbf{A} représente la fréquence d'un mot dans un document.
2. Réduire la matrice \mathbf{A} par la décomposition en valeur singulière SVD afin de prendre en compte le contexte des mots et réduire les dimensions de l'espace.

Algorithme SVD

1. La matrice \mathbf{A} est décomposée par SVD en trois nouvelles matrices.

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=0}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^T \quad (5)$$

Σ : une matrice diagonale des valeurs singulières de la matrice A, de taille $r \times r$

- $\Sigma = \text{diag}(\sigma_1, \sigma_2 \dots \sigma_r)$, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$
- U : une matrice orthonormale des vecteurs U_i (termes) de taille $t \times r$
- Les vecteurs U_i sont appelés les vecteurs singuliers gauches de A
- V^t : est la matrice orthonormale de document de taille $d \times r$
- Les vecteurs V_i^t sont appelés les vecteurs singuliers droits de A
- Les valeurs singulières et les vecteurs singuliers gauches et droits sont appelés aussi triplets singuliers.

2. On sélectionne les k premières valeurs singulières, en mettant les petites valeurs singulières de Σ à zéro, afin de réduire l'espace. Le choix de k n'est pas défini (pas de règle pour le déterminer). Par exemple, pour la langue anglaise, k est estimé empiriquement à 300 [105].

- La matrice Σ est devenue de taille $k \times k$
- $\Sigma = \text{diag}(\sigma_1, \sigma_2 \dots \sigma_k)$, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k > 0$ et $\sigma_{k+1} = \sigma_{k+2} = \dots = \sigma_r = 0$
- D'où $A_k = U_k \Sigma_k V_k^t$ tel que $k \leq r$
- A_k sont les k meilleurs rangs de la matrice originale A

LSI fait la transformation de l'espace original des termes en espace des concepts (réduit) dans la collection de documents.

Dans l'espace des concepts, les associations termes-documents les plus importantes sont représentées. Les coordonnées des documents originaux dans cet espace sont données par la matrice $V_k \Sigma_k$. On appelle l'espace $d \times k$, espace de projection de documents.

La fonction de changement de repère est donc formée par la matrice $M = U[t, k] \times \Sigma^{-1}[k, k]$. Pour évaluer une requête Q, il faut d'abord représenter la requête dans l'espace M par la formule suivante : $Q_n = Q * M$, puis calculer la similarité (distance), par exemple le cosinus, entre la requête et chaque document dans le nouvel espace vectoriel M.

3.3.1.2. Probabiliste LSA : PLSA

Le modèle PLSA est un modèle probabiliste pour LSA, en ajoutant l'interprétation probabiliste. Ce modèle est proche du modèle aspect model [106][107]. PLSA utilise une matrice P de probabilité jointe, elle est construite avec $P(w_j, d_i)$ probabilité de cooccurrences du mot w_j et du document d_i .

Soit une classe non observée $Z=\{z_1, z_2, \dots, z_r\}$. Où z est une variable de cette classe.

$$\begin{cases} W_{j,k} = p(w_j|z_k) & W \in [0, 1]^{n_W \times r} \\ \sigma_{k,k} = p(z_k) & \Sigma \in [0, 1]^{r \times r} \\ D_{i,k} = p(d_i|z_k) & D \in [0, 1]^{n_D \times r} \end{cases} \quad \Sigma = \text{diag}(\sigma_{1,1}, \sigma_{2,2}, \dots, \sigma_{r,r}) \quad (6)$$

Une SVD est appliquée sur la matrice P. Si on suppose que w et d sont indépendants conditionnellement à z , le résultat est le suivant : $P = W S D^t$

Il traduit la règle de Bayes :

$$P(w_j | d_i) = \frac{p(w_j|z_k)p(z_k)}{p(d_i|z_k)} \quad (7)$$

- La matrice S contient les probabilités des différentes classes
- La matrice W effectue un classement flou des mots dans les classes
- La matrice D effectue un classement flou des documents dans les classes

Un modèle de probabilité jointe sur $D \times W$ est obtenu en marginalisant sur les classes z_k :

$$\begin{aligned} p(w_j, d_i) &= p(d_i)p(w_j|d_i) \\ &= p(d_i) \sum_{z_k \in Z} p(w_j|z_k)p(z_k|d_i). \end{aligned} \quad (8)$$

Chaque classe peut être vue comme un concept qui génère les mots. Un document est donc associé à plusieurs concepts.

Avec les données d'apprentissage et à l'aide de l'algorithme Expectation-Maximization EM, les paramètres de modèle $P(w_j|z_k)$ et $P(z_k|d_i)$ sont obtenus en maximisant la vraisemblance des données.

$P(w_j|z_k)$ est la probabilité d'avoir le mot w_j sachant que l'on considère la classe z_k , elle permet donc d'annoter de nouveaux documents. $P(z_k|d_i)$ ne donne pas d'information sur un nouveau document.

Pour annoter un nouveau document d_{new} , on calcule la probabilité d'avoir un mot donné sachant que l'on connaît d_{new} :

$$p(w_j|d_{new}) = \sum_{z_k \in \mathcal{Z}} p(w_j|z_k)p(z_k|d_{new}). \quad (9)$$

D'autres modèles d'annotation existent dans la littérature. Nous pouvons citer les modèles multi-modals hierarchical aspect models (MoM-HAM) [108][109][110] et les modèles fondés sur la distribution Dirichlet comme le modèle Latent Dirichlet Allocation LDA [111], le modèle MoM-LDA [110], le modèle GM-Mixture ([110][119]) et le modèle Corr-LDA [85]

3.4. Survol de quelques travaux sur l'annotation d'image

Nous avons classé les travaux suivants selon le domaine intégré dans le processus d'annotation.

3.4.1. Annotation dans l'espace latent

Nous présentons les stratégies d'annotation dans l'espace latent linéaire (propagation LSA ou inférence PLSA), et l'espace latent non linéaire qui ont été étudiées par plusieurs auteurs.

3.4.1.1. Espace latent linéaire

- Analyse sémantique latente LSA: l'annotation par propagation LSA, dans [112], s'effectue après la projection de la requête dans l'espace latent. La similarité entre la requête et chaque ligne de la matrice U (représentation

de la collection dans l'espace latent) est calculée par la mesure du cosinus. L'annotation est alors propagée depuis les documents classés. L'annotation est moins fiable quand la similarité entre les documents diminue.

- Analyse sémantique latente probabiliste PLSA: l'annotation par inférence PLSA, dans [113] et [112], est un processus du calcul probabiliste. Etant donné une image non annotée, la distribution à posteriori des termes du vocabulaire est donnée par l'équation 9. Les variantes de PLSA sont : PLSA-mixed, PLSA-words et PLSA-SPLIT. La variante PLSA-mixed applique le PLSA standard sur des représentations concaténées des modalités textuelles et visuelles d'un ensemble d'images annotées.

Les auteurs dans [112] concluent que l'annotation par propagation LSA est plus performante que l'annotation par inférence PLSA et est meilleure que le modèle PLSA-MIXED.

3.4.1.2. Espace latent non linéaire

La nouvelle stratégie d'annotation d'images via IWE (Image-Word Embedding), dans [114] est proposée dans l'espace latent non linéaire. Le système non supervisé d'auto-annotation d'image comprend deux séries de procédure d'annotation NLSA (Non Latent Space Analysis) et IWE.

La méthode d'annotation basée sur NLSA comporte deux étapes : la première est le calcul de similarité, dans l'espace non linéaire, sous le mappage non linéaire entre l'image non annotée et chaque image annotée dans le corpus. La deuxième étape est l'inférence basée sur les premières images annotées selon le classement de similarité.

La méthode d'inférence IWE incorpore simultanément les images annotées et leurs termes associés dans un espace de petite dimension.

Après l'étude comparative avec d'autres méthodes d'annotation non supervisée, la méthode d'annotation basée IWE sur l'espace latent non linéaire est la plus performante.

3.4.2. Annotation avec ontologie et web sémantique

La combinaison de l'annotation d'image et le web sémantique est discutée dans [115]. Les auteurs ont proposé un Framework pour l'annotation des images en utilisant le web sémantique.

L'approche présentée utilise les ontologies pour représenter l'aspect sémantique d'une image. Les concepts de bas niveau d'une ontologie sont les images liées en utilisant les classificateurs SVM (Support Vector Machine). Les concepts de haut niveau sont représentés par les réseaux bayésiens.

La tâche d'annotation d'image est décomposée en une classification atomique de bas niveau et une classification des concepts de haut niveau dans un domaine d'ontologie spécifique. Le SVM est utilisé dans la première classification et le réseau bayésien dans la deuxième.

Après segmentation, les caractéristiques d'une image sont extraites depuis chaque objet de l'image. Les vecteurs de caractéristiques sont envoyés au classificateur SVM qui identifie les concepts de bas niveau de chaque image. L'ensemble des concepts de bas niveau est envoyé au réseau bayésien qui doit classer l'image selon des concepts sémantiques de haut niveau.

Dans [116] les auteurs montrent comment l'utilisation des ontologies avec les schémas RDF (Resource Description Framework) [117] peuvent améliorer l'annotation et la recherche dans une collection d'images. Les résultats expérimentaux sont prometteurs.

3.4.3. Annotation avec les techniques de datamining et de fouilles d'images (Image mining)

La technique des règles d'association est utilisée dans [118] pour découvrir les associations entre les caractéristiques visuelles (signal) et textuelles (symbole ou concept).

Le processus de fouille d'image FDI est proposé pour combiner l'aspect visuel et textuel à l'aide de la technique des règles d'association afin d'avoir une indexation sémantique.

Les résultats expérimentaux de FDI montrent la faisabilité et l'efficacité de l'approche [118].

Le système AnnoSearch proposé dans [47] annote les images en utilisant la recherche et les techniques de datamining.

Cette approche permet de chercher les images sémantiquement et visuellement similaires sur le web et d'extraire l'annotation depuis leurs descriptions. Le système AnnoSearch comprend trois étapes:

1. La première étape permet d'établir la recherche basée sur le texte, en donnant une requête textuelle, qui retourne les images sémantiquement similaires.
2. La deuxième étape concerne la recherche basée sur le contenu visuel, en donnant une requête image, afin d'accélérer la procédure de recherche. L'algorithme de hash coding est utilisé pour identifier les images similaires.
3. La dernière étape consiste à catégoriser les images sélectionnées et à extraire les formules marquantes depuis leurs descriptions.

L'annotation d'images est apprise et assignée à la requête image par les mêmes descripteurs.

Les expérimentations ont montré l'efficacité et le bon fonctionnement du système AnnoSearch.

3.4.4. Annotation avec les algorithmes génétiques

L'idée dans [43] est l'utilisation d'un algorithme génétique 'real coded chromosome' et la classification de k plus proches voisins comme fonction de fitness pour optimiser le poids des descripteurs des caractéristiques d'images MPEG-7. Trois schémas sont utilisés :

- L'optimisation du poids avec un algorithme génétique 'real coded chromosome'.
- La sélection de descripteur de caractéristiques optimal avec le code 'binary one'.
- La simultanéité d'optimisation du poids et la sélection de descripteur de caractéristiques optimal avec le 'bi-coded one'.

Les auteurs [43] traitent le problème d'annotation d'image comme un problème de classification. L'annotation d'image se fait par un seul mot. Les auteurs ont travaillé avec l'image entière sans segmentation. Le processus d'annotation se fait comme suit :

1. Calculer le vecteur de caractéristiques bas niveaux de l'image avec les résultats de l'algorithme génétique.
2. Classer l'image dans la catégorie avec le classificateur k plus proche voisin.
3. Finalement, le mot clé correspondant à la catégorie de l'image est propagé afin d'annoter automatiquement l'image.

Les expérimentations menées par les auteurs ont montré l'efficacité du système d'annotation.

3.4.5. Annotation avec les réseaux bayésiens

Afin d'annoter automatiquement les images, les auteurs dans [41] ont présenté une nouvelle approche bayésienne semi-naïve incorporant la catégorisation avec les contraintes pair-wise ou deux à deux.

Cette approche se décompose en deux phases : apprentissage et annotation d'image.

- Apprentissage : Le déroulement de cette phase se fait en deux étapes :
 - a. segmentation de l'image en régions.
 - b. catégorisation des régions dans des clusters de région par l'incorporation des contraintes deux à deux, qui sont dérivées du modèle de langage.

- Annotation d'image: Durant cette phase, le modèle probabiliste est utilisé afin d'établir les relations entre l'annotation et les clusters de région en utilisant le modèle bayésien semi-naïf pour la prédiction et l'inférence des concepts. Cette phase se déroule en quatre étapes :
 - a. définition / formulation du problème de probabilité.
 - b. calcul de la probabilité à posteriori des concepts à partir des clusters de région.
 - c. calcul de la probabilité à posteriori des clusters de région à partir des régions.
 - d. annotation d'une nouvelle image.

Les auteurs démontrent par expérimentation que l'approche qui combine la catégorisation avec les contraintes pair-wise et le modèle bayésien semi-naïf améliore considérablement la performance d'annotation.

3.4.6. Annotation avec SVM (séparateurs à vaste marges)

Dans [33], les auteurs ont proposé un framework d'auto-annotation qui se résume comme suit:

- Extraction automatique des caractéristiques visuelles multimodales, à partir d'une image de test, selon différentes tailles de grille.
- Classement des grilles d'image multi-résolution dans les classes d'objet les plus pertinentes.
- Fusion des images de grilles voisines de la même classe d'objet comme une seule image grille.
- Classement de l'image test dans le concept d'image le plus pertinent en connaissant des classes d'objet.
- Annotation automatique d'image multi niveaux en connaissant les classes d'objet et les concepts d'image.

Les résultats d'expérimentation obtenus, en utilisant des images naturelles, sont assez encourageants.

3.4.7. Annotation avec le modèle probabiliste

La méthode d'annotation d'image, dite DIMATEX [32], utilise le modèle probabiliste inspiré de la technique Va-Files [29] de partitionnement de l'espace.

Après le partitionnement de l'espace visuel en des clusters visuels avec la technique Va-Files, la table de description jointe entre les clusters visuels et les mots est construite afin d'estimer la distribution des mots pour un région d'une nouvelle image.

Le système DIMATEX donne des scores d'annotation appréciables, capable d'évoluer dynamiquement. DIMATEX annote très rapidement les images.

Dans [28], les auteurs proposent un système pour l'annotation d'image, qui apprend les probabilités jointes d'ensemble de régions et d'ensemble de mots.

3.4.8. Annotation avec un réseau de neurones

Un nouveau schéma d'annotation basé sur un réseau de neurones est proposé dans [27]. Le but est d'utiliser un réseau de neurones pour caractériser les associations cachées entre deux modalités, visuelles et textuelles. En outre LSA est utilisé dans le schéma d'annotation basé sur le réseau de neurones (LSA-NN) pour découvrir la corrélation contextuelle latente entre les mots clés.

Le processus d'annotation proposé se déroule en trois étapes:

- La première étape introduit LSA pour révéler la corrélation contextuelle latente entre les mots clés.
- La deuxième étape effectue l'apprentissage avec des images étiquetées. Le réseau de neurones est obtenu pour caractériser l'association cachée entre le contenu visuel et textuelle (mots clés) de l'image.
- Finalement, étant donnée une image de test, le système LSA-NN est capable de fournir efficacement les mots clés annotant l'image.

Etant donnée un ensemble d'images d'apprentissage, les caractéristiques visuelles globales de chaque image représentent l'entrée du réseau de neurones, et le vecteur contextuel de mots clés représente la sortie du réseau.

D'après les expérimentations, le schéma d'annotation basée sur LSA-NN peut obtenir une performance d'annotation appréciable.

3.4.9. Annotation avec une segmentation multiple

Un grand nombre des techniques d'annotation d'image utilisent les algorithmes de segmentation pour partitionner l'image. Mais généralement, ils choisissent un seul niveau de segmentation. La plupart des algorithmes de segmentation ne donnent pas des résultats satisfaisants.

Une nouvelle utilisation de la segmentation multiple dans le contexte d'auto annotation d'image est présentée dans [23], en variant le paramètre d'algorithme de segmentation afin de générer plusieurs niveaux de segmentation. Cette nouvelle approche est incorporée dans la technique d'annotation d'image basée sur les régions.

L'algorithme d'annotation est comme suit :

- Segmentation automatique d'image, à différents niveaux, en plusieurs régions.
- Calcul des caractéristiques descriptives de chaque région.
- Construction des espaces de caractéristiques.
- Définition des fonctions de mappage des régions d'image et « labels » dans l'espace de caractéristiques correspondant.
- Annotation d'image.

Les auteurs ont montré que la performance d'annotation peut être améliorée sur une collection de 5000 images lorsque la segmentation multiple est utilisée.

3.4.10. Annotation avec la fusion de région et d'image

Dans [26] les auteurs proposent une méthode de clustering basée sur une approche orientée concepts centrés qui combine les niveaux d'analyse de région et de l'image pour l'annotation automatique d'image.

Le processus d'annotation se déroule comme suit :

- Utiliser les descriptions de inter et intra concept de région pour :
 - identifier automatiquement les principaux clusters de région (blobs) pour chaque concept,
 - obtenir les clusters de région représentatifs et les caractéristiques typiques de chaque concept,
- Utiliser ces informations pour annoter les régions.
- Appliquer une annotation multi-niveaux par la fusion des résultats des annotations niveau région et niveau image

3.5. Conclusion

L'annotation d'image est un moyen d'exprimer la sémantique d'une image. Elle consiste à attribuer aux images des mots clés correspondant aux sens véhiculés par ces images. Cependant l'annotation d'image pose le problème du fossé sémantique et le passage de l'aspect visuel à l'aspect sémantique, qui peuvent se résumer comme suit :

- Synonymie textuelle : Différents mots ou termes peuvent être utilisés pour exprimer le même concept.
- Synonymie visuelle : Plusieurs couleurs différentes peuvent décrire le même objet.
- Polysémie textuelle : un mot clé peut correspondre à plusieurs concepts (le même mot peut avoir différent sens selon le contexte).
- Polysémie visuelle : un concept peut correspondre à plusieurs représentations visuelles (une même couleur peut décrire différents objets).

Dans ce chapitre, nous avons présenté quelques travaux existants sur l'annotation automatique des images ainsi que leurs évaluations.

Dans le chapitre suivant, nous allons présenter notre contribution qui vise à améliorer l'annotation automatique d'images.

CHAPITRE 4

SYSTEME D'AUTO ANNOTATION DES IMAGES SUR LE WEB

4.1. Introduction

Les systèmes d'auto annotation sont des systèmes très prometteurs. Cependant, la qualité des images annotées n'est pas encore satisfaisante. Parmi les stratégies les plus répandus, la stratégie qui consiste à **extraire et fusionner les informations textuelles et visuelles des images** [32], semble prometteuse. Cependant, cette stratégie doit être bien conçue. Pour cela, il faut bien définir les éléments suivants :

- **La méthode d'annotation** : les systèmes d'annotation basée sur l'image donnent des résultats assez bons mais l'annotation reste globale et parfois insuffisante. Dans les systèmes d'annotation basée sur les régions d'image, l'annotation est spécifique aux régions mais elle dépend fortement des résultats de la segmentation.
- **La segmentation** : la segmentation joue un rôle important dans l'annotation d'image basée sur les régions car elle sépare les objets dans les images; Mais cela reste théorique. Cependant, elle ne fournit pas une correspondance exacte entre un concept et une région. En général, une bonne segmentation conduit à une bonne annotation.
- **Les caractéristiques visuelles** : le choix des caractéristiques visuelles est un paramètre très important dans l'annotation d'image. Une image contient beaucoup d'informations visuelles (couleur, texture, forme...), mais le bon choix reste un problème.

Dans ce qui suit, nous allons présenter une proposition d'annotation automatique d'images en se basant sur cette stratégie qui effectue la corrélation entre le textuel et le visuel.

4.2. Système d'annotation automatique proposé

Le système d'auto-annotation d'images que nous proposons utilise un apprentissage sur un corpus d'images annotées.

Les images de corpus doivent être segmentées en régions afin de les grouper dans des classes de régions visuellement similaires appelées blobs.

Afin de corréler le textuel (annotations du corpus) avec le visuel (les blobs des régions), nous proposons d'annoter les blobs précédemment construits. Une nouvelle image peut alors être annotée automatiquement comme suit :

- Segmenter la nouvelle image en régions
- Affecter les régions de l'image aux blobs construits précédemment
- Annoter l'image par les mots des blobs auxquels les régions de cette image appartiennent.

Il suffit de trouver une méthode pour annoter les blobs et notre proposition sera complète.

Le processus d'auto-annotation se déroule en deux étapes principales : apprentissage et traitement d'une nouvelle image. La figure 14 schématise les phases du système d'annotation automatique proposé.

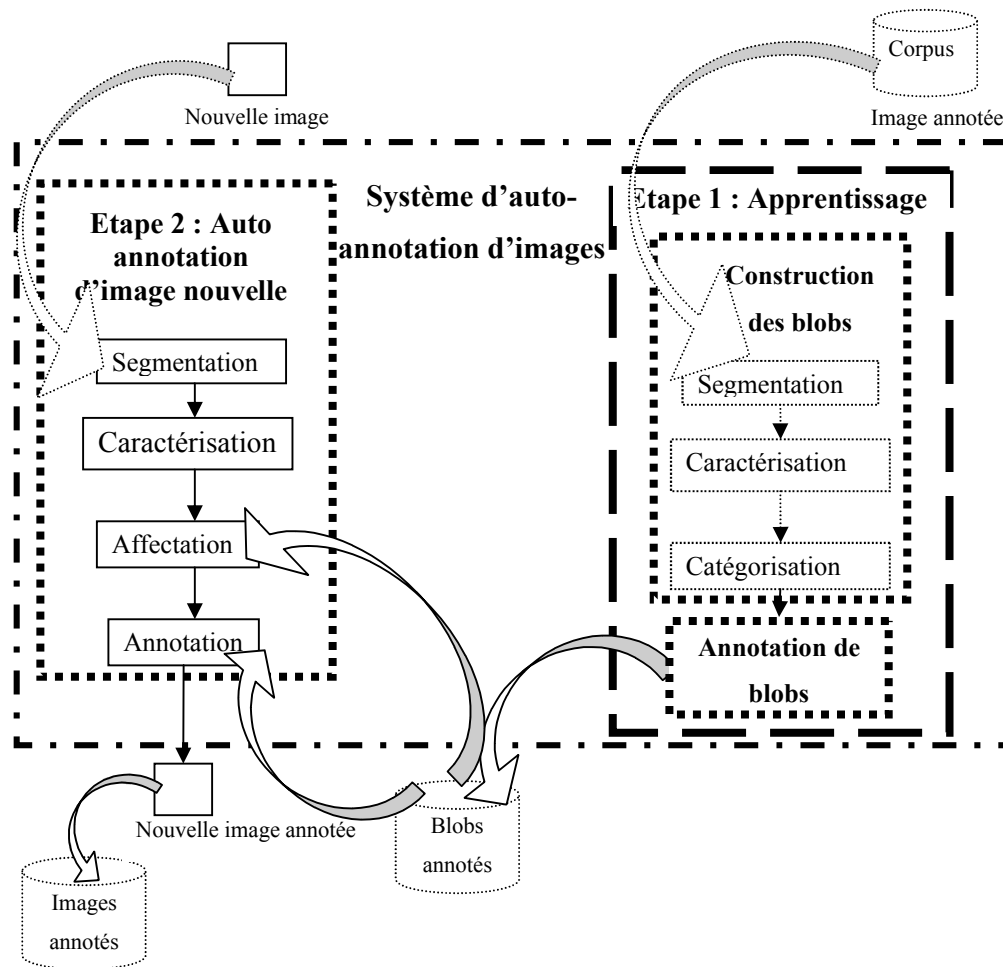


Figure 14 : Présentation des phases du système d'Auto-annotation proposé

4.2.1. Apprentissage

L'objectif d'apprentissage est la construction et l'auto-annotation des blobs. Nous segmentons les images d'un corpus d'images bien annotées. Les régions visuellement similaires sont regroupées dans des blobs, qui seront annotés.

L'étape d'apprentissage à partir d'un corpus d'images annotées comprend deux phases : construction des blobs et l'auto-annotation des blobs. Le processus d'apprentissage est représenté par le diagramme d'activité de la figure 15.

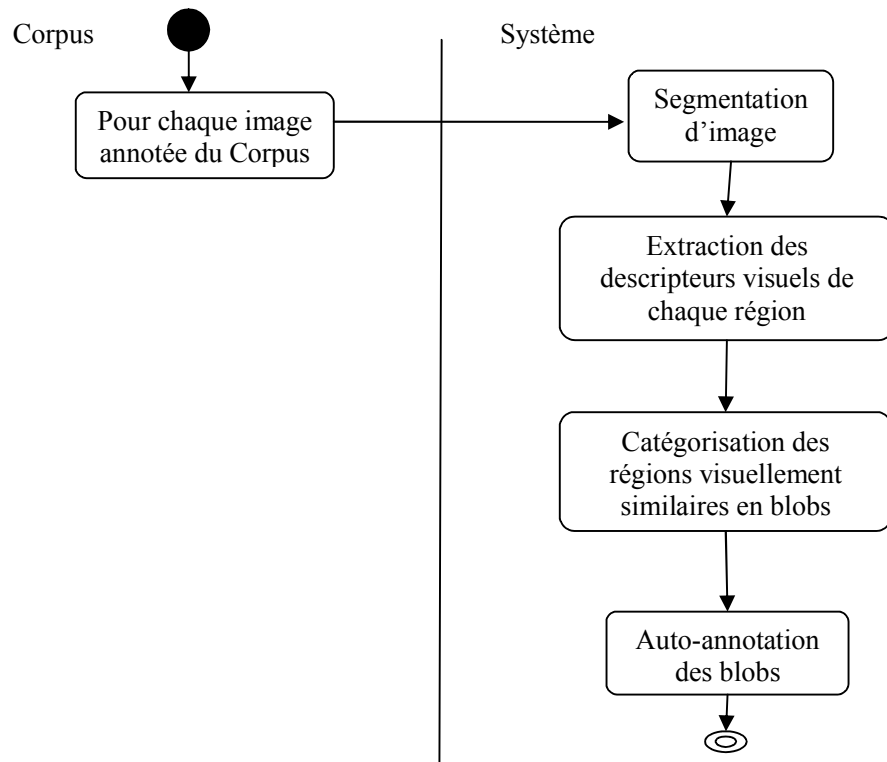


Figure 15 : Diagramme d'activité «Apprentissage»

4.2.1.1. Constructions des blobs

La première phase d'apprentissage regroupe : la segmentation d'images du corpus en régions; l'extraction des caractéristiques visuelles des régions; la catégorisation de ces dernières dans des blobs.

Nous détaillons la phase de construction des blobs dans la figure 16.

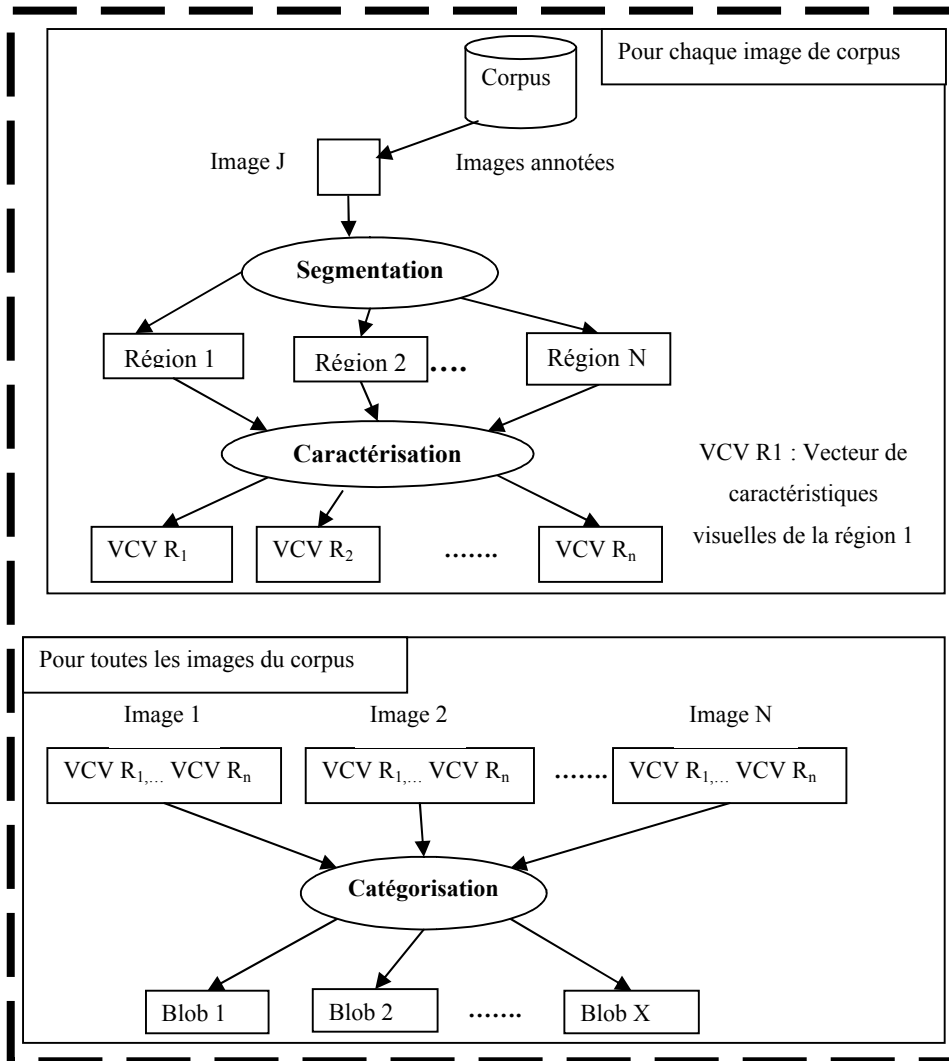


Figure 16 : Schéma d'apprentissage : Construction de blobs

A. Segmentation

Chaque image du corpus est segmentée en régions en utilisant l'algorithme Normalized-Cuts [22]. L'étude comparative menée par les chercheurs [12] a montré les performances de cet algorithme, qui est très utilisé dans les travaux d'annotation [20, 18, 11, 32]. Le diagramme de la figure 17 montre le processus de segmentation :

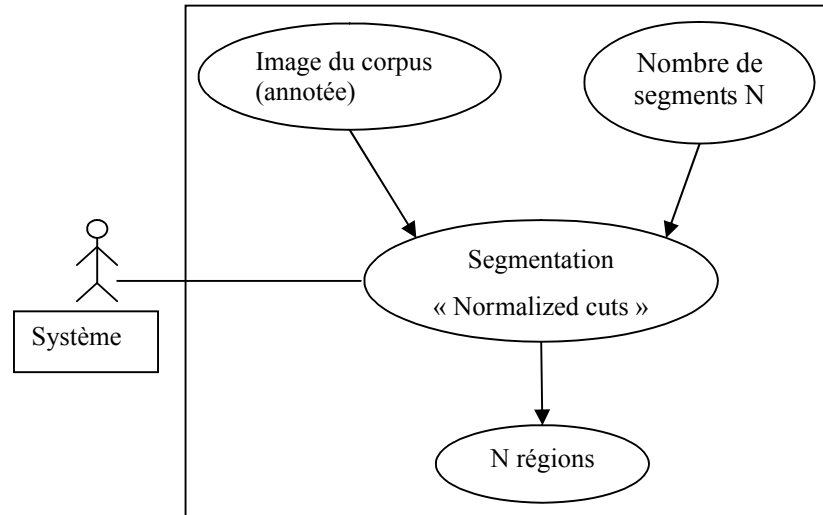


Figure 17 : Segmentation d'une image

B. Caractérisation

Dans cette étape, nous extrayons les caractéristiques visuelles de chacune des régions qui constituent l'image afin de les catégoriser dans des blobs. Les caractéristiques choisies sont : les moments statistiques de la couleur (la moyenne, la variance et les moments d'ordre trois) de chaque composante d'espace RGB et la texture (la variance, l'homogénéité, l'entropie et la contraste)...Le diagramme de la figure 18 montre l'extraction des caractéristiques visuelles.

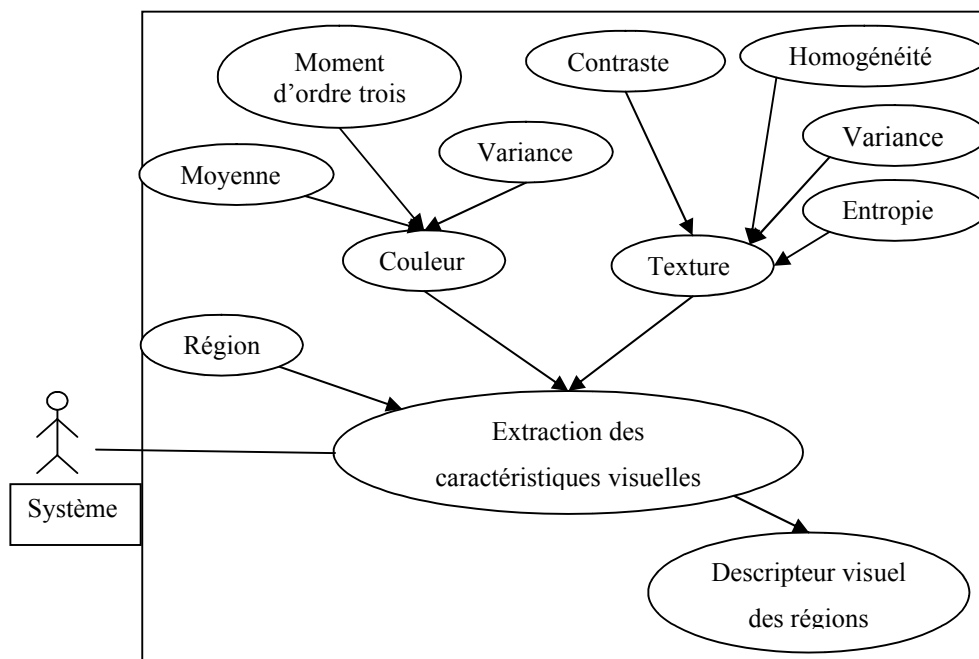


Figure 18: Extraction de caractéristiques visuelles

C. Catégorisation

Cette étape nous permet de catégoriser les régions selon leurs caractéristiques visuelles en **blobs**. La méthode choisie est la méthode de classification : K-means. Nous représentons cette étape dans le diagramme de la figure 19 :

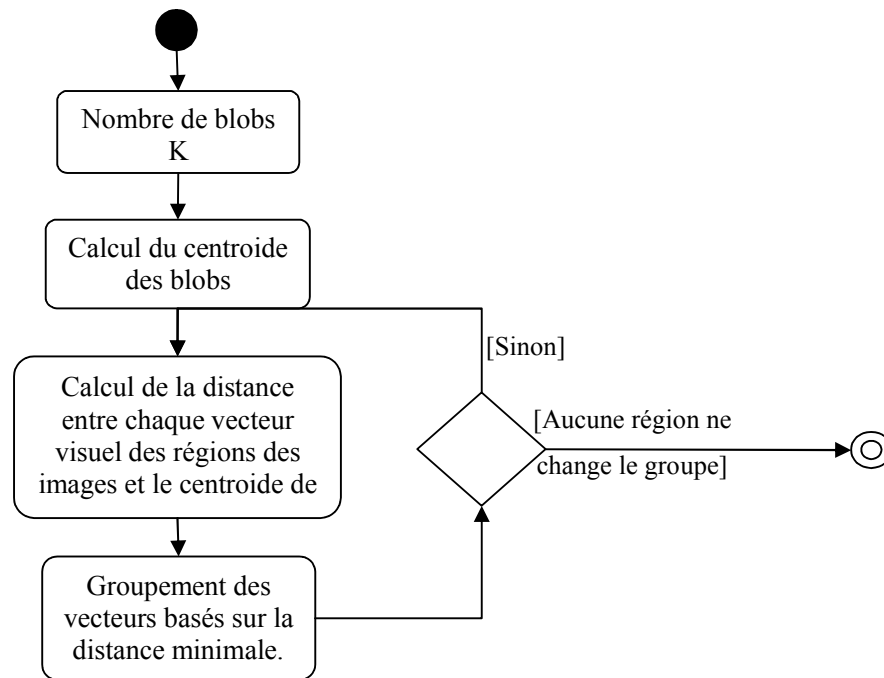


Figure 19: Diagramme d'activité de la catégorisation des régions d'images

4.2.1.2. Auto-annotation des blobs : Algorithme AnnotB-LSA

La deuxième étape de l'apprentissage permet de corrélérer le contenu visuel avec le contenu sémantique des régions des images en annotant les blobs regroupant les régions visuellement similaires, par les mots clés d'images du corpus correspondant. Ceci permettra d'annoter les nouvelles images.

Pour cela, nous proposons un algorithme, que nous appelons *AnnotB-LSA*. Pour extraire les relations sémantiques latentes dans l'espace des mots clés textuels et minimiser l'ambiguïté (polysémie, synonymie) entre les annotations des images de corpus, nous intégrons l'analyse sémantique latente LSA à l'algorithme.

A. Méthode d'annotation d'un blob

L'idée de base est la suivante : pour annoter un blob, nous lui faisons hériter les mots clé des images auxquelles les régions contenues dans ce blob appartiennent. Mais au lieu de faire un héritage intégral, nous gardons seulement les mots clé conceptuellement communs aux images (héritage sélectif).

Une fois les blobs annotés par héritage sélectif, nous pouvons annoter une nouvelle image en récupérant les mots clé des blobs auxquels les régions de cette image appartiennent.

B. Algorithme AnnotB-LSA

Entrée : Blobs

Sortie : Blobs annotés

Pour chaque blob b_i

Faire

Pour chaque région r_j du blob b_i

Extraction du numéro d'image auquel la région appartient

Extraction du vecteur de mots clés d'image correspondante

Projection du vecteur dans l'espace latent

Fin Pour

Calcul de similarité entre toutes les régions du blob b_i

Pour chaque vecteur de similarité d'une région

Chercher la similarité maximum

Si la similarité maximum entre le 0 et 1

Alors

Faire l'intersection entre les deux vecteurs de mots clés d'images auxquels les régions les plus similaires appartiennent.

Sinon

Si la similarité maximum entre le 0 et -1

Alors

Faire l'union entre les deux vecteurs de mots clés d'images auxquels les régions les plus similaires appartiennent.

Fin Si

Fin Si

Fin Pour

Fin Pour

La figure 20 montre les étapes de l'algorithme AnnotB-LSA.

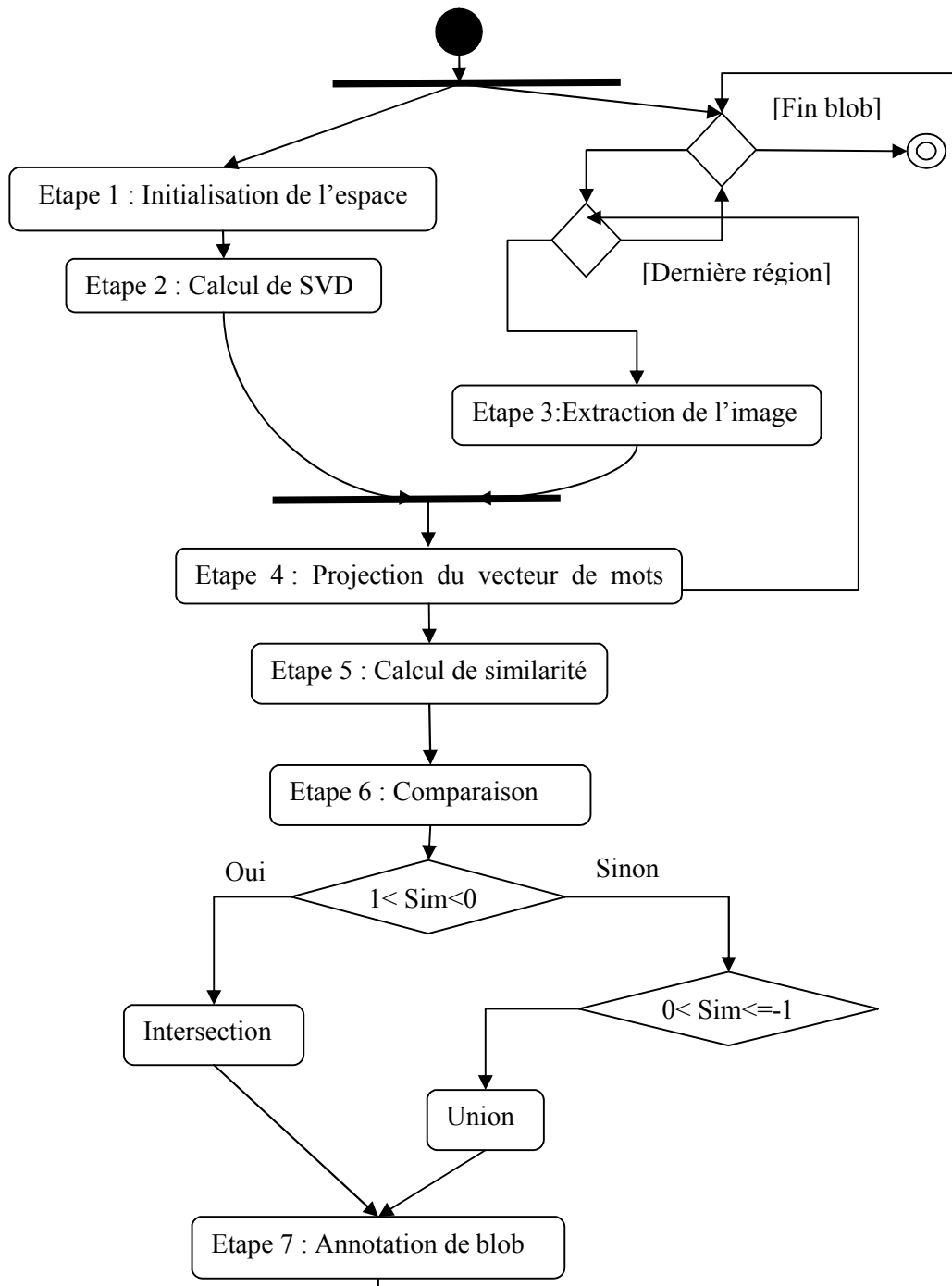


Figure 20: Diagramme d'activité de l'algorithme AnnotB-LSA

Nous désignons par :

- $C=\{J_1, J_2, \dots, J_n\}$ l'ensemble d'images d'apprentissage annotées, où n représente le nombre d'images,

- $B=\{b_1, \dots, b_k\}$ un ensemble de blobs
- $M=\{m_1, m_2, \dots, m_m\}$ l'ensemble du vocabulaire de mots clés, où m représente le nombre de mots clés,
- $W=(w_1, w_2, \dots, w_m)$ le vecteur de mots clés, où w_i représente la fréquence du mot clé m_i qui annote l'image/blob dans l'ensemble C/B.

Chaque image J appartient à la collection C contenant un sous-ensemble de M (représenté par un vecteur de mots clés dans M). Chaque blob b_i de B contient un ensemble de régions visuellement similaires.

Etape 1 : Initialisation de l'espace de vecteur

Etant donné les vecteurs termes (mots clés) associés aux images dans le corpus, nous construisons la matrice **mots clés x images** A_{mn} dont les m lignes contiennent les mots clés et les n colonnes les images. Tel que : $A_{mn} = m_m \times J_n$.

Chaque cellule A_{ij} de la matrice A correspond à la fréquence w_{ij} du mot i dans l'image j . Tel que
$$w_{ij} = \begin{cases} 1 \text{ si } m_i \in \text{au vecteur de mot clé d'image } J_j \\ 0 \text{ sinon} \end{cases} \quad (10)$$

Etape 2 : Calcul de SVD

Nous calculons le SVD sur la matrice A pour une valeur de K . La matrice A est décomposée en valeurs singulières en produit de trois nouvelles matrices (cf. équations 5) :

- une matrice diagonale des valeurs singulières Σ de la matrice A , de taille $r \times r$,
- une matrice terme U de taille $m \times r$
- et la matrice d'images V^t de taille $n \times r$.

Nous sélectionnons les k premières valeurs singulières, en éliminant les valeurs proche de zéro et les vecteurs qui leurs correspondent dans la matrice U et V^t .

Le résultat dépend fortement du paramètre k et du corpus utilisé. Ce n'est qu'à travers les expériences que nous pouvons déterminer la meilleure valeur de k .

Etape 3 : Extraction de l'image

A ce niveau nous commençons la corrélation entre les blobs, qui représentent le contenu visuel, et la matrice terme x images, qui représente le contenu textuel (sémantique), en extrayant le numéro d'image correspondant afin de mapper les blobs aux vecteurs de mots.

Nous extrayons le numéro et le vecteur de mots clés de l'image à laquelle appartient la région. Notons que les régions appartenant à la même image de même blob sont traitées une seule fois car ils ont le même vecteur de mots clés de l'image.

Etape 4 : Projection du vecteur de mots clés

Après avoir préparé l'espace latent, nous projetons les vecteurs de mots clés textuels d'images extraits dans l'espace latent (concept) par $V = W \Sigma^{-1} U'$. La projection permet de réduire la dimension de vecteur, en comparant les vecteurs V et W ($k \ll m$).

Etape 5 : Calcul de similarité

Nous calculons la similarité entre deux vecteurs de mots clés contextuels de deux images auxquelles appartiennent deux régions par la fonction cosinus (cf. tableau 1), le choix de cette fonction est dû à la simplicité du calcul.

Soit V_1 et V_2 deux vecteurs contextuels, tel que $V_1=(v_{1_1},v_{1_2},\dots,v_{1_n})$ et $V_2=(v_{2_1},v_{2_2},\dots,v_{2_n})$. La formule de similarité est la suivante :

$$\text{Sim}(V_1, V_2) = \cos(\overrightarrow{V_1}, \overrightarrow{V_2}) = \frac{\sum_{i=1}^n (v_{1_i} \times v_{2_i})}{\sqrt{\sum_{i=1}^n v_{1_i}^2} \times \sqrt{\sum_{i=1}^n v_{2_i}^2}} \quad (11)$$

Le cosinus de l'angle entre deux vecteurs de mots clés contextuels d'images appartient à l'intervalle $[1, -1]$.

Etape 6 : Comparaison

Si le résultat de similarité est égal à 1, ce qui veut dire que les deux vecteurs sont similaires (ont les mêmes concepts), alors nous affectons l'un de ces vecteurs au vecteur blob. Mais s'il est égal à -1 cela veut dire que les deux vecteurs sont entièrement dissimilaires (aucune correspondance), nous affectons alors la fusion des deux vecteurs (Union) au vecteur blob. Dans le cas d'une similarité entre 0 et 1, si les deux vecteurs ont des concepts communs, nous affectons l'intersection de deux vecteurs au vecteur blob.

Nous appliquons cette comparaison sur toutes les régions de chaque blob sauf la dernière, où nous gardons l'ancien vecteur blob dans le cas d'une similarité entre 0 et 1.

Etape 7 : Annotation de blob

Le vecteur de mots clés contextuels qui annote le blob (vecteur blob) est mis à jour à chaque région. L'annotation finale du blob est propagée par un sous ensemble de mot clés.

La figure 21 montre le déroulement de l'algorithme AnnotB-LSA.

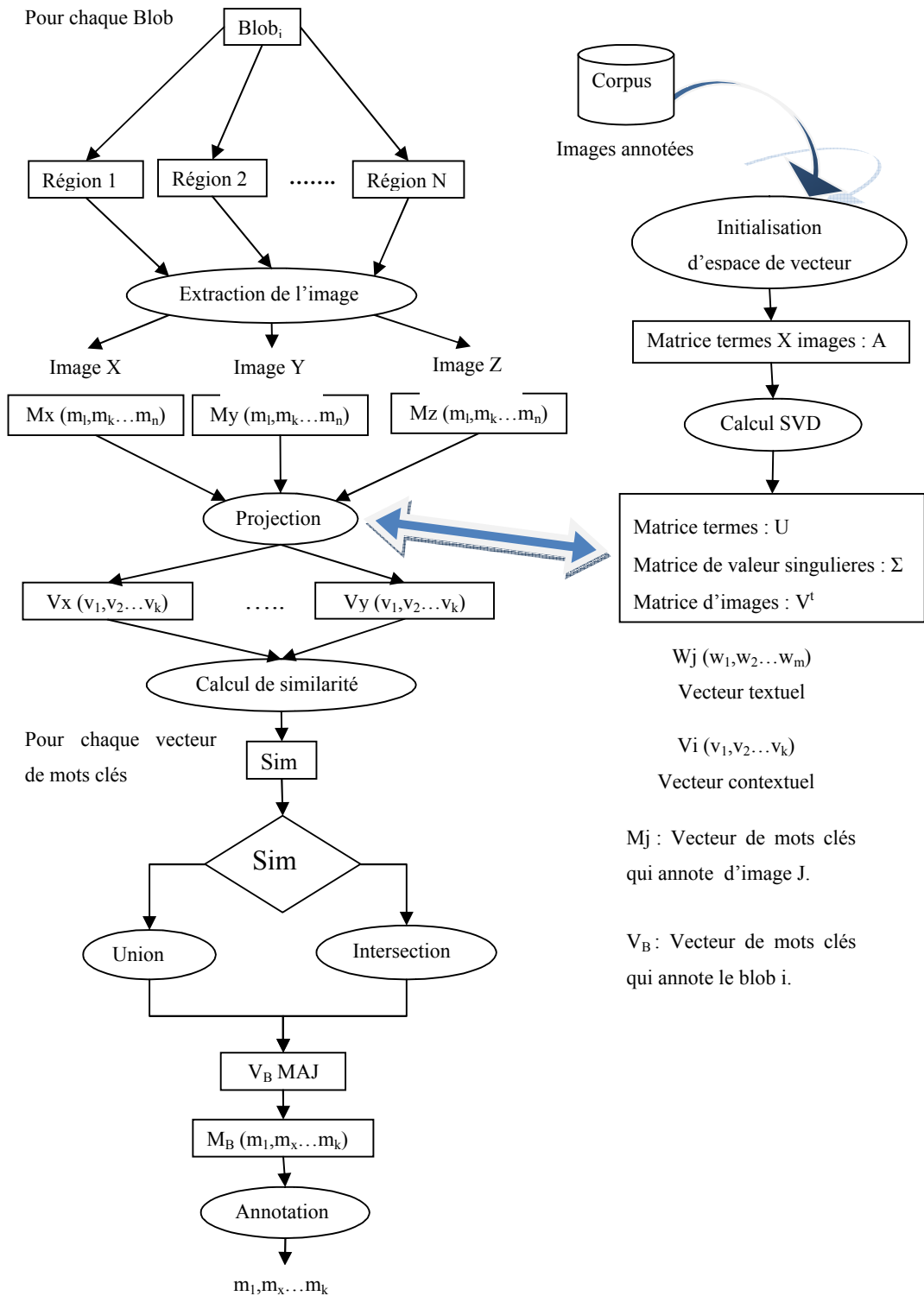


Figure 21: Schéma de déroulement de l'algorithme AnnotB-LSA

4.2.1.3. Base de données d'apprentissage

Nous représentons la base d'apprentissage dans le diagramme de classe de la figure 22.

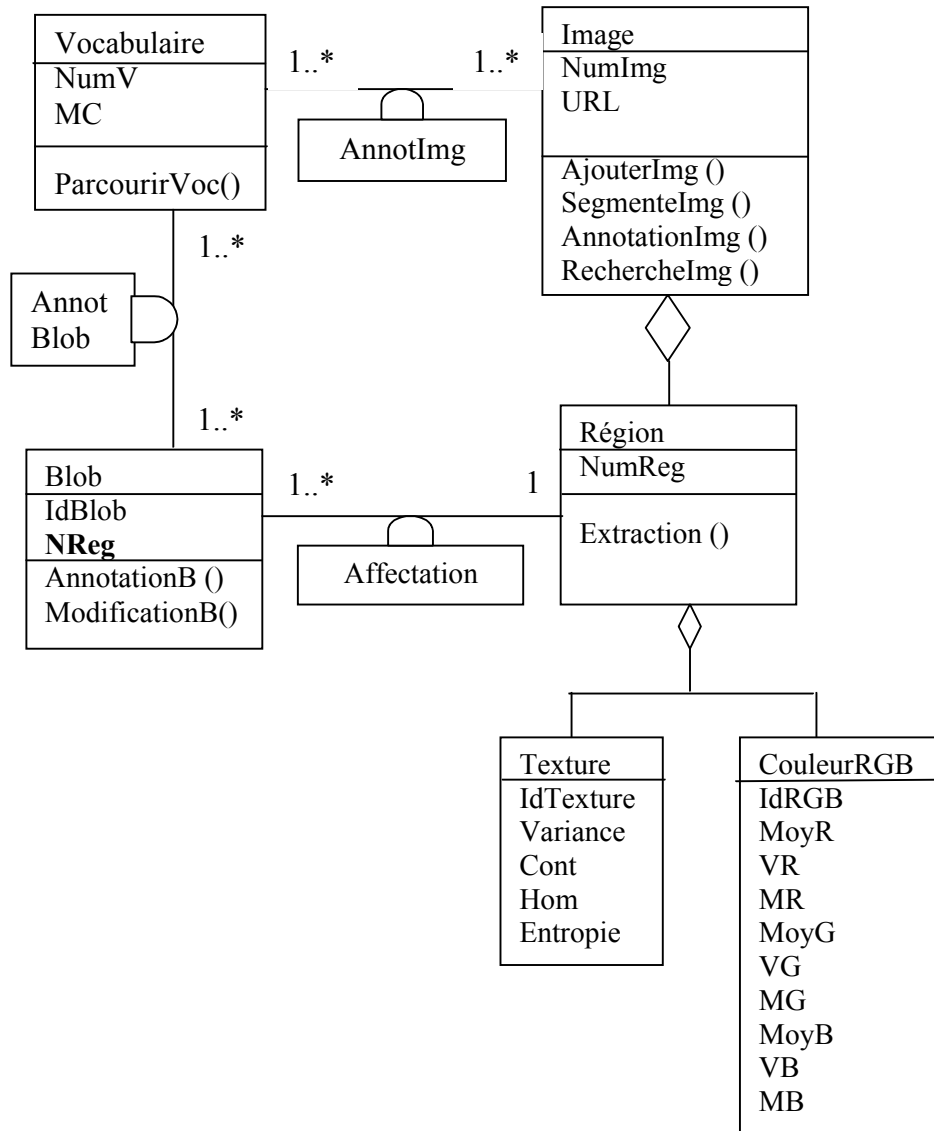


Figure 22 : Diagramme de classes du système d'auto annotation d'images

La description des classes est représentée par le tableau 4 suivant :

Classe	Attributs	Code Attribut	Type
Image	Numéro d'image URL de l'image	NumImg URL	Numérique Alphabétique
Région	Numéro de région	NumReg	Numérique
Blob	Identifiant du blob	IdBlob	Numérique

	Nombre de régions	NReg	Numérique
Vocabulaire	Numéro du Vocabulaire	NumV	Numérique
	Mots clés	MC	Caractère
CouleurRGB	Identifiant de la couleur d'espace RGB	IdRGB	Numérique
	Moyenne de la composante R	MoyR	Numérique
	Variance de la composante R	VR	Numérique
	Moment d'ordre 3 de la composante R	MR	Numérique
	Moyenne de la couleur de la composante G	MoyG	Numérique
	Variance de la composante G	VG	Numérique
	Moment d'ordre 3 de la composante G	MG	Numérique
	Moyenne de la couleur de la composante B	MoyB	Numérique
	Variance de la composante B	VB	Numérique
	Moment d'ordre 3 de la composante R	MB	Numérique
Texture	Identifiant de la texture	IdTexture	Numérique
	Variance	Variance	Numérique
	Contraste	Cont	Numérique
	Homogénéité	Hom	Numérique
	Entropie	Entropie	Numérique

Tableau 4: Description des classes

Les méthodes sur les classes sont brièvement expliquées comme suit :

La classe **Image** contient quatre méthodes :

AjouterImg () : elle permet d'ajouter une image dans la base d'images annotées.

SegmentImg () : elle permet de segmenter l'image en régions.

AnnotationImg () : elle permet d'annoter l'image.

RechercheImg () : elle permet de rechercher les images pertinentes.

La classe **Région** contient une méthode, **Extraction ()**, qui permet d'extraire les caractéristiques visuelles d'une région.

La classe **Blob** possède deux méthodes :

AnnotationB () : elle permet d'annoter les blobs visuellement similaires.

ModificationB () : elle permet de modifier un blob par l'affectation d'une région.

La classe **Vocabulaire** contient une opération qui parcourt l'ensemble du vocabulaire : **ParcourirVoc ()**.

4.2.2. Auto annotation d'une nouvelle image

Une fois la phase d'apprentissage effectuée, le système dispose d'une base de blobs annotés. Le système est alors prêt à annoter une nouvelle image. Cette phase se déclenche dès qu'une nouvelle image est collectée depuis le Web. Elle permet d'indexer textuellement et automatiquement l'image collectée.

Dans cette phase, la nouvelle image est segmentée en régions. Une extraction des caractéristiques visuelles de chaque région est nécessaire afin de les affecter aux blobs annotés issus de la phase d'apprentissage. L'image collectée sera annotée par les mots clés des blobs auxquels ses régions appartiennent.

Le processus d'auto-annotation d'une nouvelle image est présenté dans le diagramme d'activité de la figure 23.

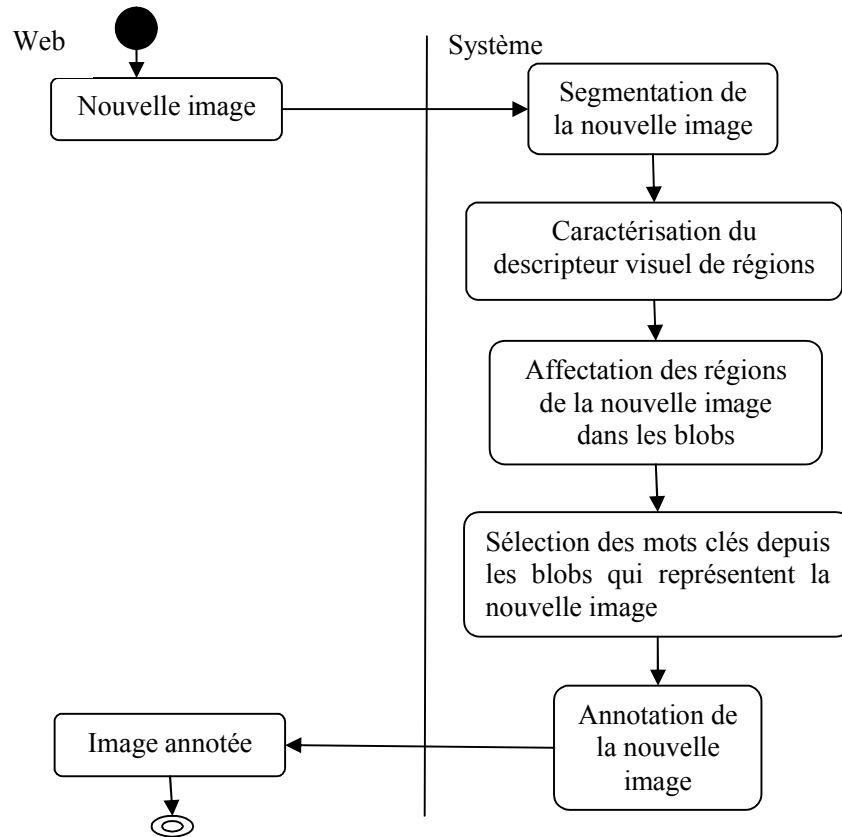


Figure 23: Diagramme d'activité global d'annotation automatique d'images

A. Segmentation

Chaque image collectée doit être segmentée par l'algorithme Normalized-Cuts en régions. Cela est présenté par le diagramme de la figure 24.

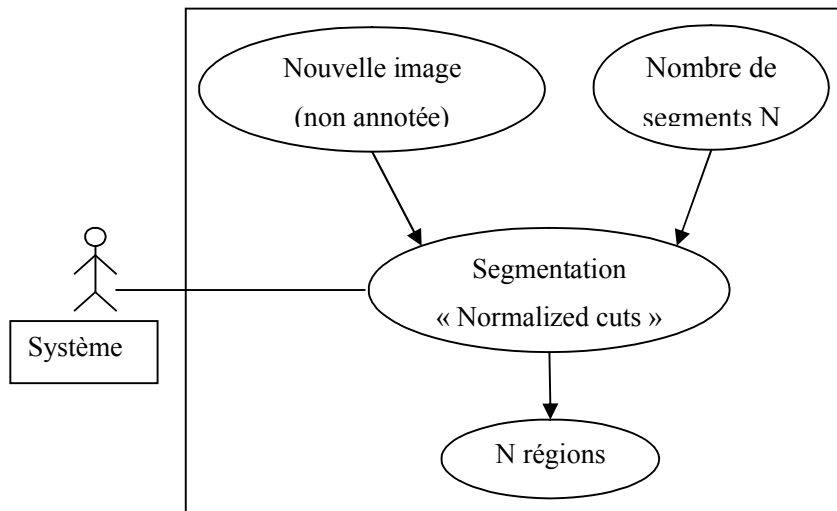


Figure 24: Segmentation d'une nouvelle image

B. Caractérisation

Pour chaque région d'image collectée, on extrait un vecteur de descripteur visuel. Les caractéristiques visuelles sont les mêmes que celles de l'étape d'apprentissage. Cela est présenté par le diagramme de la figure 18.

C. Affectation

Les régions de la nouvelle image seront affectées aux blobs annotés par l'algorithme suivant :

1. *Calcul des distances euclidiennes entre les régions de la nouvelle image et les vecteurs des centroïdes de blobs.*
2. *Sélection de la distance minimale pour chaque région.*
3. *Pour chaque région, sélection du blob qui correspond à la distance minimale.*
4. *Affectation de la région au blob sélectionné.*

Nous montrons le processus d'affectation par le diagramme d'activité de la figure 25.

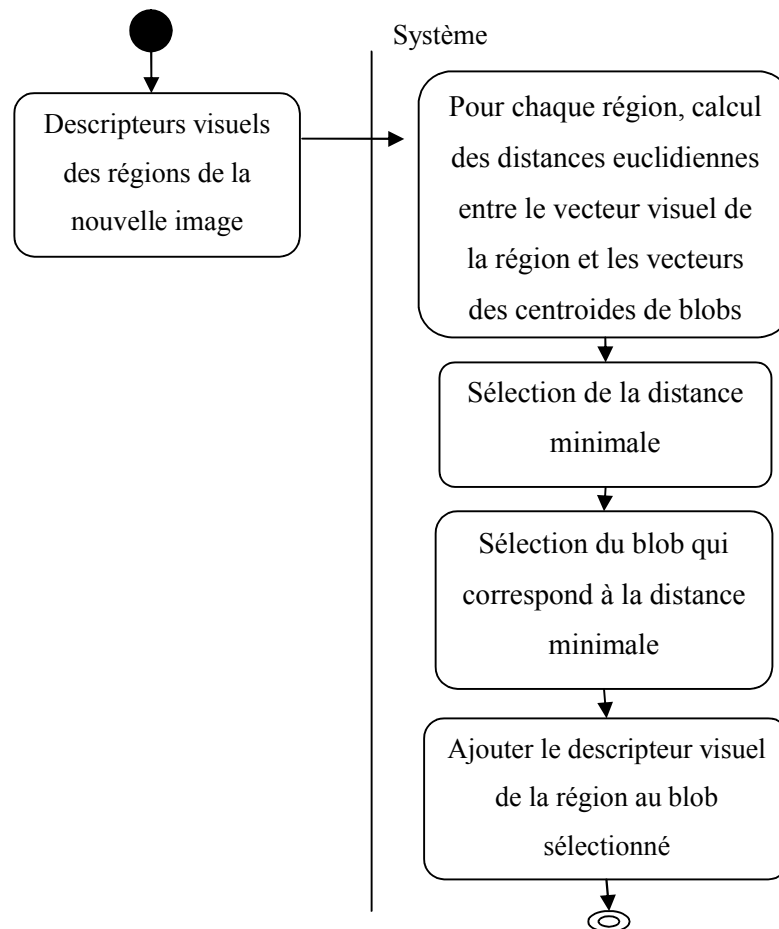


Figure 25 : Diagramme d'activité d'affectation

D. Annotation

Soit I une image de test représentée par un ensemble de blobs $I = \{b_x, b_y, b_i, b_z\}$, à chaque blob b correspondant un vecteur de mots clés textuels. Tels que $M_x (m_1, m_k \dots m_n)$, $M_y (m_1, m_k \dots m_n)$, $M_i (m_1, m_k \dots m_n)$, $M_z (m_1, m_k \dots m_n)$ correspondant respectivement aux b_x, b_y, b_i et b_z . Nous voulons sélectionner automatiquement l'ensemble de mots clés $\{m_1, \dots, m_n\}$ qui reflètent exactement le contenu de l'image.

La sélection des mots sémantiquement n'est pas évidente car il est difficile de déterminer le critère de choix efficace.

Nous représentons cette étape par le diagramme de la figure 26.

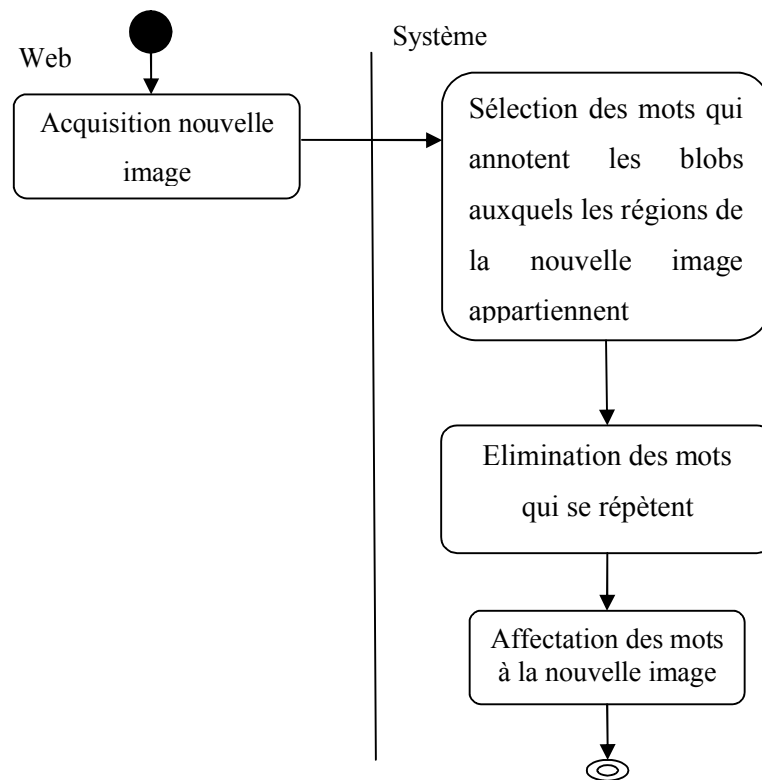


Figure 26: Diagramme d'activité d'annotation

La figure 27 présente le déroulement de la phase d'auto annotation d'une nouvelle image.

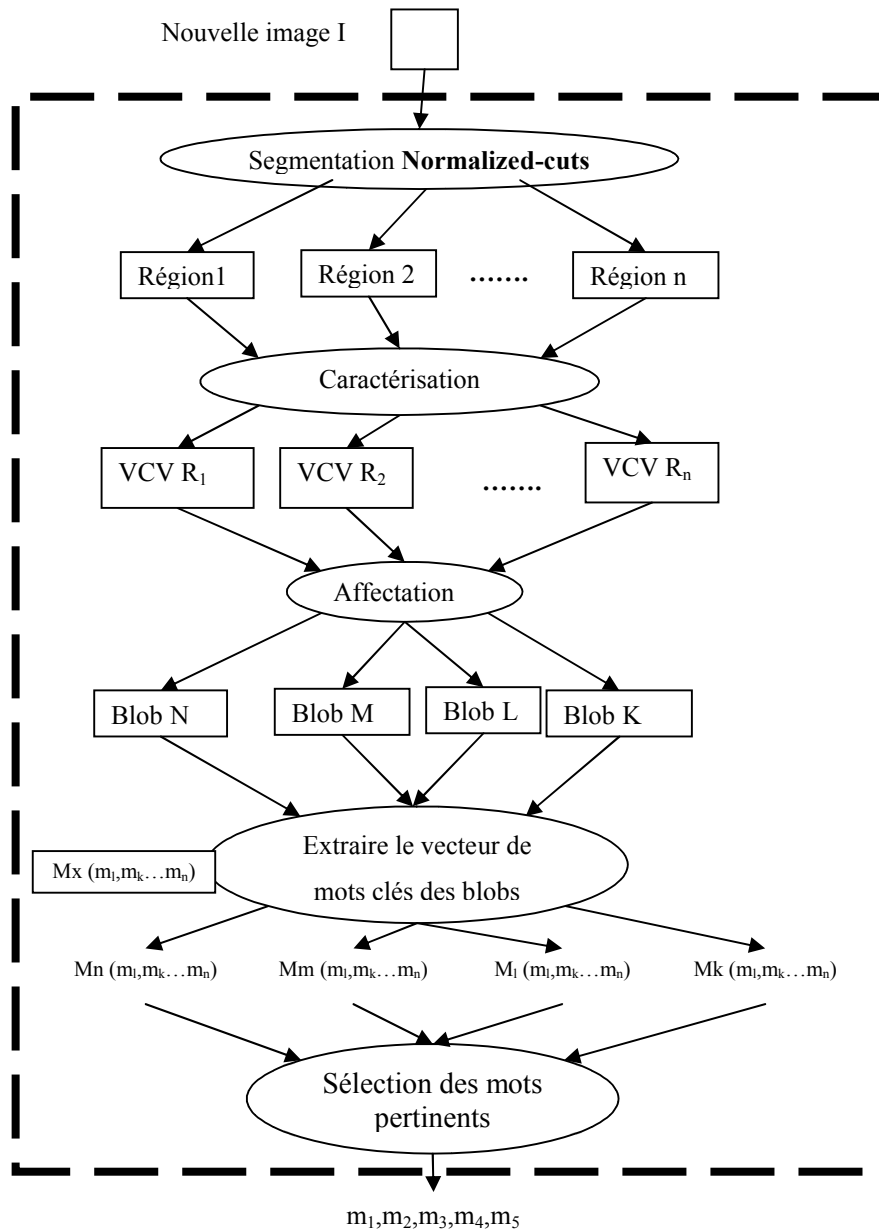


Figure 27: Auto-annotation d'une nouvelle image

4.3. Intégration du système d'auto-annotation d'images dans un moteur de recherche

Dans cette section nous allons montrer comment intégrer notre proposition dans un moteur de recherche d'images sur le web classique. Rappelons qu'un moteur de recherche classique comprend généralement les composants suivants : un système de Crawler et un système de recherche (cf. Chapitre 1).

Le système collecte les images à partir du Web, les annote et fournit un moyen de recherche de ces images par mots clé. Il s'agit donc d'un moteur de recherche d'images classique (figure 28).

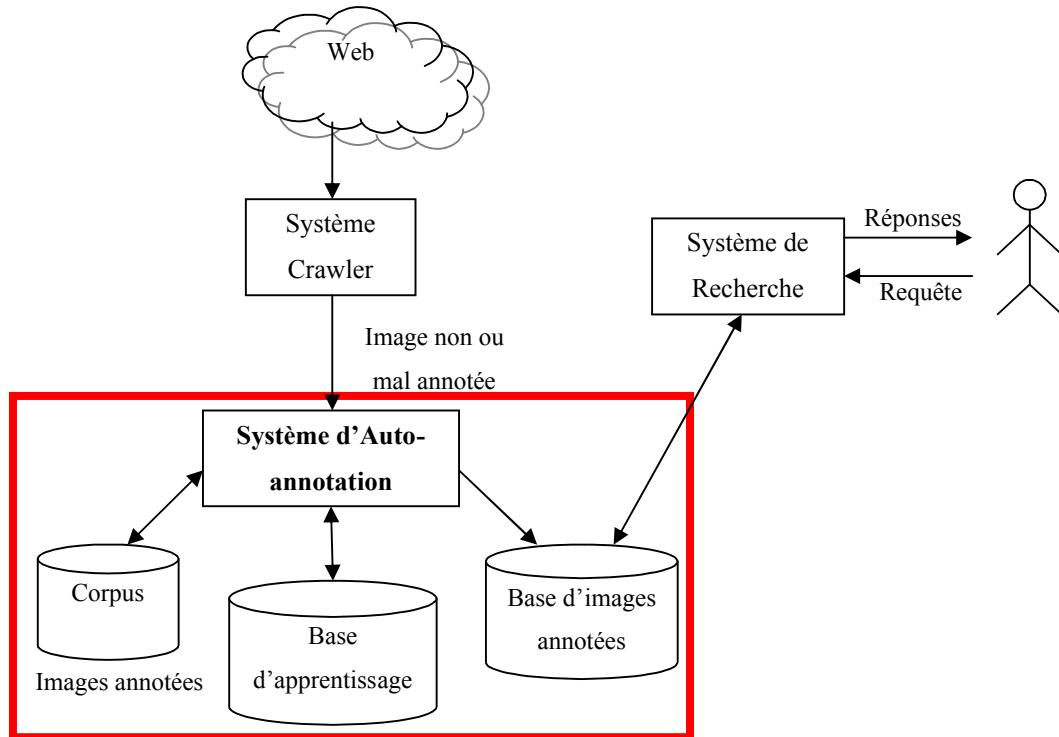


Figure 28: Moteur de recherche d'images classique

4.3.1. Système de collecte des images

Nous pouvons intégrer un système de collecte d'images classique (tel un Spider sur le Web) à notre système d'auto annotation. Ainsi, chaque image collectée depuis le Web par le Spider pourra être annotée par notre système.

4.3.1.1. Acquisition des images depuis le web

Dans ce système, un robot logiciel (Spider, Crawler) fait l'exploration du Web afin de rechercher et acquérir les images de différents types (JPG, BMP, GIF, PNG, JPE, JPEG...). La figure 29 montre un tel système :

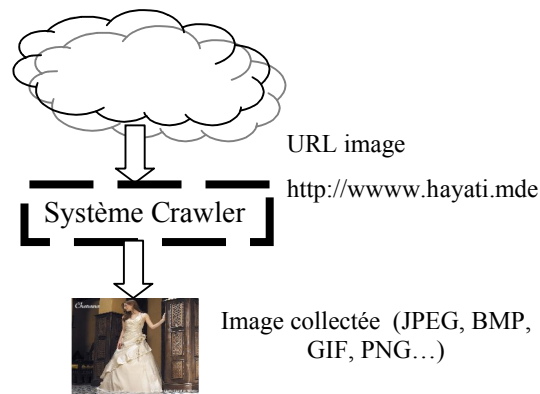


Figure 29 : Présentation du système de Collecte d'images

4.3.1.2. Annotation des images collectées

Le système d'auto-annotation d'images permet d'annoter les images collectées depuis le Web, afin de faciliter la recherche à l'utilisateur. Nous détaillons le système dans la figure 30.

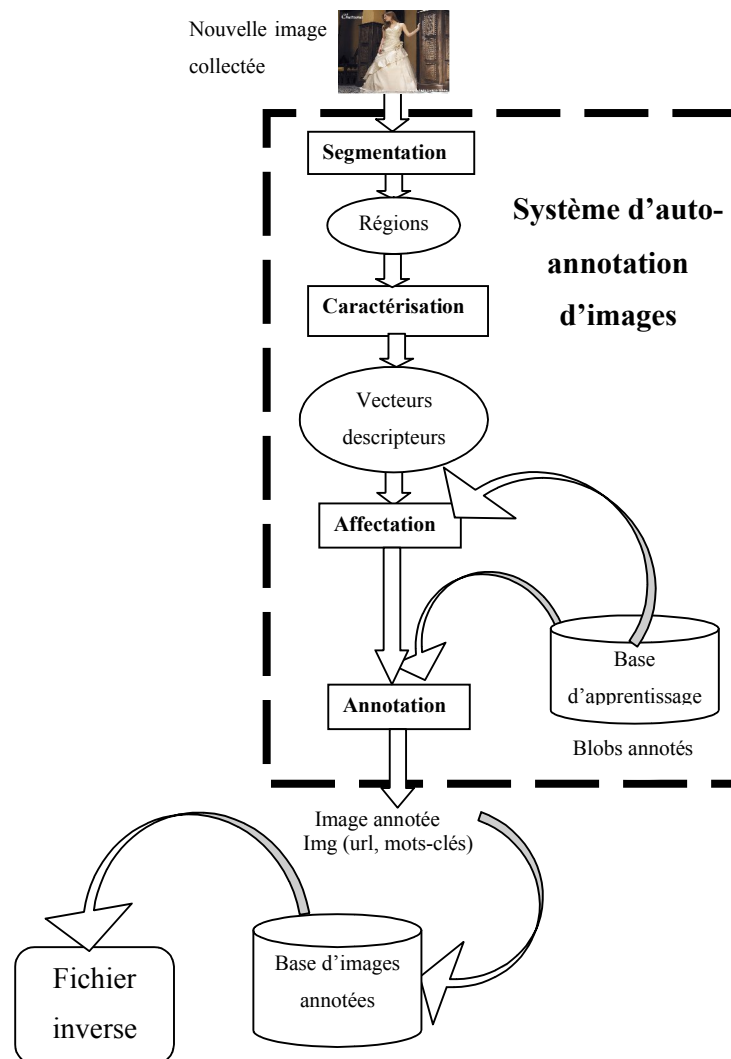


Figure 30 : Présentation du système d'auto-annotation d'images

Le schéma de la base d'images annotées est montré par le diagramme 31.

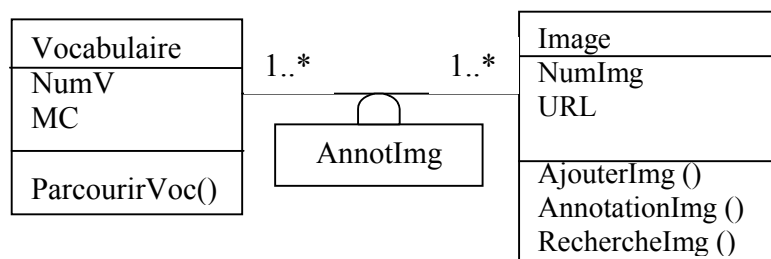


Figure 31 : Diagramme de classe de la base d'images annotées

Nous représentons physiquement les données (index) image de la base d'images par un fichier inverse qui utilise des blocs d'adressage [51].

4.3.2. Système de recherche des images annotées

Un système de recherche classique comprend trois phases : Interrogation, recherche et restitution. Nous allons intégrer l'index (fichier inverse) de la base d'images annotées dans un système de recherche classique (un moteur de recherche générique).

Le système de recherche classique d'image est représenté dans la figure 32.

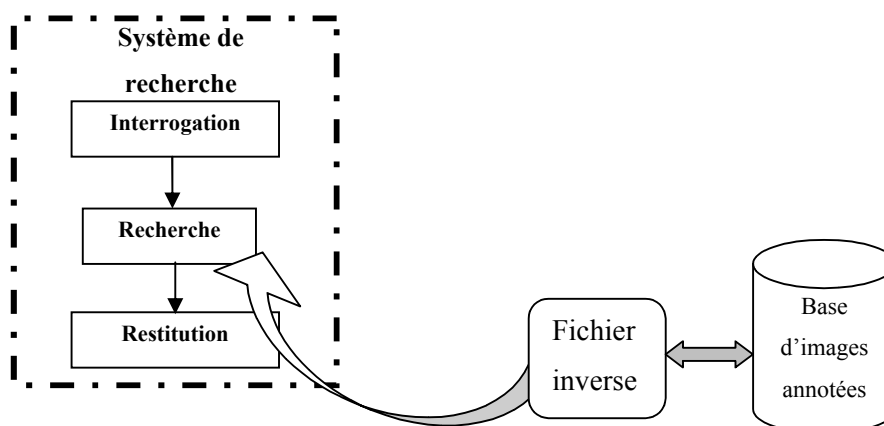


Figure 32 : Présentation du système de recherche d'images

4.3.2.1. Phase d'interrogation

A travers l'interface d'interrogation, l'utilisateur doit définir précisément ses besoins. La formulation d'un besoin par une requête déclenche le processus de recherche d'images.

Dans le but de permettre l'accès au fichier inverse, la requête doit être textuelles et contenir un ensemble de mots clés.

4.3.2.2. Phase de recherche

Cette phase permet de rechercher des images annotées. Le processus de recherche se déclenche dès qu'une requête est soumise, en comparant les mots de la requête avec ceux du fichier inverse (index). Selon le modèle utilisé, un appariement est appliqué pour calculer la similarité entre les mots. Nous prenons l'exemple du modèle booléen où l'appariement est basé sur la présence ou l'absence des mots clés de la requête dans les images. Dans le cas favorable où les images répondent exactement à la requête, une sélection de leurs numéros est faite afin d'extraire leurs URL.

4.3.2.3. Phase de restitution

La dernière étape dans un système de recherche d'images est la restitution des résultats de recherche. Cette phase permet à l'utilisateur de trouver rapidement les réponses à sa requête.

Comme les résultats retournés sont des images, nous affichons les images avec leurs URL et mots clés qui les annotent sous forme d'une présentation par liste triée par la fréquence d'apparition des mots de la requête.

La figure 33 donne un aperçu d'une restitution des résultats de recherche Flower.



Flower Glitter
500 x 500 - 172 ko - gif
myspace.com



Flower
1024 x 768 - 475 ko - jpg
pkarus-team.blogspot.com



[affection], **Flower**
320 x 300 - 13 ko - jpg
affection.net



Figure 33: Aperçu d'une restitution des résultats

4.3.3. Aperçu du moteur de recherche images avec le système d'auto-annotation

Nous présentons le processus complet du moteur de recherche images dans la figure 34

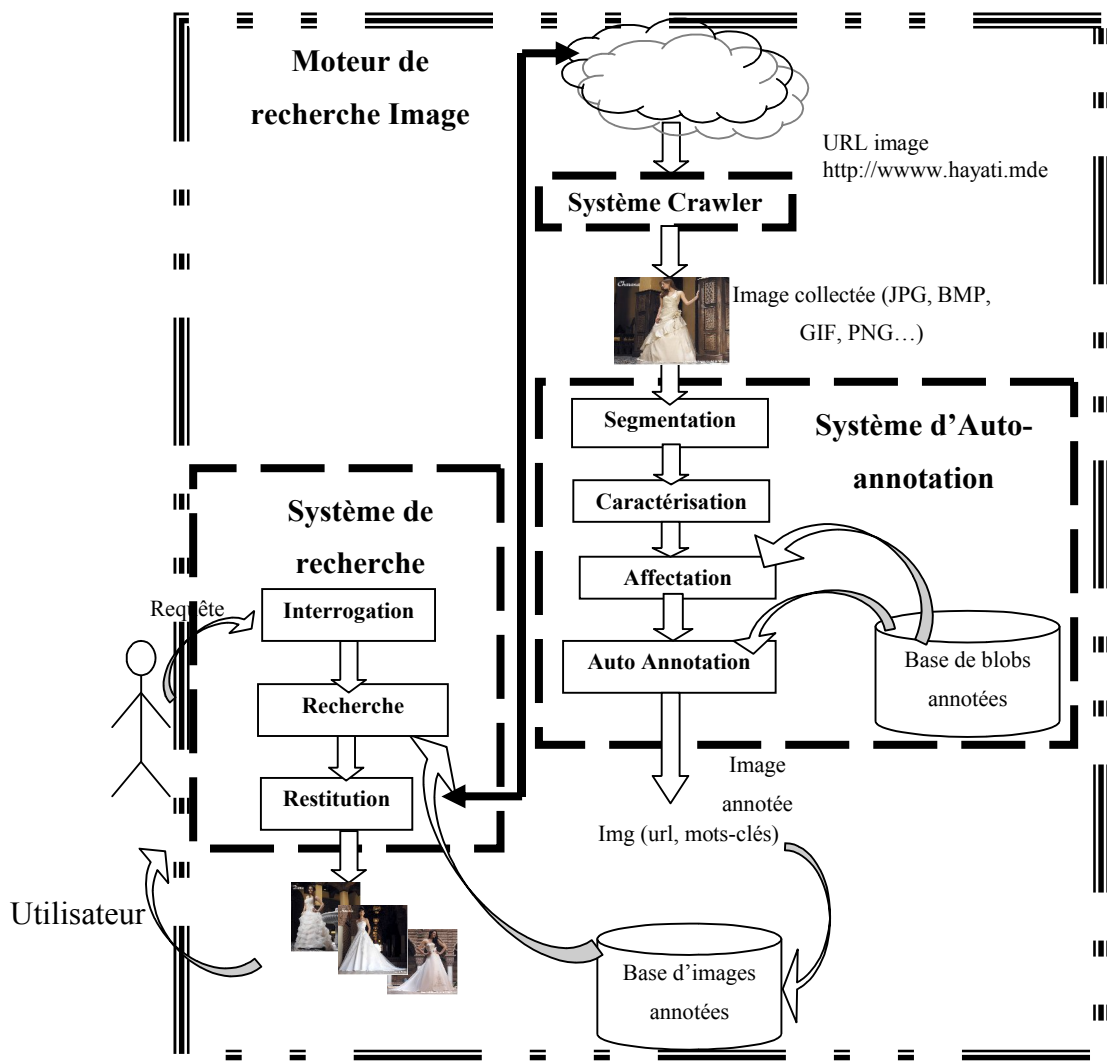


Figure 34: Présentation du processus complet du moteur de recherche images

Le processus du moteur de recherche images est résumé dans le diagramme de collaboration de la figure 35.

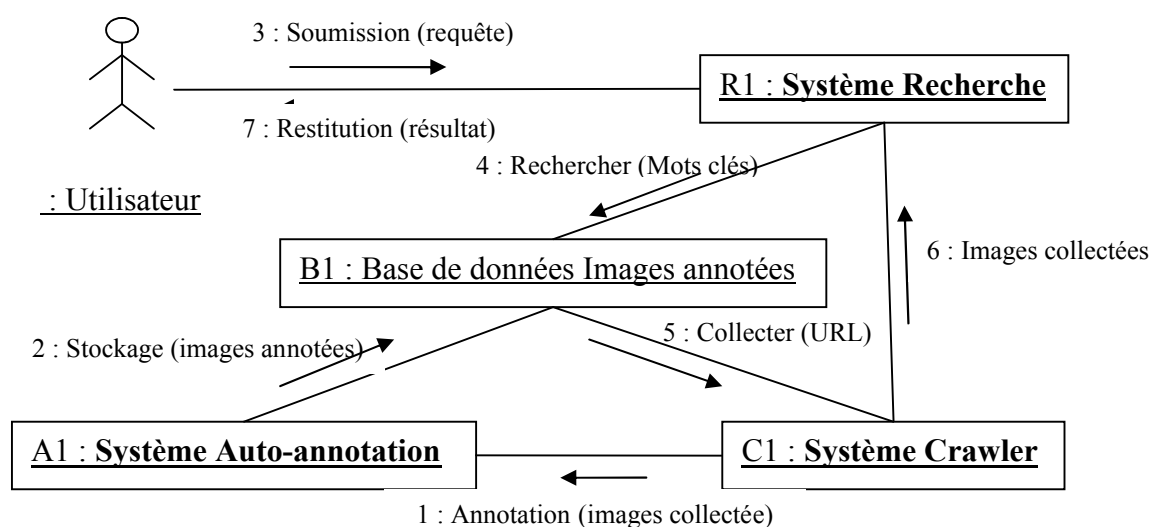


Figure 35 : Diagramme de collaboration du processus complet du moteur de recherche images

4.4. Conclusion

Dans ce chapitre, nous avons présenté un système d'auto annotation des images en utilisant un apprentissage par un corpus d'images annotées. Le système est basé sur la corrélation entre les blobs de régions d'images (aspect visuel) et les annotations des images du corpus (aspect textuel). Les blobs sont annotés par un nouvel algorithme AnnotB-LSA basé sur l'analyse sémantique latente.

Ce système peut être intégré dans un moteur de recherche d'images sur le Web par le biais d'une base d'images annotées qui joue le rôle d'index.

Dans le chapitre suivant, nous présenterons la réalisation d'un prototype de notre proposition afin d'évaluer la qualité d'annotation d'images par notre système

CHAPITRE 5

MISE EN ŒUVRE DU SYSEME ET APPLICATION AU CORPUS COREL

5.1. Introduction

Dans ce chapitre, nous allons d'abord décrire l'implémentation de notre proposition à travers un système logiciel d'annotation. Par la suite, nous réalisons un prototype afin de tester notre système en comparant les résultats obtenus avec ceux du corpus Corel.

5.2. Description du corpus Corel

Afin d'expérimenter notre travail, nous avons travaillé sur le même Corpus Corel (images et données) que Barnard et al [110]. Corel est une base d'image contenant 16000 images annotées manuellement de 1 à 5 mots. Ces derniers sont pris d'un vocabulaire de 267 mots dont les plus fréquents sont au nombre de 157 mots.

Les données du corpus [127] sont organisées dans 10 échantillons différents. Chaque échantillon comporte trois sous-ensembles disjoints correspondant aux données d'apprentissage, données de test (test_1) et autres données de test sévère (test_3).

Pour la validation de notre système, nous avons choisi d'utiliser l'échantillon numéro 009. Afin de construire les blobs annotés, nous avons considéré l'ensemble d'apprentissage contenant 5239 images et l'ensemble de test (test_1) contenant 1801 images.

Comme tous les autres échantillons, l'échantillon 009 contient un ensemble de fichiers. Nous avons travaillé avec les quatre fichiers suivants :

- Words : il contient un vocabulaire de 154 mots.
- Document_words, test_1_document_words et test_3_document_words : ces trois fichiers contiennent les annotations d'images. Chaque ligne est une liste des numéros de cinq colonnes correspondant aux indices dans le fichier vocabulaire Words. Comme les images sont annotées de 1 à 5 mots, les colonnes qui ne contiennent pas d'indice de vocabulaire sont représentées par le numéro -99.
- Word_counts, test_1_word_counts et test_3_word_counts : ces fichiers représentent le nombre des mots pour chaque image.
- Image_nums, test_1_image_nums et test_3_image_nums : ces fichiers contiennent les numéros d'image du corpus Corel.

5.3. Environnement du système d'auto-annotation d'images

L'environnement d'implémentation de notre système comprend les éléments suivants :

- Une source d'images où se trouve un spider afin de collecter les images depuis le web.
- Une base d'images annotées où nous stockons les images annotées avec leurs URL.
- Une machine d'application où se trouve notre système d'auto-annotation d'images.
- Un système de recherche d'images qui joue le rôle d'intermédiaire entre la base d'images annotées et le client.
- Un client pour rechercher les images à travers un navigateur web.

Nous avons développé notre système d'auto-annotation avec le framework Matlab version 7 en raison de sa facilité de développement et de ses possibilités de traitement d'images.

Les données du processus d'apprentissage sont sauvegardées dans une base de données relationnelle sous le SGBD SQL Server. La base des images annotées est également sauvegardée avec le même SGBD.

La description de l'environnement de notre système est représentée dans la figure 36.

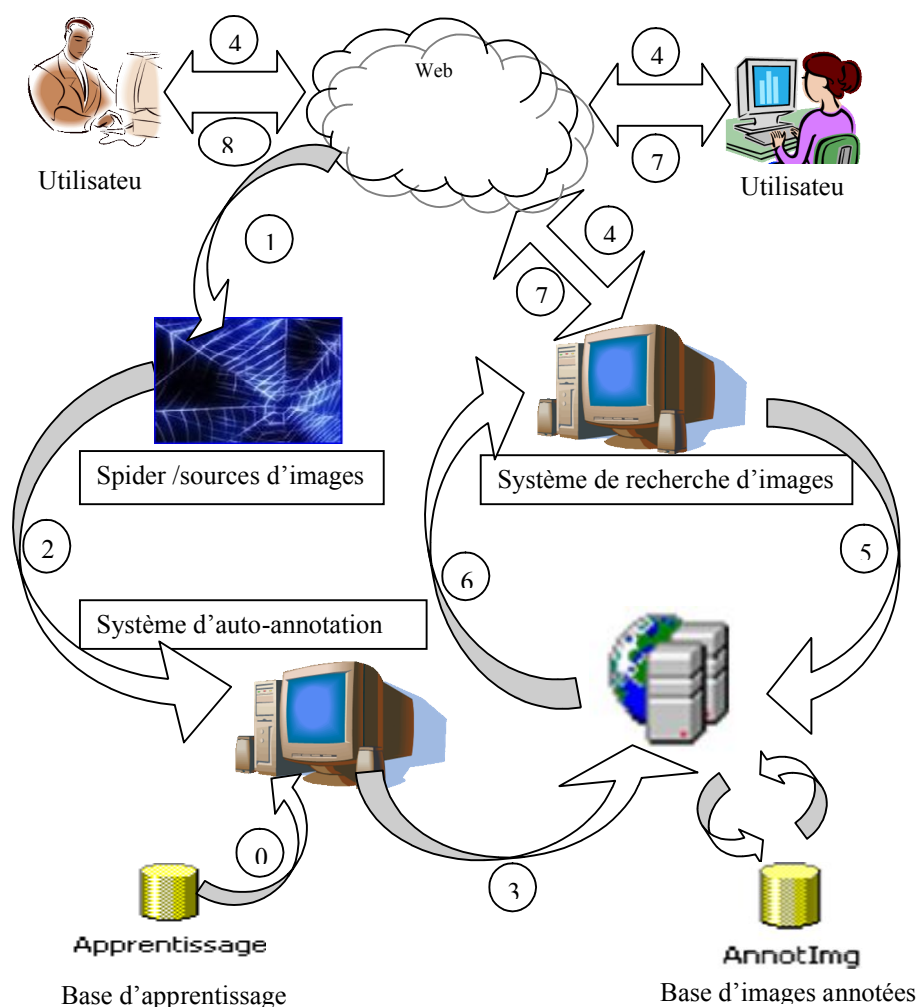


Figure 36 : Environnement du système

L'étape initiale est l'apprentissage (0) qui utilise le corpus Corel contenu dans la base d'apprentissage. Le spider collecte les images (1) depuis le web et les envoie au système d'auto-annotation d'images (2). Une fois l'image annotée, elle doit être stockée (3) dans la base d'images annotées qui sera utilisée pour la recherche.

Afin de rechercher les images sur le web, un utilisateur soumet une requête textuelle (4). Le système de recherche décompose la requête en un ensemble de

mots clés et lance la recherche (5) dans la base d'images annotées (fichier inverse) pour extraire les URLs des images pertinentes (6). Les URLs des images trouvées ainsi que leurs aperçus sont envoyées à l'utilisateur(7).

5.4. Architecture logicielle du système d'auto-annotation d'images

Notre système d'auto-annotation d'images est orienté réseau pour permettre d'annoter les images collectées depuis le Web. De plus, la recherche sur la base d'images annotées s'effectue à partir de n'importe quel navigateur Web. Pour cela, nous allons utiliser SQL server [123] pour les données et Matlab [124] pour les traitements.

Le système se compose d'un corpus d'images annotées pour l'apprentissage, un ensemble de modules pour le traitement et une base de données relationnelle d'images afin de stocker les nouvelles images annotées.

Notons que la base de données des images annotées ne stocke pas les images (en tant que fichier image) mais seulement leur URL et les mots qui les annotent.

Le système se déroule en deux phases : la phase d'apprentissage qui regroupe la construction des blobs et l'auto-annotation des blobs, et la phase d'auto-annotation d'une nouvelle image collectée (figure 37).

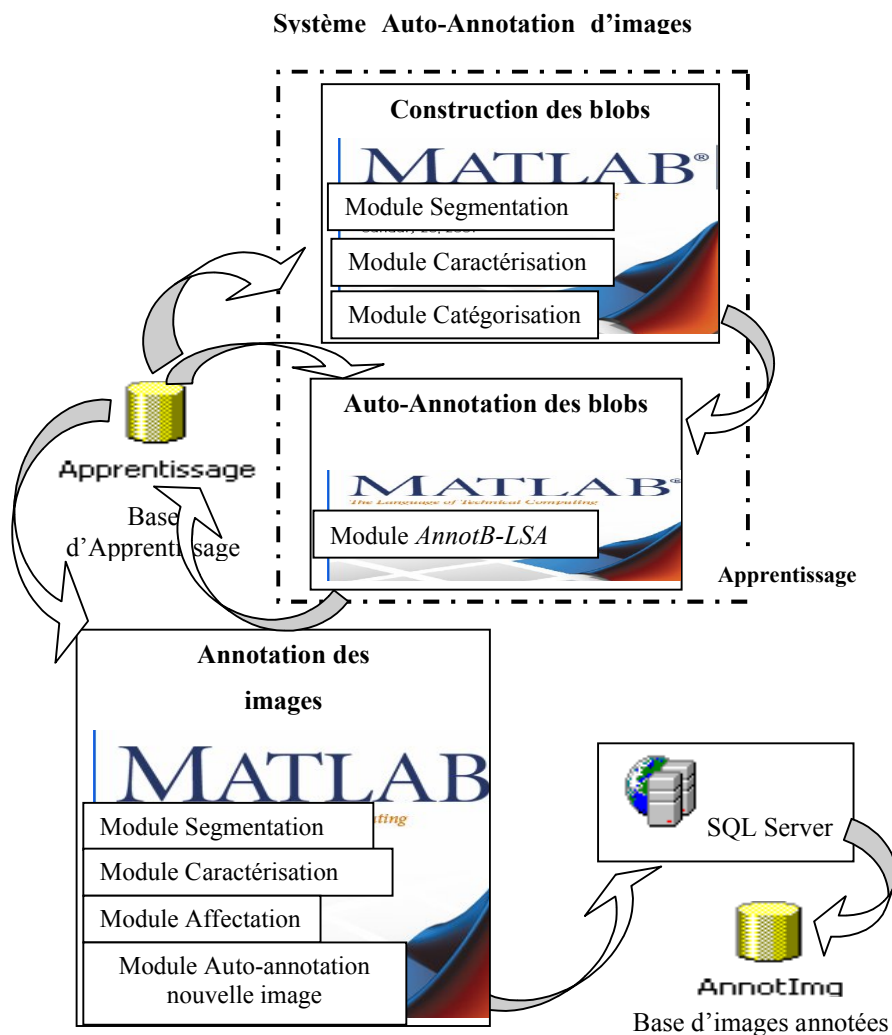


Figure 37 : Architecture logicielle du système d'auto-annotation d'images

5.4.1. Phase d'apprentissage

L'apprentissage est appliqué sur 5239 images. Pour une image donnée, le processus se déroule comme suit:

5.4.1.1. Construction des blocs

Cette étape permet de créer les blocs qui regroupent les régions visuellement similaires. Elle comprend les modules de segmentation, d'extraction et de catégorisation.

A. Module segmentation

Ce module permet de segmenter les images du corpus en régions. Il est le premier à être exécuté (figure 38).

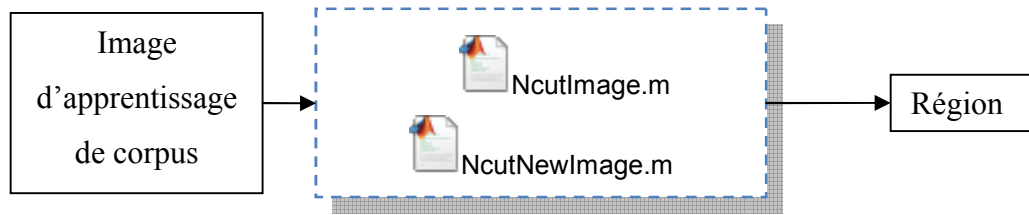


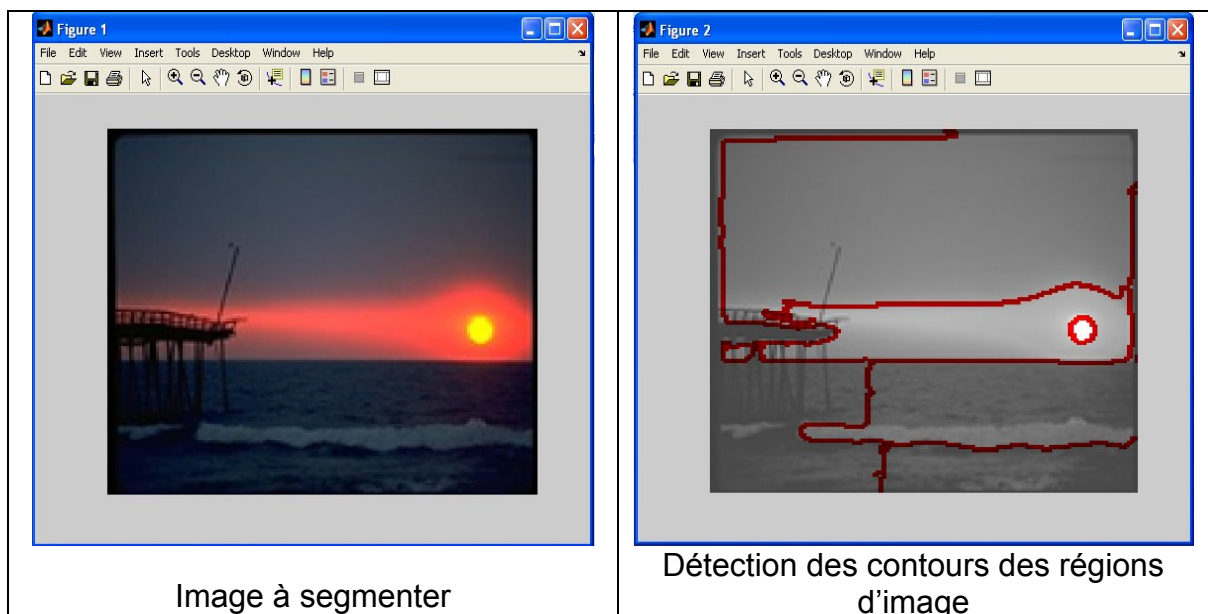
Figure 38 : Module de segmentation

Dans ce module, nous utilisons l'algorithme NormalizedCut [125] pour la segmentation des images. Cet algorithme se résume en quatre étapes :

1. *Construction d'un graphe pondéré G à partir de l'image, tel que les nœuds représentent les pixels.*
2. *Calcul de la matrice des poids W entre chaque pixel*
3. *Bipartition du graphe en utilisant le vecteur propre avec la seconde plus petite valeur propre.*
4. *Répéter les étapes 2 jusqu'à 4. Arrêter si la valeur $Ncut$ est plus grande qu'une valeur seuil prédéfinie. En outre, arrêter si le nombre total de nœuds dans la partition est plus petit qu'une valeur seuil prédéfinie.*

Comme la majorité des travaux utilisent l'algorithme Ncuts en segmentant l'image en cinq régions, nous avons gardé le même nombre de régions.

Le résultat d'exécution de ce module est montré dans la figure 39 :



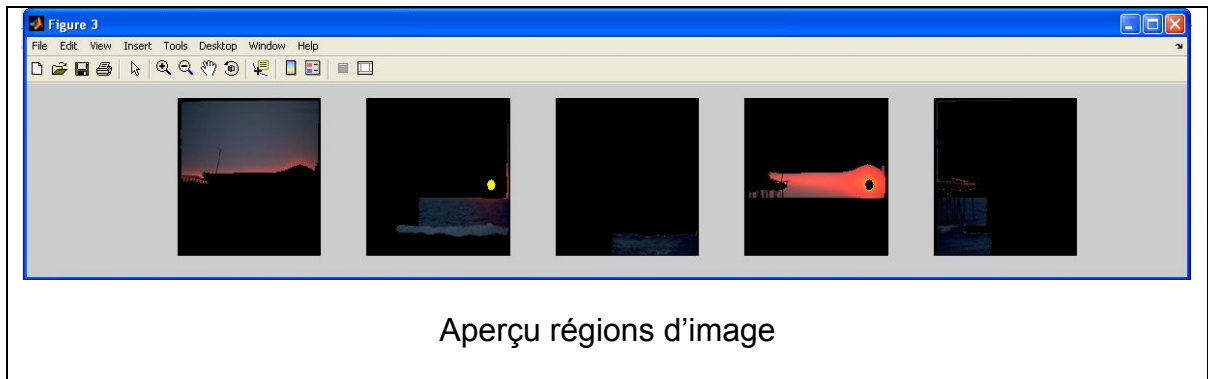


Figure 39 : Segmentation d'image

B. Module caractérisation

Ce module permet de caractériser les régions obtenues par segmentation (figure 40). Pour chaque région obtenue, nous extrayons les moments statistiques de la couleur : la moyenne, la variance et les moments d'ordre trois de chaque composante d'espace RGB. Nous extrayons également les cinq paramètres de la caractéristique visuelle Texture : la variance, l'homogénéité, le contraste et l'entropie.

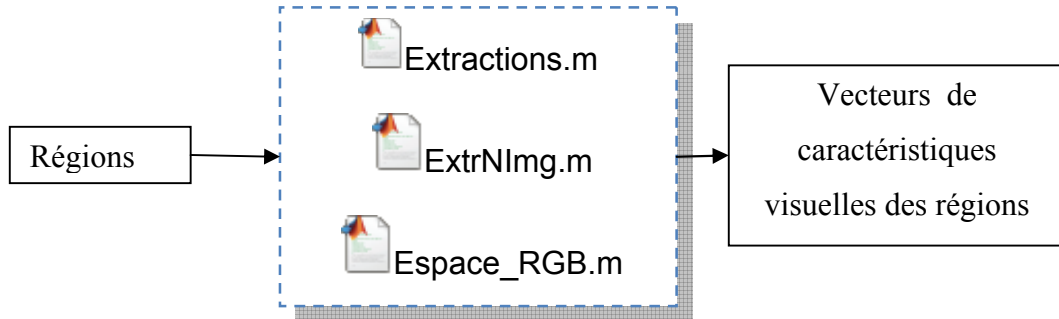


Figure 40 : Module extraction

Les résultats de l'exécution de ce module pour toutes les régions de l'image sont résumés dans la figure 41 :


```

Command Window
i To get started, select MATLAB Help or Demos from the Help menu.

Calcul de la matrice de cooccurrence de la région 1 patientez SVP...
La variance de la région 1 est : 21777876088746
La contraste de la région 1 est : 1913344
Homogeniete de la région 1 est : 44642.4133
Entropie de la région 1 est : -392079.4152
La correlation de la région 1 est : -1.304e-022
Calcul de la matrice de cooccurrence de la région 2 patientez SVP...
La variance de la région 2 est : 75891145569111
La contraste de la région 2 est : 3720606
Homogeniete de la région 2 est : 45662.8765
Entropie de la région 2 est : -472221.2914
La correlation de la région 2 est : -1.5466e-021
Calcul de la matrice de cooccurrence de la région 3 patientez SVP...
La variance de la région 3 est : 101812675655833
La contraste de la région 3 est : 150118
Homogeniete de la région 3 est : 48468.5724
Entropie de la région 3 est : -510984.5641
La correlation de la région 3 est : -1.087e-020
Calcul de la matrice de cooccurrence de la région 4 patientez SVP...
La variance de la région 4 est : 85447628404320
La contraste de la région 4 est : 3775776
Homogeniete de la région 4 est : 47821.5275
Entropie de la région 4 est : -488674.0753
La correlation de la région 4 est : -3.9983e-022
Calcul de la matrice de cooccurrence de la région 5 patientez SVP...
La variance de la région 5 est : 74370623887389
La contraste de la région 5 est : 1136258
Homogeniete de la région 5 est : 45766.5913
Entropie de la région 5 est : -472618.4111
La correlation de la région 5 est : -3.3562e-021
>> |

```

Figure 41 : Valeurs des paramètres texturaux pour les régions considérées

C. Module catégorisation

Ce dernier module regroupe les régions visuellement similaires dans des blobs et s'exécute après le module d'extraction (figure 42). Nous utilisons l'algorithme de k-means sur les vecteurs de caractéristiques visuelles des régions (données numériques) afin de construire les blobs de régions visuellement similaires.

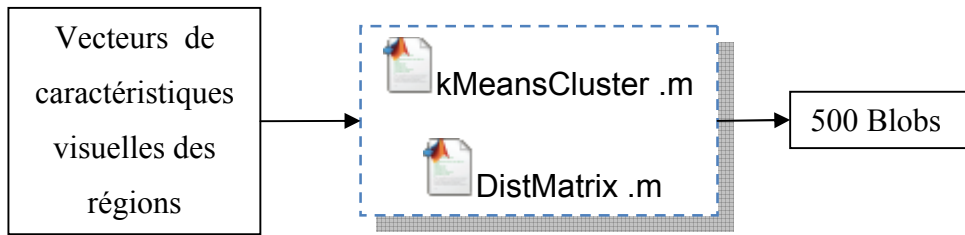


Figure 42 : Module catégorisation

Nous avons fixé le paramètre K à 500 afin de catégoriser les régions dans 500 blobs en calculant les distances euclidiennes entre les vecteurs des caractéristiques visuelles des régions. Nous avons utilisé l'algorithme de K-means et de distance euclidienne qui se trouvent dans la référence [126].

Dans la figure 43, nous montrons les résultats de l'exécution ce module sur un seul blob (par exemple le blob numéro 222).

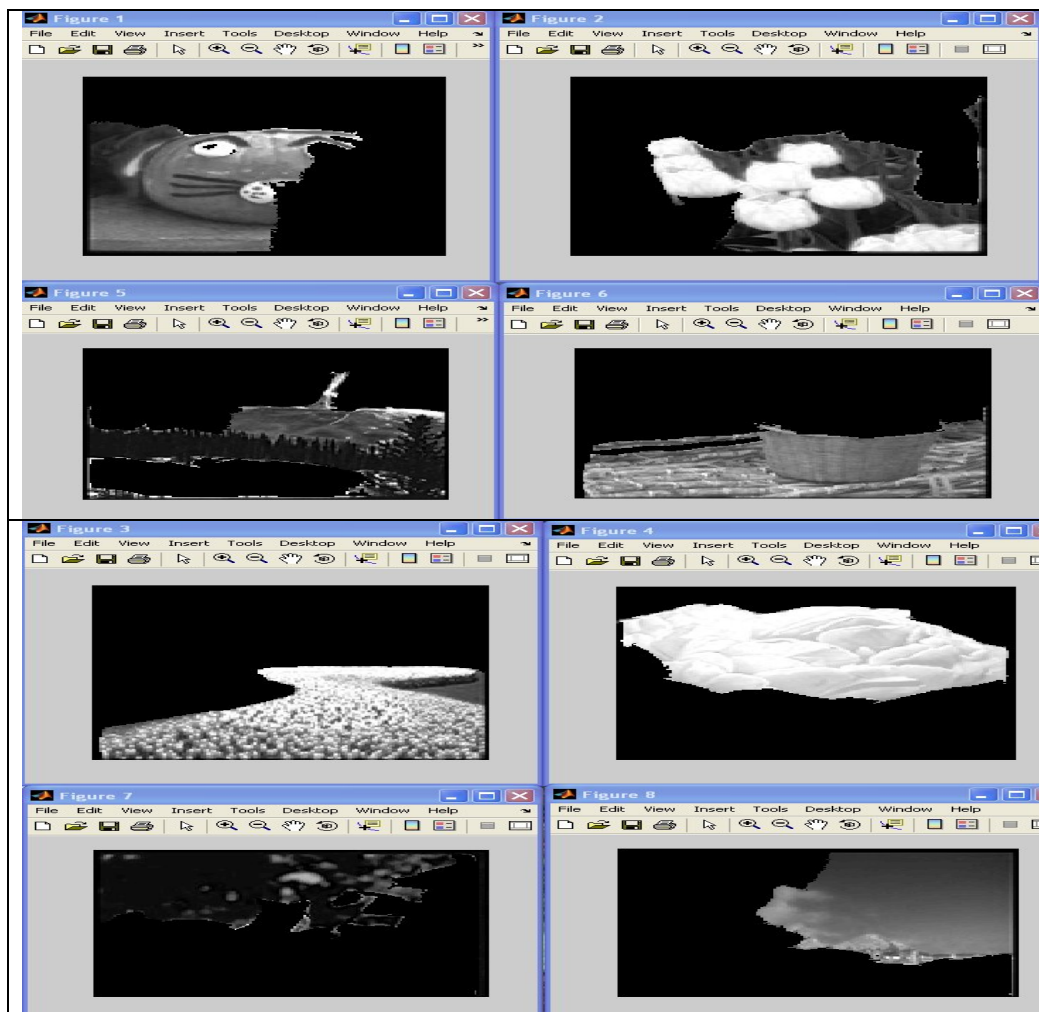


Figure 43 : Représentation des régions du blob 222

5.4.1.2. Auto-annotation de blobs

L'auto-annotation de blobs correspond à l'algorithme *AnnotB-LSA*. Elle comprend un seul module qui est le dernier à exécuter dans la phase d'apprentissage (figure 44). L'algorithme *AnnotB-LSA* est implémenté sur les 500 blobs.



Figure 44 : Module auto-annotation blobs

Une partie (16 blobs) du résultat de l'exécution de ce module est montrée dans la figure 45.

Command Window	Command Window	Command Window
<pre> To get started, select MATLAB Help or Demos from the Help menu. Blob numéro : 1 annoté par : field Blob numéro : 1 annoté par : flag Blob numéro : 1 annoté par : food Blob numéro : 1 annoté par : people Blob numéro : 1 annoté par : petals Blob numéro : 1 annoté par : plants Blob numéro : 1 annoté par : reef Blob numéro : 1 annoté par : tree Blob numéro : 1 annoté par : vegetables Blob numéro : 1 annoté par : water Blob numéro : 2 annoté par : water Blob numéro : 2 annoté par : vegetables Blob numéro : 2 annoté par : tree Blob numéro : 2 annoté par : pumpkin Blob numéro : 2 annoté par : plants Blob numéro : 2 annoté par : people Blob numéro : 2 annoté par : flag Blob numéro : 2 annoté par : flowers Blob numéro : 3 annoté par : flowers Blob numéro : 3 annoté par : cactus Blob numéro : 3 annoté par : coast Blob numéro : 3 annoté par : display Blob numéro : 3 annoté par : leaves Blob numéro : 3 annoté par : people Blob numéro : 3 annoté par : pattern Blob numéro : 3 annoté par : polar Blob numéro : 3 annoté par : sky Blob numéro : 3 annoté par : snow Blob numéro : 3 annoté par : tree Blob numéro : 3 annoté par : town Blob numéro : 3 annoté par : walls </pre>	<pre> To get started, select MATLAB Help or Demos from the Help menu. Blob numéro : 5 annoté par : clouds Blob numéro : 5 annoté par : fish Blob numéro : 5 annoté par : plants Blob numéro : 5 annoté par : vegetables Blob numéro : 6 annoté par : temple Blob numéro : 6 annoté par : water Blob numéro : 6 annoté par : rabbit Blob numéro : 6 annoté par : snow Blob numéro : 6 annoté par : stone Blob numéro : 6 annoté par : flowers Blob numéro : 6 annoté par : clouds Blob numéro : 6 annoté par : cactus Blob numéro : 6 annoté par : building Blob numéro : 6 annoté par : cliffs Blob numéro : 6 annoté par : leaves Blob numéro : 6 annoté par : mountains Blob numéro : 6 annoté par : ocean Blob numéro : 7 annoté par : field Blob numéro : 7 annoté par : petals Blob numéro : 7 annoté par : people Blob numéro : 7 annoté par : tree Blob numéro : 8 annoté par : walls Blob numéro : 8 annoté par : people Blob numéro : 8 annoté par : polar Blob numéro : 8 annoté par : flowers Blob numéro : 9 annoté par : flowers Blob numéro : 9 annoté par : pumpkin Blob numéro : 9 annoté par : people Blob numéro : 9 annoté par : tree Blob numéro : 9 annoté par : water Blob numéro : 10 annoté par : water Blob numéro : 10 annoté par : tree Blob numéro : 10 annoté par : people Blob numéro : 10 annoté par : pumpkin Blob numéro : 10 annoté par : flowers </pre>	<pre> To get started, select MATLAB Help or Demos from the Help menu. Blob numéro : 11 annoté par : currency Blob numéro : 11 annoté par : mountains Blob numéro : 11 annoté par : street Blob numéro : 11 annoté par : people Blob numéro : 11 annoté par : tree Blob numéro : 12 annoté par : water Blob numéro : 12 annoté par : rapids Blob numéro : 12 annoté par : flowers Blob numéro : 12 annoté par : fish Blob numéro : 13 annoté par : flowers Blob numéro : 13 annoté par : desert Blob numéro : 13 annoté par : clouds Blob numéro : 13 annoté par : polar Blob numéro : 13 annoté par : people Blob numéro : 13 annoté par : tree Blob numéro : 13 annoté par : walls Blob numéro : 14 annoté par : walls Blob numéro : 14 annoté par : people Blob numéro : 14 annoté par : polar Blob numéro : 14 annoté par : flowers Blob numéro : 15 annoté par : flowers Blob numéro : 15 annoté par : people Blob numéro : 15 annoté par : plain Blob numéro : 15 annoté par : plants Blob numéro : 15 annoté par : walls Blob numéro : 15 annoté par : village Blob numéro : 15 annoté par : trunk Blob numéro : 16 annoté par : trunk Blob numéro : 16 annoté par : water Blob numéro : 16 annoté par : plane Blob numéro : 16 annoté par : people Blob numéro : 16 annoté par : reef Blob numéro : 16 annoté par : fish Blob numéro : 16 annoté par : food </pre>

Figure 45 : Résultat de l'annotation de 16 blobs

5.4.1.3. Base de données d'apprentissage

Les données de l'apprentissage et le corpus Corel sont stockées dans la base d'images apprentissage. Le schéma de la figure 46 décrit cette base de données dans le modèle relationnel du SGBD SQL Server.

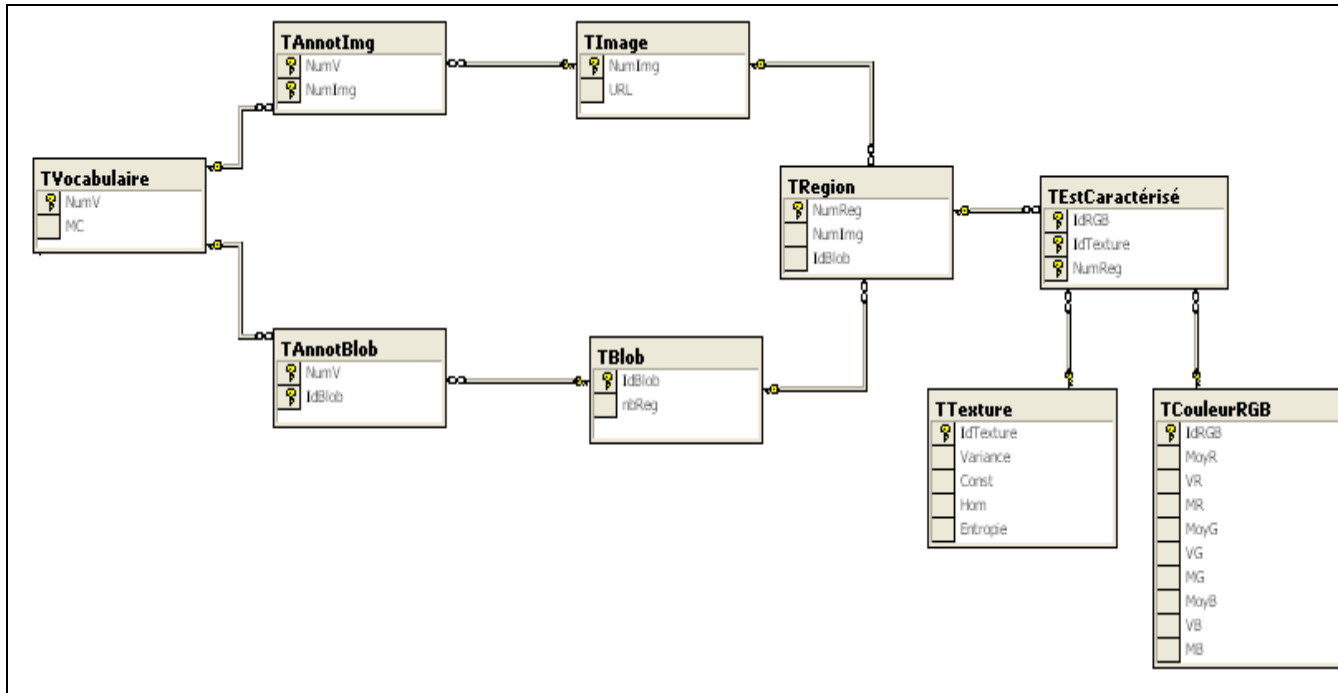


Figure 46 : Schéma de la base de données apprentissage

5.4.2. Phase de traitement d'une nouvelle image

Cette dernière phase permet d'annoter automatiquement les nouvelles images à partir des blobs annotés. Elle regroupe les modules segmentation, extraction, affectation et auto-annotation.

A. Module segmentation

Ce module est similaire à celui de la phase d'apprentissage. Le résultat d'exécution de ce module est montré dans la figure 47 :



Figure 47 : Segmentation de la nouvelle image

B. Module caractérisation

Ce module est similaire à celui de la phase d'apprentissage. Les résultats d'exécution de ce module pour toutes les régions de la nouvelle image sont résumés dans la figure 48.

```

Command Window
i To get started, select MATLAB Help or Demos from the Help menu.

Calcul de la matrice de cooccurrence de la région 1 patientez SVP...
La variance de la région 1 est : 37903838789814
La contraste de la région 1 est : 15418364
Homogeniete de la région 1 est : 36213.4522
Entropie de la région 1 est : -366500.3407
La corrélation de la région 1 est : -1.6265e-022
Calcul de la matrice de cooccurrence de la région 2 patientez SVP...
La variance de la région 2 est : 55559378282685
La contraste de la région 2 est : 12072078
Homogeniete de la région 2 est : 40265.98
Entropie de la région 2 est : -415131.1908
La corrélation de la région 2 est : -2.3059e-022
Calcul de la matrice de cooccurrence de la région 3 patientez SVP...
La variance de la région 3 est : 121620957540392
La contraste de la région 3 est : 8652444
Homogeniete de la région 3 est : 49736.7347
Entropie de la région 3 est : -536236.5794
La corrélation de la région 3 est : -5.4137e-021
Calcul de la matrice de cooccurrence de la région 4 patientez SVP...
La variance de la région 4 est : 101255618643430
La contraste de la région 4 est : 6457864
Homogeniete de la région 4 est : 47284.6324
Entropie de la région 4 est : -502463.265
La corrélation de la région 4 est : -8.5995e-022
Calcul de la matrice de cooccurrence de la région 5 patientez SVP...
La variance de la région 5 est : 42025672049954
La contraste de la région 5 est : 7923456
Homogeniete de la région 5 est : 38326.0644
Entropie de la région 5 est : -390374.3211
La corrélation de la région 5 est : -4.8311e-022
>>

```

Figure 48 : Résultats de la caractérisation texturale des différentes régions

C. Module affectation

L'intérêt de ce module est d'affecter les régions de la nouvelle image aux blobs existants selon la similarité visuelle des régions (figure 49). La similarité est déterminée en calculant les distances euclidiennes entre les vecteurs de caractéristiques visuels des centres des blobs et ceux des régions de la nouvelle image.

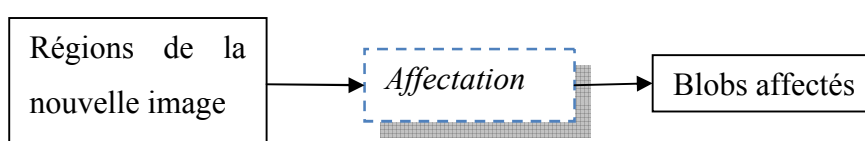


Figure 49 : Module affectation

La figure 50 montre les numéros des blobs auxquels les régions appartiennent.



```

Command Window
To get started, select MATLAB Help or Demos from the Help menu.

Affectation des régions aux blobs
Les régions de la nouvelle image appartiennent aux blobs suivants :

g =

    248
     88
    305
    393
    387
  
```

Figure 50 : Affectation des régions aux blobs

D. Module auto-annotation des nouvelles images

L'objectif de ce module est d'annoter automatiquement une nouvelle image à partir des blobs annotés (figure 51). Une première approche consiste à sélectionner un sous-ensemble de mots clés des blobs auxquels la nouvelle image appartient en éliminant les mots qui se répètent.

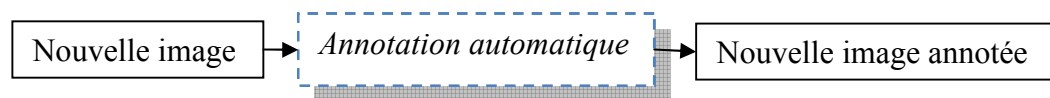


Figure 51 : Module auto-annotation d'une nouvelle image

La figure 52 présente le résultat sous forme d'un sous-ensemble de termes qui annotent la nouvelle image.

```

Command Window
i To get started, select MATLAB Help or Demos from the Help menu.

Les mots qui annotent le blob 248

Q1 =

    'flag'
    'people'
    'plants'
    'tree'
    'vegetables'

Les mots qui annotent le blob 88

Q1 =

    'water'

Les mots qui annotent le blob 305

Q1 =

    'flowers'
    'people'
    'polar'
    'walls'

Les mots qui annotent le blob 393

Q1 =

    'water'

Les mots qui annotent le blob 387

Q1 =

    'costume'
    'people'
    'snow'
    'train'

```

Figure 52 : Annotation de la nouvelle image

5.4.2.1. Base de données d'images annotées

La figure 53 décrit la base de données des images annotées dans le modèle relationnel du SGDB SQL Server. Cette base permet de stocker et d'interroger les images annotées par notre système.

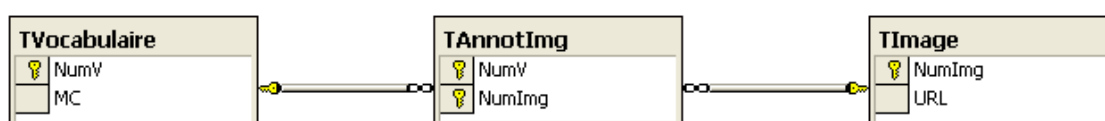


Figure 53 : Base d'images annotées

5.5. Tests et validation

Pour réaliser les tests, nous utilisons les images du deuxième sous-ensemble de la base Corel (test_1) (figure 54).

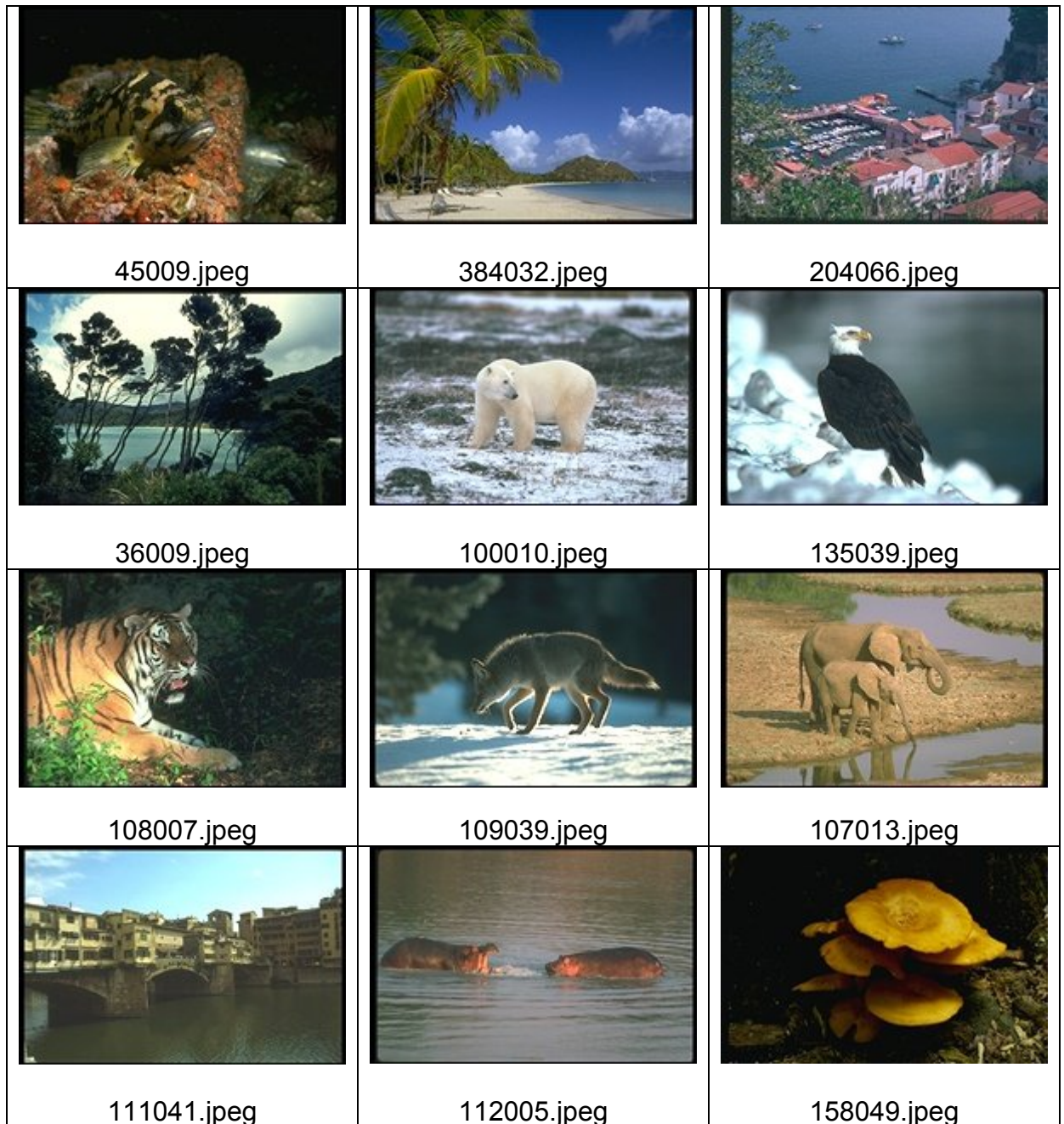








Figure 54 : Images à annoter

Nous comparons ensuite les résultats (ensemble de mots clés) de notre système avec ceux du corpus Corel. Le système va prendre chaque image de test comme une nouvelle image. Dans cette première expérimentation, nous avons

tenu compte seulement de la texture de l'image. Les résultats sont résumés dans la figure 56.

		
<p>45009.jpeg</p> <p>flowers, fish, people, plants, race, tree, vegetables, water.</p>	<p>384032.jpeg</p> <p>flowers, frost, rocks, water, food, people, polar, reef, tree, walls.</p>	<p>204066.jpeg</p> <p>field, flight, flowers, people, plain, plats, polar, trunk, village, walls, water, fungus, rocks, tree, fish, plane.</p>
		
<p>36009.jpeg</p> <p>flag, flowers, people, plants, polar, vegetables, walls, tree, building, rabbit, sky, water, frost, rocks, costume, field, petals, train.</p>	<p>100010.jpeg</p> <p>flight, flowers, people, plats, rabbit, tree, village, fungus, rocks, field, petals, reef, polar, walls, water.</p>	<p>135039.jpeg</p> <p>closeup, feline, flowers, frost, mushrooms, people, polar, rocks, snow, sunset, tree, walls, water, rabbit, food, reef.</p>



108007.jpeg

clouds, dog, fish, flowers,
 food, mountains, palace,
 people, plants, polar, reef,
 tiger, tree, vegetables,
 walls, water, building,
 closeup, coast, elephant,
 leaf, park, pattern, sky,
 smoke, street, tower, town,
 reef, cars, city, field,
 locomotive, petals, rapids,
 snow, stone, trunk.



109039.jpeg

flowers, fungus, rocks,
 water, people, fish, plants,
 walls, polar, tree, flag,
 vegetables.



107013.jpeg

city, mountaines, sky,
 stone, fish, flowers, plants,
 polar, vegetables, walls,
 field, trunk, flight, railroad.



111041.jpeg

flowers, people, race,
 water, polar, walls, fungus,
 rocks, tree.



112005.jpeg

flowers, fungus, rocks,
 water, clouds, desert, field,
 people, plane, pumpkin,
 tree, trunk, food, reef,
 frost.



158049.jpeg







flowers, people, polar,
 tree, walls, railroad, water,
 fungus, rocks, bear, boat,
 cars, close-up, clouds,
 doors, house, hazard,
 mountains, museum,
 ocean, pumpkin, shore,
 sky, snow, street, sunset,

		tables, woman
--	--	---------------

Figure 55 : Résultats de notre Système

Nous avons comparé les résultats de notre système avec l'annotation du corpus. La figure 56 présente un sous-ensemble des images du corpus annotées par notre système. Dans cette figure, l'annotation 1 représente les annotations du Corel, et l'annotation 2 représente celle de notre système.

 <p>45009.jpeg</p> <p>1 : fish, ocean, racks</p> <p>2 : flowers, fish, people, plants, race, tree, vegetables, water.</p>	 <p>384032.jpeg</p> <p>1 : sand, sky, tree</p> <p>2 : flowers, frost, rocks, water, food, people, polar, reef, tree, walls.</p>	 <p>204066.jpeg</p> <p>1 : building, house</p> <p>2 : field, flight, flowers, people, plain, plats, polar, trunk, village, walls, water, fungus, rocks, tree, fish, plane.</p>
 <p>36009.jpeg</p> <p>1 : clouds, tree, water</p> <p>2 : flag, flowers, people,</p>	 <p>100010.jpeg</p> <p>1 : bear, polar, snow</p> <p>2 : flight, flowers, people,</p>	 <p>135039.jpeg</p> <p>1 : bird, eagle, mountains, snow</p>

<p>plants, polar, vegetables, walls, tree, building, rabbit, sky, water, frost, rocks, costume, field, petals, train.</p>	<p>plats, rabbit, tree, village, fungus, rocks, field, petals, reef, polar, walls, water.</p>	<p>2 : closeup, feline, flowers, frost, mushrooms, people, polar, rocks, snow, sunset, tree, walls, water, rabbit, food, reef.</p>
		
<p>108007.jpeg</p> <p>1 : bengal, cat, forest,</p> <p>2 : clouds, dog, fish, flowers, food, mountains, palace, people, plants, polar, reef, tiger, tree, vegetables, walls, water, building, closeup, coast, elephant, leaf, park, pattern, sky, smoke, street, tower, town, reef, cars, city, field, locomotive, petals, rapids, snow, stone, trunk.</p>	<p>109039.jpeg</p> <p>1 : coyote, horizon, snow,</p> <p>2 : flowers, fungus, rocks, water, people, fish, plants, walls, polar, tree, flag, vegetables.</p>	<p>107013.jpeg</p> <p>1 : elephant, reflection,</p> <p>2: city, mountains, sky, stone, fish, flowers, plants, polar, vegetables, walls, field, trunk, flight, railroad.</p>
		
<p>111041.jpeg</p>	<p>112005.jpeg</p>	<p>158049.jpeg</p>

1 : bridge, building, sky, water	1 : bulls, hippos, water	1 : fungus, ground, mushrooms
2 : flowers, people, race, water , polar, walls , fungus, rocks, tree.	2 : flowers, fungus, rocks, water , clouds, desert, field, people, plane, pumpkin, tree, trunk, food, reef, frost.	2 : flowers , people, polar, tree, walls, railroad, water, fungus , rocks, bear, boat, cars, close-up, clouds, doors, house, hazard, mountains, museum, ocean, pumpkin, shore , sky, snow, street, sunset, tables, woman

Figure 56 : Exemple de comparaison des résultats

5.6. Discussion

L'expérimentation effectuée a pris en compte seulement la texture de l'image. Les résultats obtenus sont donc incomplets. Nous pouvons résumer les résultats préliminaires dans le tableau 5.

Images	Corpus	Système
1	3	2
2	3	2
3	2	2
4	3	2
5	3	3
6	4	3
7	3	3
8	3	1
9	2	0
10	4	2
11	3	1
12	3	3

Tableau 5 : Données de comparaison des annotations

Ces données peuvent être reportées dans le graphe de la figure 57.

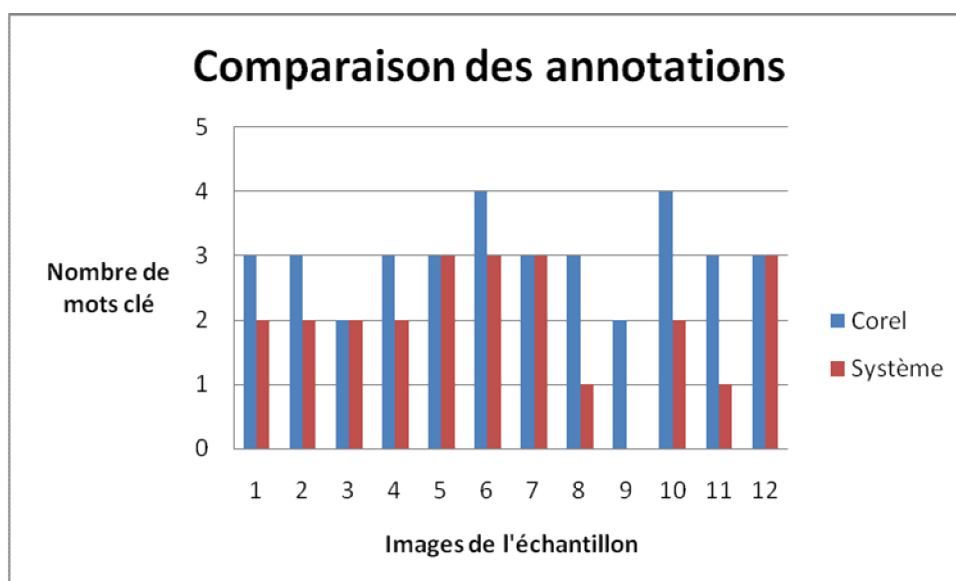


Figure 57 : Graphe de comparaison des annotations

Nous remarquons que l'annotation fournie par le système est moyenne. Ces résultats ne concernent qu'un petit échantillon que nous avons choisi de montrer et en considérant seulement la dimension texture dans les calculs. Nous sommes en train d'intégrer la couleur dans le programme. Nous pensons que l'ajout de ce paramètre important aura une incidence appréciable sur les résultats.

5.7. Conclusion

Dans ce chapitre, nous avons présenté l'implémentation d'un prototype de notre système d'auto-annotation et son expérimentation. Nous avons présenté les différents modules dans la plate-forme Matlab et les modèles de données dans les bases de données SQL Server.

Pour les tests, nous avons utilisé un benchmark fourni par Corel afin de comparer les résultats obtenus par notre système avec ceux fournis par le corpus Corel.

Les résultats que nous avons obtenus dans un échantillon d'images sont encourageants. Par conséquent, nous estimons que le système d'auto-annotation d'images proposé est prometteur, notamment si nous affinons le processus d'apprentissage et le processus d'extraction visuelle des images (en ajoutant la dimension couleur).

CONCLUSION

Dans ce mémoire nous avons présenté le domaine de la recherche d'informations en insistant sur le processus de la recherche ainsi que les systèmes de recherche d'informations. Nous avons ensuite donné un aperçu sur la recherche d'informations sur le web.

La recherche d'images sur le Web est un cas de recherche très répandu. Nous avons présenté le concept d'image et les différents systèmes de recherche d'images.

L'annotation d'images est la dernière partie de l'état de l'art. Nous avons effectué un survol des travaux de recherche dans le domaine d'annotation d'images.

Notre contribution consiste en un système d'auto-annotation d'images qui utilise un apprentissage sur un corpus d'images annotées. Des blobs sont construits afin de regrouper les régions visuellement similaires des images de corpus. Nous avons proposé de corréliser l'aspect visuel (les blobs obtenus) et l'aspect textuel (les annotations du corpus), à travers un nouvel algorithme **AnnotB-LSA** qui annote les blobs afin d'annoter de nouvelles images.

L'intégration du modèle d'analyse sémantique latente LSA dans notre algorithme a pour but d'extraire les relations sémantiques latentes dans l'espace des mots clés textuels et de minimiser l'ambiguïté (polysémie, synonymie) entre les annotations de corpus. Une nouvelle image peut être annotée par les mots clé des blobs auxquels les régions de cette image appartiennent.

Le système d'auto-annotation d'images basé sur l'algorithme **AnnotB-LSA** pourra être intégré dans un moteur de recherche sur le Web afin de fournir

un système de recherche d'images à base de base d'images annotées. Un prototype est réalisé afin de valider notre système. Nous l'avons testé sur un échantillon du corpus Corel. Nous avons réalisé l'apprentissage sur 1801 images de Corel afin de construire 400 blobs qui sont annoté par l'algorithme **AnnotB-LSA**.

Les résultats obtenus sont comparés avec les annotations de Corel Nous estimons que les résultats sont prometteurs.

Nous envisageons d'améliorer notre système d'auto-annotation d'images en :

- Variant le nombre de régions d'image à segmenter.
- Ajoutant d'autres caractéristiques visuelles (forme par exemple)
- Variant le nombre de blobs dans la classification (le paramètre k de la méthode de k-means).
- Variant le paramètre de réduction dans le modèle LSA.
- Renforçant l'algorithme AnnotB-LSA par un thésaurus afin de traiter la synonymie entre les mots qui annotent les blobs.
- Sélectionnant les cinq mots clé les plus pertinents depuis le sous ensemble de mots clé des blobs auxquels les régions d'une nouvelle image appartiennent.

REFERENCES

1. C.N. Mooers. Application of Random Codes to the Gathering of Statistical Information, MIT Master's Thesis, 1948.
2. C.W. Cleverdon. The Cranfield tests on index language devices. *Aslib Proceedings* 19(6), 173-193, 1967.
3. F.W. Lancaste. Evaluation of the MEDLARS Demand Search Service, National Library of Medicine, Bethesda, Maryland, 1968.
4. G. Salton. The SMART Retrieval System. Prentice Hall, Englewood Cliffs, NJ, 1971.
5. D.C. Blair et M.E. Maron. An evaluation of retrieval effectiveness for a full-text document- *Transactions on Neural Networks*, Special Issue on Neural Networks for Data Mining and Knowledge Discovery, pages pages 574–585, 2000.
6. D. K. (ed.), Harman, NIST Special Publication 500-207: The First Text REtrieval Conference (TREC-1), 1992.
7. G. Adamson et J. Boreham. The use of an association measure based on character structure to identify semantically related pairs of words and document titles. *Information Storage and Retrieval*, 10:pages 253–60, 1974.
8. R. Baea-Yates et B. Riberto-Neto. *Modern Information Retrieval*. New- York : ACP Press, Addison-Wesley, 1999.
9. J.-P. Balpe, A. Lelu, et I. Saleh. *Hypertextes et hypermédias :Réalisations, outils et méthodes*. Paris : Hermès, 1995.

10. N. J. Belkin et W. Croft. Information filtering and information retrieval: two sides of the same coin? *Communications of the ACM*, 35(12), December 1992.
11. G. Mecca, S. Raunich, A. Pappalardo. A new algorithm for clustering search results. *Data & Knowledge Engineering* 62, 504–522, 2007.
12. K. Barnard, P. Duygulu, R. Guru, P. Gabbur and D. Forsyth. The effects of segmentation and feature choice in a translation model of object recognition. *Computer Vision and Pattern Recognition*, 675-682.2003a.
13. C. W. Cleverdon, J. Mills, et M. Keen. Factors determining the performance of indexing systems. ASLIB Cranfield Research Project, Cranfield (UK), 1966.
14. C. Crouch, D. Crouch, Q. Chen et S. Holz. Improving the retrieval effectiveness of very short queries. *Information Processing and Management*, 38 :pages 1–36, 2002.
15. C. Fox. Lexical analysis and stop lists, pages 102–130. Frakes WB, Baeza-Yates R (eds) Prentice Hall, New jersey, 1992.
16. W. B. Frakes. Stemming Algorithms, pages 131–160. Frakes W B, Baeza-Yates R (eds) Prentice Hall, New jersey, 1992.
17. G. Furnas, S. Deerwester, S. Dumais, T. Landauer, R. Harshman, L. Streeter et K. Lochbaum. Information retrieval using a singular value decomposition model of latent semantic structure. In *Proceedings of ACM SIGIR 88*, pages 465–480, 1988.
18. J. Tang and Paul H. Lewis. Using Segmentations for Image Auto-Annotation. *ACM SIGIR Conference*, 978-1-59593-733-9/07/0007, 2007.
19. D. M. Blei. Probabilistic models of text and images. These of doctorat, university of California, Berkeley. 2004.

20. J. Jeon, V. Lavrenko and R. Manmatha. Automatic Image Annotation and Retrieval using Cross-Media Relevanve Models. ACM SIGIR Conference, 1-58113-646-3/03/0007, 2003.
21. R. Korhage. Information storage and retrieval. John Wiley and Sons, Inc. 1997.
22. J. Shi and J. Malik. Normalized Cuts and Image Segmentation. IEEE Transactions on pattern Analysis and Machine Intelligence, 22(8):888-905, 2000.
23. J. Tang, P. -H. Lewis. Using Multiple Segmentations for Image Auto-Annotation. ACM 978-1-59593-733-9/07/0007. CIVR'07, Amsterdam, The Netherlands, 2007.
24. J. Maniez et E. de Grolier. A decade of research in classification. International Classification 18, 2 , 73–77, 1991.
25. M. Maron et J. Kuhns. On relevance, probabilistic indexing and information retrieval. Journal of the Association for Computing Machinery, 7 :pages 216–244, 1960.
26. Y. Xiao, T.-S. Chua and C.-H. Lee. Fusion of Region and Image-Based Techniques for Automatic Image Annotation. MMM 2007, LNCS 4351, Part I, pp. 247-258, 2007.
27. Y. Zhao, Y. Zhao, Z. Zhu and J.-S. Pan, A Novel Image Annotation Scheme Based on Neural Network, IEEE. Eighth International Conference on Intelligent Systems Design and Applications, 978-0-7695-3382-7/08. 2008 IEEE. DOI 10.1109/ISDA.2008.55
28. J. Jeon, V. Lavrenko and R. Manmatha. Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. ACM 1-58113-646-3/03/0007, SIGIR'03, Toronto, Canada, 2003.
29. R. Weber, H.-J. Schek and S. Blott. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional

- Space. Pages 194-205 of : Proceedings of Internationale Conference of Very Large Data Bases (VLDB) (24-27). 1998.
30. J. Pearl. Probabilistic reasoning in Intelligent Systems : Networks of Plausible Inference. Morgan Kaufmann Publishers, Inc., 1988.
 31. M. F. Porter. An algorithm for suffix stripping. Program Vol. **14** No. 3, pp 130-137, July 1980.
 32. S. Tollari. Indexation et Recherche d'Images par Fusion d'Informations Textuelles et Visuelles. . Thèse de doctorat. Université du sud Toulon-Var, Octobre 2006.
 33. Y. Gao, J. Fan, H. Luo, X. Xue and R. Jain. Automatic Image Annotation by Incorporating Feature Hierarchy and Boosting to Scaleup SVM Classifiers, ACM 1-58113-737-0/03/0008, MM'06, October 23–27, Santa Barbara, California, USA, 2006.
 34. S. Robertson, S. Walker, S. Jones, and M. H.-B. And M. Gatford. Okapi at TREC 3. In Proceedings of the 3rd Text REtrieval Conference (TREC-3), pages 109–126, 1994.
 35. S. E. Robertson et S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In Proceedings of SIGIR 1994, pages 232–241, 1994.
 36. G. Salton. A comparison between manual and automatic indexing methods. Journal of American Documentation, 20(1) : pages 61–71, 1971.
 37. G. Salton, E. Fox, et H. Wu. Extended boolean information retrieval. Communications of the ACM, 31(2) :1002–1036, November 1983.
 38. G. Salton and M. McGill. Introduction to modern information retrieval. McGraw-Hill Int. Book Co, 1984.

39. W. Shaw, R. Burgin, et P. Howell. Performance standards and evaluations in IR test collections : Cluster-based retrieval models. *Information Processing and Management*, 33(1) : pages 1–14, 1997.
40. A. Singhal, G. Salton, M. Mitra, et C. Buckley. Document length normalization. *Information Processing and Management*, 32(5) : pages 619– 633, 1996.
41. W. Jin, R. Shi and T-S. Chua. A Semi-Naïve Bayesian Method Incorporating Clustering with Pair-wise Constraints for Auto Image Annotation, ACM 1-58113-893-8/04/0010, MM'04, New York, New York, USA. October 10–16, 2004.
42. C. ed. Fellbaum. *WordNet – An Electronic Lexical Database*. Bradford books. May,1998.
43. J. Lu, T. Zhao and Y. Zhang. Feature Selection Based-on Genetic Algorithm for Image Annotation, *Knowledge-Based Systems*. 0950-7051, 2008.
44. C. J. Van Rijsbergen, *Information Retrieval*, Butterworth-Heinemann, Newton, MA, 1979.
45. S. Walker, S. Robertson, M. Boughanem, G. Jones, et K. S. Jones. Okapi at TREC-6 automatic and ad hoc, VLC, routing, filtering and QSDR. In *Proceedings of TREC-6*, pages 125–136, 1997.
46. S. Wong, W. Ziarko, et P. Wong. Generalized vector space model in information retrieval. In *Proceedings of the 8th ACM SIGIR Conference on Research and Development in information retrieval*, New-York, USA, pages 18–25, 1985.
47. X-J. Wang, L. Zhang, F. Jing, and W-Y. Ma. AnnoSearch: image auto-annotation by search. *Proc. of IEEE Conf. CVPR*, New York, June, 2006.
48. G. Zipf. *Human Behaviour and the Principle of Least Effort*. Addison-Wesley, Cambridge, États-Unis, 1949.

49. G. Salton. The SMART retrieval system : Experiments in automatic document processing. Prentice Hall, Inc., Upper Saddle River, NJ, 1971 1970.
50. G. Salton and M. J. McGill. Introduction to Modern Information Retrieval. McGraw-Hill, Inc., New York, NY, 1983.
51. R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. New York : ACP Press, Addison-Wesley, 1999.
52. N. Bonnel et F. Moreau. Quel avenir pour les moteurs de recherche ? Proceedings of the Majec STIC workshop, pages 291—299, Nov. 2005.
53. F. Picarougne. Recherche d'information sur Internet par algorithmes évolutionnaires. Thèse de doctorat, Université, Toulouse, France, Novembre 2004.
54. J.W. Sammon. A non linear mapping for data structure analysis. IEEE Transactions on Computers, C-18(5) :401—409, 1969.
55. M. Carey, F. Kriwaczek, et S.M. Rüger. A visualization interface for document searching and browsing. In Proceedings of CIKM 2000 Workshop on New Paradigms in Information Visualization and Manipulation, 2000.
56. N. Bonnel, A. Cotarmanac'h et A. Morin. Visualisation 3D des résultats de recherche : quel avenir ? Actes de la huitième conférence Hypermedias Hypertexts, Products. Tools and Methods (H²PTM'05). Pages 325-339. Paris, France, Novembre 2005.
57. K. W. Boyack, B. N. Wylie et G. S. Davidson. Domain visualization using Vx Insight for science and technology management, JASIST, vol. 53, no 9, p.764—774, John Wiley & Sons, Inc. 2002.
58. F. Sparacino, A. Pentland, G. Davenport, M. Hlavac and M. Obelnicki. City of News, Proc. of the Ars Electronica Festival, 1997.

59. F. Sparacino, C. Wren, A. Azarbajegani et A. Pentland. Browsing 3-D spaces with 3-D vision : body-driven navigation through the Internet city, Proc. of Int. Symp. 3DPVT, p. 224-233, 2002.
60. J. Cugini, S. Laskowski et M. Sebrechts. Design of 3D Visualization of Search Results : Evolution and Evaluation, Proc. of IST/SPIE's Int. Symp. : Electronic Imaging 2000 : Visual Data Exploration and Analysis, 2000.
61. M. Chevalier. Interface adaptative pour l'aide à la recherche d'information sur le web, Thèse de doctorat, Université Paul Sabatier, Toulouse, France, 2002.
62. S. Benford, D. Snowdon, C. Greenhalgh, R. Ingram, I. Knox et C. Brown. VR-VIBE : A Virtual Environment for Co-operative Information Retrieval, Computer Graphics Forum (Proc.Eurographics'95), vol. 14, no 3, p. 349-360, 1995.
63. J. Mothe, C. Chrisment, B. Dousset et J. Alaux. DocCube : multidimensional visualisation and exploration of large document sets, JASIST, vol. 54, no 7, p. 650–659, John Wiley & Sons, Inc, 2003.
64. M. Hearst et C. Karadi. Cat-a-Cone : An Interactive Interface for Specifying Searches and Viewing Retrieval Results using a Large Category Hierarchy, Proc. of the 20th Ann.Int. ACM/SIGIR Conference, p. 246-255, 1997.
65. J. Rekimoto et M. Green. The Information Cube : Using Transparency in 3D Information Visualization, Proc. of the Third Ann. Workshop Information Technologies & Systems (WITS'93), p. 125-132, 1993.
66. S. K. Card, G. G. Robertson et W. York. The Web Book and the Web Forager: An Information Work space for the World-Wide Web, CHI'96:Proc.of the SIGCHI conference on Human factors in computing systems, ACM Press, p. 111–ff, 1996.

67. K. Sauvagnat. Modèle flexible pour la Recherche d'Information dans des corpus de documents semi-structurés. Thèse de doctorat, Université Paul Sabatier, Toulouse, France, 2005.
68. M. Boughanem, W. Kraaij, et J.-Y. Nie. Modèles de langue pour la recherche d'information. In *Les systèmes de recherche d'informations*, pages 163–182. Hermes-Lavoisier, 2004.
69. S. Lawrence and C-L. Giles. Accessibility of information on the web. *Nature*, vol.400, pp. 107—109, 1999.
70. R. Jain, Workshop report: NSF workshop on visual information management systems. in *Storage and Retrieval for Image and Video Databases* (Niblack, W R and Jain, R C, eds), Proc SPIE 1908, 198-218, 1993.
71. V. Lozano. Contribution de l'analyse d'image couleur au traitement des images textile. Thèse de doctorat, Université Jean-Monnet, Saint-Étienne, janvier 1998.
72. T. Gevers. Color in image databases. Technical report, University of Amsterdam, 2000.
73. CIE. Compte rendu de la huitième session. Technical report, Cambridge University Press, Cambridge, 1931.
74. D. L. MacAdam. Color measurement, theme and variation. Optical Sciences, Springer-Verlag, 1985.
75. CIE. Colorimetry. Technical Report 15.2, Bureau central de la CIE, Vienna, 1986.
76. Ph. Bolon, J.-M. Chassery, J. -P. Cocquerez, D. Demigny, C. Graffigne, A. Montanvert, S. Philipp, R. Zeboudj and J. Zerubia. *Analyse d'Image : Filtrage et segmentation*. Paris, 1998.
77. J.-G. Postaire. *De l'image à la décision*. Dunod informatique, 1987.

78. B. Burke-Hubbard. Ondes et ondelettes, la saga d'un outil mathématique. Pour la science, 1995.
79. S. Mallat. A Wavelet Tour of Signal Processing. Second Edition, Academic Press, 1999.
80. M. K. Hu. Visual pattern recognition by moments invariants, computer methods in image analysis. Transactions on Information Theory, 8, 1962
81. S. Derrode, R. Mezhoud et F. Ghorbel – Reconnaissance de formes par invariants complets et convergents: Application à l'indexation de bases d'objets à niveaux de gris. GRETSI'99. – Vannes, France, 13-17 Septembre 1999.
82. M. Sonka, V. Hlavac, and R. Boyle. Image Processing, Analysis and Machine Vision. PWS Publishing, seconde edition, 1999
83. E. Persoon and K.S. Fu – Shape discrimination using fourier descriptors. IEEE Trans. on Man and Cybernetics, vol. 7, n° 3, pp. 629–639, 1977.
84. Y. Rui, A.C. She and T.S. Huang – Modified fourier descriptors for shape representation - a practical approach. Proc. of First International Workshop on Image Databases and Multi Media Search, 1996.
85. D. M. Blei and M. Jordan. Modeling Annotated Data. Pages 127-134 of : ACM SIGIR. ACM Press, 2003.
86. G. Govaert. Analyse de données. IC2, Hermes/Lavoisier, Paris, p 151-168, 2003.
87. F. aillez et J. Pages. Introduction à l'analyse des données, Soc. De Math appliquées et de sciences humaines, 1976.
88. E. Diday, J. Lemaire, J. Bagat et F. Testu. Eliments d'analyse de données, Dunod-Bordas, Paris, 1982.

89. M. G0 Krauze, Intellectual problems of indexing picture collection. *Audiovisual Librarian*, 14(4), p 73-81, 1988.
90. T. Almeida Souza Coelho, P. Peireira Calado, L. Vieira Souza, B. Ribeiro-Neto and R. Muntz. Image Retrieval Using Multiple Evidence Ranking. *IEEE Transactions on Knowledge and Data Engineering*, p. 408-417, 2004.
91. L. Jayaratne, Z. Jiang and A. Ginige. A unified approach to indexing multimedia on the web. In : 12th International World Wide Web Conference. 2003.
92. R. K. Srihari and D. T. Burhans. Visual semantics : extracting visual information from text accompanying pictures. Pages 793-798 of : National Conference on Artificial Intelligence (AAAI). 1994.
93. A. W. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(12) :1349–1380, December 2000.
94. C.H. Chen, L.F. Pau, and P.S.P. Wang. *The Hand book of Pattern Recognition And Computer Vision (2nd Edition)*. World Scientific Publishing, 1998.
95. J. Martinet. Un Modèle Vectoriel Relationnel de Recherche d'Information Adapté aux Images. Phd Décembre 2004.
96. S. BISSOL. Indexation symbolique d'images : une approche basée sur l'apprentissage non supervisé de régularités. Phd. UNIVERSITE JOSEPH FOURIER – GRENOBLE 1. Octobre 2005.
97. W. Niblack, W. Barber, M. Equitz, M. Flicknet, E. Glasman, D. Petkovic and P. Yanker. The QBIC project: querying images by content using color, texture and shape. *Proceedings SPIE: Storage and Retrieval for Image and Video Database*, 173-181. 1993.

98. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele and P. Yanker. Query by image and video content : The QBIC System. *Computer*, 28(9), 23-32. 1995.
99. J. Li and J. Z. Wang. Automatic Linguistic Indexing of Pictures by a Statistical Modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25 (9), 1075-1088. 2003.
100. R. K. Srihari. Automatic Indexing and Content-Based Retrieval of Captioned Images. *IEEE Computer*, 28(9), 49-56. 1995.
101. A. Pentland, R. W. Picard & S. Sclaroff. Photobook : Content-Based Manipulation Of Image Databases. *International Journal of Computer Vision*, 18(3), 233-254. 1996.
102. S. Sclaroff, L. Taycher and M. La. Cascia. ImageRover : A Content-Based Image Browser for the World Wide Web. In : *Proceedings of IEEE Workshop on Content-based Access of Image and video Libraries*. 1997.
103. J. Cox, M. L. Miller, T.P. Minka, T. V. Papathomas and P. N. Yianilos. The Bayesian Image Retrieval System, PicHunter : Theory, Implementation, and Psychological Experiments. *IEEE Transactions on Image Proceeding*, 9(1), 20-37. 2000.
104. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41 (6), 391-407. 1990.
105. T. K. Landauer and S. T. Dumais. A Solution to Plato's Problem : The Latent Semantic Analysis Theory of Acquisition, Induction and Representation of Knowledge. *Psychological review*, 104, 211-240. 1997.
106. T. Hofmann. Learning and Representing Topic – A Hierarchical Mixture Model for Word Occurrences in Document Databases., In : *Workshop on learning from text and the web*, 1998.

107. T. Hofmann, J. Puzicha and L.I. Jordan. Unsupervised learning from dyadic data. Tech. rept. TR-98-042. International Computer Science Institute, Berkeley, 1998.
108. K. Barnard and D. Forsyth. Learning the Semantics of Words and Pictures. Pages 408-415 of : Proceedings of International Conference on Computer vision, vol. 2, 2001.
109. K. Barnard, P. Duygulu and D. Forsyth. Clustering Art. Pages 434-439 of : Proceedings of Computer Vision and Pattern Recognition, vol. II, 2001.
110. K. Barnard, P. Duygulu, N. Freitas, D. Forsyth, D. Blei and M. I. Jordan. Matching words and pictures. Journal of Machine Learning Research, 3:1107-1135, 2003.
111. D. M. Blei, A. Y. Ng and M. I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research, 3, 993-1022. 2003.
112. F. Monay and D. Gatica-Perez. On Image Auto-Annotation with Latent Space Models. In Proc. ACM Int. Conf. on Multimedia (ACM MM), Berkeley, California, USA, Nov. 2003.
113. F. Monay and D. Gatica-Perez. PLSA-based Image Auto-Annotation: Constraining the Latent Space. In Proc. ACM Int. Conf. on Multimedia (ACM MM), NewYork, USA, Oct. 2004.
114. W. Liu and X. Tang. Learning an Image-Word Embedding for Image Auto-Annotation on the Nonlinear Latent Space, ACM 1-59593-044-2/05/0011, MM'05, Singapore, November 6–11, 2005.
115. L. Khan. Standards for Image Annotation using Semantic Web. Computer Standards & Interfaces 29, 169-204, 2007.
116. L. Hollink, G. Schreiber, J. Wiele maker and B. Wielinga. Semantic Annotation of Image Collections. In Proceedings of Knowledge Capture - Knowledge Markup and Semantic Annotation, Workshop 2003. Workshop 2003.

117. D. Brickley and R. V. Guha. Resource Description Framework (RDF) schema specification 1.0. Candidate recommendation, W3C Consortium, 27 March 2000. See: <http://www.w3.org>.
118. H. Haddad and P. Mulhem. Utilisation de la Fouille de Données Images pour l'Indexation Automatique des Images. *INFORSID 2001*, P :405--418, Genève, Suisse, jun 2001.

Webographie

119. Moteur de recherche Vivisimo. <http://www.vivisimo.com>.
120. Moteur de recherche Google, <http://www.google.com>.
121. Moteur de recherche Altavista, <http://www.altavista.com>.
122. Moteur de recherche Grokker, <http://www.grokker.com>.
123. SQL server, [http:// www.sqlserver.com/](http://www.sqlserver.com/)
124. Matlab, <http://www.mathworks.com>.
125. Algorithme NormalizedCut, <http://note.sonots.com/SciSoftware/NcutImageSegmentation.html>
126. Code source de catégorisation, <http://people.revoledu.com/kardi/tutorial/kMean/NumericalExample.htm>.
127. Données de Corel, <http://Kobus.ca/research/data>.

LEXIQUE

Donnée : Chaîne de caractères associée à des objets, des personnes et des événements.

Information : La signification (explication/description) des données interprétées par une personne ou un système.

Connaissance : Information apprise, découverte, comprise et partagée par une communauté.

La Recherche d'Information (RI) :

- Elle sous-entend une demande vague, imprécise, floue, ...
- Elle comporte deux acteurs principaux:
 - Un être humain, qui a un **besoin d'information**.
 - Un **stock de données**.
- La **recherche d'information** automatisée est une discipline qui met en jeu le **stockage** et la **représentation** de l'information d'une part, l'**analyse** et la satisfaction d'un **besoin d'information** d'autre part.
- La RI est un processus **incertain** et **imprécis** :
 - Imprécision dans l'**expression des besoins**,
 - Incertitude dans **la représentation des informations**.
- Il y a trois étapes fondamentales pour la recherche d'information :
 - Comment représenter un document ? Indexation,
 - Comment exprimer une requête au besoin d'information ? Formulation de la requête,
 - Comment apparier la requête avec les représentations des documents ? Recherche.

Besoin en information : expression mentale de ce que recherche l'utilisateur.

Requête : représentation structurée du besoin en information.

Expression des besoins (Langage de requêtes) : est un texte libre, une liste de mots clés, avec ou sans opérateurs booléens (AND, OR, NOT), avec ou sans variables linguistiques ((plus) récent, (plus) important, ...), une image ou une navigation dans une liste de concepts (Yahoo,...).

La pertinence : est la relation entre un document et une **requête** selon **le système** ou un document et un **besoin en information** selon **l'utilisateur**.

Types de pertinences : il existe différents types de pertinence :

- **Pertinence système**: relation entre la requête et l'information trouvée,
- **Pertinence au sujet** : relation entre le sujet exprimé dans la requête et le sujet couvert dans le document,
- **Pertinence cognitive** : relation entre l'état de la connaissance de l'utilisateur et l'information sélectionnée,
- **Pertinence situationnelle (utilité)** : relation entre la tâche ou le problème posé par l'utilisateur et l'information retrouvée.