Université de Montréal

INTÉGRATION D'UNE NOUVELLE APPROCHE SÉMANTIQUE BASÉE SUR LES CARACTÉRISTIQUES VISUELLES DES CONCEPTS DANS UN SYSTÈME DE RECHERCHE D'IMAGES PAR CONTENU ET PAR TEXTE

par
Ahmed ID-OUMOHMED

Département d' Informatique et de Recherche Opérationnelle
Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de Maître ès sciences (M.Sc.)
en Informatique et Recherche Opérationnelle

Août, 2005

Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé:

INTÉGRATION D'UNE NOUVELLE APPROCHE SÉMANTIQUE BASÉE SUR LES CARACTÉRISTIQUES VISUELLES DES CONCEPTS DANS UN SYSTÈME DE RECHERCHE D'IMAGES PAR CONTENU ET PAR TEXTE

présenté par:

Ahmed ID-OUMOHMED

a été évalué par un jury composé des personnes suivantes:

Philippe Langlais
président-rapporteur

Max Mignotte
directeur de recherche

Jian-Yun Nie
codirecteur

Pierre Poulin
membre du jury

Mémoire accepté le: 14 novembre 2005

RÉSUMÉ

De nos jours, on assiste de plus en plus à une croissance fulgurante des médias et des ressources électroniques disponibles (documents textes, banques d'images généralisées ou spécifiques à un domaine, texte descriptif des images, pages Web, etc.). D'où l'intérêt croissant pour les systèmes de recherche d'information trans-médiatiques qui utilisent aussi bien les techniques spécifiques à un type de documents (modèles de recherche d'information et de traduction pour les documents de texte, techniques de caractérisation et de comparaison pour la recherche d'images) que de nouvelles approches établissant des relations sémantiques entre différents types de média par le biais d'un processus d'apprentissage non-supervisé.

En plus de la mise en oeuvre de nouvelles méthodes de recherche d'images et l'amélioration de certaines méthodes pré-existantes, nous nous intéressons à une nouvelle approche qui permettra d'étendre les critères de recherche usuels (par texte ou par contenu de l'image) par la **définition automatique** des caractéristiques visuelles (couleur, texture, contours ou forme) les plus représentatives d'un mot. Pour ce, notre système d'apprentissage se base sur la présence simultanée des images et du texte dans les documents d'un corpus (images annotées par du texte, pages Web multimédia, etc.). Toutes les images associées à un même mot ou concept sont modélisées par un ensemble de vecteurs multi-dimensionnels qui correspondent à la représentation compacte et discrétisée de ces images selon les caractérisations visuelles utilisées. On cherche alors à déterminer des méthodes de groupement (*clustering*) et des paramètres optimaux pour arriver à associer (avec un certain degré de confiance) un mot ou concept donné à une caractéristique visuelle. L'idée de base étant de chercher à localiser les agglomérations de vecteurs associés à un mot ou un concept qui soient compacts (au sens géométrique) et qui interfèrent peu avec les autres vecteurs associés à toutes les images du corpus.

Mots clés : Recherche d'images, contenu image, segmentation, mesure de similarité, attribut visuel, groupement vectoriel, approche sémantique.

ABSTRACT

Nowadays, one attends more and more dazzling growth of the available media and electronic resources (text documents, generalized or specific-field image databases, descriptive text of the images, Web pages, etc.). Thus, there is a growing interest for trans-media systems of information retrieval which use specific techniques to a certain type of documents (models of information retrieval and translation for the text documents, techniques of characterization and comparison for the image retrieval engines), as well as new approaches establishing semantic relations between various types of media by the means of a process of unsupervised training.

In addition to the implementation of new methods of image retrieval and the improvement of certain preexistent methods, we are interested in a new approach which will make it possible to extend the usual search criteria (text or image content) by an **automatic definition** of the most representative visual characteristics (color, texture, contours, or shape) of a word. For this, our training system is based on the simultaneous presence of the images and the text in the documents of a corpus (images annotated by text, multi-media Web pages, etc.). All the images associated with the same word or concept are modeled by a set of multidimensional vectors which correspond to the compact and discretized representation of these images according to the used visual characterizations. One then seeks to determine appropriate methods of grouping (*clustering*) and optimal parameters in order to associate (with a certain degree of confidence) a given word or concept to a visual characteristic. The basic idea is to seek to locate the agglomerations of vectors associated with a word or concept which are compact (in a geometrical sens) and which interfere little with the other vectors associated with all the images within the corpus.

Keywords : Image retrieval, image content, segmentation, similarity measure, visual feature, clustering, semantical approach.

TABLE DES MATIÈRES

DÉDICACE v

REMERCIEMENTS vi

RÉSUMÉ vii

ABSTRACT viii

TABLE DES MATIÈRES ix

LISTE DES FIGURES xii

LISTE DES TABLEAUX xvii

LISTE DES ANNEXES xviii

CHAPITRE 1 :INTRODUCTION 1

1.1 Contexte général 1

1.2 Cadre du projet et corpus utilisés 6

1.3 Objectifs 10

1.4 Organisation du mémoire 11

CHAPITRE 2 :ÉTAT DE L'ART 12

2.1 Notions fondamentales de la colorimétrie 12

2.1.1 Définition quantitative de la couleur et espace RGB 13

2.1.2 L'espace CIE XYZ 15

2.1.3 L'espace HSV 16

2.1.4 L'espace CIE $L^*u^*v^*$ 18

2.2 Mesures de similarités entre histogrammes 18

2.2.1 Intersection des histogrammes 20

2.2.2 Distance euclidienne 20

2.2.3 Distance de Hamming sur les histogrammes binaires des couleurs . . . 21
2.2.4 Distance quadratique . . . 21
2.3 Systèmes de recherche d'images . . . 22
2.3.1 QBIC : Query By Image Content . . . 22
2.3.2 Chabot . . . 23
2.3.3 MARS (Multimedia Analysis and Retrieval System) . . . 23
2.3.4 IKONA . . . 24
2.3.5 Blobworld . . . 24

CHAPITRE 3 : IMPLANTATION . . . 26
3.1 Structure générale du système . . . 26
3.2 Évaluation par la méthode de précision/rappel . . . 28
3.3 Couleurs . . . 29
3.3.1 Quantification régulière des espaces de couleurs RGB et HSV 29
3.3.2 Segmentation adaptative et distances inter-régions . . . 35
3.4 Les propriétés texturales de Tamura . . . 41
3.5 Caractérisation des contours par les ondelettes . . . 50
3.6 Formes . . . 56
3.6.1 Segmentation $L^*u^*v^*$ par voisinage . . . 59
3.6.2 Caractérisation des contours . . . 60

CHAPITRE 4 : ARTICLE : TOWARD CROSS-LANGUAGE AND CROSS-MEDIA IMAGE RETRIEVAL . . . 63
4.1 Introduction . . . 65
4.2 Image processing-based learning procedure . . . 68
4.2.1 Edge class and its measure . . . 69
4.2.2 Texture class and its measure . . . 70
4.2.3 Shape class and its measure . . . 72
4.2.4 The learning procedure . . . 74
4.3 Cross-language text retrieval . . . 77

4.3.1 Translation models 77
4.3.2 CLIR process 78
4.4 Combining text and images in image retrieval 79
4.4.1 The image relevance score based on clustering 79
4.4.2 Combining the five image relevance scores 79
4.4.3 Filtering the list of images based on location, photographer, and date 80
4.5 Experimental results and conclusion 80

CHAPITRE 5 : ARTICLE : SEMANTIC-BASED CROSS-MEDIA IMAGE RETRIEVAL 82
5.1 Introduction 83
5.1.1 Related Work 84
5.1.2 Our Approach 86
5.1.3 Outline of the Paper 87
5.2 Image Processing Retrieval Techniques 88
5.3 Associating words with representative images and features 89
5.4 Experimental Results and Conclusion 91

CONCLUSION 96

BIBLIOGRAPHIE 115

LISTE DES FIGURES

1.1 Recherche *"Google TM Images"* pour la requête *"animal"*. 4

1.2 Recherche *"Google TM Images"* pour la requête *"animal grass"*. 4

1.3 Le résultat de recherche par contenu de l'image effectuée par le système *IKONA*. L'image exemple est située au coin haut vers la gauche. 5

1.4 Le résultat de recherche par contenu de l'image effectuée par le système *SIMPLIcity*. L'image exemple est située au coin haut vers la gauche. 5

1.5 Un exemple de 6 images de la base donnée *Corel* ainsi que leurs champs textuels associés, sauf pour l'image *91057.jpg* qui entre dans la catégorie des images qui n'ont que le champ *catégorie* comme texte d'annotation. 7

1.6 L'image *stand03_16658_big.jpg* ainsi que son texte associé en format *TREC*. . . 9

2.1 Courbes des couleurs $\overline{r}(\lambda)$, $\overline{g}(\lambda)$ et $\overline{b}(\lambda)$ correspondants aux composantes tri-chromatiques spectrales du système RGB de la CIE. (source : http ://escience.anu.edu.au/lecture/cg/Color/) 14

2.2 Cube des couleurs du modèle RGB. (source : http ://escience.anu.edu.au/lecture/cg/Color/) 14

2.3 Composantes tri-chromatiques spectrales du système CIE XYZ. (source : http ://escience.anu.edu.au/lecture/cg/Color/) 15

2.4 Modèles de couleurs HSV et HLS. (source : http ://escience.anu.edu.au/lecture/cg/Color/) 17

3.1 Schéma descriptif du système de recherche d'images par contenu. 28

3.2 La transformation de l'espace RGB vers l'espace HSV et la quantification de ce dernier en 166 régions : 18 teintes (H) × 3 saturations (S) × 3 luminosité (V) + 4 niveaux de gris. (source : thèse de Smith [1]) 31

3.3 Recherche des images similaires à l'image exemple 105072 (coin haut gauche) en utilisant la quantification en 125 régions de l'espace RGB. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images. 32

3.4 Recherche des images similaires à l'image exemple 105072 (coin haut gauche) en utilisant la quantification en 166 régions de l'espace HSV. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images. 33

3.5 Chaque colonne d'images illustre les 6 premières images de la recherche par rapport à l'image exemple située en haut en utilisant la quantification RGB ou la quantification HSV. 34

3.6 Chaque colonne contient l'image exemple, l'image segmentée en régions couleurs de l'espace HSV, les histogrammes 6xH+3xS+3xV+4 de chaque région. La segmentation de l'image de l'avion a été effectuée en 2 régions, mais nous excluons la région dont la proportion dans l'image est inférieure à 5% pour ne retenir qu'un seul histogramme. 39

3.7 Exemple de quelques images de la collection *"St. Andrews"* avec des niveaux de contrastes très variés. La plage de concentrations des bâtonnets de l'histogramme des niveaux de gris est indiquée en dessous de chaque image. 43

3.8 Exemples de quelques images des collections *St. Andrews* et *Corel* avec leurs histogrammes de granularité à 7 dimensions (7 résolutions de textures) et de directivité (8 directions). 49

3.9 Recherche des images similaires à l'image exemple *7753.jpg* (coin haut gauche) en utilisant le descripteur composé des histogrammes de la granularité et de la directionnalité. La distance de similarité utilisée est la distance Jeffrey-divergence. La recherche est effectuée sur un ensemble de 28133 images. 51

3.10 Les fonctions de la base de V^2 qui sont obtenues à partir de la fonction porte ϕ. (source : http ://diuf.unifr.ch/courses04-05/improc/Annexes/). 54

3.11 Les ondelettes de Haar relatives à l'espace W^1. (source : http ://diuf.unifr.ch/courses04-05/improc/Annexes/) 54

3.12 *a) L'image originale* 26737.JPG *de taille* 238×308. *b) Les coefficients d'ondelettes de Haar de l'image a) après recallage entre* 0 *et* 255. . . 56

3.13 Recherche des images similaires à l'image exemple *10000.jpg* (coin haut gauche) en utilisant le descripteur des coefficients d'ondelettes. La distance de similarité utilisée est la distance D_{WMV}. La recherche est effectuée sur un ensemble de 20000 images. 57

3.14 Exemples de quelques images couleurs avec les formes obtenues en utilisant les espaces de couleurs RGB et $L^*u^*v^*$. 58

3.15 Dans chaque ligne, une image exemple est présentée avec l'image de ses contours issus d'une segmentation en 2 régions et avec l'image contour résultant de l'addition des contours résultant des segmentations en 2 et 3 régions. 61

3.16 Recherche des image similaires à l'image exemple 56026 (coin haut gauche) en utilisant l'histogramme des 4 directions associées aux contours des formes obtenues par segmentation $L^*u^*v^*$ avec un voisinage de 4 × 4. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images. 62

4.1 Workflow of image retrieval. 68

4.2 a) The original image STAND03_1028/STAND03_26737_BIG.JPG of size 238× 308. b) The Haar wavelets coefficients after the image a) is adjusted to size 256×256. 70

4.3 a) The original image STAND03_2093/STAND03_7363_BIG.JPG b) 4×4 pixel blocks and clustering result into 2 regions. c) 4 × 4 pixel blocks and clustering result into 3 regions. 73

4.4 Results of learning procedure applied to the word "garden". Below each image, we can read its identifier key in the database and the score (similarity measure) obtained after normalization. The images which are not annotated by the word "garden" have their identifier key written in a gray box. 76

5.1 For each word, the training data is the set of corresponding annotated images which yield to three sets of descriptors (vectors) according to each high-level visual feature. Each set of descriptors is clustered in several regions. The figure shows an example of clustering in 2 regions for the set associated to the texture feature. 86

5.2 A list of concepts with their discriminative features ranked by the sum of $top20^{feature}$ over all the clusters of the feature (criterion used to choose the most discriminative feature or eventually to combine several features). 93

5.3 Semantic query results for concepts *flower* (shape), *canal* (texture), and *grass* (contours). The last query is made according to the best cluster of feature *shape*. The identification number is shown above each image. Annotated images are marked by a W box. Visually related images to the concept are marked by V box. Reference images have their identification number in a gray box. 95

I.1 Interface d'exploitation du système de recherche d'images C-T-S IR (*"Content-Text-Semantic based Image Retrieval"*). 101

II.1 Les 25 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la quantification en 125 régions de l'espace RGB. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images. 103

II.2 Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la quantification en 166 régions de l'espace HSV. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images. 104

II.3 Suite de la figure II.2. 105

II.4 Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la segmentation adaptative des couleurs HSV de l'image et la quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs. La distance de similarité utilisée est la distance $D_{region\_a\_image}$. La recherche est effectuée sur un ensemble de 20000 images. 106

II.5 Suite de la figure II.4. 107

II.6 Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la segmentation adaptative des couleurs HSV de l'image et la quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs. La distance de similarité utilisée est la distance $D_{region\_a\_region}$. La recherche est effectuée sur un ensemble de 20000 images. 108

II.7 Suite de la figure II.6. 109

III.1 Aperçu du classement à l'issu de notre participation au *Workshop ImageCLEF2004* (source : Archives du *CLEF Forum2004* à l'adresse http ://clef.isti.cnr.it/). . . 111

LISTE DES TABLEAUX

5.1 Some statistics about the top retrieved images for some words. topX is the number of images annotated by the word among the first X retrieved images. Identically, refX and visX are related respectively to reference images and visually accepted images (a subjective judgment). 94

LISTE DES ANNEXES

Annexe I : Interface et fonctionnalités du système de recherche d'images C-T-S IR 100

I.1 Zone 1 100

I.2 Zone 2 100

I.3 Zone 3 100

I.4 Zone 4 102

I.5 Zone 5 102

I.6 Zone 6 102

I.7 Zone 7 102

Annexe II : Résultats comparatifs de 4 méthodes de recherche d'images par contenu couleur 103

II.1 Quantification régulière de l'espace de couleurs RGB en 125 sous-cubes en utilisant la distance euclidienne D_2 103

II.2 Quantification régulière de l'espace de couleurs HSV en 166 blocs en utilisant la distance euclidienne D_2 104

II.3 Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des histogrammes des régions par la distance $D_{region\_a\_image}$ 106

II.4 Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des histogrammes des régions par la distance $D_{region\_a\_region}$ 108

Annexe III : Participation *ImageCLEF 2004* 110

III.1 Liste des 25 requêtes de référence 110

III.2 Classement des résultats soumis au *Workshop ImageCLEF2004* . . 111

III.3 Liste de quelques mots de la collection *St. Andrews* avec leurs sens sémantiques 111

" Des fois, lassant de chercher, je suppose que je trouve, j'agite avec bonheur ce qui n'est pas encore vrai : je remue en moi-même les innombrables chances de la méditation, et prophétise ; parce qu'une sorte de réponse légère, visiblement fragile, accompagne les problèmes au moment qu'ils apparaissent : tous ne se montrent que dans l'alliance d'une solution provisoire ailée, où le sentiment de la vérité commence. "
PAUL VALERY, Agathe.

Aux trois femmes de ma vie :
la petite gazelle RIM,
le symbole de la pureté SOPHIA
et leur généreuse mère KARIMA.

REMERCIEMENTS

Un travail de recherche, bien qu'il soit le fait de quelques individus, implique le concours de nombreuses compétences. Je tiens donc à exprimer ma profonde gratitude à toutes les personnes dont l'intervention au cours de ce projet a favorisé son aboutissement :

- Max Mignotte -Directeur du projet- pour sa disponibilité, ses directives précieuses et ses conseils fructueux.
- Jian-Yun Nie -Co-Directeur du projet- pour son aide sincère et efficace. Sa compétence et son expérience m'ont beaucoup motivé et ont suscité mon intérêt pour le domaine de la recherche d'information.
- Carmen Alvarez pour sa précieuse collaboration.
- Said Benameur pour sa gentillesse et ses aimables critiques quant à l'élaboration de ce mémoire.

Je ne puis omettre d'exprimer ma gratitude aux Laboratoires Universitaires de Bell pour leur aide financière.

Que toutes les personnes qui ont contribué, de près ou de loin, à l'accomplissement du présent travail, trouvent ici le témoignage de ma profonde reconnaissance.

CHAPITRE 1

INTRODUCTION

Dans les dernières décennies, la quantité d'images numériques disponibles sur le Web a connu une profusion fulgurante au même titre que le nombre d'unités d'appareils photos numériques vendues à travers le monde. À titre d'exemple, le moteur de recherche "Google Image Search" permet une recherche parmi plus de 350 millions d'images à travers le Web [1]. À côté des moteurs de recherche d'images, des banques d'images sont aussi disponibles dans des sites spécialisés tels *FotoSearch*, *Gallica* ou *Corbis*. Par ailleurs, le nombre d'unités d'appareils photos numériques vendues dans le monde s'est élevé à 10 millions pour l'année 2003 [2]. Avec une telle quantité d'images, et vue la richesse sémantique présente dans chaque image, l'utilisateur se trouve dans le besoin d'un système de recherche d'images efficace qui utilise aussi bien l'information textuelle associée aux images que leurs contenus visuels.

1.1 Contexte général

En réalité, il y a peu d'images annotées avec suffisamment de texte par rapport au nombre total d'images disponibles. En fait, si la prise d'une photo est plus ou moins un geste simple, il n'en est pas moins pour l'annotation manuelle des images qui, tout en étant subjective, représente encore un grand fardeau. Paradoxalement, pour effectuer une recherche d'images, il est plus simple pour le commun des gens de formuler une requête par un ensemble de mots que de spécifier ou utiliser des critères relevant des techniques de traitement d'images.

Les systèmes de recherche d'images traditionnels permettent aux utilisateurs de rechercher des images dans une collection (appelée aussi corpus, banque d'images

[1] http ://c.asselin.free.fr/french/images.htm

[2] Article de presse : *"les appareils reflex enfin à la fête"* le point 05/09/03 - N1616 - Page 90

ou base de données d'images) selon deux méthodes qui se complémentent au niveau de leurs intérêts ; à savoir, une recherche basée sur le texte (*TBIR : Text-Based Image Retrieval*) et une recherche basée sur le contenu des images (*CBIR : Content-Based Image Retrieval*). Dans les deux cas, des techniques de rétroaction (*feedback*) peuvent être utilisées pour améliorer les résultats de la recherche et pour éviter à l'utilisateur d'être trop précis sur sa requête initiale. L'utilisateur est ainsi amené à estimer les images retournées par des réponses positives, négatives ou neutres quant à leurs pertinences ; combinées avec la requête initiale, ces réponses permettront de raffiner et d'améliorer les résultats de la prochaine recherche.

Recherche basée sur le texte :

Les techniques de recherche d'information offrent plusieurs modèles pour essayer de retrouver les images qui correspondent le mieux aux mots clés d'une question (ou requête) de l'utilisateur en se basant sur les mots qui annotent les images. En général, on traite les mots de la requête comme des entités à part entière, sans essayer de donner un sens sémantique à certains groupements de mots. Cependant, cette approche ne permet pas de retrouver avec succès les images qui ne peuvent pas être décrites ou qui ne peuvent être décrites que de façon ambigu avec des mots (description des couleurs ou des motifs de texture par exemple).

Recherche basée sur le contenu de l'image :

Un système de recherche d'images basé sur le contenu de l'image essaye de retrouver les images similaires à une image exemple (*image query*) soumise par l'utilisateur ; la similarité entre deux images se définit en fonction de leurs caractéristiques de bas niveaux comme l'histogramme des couleurs, la texture, les contours, etc. En fait, il n'est pas toujours aisé d'extraire des caractéristiques visuelles de haut niveau d'une image et de leur associer une syntaxe et une sémantique afin de les utiliser dans la comparaison des images. Le résultat de recherche par contenu de l'image donne souvent des images non pertinentes pour plusieurs raisons : entre autres, les images sont décrites par leurs contenus globaux, chaque objet présent dans une image a

ses propres caractéristiques, un même objet peut prendre différentes formes d'une image à une autre et une grande variété d'images dans une collection de grande taille peut réduire la performance du système.

Nous avons choisi quelques exemples de recherche d'images pour mettre l'emphase sur les difficultés qui peuvent être encourues lorsqu'on utilise une recherche basée sur le texte ou une recherche basée sur le contenu de l'image :

- La figure 1.1 montre le résultat de la recherche par le moteur de recherche d'images *"Google TM Images"* pour la requête *animal* : 9 images sur les 20 premières images retournées n'ont pas de contenu lié au sujet *animal*.
- De même (figure 1.2), la recherche par *"Google TM Images"* associée à la requête *"animal grass"* donne 12 images infructueuses sur les 20 premières images.
- Les figures 1.3 et 1.4 montrent clairement l'échec (au sens de la sémantique de l'image) de deux recherches par le contenu de l'image effectuées par deux systèmes de recherche d'images différents.

Actuellement, on porte plus d'intérêt pour les méthodes qui, en plus des techniques de recherche d'images basées sur le texte et/ou sur le contenu visuel de l'image, permettent d'étendre les critères de recherche par un ensemble de connaissances sémantiques obtenues par un processus d'apprentissage automatique. Le présent travail entre dans ce cadre et nous allons essayer, entre autres, de donner une *"sémantique visuelle"* aux mots qui annotent les images.

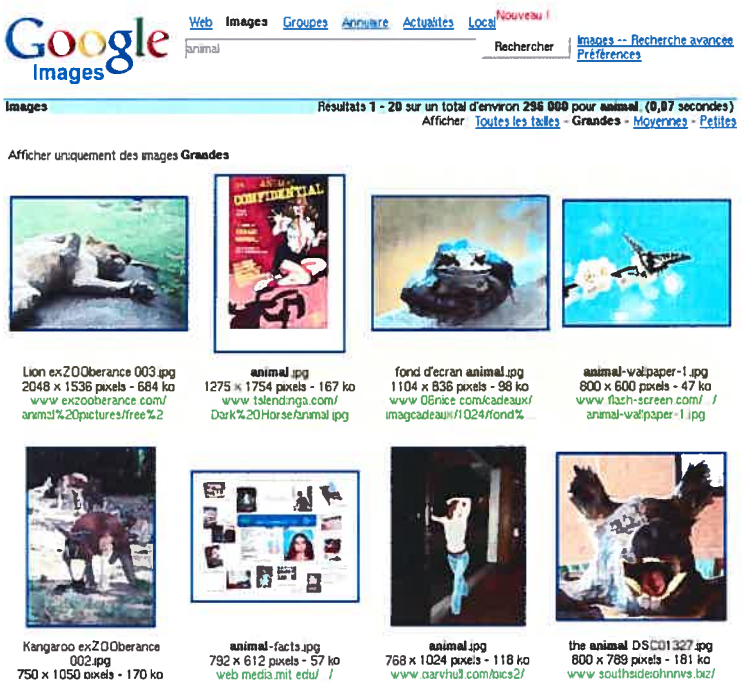


Figure 1.1: Recherche *"Google TM Images"* pour la requête *"animal"*.

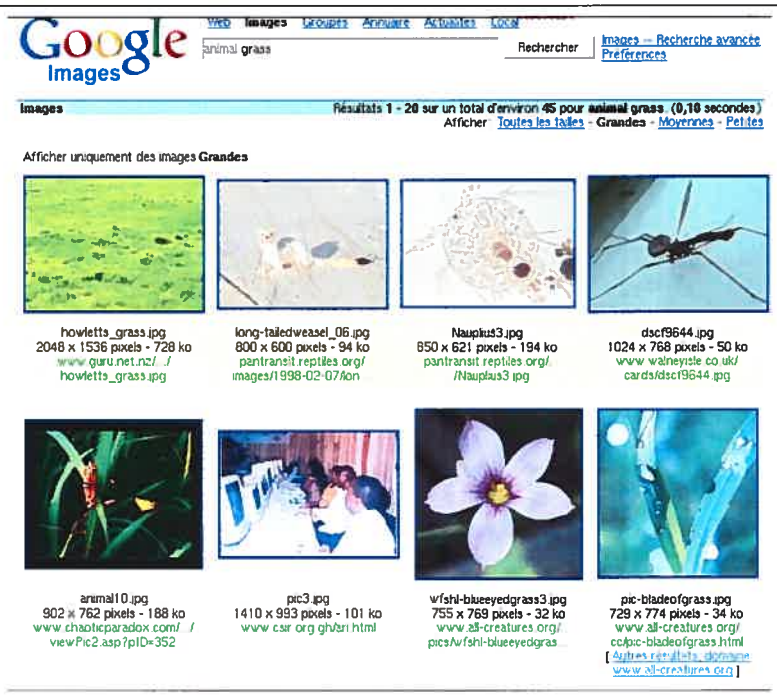


Figure 1.2: Recherche *"Google TM Images"* pour la requête *"animal grass"*.

Figure 1.3: Le résultat de recherche par contenu de l'image effectuée par le système *IKONA*. L'image exemple est située au coin haut vers la gauche.

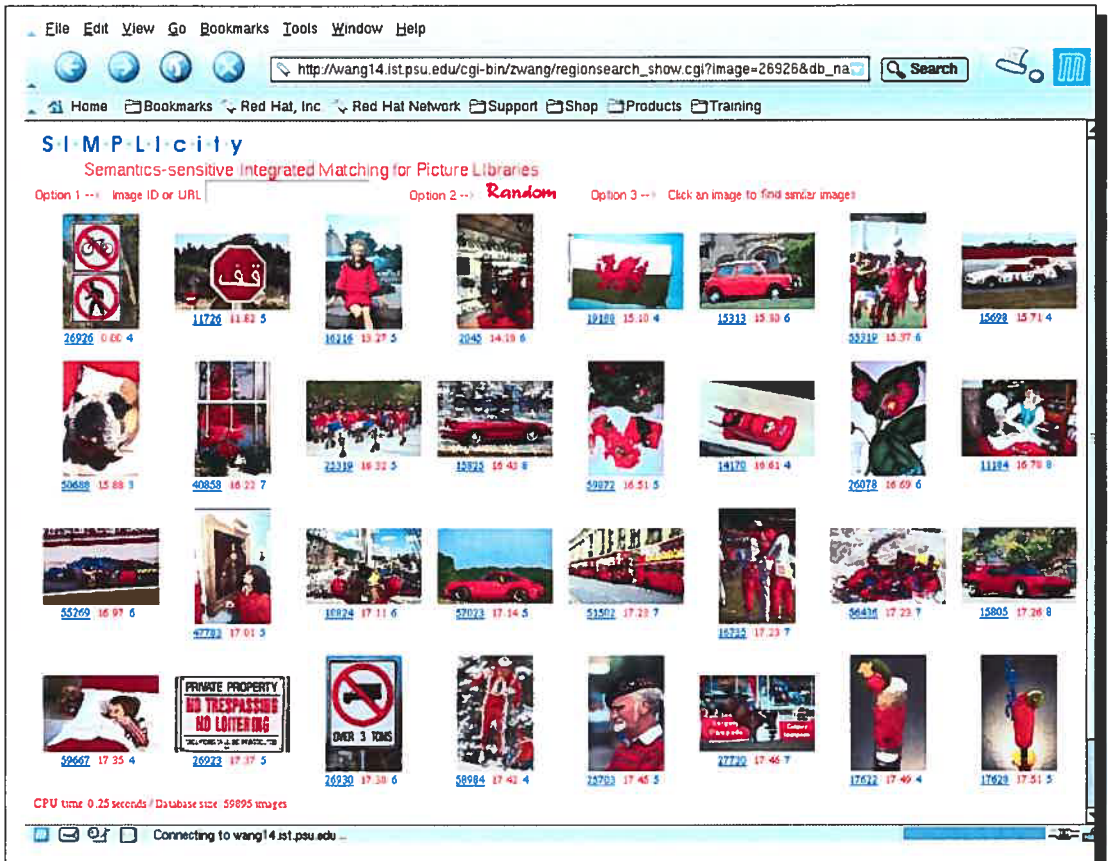


Figure 1.4: Le résultat de recherche par contenu de l'image effectuée par le système *SIMPLIcity*. L'image exemple est située au coin haut vers la gauche.

1.2 Cadre du projet et corpus utilisés

Le présent travail entre dans le cadre du projet "Recherche multilingue des images" supervisé par la Chaire Bell en recherche interdisciplinaire sur les technologies émergentes et a été financé par les laboratoires universitaires de Bell (LUB). Ce projet a été réalisé dans le cadre d'une collaboration entre le laboratoire du traitement d'images et le laboratoire RALI (Recherche Appliquée en Linguistique Informatique) et vise à élaborer un système de recherche d'images avec deux fonctionnalités majeures :

- Utiliser des techniques de recherche d'images par le contenu visuel des images et par le texte associé aux images,
- Déterminer la caractéristique visuelle la plus représentative d'un mot et l'utiliser comme un autre critère de recherche pour améliorer le résultat des recherches textuelles.

Pour la réalisation de ce projet, nous utilisons deux corpus :

Les images artistiques *Corel* : La banque d'images *Corel Professional Photos(TM)* de *Corel* qui ne comporte pas moins de 80000 images artistiques de grande qualité visuelle et qui sont annotées. Elle est très bien reconnue dans la communauté scientifique et tend à être une référence dans le domaine. Malheureusement, elle n'est plus mise en vente depuis l'année 2003. En fait, *Corel* a cédé sa division *GraphicCorp* ainsi que sa bibliothèque d'images à *Hemera* qui est une société spécialisée dans les images numériques libres de droit [3]. L'accès aux images de cette collection peut être fait par le biais du système de recherche d'images *Digital Project Library* de l'Université de *Berkeley* [4]. L'application *Corel Gallery (TM) magic* que nous avons utilisée permet l'accès à 80000 images artistiques et est dotée d'une interface pour la visualisation et la recherche de ces images en fonction d'un mot clé. Malheureusement, il n'est pas possible d'en extraire les images originales avec

[3] http ://www.graphiland.fr/news_t/news_t.aspCode=1003&Code_Cat=ACT&month=01/7/2000&rn=5&CS=Yes

[4] http ://elib.cs.berkeley.edu/photos/corel/

leurs annotations à cause du format propriétaire des données utilisé par *Corel* ; cependant, l'interface *Prelude Browser* permet d'extraire des versions réduites de ces images en format *jpeg* ou *BMP* ; les images ainsi obtenues ont une taille approximative de 373×251 ou 251×373 pixels. Avec cette interface, nous avons récupéré un fichier texte où chaque image est associée au nom du CD qui la contient (les noms de CD sont associés aux catégories des images et chaque CD contient une centaine d'images), ce qui représente peu d'informations textuelles à ce niveau. Nous avons aussi récupéré un fichier texte *corel.txt* [5] qui contient l'annotation de 40000 images ; ce texte annotatif correspond à deux champs : *caption* (légende) et *keywords* (mots clés). En somme, chaque image se trouve associée à quelques mots (de l'ordre d'une dizaine). La figure 1.5 montre un exemple de l'annotation de quelques images *Corel*.

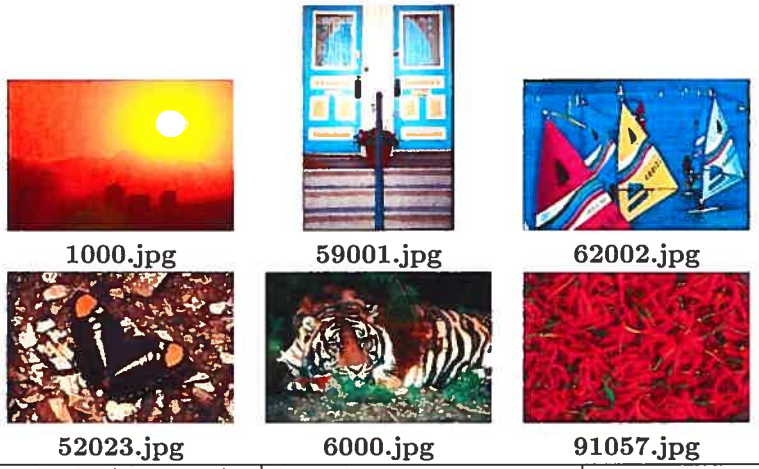


image	nom du CD(catégorie)	légende	mots clés
1000.jpg	Sunrises & Sunsets	bright sunset over city	sun city sky mountain
59001.jpg	Doors Of San Francisco	the blue and gold	doors stairs flowers home
62002.jpg	Windsurfing	light winds	sailboards people water sails
52023.jpg	Butterflies	california sister	butterfly sister wings gravel
6000.jpg	Wild Animals	tiger resting head foreleg	animal branches ground
91057.jpg	Fruits Vegetables	∅	∅

Figure 1.5: Un exemple de 6 images de la base donnée *Corel* ainsi que leurs champs textuels associés, sauf pour l'image *91057.jpg* qui entre dans la catégorie des images qui n'ont que le champ *catégorie* comme texte d'annotation.

[5] http ://elib.cs.berkeley.edu/photos/corel/corel.txt

Les images historiques *St. Andrews* : La collection d'images *St. Andrews* que nous avons utilisée dans le cadre de notre participation au *Workshop CLEF2004* et qui est constituée par un ensemble de 28133 images historiques annotées avec du texte. Bien que la plupart des images soient monochromes et que leur qualité visuelle laisse à désirer, le texte associé à chaque image est composé de plusieurs dizaines de mots, voir des centaines. Ce texte est organisé en trois champs : *headline* (lettrine), *categories* (catégorie) et *caption* (légende). La figure 1.6 montre l'exemple d'une image de la collection *St. Andrews* ainsi que son texte associé.

stand03_16658_big.jpg

<DOC>
<DOCNO>stand03_1897/stand03_16658.txt</DOCNO>
<HEADLINE> Renfrew. Swing Bridge. </HEADLINE>
<TEXT>
<RECORD_ID>JV-A.003851</RECORD_ID>
Swing Bridge, Renfrew. Iron girder bridge with towers and curved structure at end, reflected in river with grass banks and trees. Registered 2 June 1936 J Valentine & Co Renfrewshire, Scotland JV-A3851 jf/pc/mbDETAIL : The towers are single storey huts set on high iron girders with cross braces. The bridge is built with pony trusses and the solid curve holds the counterbalance for when the bridge is lifted. ADD : Renfrew was the principal port on the Clyde in 1614. The first shipbuilding firm was established in 1844 and after that other shipbuilding and heavy industry firms were founded here or moved to Renfrew. The Hillingdon Industrial Estate was founded in 1938 and was the first in Scotland.
<CATEGORIES> [bridges - metal],[rivers & streams],[reflections],[bridges - vertical lift],[Renfrews all views],[Collection - J Valentine & Co] </CATEGORIES>
<SMALL_IMG> stand03_1897/stand03_16658.jpg </SMALL_IMG>
<LARGE_IMG> stand03_1897/stand03_16658_big.jpg
</LARGE_IMG>
</TEXT>
</DOC>

Figure 1.6: L'image *stand03_16658_big.jpg* ainsi que son texte associé en format *TREC*.

1.3 Objectifs

Pour ce projet, nous nous sommes fixés les objectifs suivants :

Système de recherche d'images basé sur le contenu de l'image :
Nous implantons un système fonctionnel de recherche d'images par contenu dans lequel nous utilisons des méthodes courantes et de nouvelles méthodes pour la caractérisation des images (couleur, texture, contours ou forme) ainsi que de nouvelles mesures de similarités. En particulier, nous proposons une méthode basée sur la segmentation des images en couleurs ainsi qu'une mesure de similarité basée sur les régions.

Extension de la recherche textuelle par la sémantique visuelle d'un mot :
En plus des techniques de recherche d'information que nous utilisons en collaboration avec le groupe RALI, nous proposons une nouvelle approche pour associer une caractéristique visuelle (couleur, texture, contours ou forme) à un mot. Ensuite nous définissons la représentation associée à la caractéristique visuelle obtenue. Ceci revient à donner une sémantique visuelle à un mot par un processus d'apprentissage basé sur le regroupement des images selon leurs similarités visuelles. Par exemple, notre objectif sera d'arriver à affirmer que la caractéristique visuelle la plus discriminante du mot *sea* est la texture et de définir une description de cette caractéristisation. Une fois l'association "mot-caractéristique visuelle" définie, nous essayons de voir dans quelle mesure on peut étendre les requête textuelles par cette association afin d'en améliorer la performance.

Recherche d'images basée sur la sémantique des mots :
Nous utilisons ensuite cette approche pour représenter les classes d'images qui peuvent être associées à un mot (*bateau* dans la mer, gros plan d'un *bateau*, etc.). Nous intégrons ensuite cette notion de classe d'images dans notre système de recherche d'images de telle sorte que l'utilisateur puisse choisir cette option pour retrouver éventuellement des images qui ne sont pas annotées avec un mot mais

qui sont visuellement associées à celui-ci. Nous nous référons à cette approche par le terme SBIR (*SBIR : Semantic-Based Image Retrieval*).

1.4 Organisation du mémoire

Dans le chapitre 2, nous abordons quelques notions de base concernant les espaces de couleurs ainsi que quelques mesures de similarités relatives à la comparaison des histogrammes couleurs en particulier. Nous énumérons également quelques systèmes de recherche d'images existants avec leurs principales caractéristiques. Le chapitre 3 sera consacré à la description des méthodes que nous utilisons pour l'implantation de notre système de recherche d'images par contenu, et ce en égard aux attributs des couleurs, de textures, des contours et des formes. Les articles [2] et [3] sont présentés respectivement dans les chapitres 4 et 5.

L'annexe I est consacrée à une comparaison visuelle des différentes méthodes de recherche d'images par couleur. Dans l'annexe II, nous présentons l'interface et les différentes fonctionnalités implantées dans notre système de recherche d'images. Dans l'annexe III, nous présentons quelques résultats additionnels liés à l'approche présentée dans le chapitre 4 et que nous commentons dans la conclusion de ce mémoire.

CHAPITRE 2

ÉTAT DE L'ART

Lors de la phase d'indexation des images d'un système de recherche d'images par contenu, chaque image est associée à une description quantitative qui va servir à retrouver les images qui se rapprochent le plus de l'image exemple au sens d'une certaine mesure de similarité. Pour caractériser une méthode de recherche, deux termes étroitement liés sont utilisés ; en l'occurrence : attribut (*feature*) et descripteur (signature ou index). L'attribut réfère à une caractéristique visuelle de l'image (couleur, texture, contour, etc.), alors que le descripteur est une représentation numérique de cet attribut qui en définit la syntaxe et la sémantique. La nature des données qui composent un descripteur dépend de ce qu'on décide de quantifier, tel un histogramme, une matrice de relations entre composantes de l'image, pourcentages des régions dans l'image, etc. D'où l'importance de faire un choix judicieux du descripteur ainsi que d'une mesure de similarité entre descripteurs qui soit la plus appropriée et la plus adaptée.

La section 2.1 est une présentation sommaire de quelques notions fondamentales de la colorimétrie où nous décrivons les espaces de couleurs utilisés dans ce projet. Dans la section 2.2, nous discutons de quelques mesures de similarités entre histogrammes. La section 2.3 est consacrée à l'exposition de quelques exemples de systèmes existants de recherche d'images par contenu.

2.1 Notions fondamentales de la colorimétrie

L'attribut couleur d'une image peut apparaître comme étant la caractéristique visuelle la plus simple et la plus intuitive pour caractériser une image visuellement et ainsi définir un critère discriminatoire pour la recherche d'images. Cependant, il y a une multitude d'approches et de méthodes qui peuvent être utilisées : le modèle de représentation de la couleur, la caractérisation globale ou par régions,

une certaine méthode de segmentation de l'image en couleurs, la localisation des régions à travers l'image, etc. De plus, et hormis la difficulté de décrire une couleur par un modèle de représentation particulier, les couleurs sont souvent altérées par la texture de la surface et les conditions d'illumination des différents objets contenus dans l'image.

2.1.1 Définition quantitative de la couleur et espace RGB

On peut percevoir la couleur comme une radiation électromagnétique dans le spectre visible, soit une fonction $F(\lambda)$ avec λ une longueur d'onde comprise entre 380 nm et 780 nm. L'oeil humain comporte trois types de cônes sensibles aux trois lumières rouge, verte et bleue. Des expérimentations portant sur l'absorption de ces lumières par les cônes de l'oeil humain ont montré que les trois types de cônes présentent des réponses maximales pour des longueurs d'ondes qui se rapprochent de celles définies par la CIE (Commission Internationale de l'Éclairage) en 1931 ; à savoir, $\lambda_r = 700.0$ nm pour le rouge, $\lambda_g = 546.1$ nm pour le vert et $\lambda_b = 435.8$ nm pour le bleu. Ces dernières valeurs ont été utilisées dans d'autres expérimentations qui concernent l'égalisation (jumelage ou appariement) des couleurs spectrales pures par la combinaison des trois couleurs primaires R (rouge), G (vert) et B (bleu) associées respectivement aux longueurs d'onde λ_r, λ_g et λ_b, ce qui a mené à la définition de trois fonctions colorimétriques $\overline{r}(\lambda)$, $\overline{g}(\lambda)$ et $\overline{b}(\lambda)$ qui permettent de calculer facilement les composantes tri-chromatiques d'une lumière spectrale (figure 2.1). La courbe $\overline{r}(\lambda)$ est négative sur une partie du spectre du fait que certaines couleurs très saturées ne peuvent pas être créées par une synthèse additive (la solution apportée est de superposer une faible quantité de la composante primaire rouge en complémentarité à la couleur à créer afin de la dé-saturer).

Ainsi, il est possible de représenter un stimulus de couleur $S(\lambda)$ avec les fonctions colorimétriques primaires de base $\overline{r}(\lambda)$, $\overline{g}(\lambda)$ et $\overline{b}(\lambda)$. Les composantes tri-

chromatiques sont calculées avec les formules suivantes :

$$\begin{cases} R = \int_{\lambda_{min}}^{\lambda_{max}} S(\lambda)\ \overline{r}(\lambda)\ d\lambda \\ G = \int_{\lambda_{min}}^{\lambda_{max}} S(\lambda)\ \overline{g}(\lambda)\ d\lambda \\ B = \int_{\lambda_{min}}^{\lambda_{max}} S(\lambda)\ \overline{b}(\lambda)\ d\lambda \end{cases}$$

et définissent les composantes du modèle RGB, qui sont représentées dans un système de coordonnées cartésien (figure 2.2).

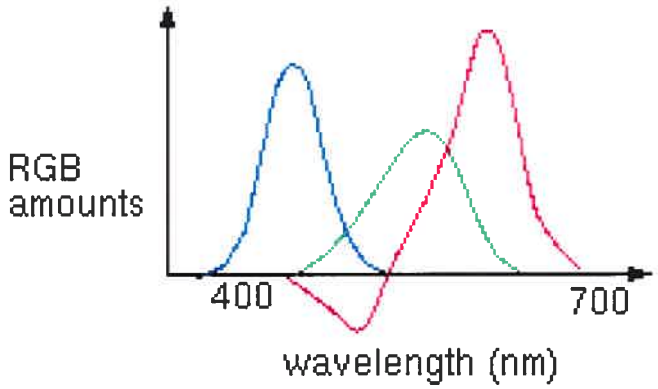


Figure 2.1: Courbes des couleurs $\overline{r}(\lambda)$, $\overline{g}(\lambda)$ et $\overline{b}(\lambda)$ correspondants aux composantes trichromatiques spectrales du système RGB de la CIE. (source : http ://escience.anu.edu.au/lecture/cg/Color/)

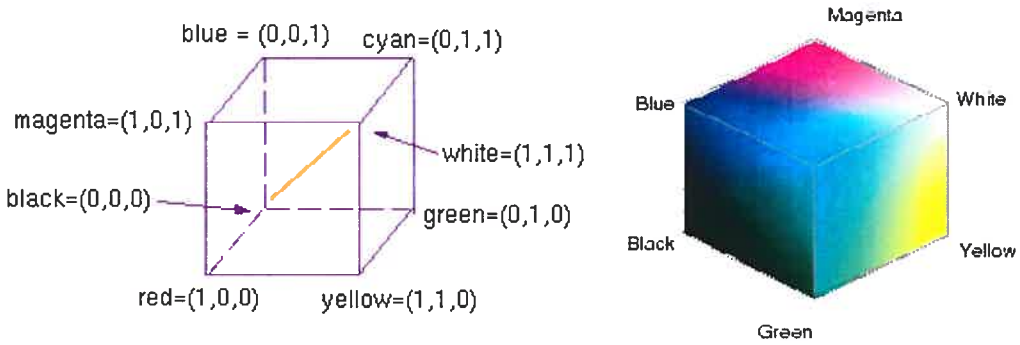


Figure 2.2: Cube des couleurs du modèle RGB. (source : http ://escience.anu.edu.au/lecture/cg/Color/)

2.1.2 L'espace CIE XYZ

L'espace CIE XYZ a été défini afin de corriger certains défauts de l'espace RGB ; cet espace est constitué de trois primaires X, Y et Z, dites virtuelles. De même qu'avec l'espace RGB, les composantes X, Y et Z sont obtenues par intégration du produit du stimulus de couleurs $S(\lambda)$ avec les fonctions colorimétriques $x(\lambda)$, $y(\lambda)$ et $z(\lambda)$ qui sont présentées dans la figure 2.3. L'espace CIE XYZ présente les propriétés suivantes :

- les triplets décrivant chaque couleur en fonction de ses primaires ont tous des valeurs positives pour le spectre visible,
- la fonction $y(\lambda)$ représente approximativement la sensibilité de l'oeil humain à la luminosité [4].

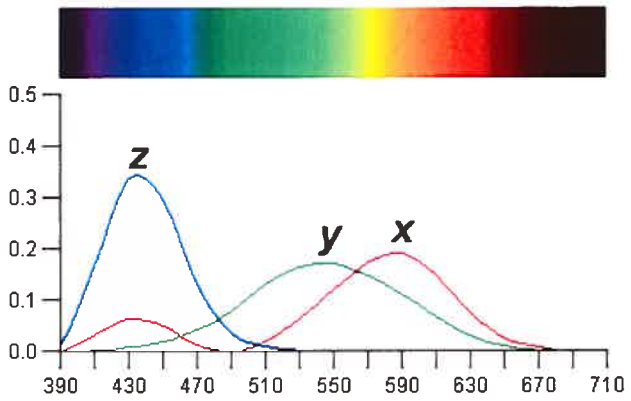


Figure 2.3: Composantes tri-chromatiques spectrales du système CIE XYZ. (source : http :// escience.anu.edu.au/lecture/cg/Color/)

Le passage de l'espace RGB à l'espace CIE XYZ s'effectue simplement par une transformation linéaire qui dépend du choix d'une couleur blanche de référence (appelée aussi point illuminant) et de la distance de l'observateur. La CIE recommande l'emploi des sources normalisées suivantes

- Illuminant A : version normalisée de l'éclairage à incandescence.
- Illuminant B : lumière directe du soleil.
- Illuminant C : lumière moyenne du jour, sans apport des rayons ultra-violets.

- Illuminant D65 : lumière moyenne du jour, avec apport des rayons ultra-violets.

Ainsi, en considérant le point illuminant D65 ($X_w = 95.047, Y_w = 100, Z_w = 108.883$) par rapport à un observateur à un angle de 2^o, la transformation des coordonnées RGB, comprises entre 0 et 1, vers l'espace CIE XYZ se fait par la formule suivante

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{pmatrix} * \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

2.1.3 L'espace HSV

Ce modèle fait partie des espaces perceptuels qui essayent de représenter la couleur sous une forme qui se rapproche de la perception humaine des couleurs ; ainsi, ce modèle représente la couleur par trois entités :

- H : tonalité, teinte ou couleur (*Hue*).
- S : saturation ou pureté.
- V : luminosité.

Les valeurs de H, S et V sont obtenues par une dérivation géométrique de l'espace RGB : le pseudo cône à six faces de la figure 2.4 correspond au cube RGB vu selon sa diagonale, le plan supérieur correspond aux faces visibles et les autres plans correspondent aux faces des différents sous-cubes.

Le modèle HLS (H : teinte (*Hue*) , L : luminance (*lightness*) et S : saturation) est une variante du modèle HSV qui se présente sous forme d'un double cône hexagonal comme le montre la figure 2.4. Il diffère du modèle HSV par la position de la couleur blanche et des interpolations qui en résultent.

Soit $(r, g, b) \in [0, 1]^3$ un triplet associé aux coordonnées d'une couleur dans l'espace RGB, la transformation dans l'espace HSV résulte en un triplet (h, s, v) tel que $h \in [0, 360[$, $(s, v) \in [0, 1]^2$ et h est indéfinie quand $s = 0$. L'algorithme de la conversion RGB $\rightarrow$ HSV se présente comme suit [4] :

- $max = \max(r, g, b)$, $min = \min(r, g, b)$, $delta = max - min$
- $v = max$
- si $(\max = 0)$

  $s = 0$

  h est indéfini $\rightarrow$ fin de la procédure

  sinon

  $s = delta/max$

  finsi
- si $(r = max)$, $h = (g - b)/delta$

  si $(g = max)$, $h = 2 + (b - r)/delta$

  si $(b = max)$, $h = 4 + (r - g)/delta$

  $h = h * 60$

  si $(h < 0)$, $h = h + 360$

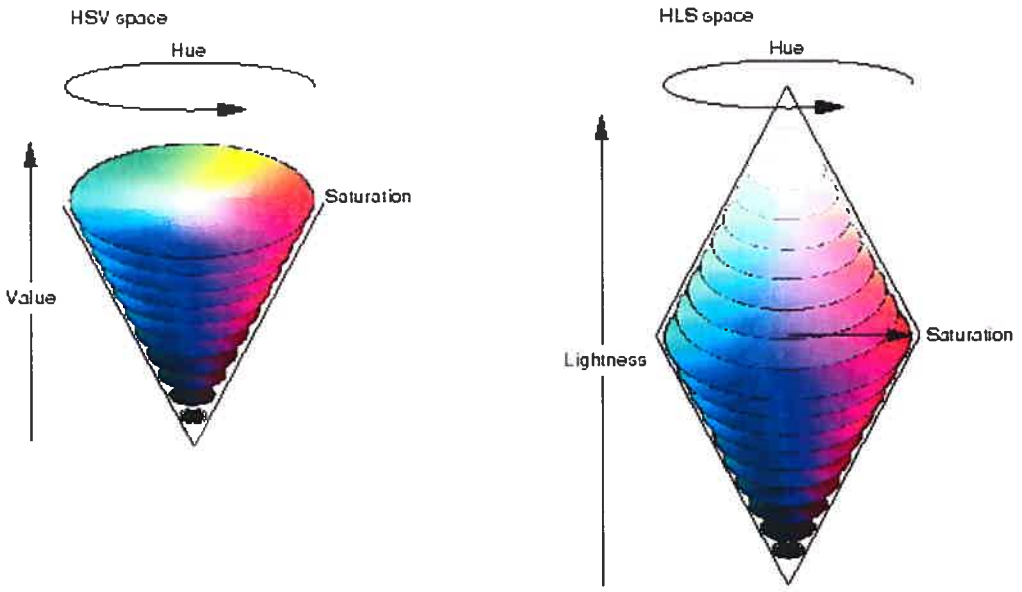


Figure 2.4: Modèles de couleurs HSV et HLS. (source : http ://escience.anu.edu.au/lecture/ cg/Color/)

2.1.4 L'espace CIE $L^*u^*v^*$

Plusieurs travaux ont été menés dans le but de produire des espaces de couleurs perceptuellement uniformes. L'espace CIE $L^*u^*v^*$ a été introduit en 1976 et est reconnu et utilisé comme standard. Les composantes chromatiques de l'espace CIE $L^*u^*v^*$ sont : L^* qui représente la luminosité ou la clarté, u^* qui représente l'opposition de couleurs vert-rouge et v^* qui représente l'opposition de couleurs bleu-jaune [4].

Le passage des coordonnées XYZ aux coordonnées $L^*u^*v^*$ est effectué par rapport à une couleur blanche de référence caractérisée par ses trois composantes trichromatiques (X_w, Y_w, Z_w) prises dans l'espace XYZ. Les formules de conversion sont les suivantes

$$L^* = \begin{cases} 116 * \left(\frac{Y}{Y_w}\right)^{\frac{1}{3}} - 16 & \text{si } \frac{Y}{Y_w} > 0.008856, \\ 903.3 * \frac{Y}{Y_w} & \text{si } \frac{Y}{Y_w} \leq 0.008856. \end{cases}$$

où

$$u^* = 13L^*(u' - u_w{}') \text{ et } v^* = 13L^*(v' - v_w{}')$$

avec

$$u' = \frac{4X}{(X + 15Y + 3Z)} \text{ et } v' = \frac{9Y}{(X + 15Y + 3Z)}.$$

2.2 Mesures de similarités entre histogrammes

Soient I une image de taille $W \times H$ (W et H désignent respectivement les dimensions horizontales et verticales de l'image) et $I(i,j) = (r,g,b)$ la couleur RGB du pixel se trouvant à la position (i,j). Soit une fonction de transformation T telle que $T(i,j) = T(r,g,b) \in \mathbb{R}^n$ qui associe à chaque pixel une certaine entité scalaire ($n = 1$) ou vectorielle ($n \geq 2$) : par exemple, la transformation des coordonnées (r,g,b) dans un autre espace de couleurs, le niveau de gris associé au triplet (r,g,b), le gradient de $I(i,j)$, l'angle de la direction du contour à la position (i,j), etc. L'ensemble $E = T([0,255]^3)$ résultant de la transformation de l'espace RGB par

T est de dimension n. En général, on quantifie (partitionne) l'ensemble E en un ensemble réduit de régions afin d'en déduire des histogrammes.

La quantification d'un espace multidimensionnel peut se faire soit par une quantification scalaire et régulière de chacune de ses dimensions, soit directement par une quantification vectorielle (*vector quantization*). Nous parlons dans les deux cas du quantificateur vectoriel Q (*vector quantizer*) de dimension $n = \dim(E)$ qui se définit comme une application (*mapping*) $Q : E \rightarrow C$ où $C = \{y_0, y_1, .., y_{S-1}\}$ est un ensemble fini à S éléments, chaque y_s est un vecteur à n dimensions et S est associé au nombre de régions désiré dans le partitionnement. L'ensemble C est communément appelé livre de codes (*codebook*) et ses éléments sont appelés des codes (*code words*). Le quantificateur Q partitionne alors l'espace E en S régions disjointes R_0, R_2,...,R_{S-1} telles que

$$R_s = \{v \in E : Q(v) = y_s\} \quad \text{et} \quad \bigcup_{s=0}^{S-1} R_s = E \quad \text{et} \quad R_i \bigcap R_j = \varnothing, \quad \forall\, i \neq j.$$

L'histogramme h associé à cette quantification se définit alors comme suit —

$$h\,[s] = \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} \delta(Q(T(i,j)), y_s) \quad \text{avec} \quad s = 0, ..., S-1$$

où $Q(T(i,j))$ est le code associé à $T(i,j)$ après quantification et $\delta(.,.)$ est la fonction delta de Kronecker qui est égale à 1 si ses deux arguments sont égaux et à 0 sinon. Ensuite h peut être normalisé par $\left(\sum_{s=0}^{S-1} |h[s]|^r\right)^{\frac{1}{r}}$ de telle sorte à avoir une distribution empirique ($r = 1$) ou un vecteur unitaire dans l'espace euclidien à n dimensions ($r = 2$). La distance L_p entre deux histogrammes h_q et h_t de même taille S est définie par :

$$L_p(h_q, h_t) = \left[\sum_{s=0}^{S-1} |h_q(s) - h_t(s)|^p\right]^{\frac{1}{p}}.$$

2.2.1 Intersection des histogrammes

En se basant sur une recherche d'images par couleurs, et pour mesurer le degré de présence d'un objet ayant un histogramme h_q dans une image ayant un histogramme h_t, Swain et al. [5] ont utilisé la mesure suivante

$$D_{\cap}(h_q, h_t) = 1 - \frac{\sum_{s=0}^{S-1} \min(h_q(s), h_t(s))}{|h_q|},$$

où $|h_q| = \sum_{s=0}^{S-1} h_q(s)$ et $|h_q| < |h_t|$. Cette mesure n'est pas symétrique et les histogrammes h_q et h_t ne sont pas normalisés de telle sorte à ce qu'on tienne compte de la taille de l'objet recherché. Cette mesure peut être rendue symétrique et utilisable pour la comparaison de deux images ayant les histogrammes h_1 et h_2 en utilisant la formulation suivante

$$D_1(h_1, h_2) = 1 - \frac{\sum_{s=0}^{S-1} \min(h_1(s), h_2(s))}{\min(|h_1|, |h_2|)}.$$

Quand h_1 et h_2 sont normalisés, cette mesure d'intersection des histogrammes devient une distance L_1 [5]

$$D_1(h_1, h_2) = L_1(h_1, h_2) = \sum_{s=0}^{S-1} |h_1(s) - h_2(s)|.$$

2.2.2 Distance euclidienne

La distance euclidienne (L_2) entre deux histogrammes h_q et h_t est définie par

$$D_2(h_q, h_t) = \sqrt{(h_q - h_t)^T (h_q - h_t)} = \sqrt{\sum_{s=0}^{S-1} (h_q(s) - h_t(s))^2}.$$

Quand les histogrammes h_q et h_t sont normalisés en des vecteurs unitaires, $D_2(h_q, h_t)$ peut être simplifiée comme suit

$$D_2^2(h_q, h_t) = (h_q - h_t)^T (h_q - h_t) = h_q^T h_q + h_t^T h_t - 2h_q^T h_t = 2 - 2h_q^T h_t$$

du fait que $\|h_q\| = h_q^T h_q = 1$ et $\|h_t\| = h_t^T h_t = 1$. Cette simplification permet de réduire le coût du calcul en se ramenant à une distance angulaire entre h_q et h_t et constitue une approximation efficace de la distance L_2 [6].

2.2.3 Distance de Hamming sur les histogrammes binaires des couleurs

Dans sa thèse [1], Smith a proposé une représentation binaire des histogrammes des couleurs qui a l'avantage de réduire le coût de comparaison des images dans les grandes collections d'images, et de ce fait, elle peut être utilisée pour un premier filtrage dans une recherche basée sur les couleurs. Soit h un histogramme non normalisé, l'histogramme binaire h^b associé à h a les valeurs binaires 0 ou 1 en fonction d'un seuil T : $h^b(s) = 1$ si $h(s) > T$ et $h^b(s) = 0$ sinon. La distance de Hamming entre deux histogrammes binaires h_q^b et h_t^b mesure le nombre de positions où les bits sont différents dans h_q^b et h_t^b. Elle s'exprime par

$$D_3(h_q^b, h_t^b) = \frac{|h_q^b - h_t^b|}{|h_q^b||h_t^b|} = \frac{\sum_{s=0}^{S-1} |h_q^b(s) - h_t^b(s)|}{|h_q^b||h_t^b|}$$

et peut être déduite de la formule suivante $D_3(h_q^b, h_t^b)|h_q^b||h_t^b| = h_q^b \oplus h_t^b$ où $\oplus$ est l'opérateur binaire "ou exclusif".

2.2.4 Distance quadratique

Le système IBM QBIC [7] utilise une métrique quadratique entre deux histogrammes de couleurs h_q et h_t qui est définie par

$$D_4(h_q, h_t) = \sqrt{(h_q - h_t)^T A (h_q - h_t)}$$

où $A = [a_{ij}]$ est une matrice symétrique avec $a_{ii} = 1$ et dont les coefficients a_{ij} représentent les similitudes entre les éléments (couleurs exprimées dans l'espace RGB) y_i et y_j de l'ensemble C résultant de la quantification. Plus précisément, a_{ij}

se définit par

$$a_{ij} = 1 - \frac{d_{ij}}{d_{max}}$$

où d_{ij} est la distance L_2 entre les couleurs i et j de l'ensemble C et

$$d_{max} = \max_{i,j} d_{ij}.$$

2.3 Systèmes de recherche d'images

À part les moteurs de recherche d'images sur le Web, tels *Google* ou *MSN Search*, d'autres systèmes de recherche d'images existent et utilisent des méthodes différentes. Nous présentons une description sommaire de quelques-uns de ces systèmes.

2.3.1 QBIC : Query By Image Content

QBIC [7] est un système commercial qui a été développé en 1995 par le centre *"IBM Almaden Research Cente"* [1]. La caractérisation de la couleur se fait, soit par le vecteur couleur moyen de dimension 3 dans les espaces RGB, YIQ, Lab ou Munsell, soit par un histogramme des couleurs de dimension 256. Le vecteur couleur moyen h_{avg} d'un histogramme h de dimension S se définit comme le produit matriciel Ch où $C = [c_1 c_2 ... c_S]$ est une matrice $3 \times S$ dont une colonne c_i représente la $i^{\text{ème}}$ couleur 3D de la $i^{\text{ème}}$ entrée de h. Les textures sont décrites par les propriétés de Tamura [8] : granularité (*coarsness*), la directivité (*directionality*) et le contraste. Les formes sont caractérisées par la surface, la circularité, l'excentricité, l'orientation de l'axe directeur et un ensemble de moments algébriques invariants ; l'excentricité et l'orientation de l'axe directeur sont calculées à partir des valeurs de la matrice de covariance du second ordre de l'image contour.

La recherche d'images se fait soit par une image exemple, soit par le croquis d'une forme dessinée par l'utilisateur, soit par une sélection pondérée des couleurs et des motifs de texture. La distance de similarité entre deux vecteurs couleurs

[1] http ://wwwqbic.almaden.ibm.com/

moyens est une distance euclidienne pondérée où le facteur de pondération d'une composante est sa variance par rapport à toute la base de données. Pour les histogrammes de couleurs, une première distance est utilisée sur les vecteurs couleurs moyens comme filtre, ensuite une distance quadratique est effectuée sur les histogrammes de couleurs.

2.3.2 Chabot

Le système Chabot [9] a été développé en 1995 par le *"Departement of Computer Scienc"* de l'université de Californie à Berkeley [2]. Ce système combine les informations textuelles qui annotent les images avec les techniques de recherche d'images par contenu. Pour chaque image, un histogramme de couleurs de 20 entrées est calculé. L'utilisateur peut définir des concepts en combinant des critères textuels et visuels. Par exemple, le concept *"sunse"* est défini par le mot *sunset* et une grande présence de la couleur rouge ; une image répond au critère "grande présence de la couleur rouge" si plus de 50% des pixels de l'image sont de couleur rouge, le critère "faible présence de la couleur rouge" est satisfait si une ou deux entrées de l'histogramme correspondant à la couleur rouge ne sont pas nulles.

2.3.3 MARS (Multimedia Analysis and Retrieval System)

Le système MARS [10] a été développé en 1997 par le *"Department of Computer Scienc"* de l'université d'Illinois à Urbana-Champaign et dans le *"Department of Information and Computer Science"* de l'université de Californie en Irvine [3]. Ce système combine l'information textuelle associée aux images avec les caractéristiques visuelles de bas niveau de l'image (couleurs, textures et formes). La couleur est représentée par un histogramme 2D en utilisant les composantes H et S de l'espace de couleurs HSV. La texture est représentée par deux histogrammes, l'un mesurant la granularité, l'autre mesurant la directivité. Les images de la base de données

[2] http ://http.cs.berkeley.edu/~ginger/chabot.html
[3] http ://www-db.ics.uci.edu/pages/research/mars.shtml

utilisée contiennent un seul objet et la forme de l'objet extrait est caractérisée par les descripteurs de Fourier [11].

2.3.4 IKONA

Succédant au système Surfimage [12], IKONA [13] a été développé par l'*"INRIA, Rocquencourt, France"* [4] dans le cadre du projet *"IMEDIA project"* en 2001. Une version de démonstration de ce système est disponible en ligne [5]. Ce système combine des caractéristiques visuelles de bas niveau et de haut niveau de l'image. Les caractéristiques de bas niveau sont :

- l'histogramme de couleurs RGB.
- l'histogramme des orientations des contours obtenus par le détecteur de Canny [11].
- la texture qui est caractérisée par la matrice de co-occurrence des niveaux de gris de l'image, la transformée de Fourier et les ondelettes.

Plusieurs mesures de similarités sont utilisées telles les distances L_p (L_1 et L_2), la distance du cosinus et la distance de Hellinger. La distance d'intersection est utilisée entre les histogrammes de couleurs.

2.3.5 Blobworld

Le système Blobworld [14] a été développé en 1999 par la division *"Developer Computer Science Division"* de l'Université de Californie à Berkeley [6]. Une version de démonstration de ce système est disponible en ligne [7]. Les caractérisations de l'image utilisées sont : la couleur, la texture, la localisation des objets, les formes des régions de l'image (*blobs*). La couleur est décrite par un histogramme de couleurs à 218 entrées de l'espace de couleurs $L^*a^*b^*$. La texture d'une région est représentée par le contraste et l'anisotropie, tandis que les formes sont représentées par leur

[4] http ://wwwrocq.inria.fr/imedia/index_UK.html
[5] http ://www-rocq.inria.fr/cgi-bin/imedia/ikona
[6] http ://elib.cs.berkeley.edu/photos/blobworld/
[7] http ://elib.cs.berkeley.edu/photos/blobworld/start.html

surface, leur excentricité et leur orientation. Partant d'une image donnée, l'utilisateur commence par sélectionner une région (*blob*) tout en indiquant son importance dans la recherche, ensuite il spécifie l'importance qui sera accordée aux couleurs, à la texture, à la localisation et à la forme de la région choisie. La mesure de similarité utilisée entre les histogrammes de couleurs est une distance quadratique, et celle utilisée pour les textures est une distance euclidienne. Ces deux mesures sont combinées en une seule mesure.

Les figures 1.3 et 1.4 du chapitre 1 montrent les résultats trouvés par les systèmes *IKONA* et *SIMPLIcity* [8] pour une recherche par une image exemple. Nous avons choisi ces exemples pour souligner le fait que la qualité d'une recherche par image exemple dépend essentiellement de(s) méthode(s) utilisée(s) pour caractériser un ou plusieurs aspects visuels de l'image, comme elle dépend aussi des mesures adoptées pour évaluer la similarité entre les images.

[8] http ://wang.ist.psu.edu/IMAGE/

CHAPITRE 3

IMPLANTATION

L'objectif d'un système de recherche d'information (recherche d'images en particulier) est de déterminer une liste classée de documents (images) pertinents par rapport à une requête de l'utilisateur qui peut être une requête texte, une image exemple, une région d'une image, un croquis dessiné par l'utilisateur ou une combinaison de certaines caractéristiques visuelles. L'implantation et l'exploitation d'un tel système se compose alors de deux modules : une phase d'indexation des documents dans une base de données et une phase d'interrogation de cette base de données.

La section 3.1 décrit le système CBIR que nous avons implanté. Le critère d'évaluation précision/rappel d'un système de recherche d'information fait l'objet de la section 3.2. Dans les sections 3.3, 3.4, 3.5 et 3.6, nous présentons respectivement les techniques que nous adoptons pour caractériser les attributs couleurs, textures, contours et formes, ainsi que les mesures de similarité associées. En particulier, nous proposons une nouvelle méthode de recherche par couleur ainsi que deux nouvelles mesures de similarité dans la section 3.3, une amélioration en deux points d'une méthode préexistante d'analyse de la texture dans la section 3.4 et une nouvelle méthode de caractérisation des formes dans la section 3.6.

3.1 Structure générale du système

Nous associons à chaque image un numéro d'identification (*id-image*) unique qui va servir comme clé primaire pour l'accès aux différentes tables de la base de donnée. Pour chacune des deux collections de ce projet, nous utilisons le langage MySQL pour stocker et manipuler deux types de tables :

- une table qui contient des informations générales sur l'image : localisation physique sur le disque, nature de l'image (couleur ou monochrome) et taille

ainsi que les champs correspondant à son annotation textuelle,

- une table par méthode de caractérisation visuelle de l'image où nous stockons les descripteurs des images (en général, il s'agit d'histogrammes).

Nous précisons donc que nous utilisons une méthode d'indexation simple et que nous n'utilisons pas une organisation prédéfinie des images en classes, c'est-à-dire que nous n'utilisons aucune information *a priori* sur les images. Le fait d'organiser les images en classes est souvent utilisé pour améliorer la performance d'un système de recherche d'images en effectuant un premier filtrage rapide lors de la phase d'interrogation ou encore pour entraîner des systèmes d'apprentissage de classification et/ou d'annotation automatique. Par ailleurs, la plupart des systèmes de recherche d'images que nous avons rencontrés dans la littérature utilisent des collections de quelques milliers d'images : entre 1000 et 4000 images.

Le schéma de notre système de recherche d'images par contenu est illustré par la figure 3.1. Une fois que la requête utilisateur et la méthode de caractérisation des images sont spécifiées, le système utilise une distance (mesure de similarité) appropriée entre les descripteurs en fonction de leur nature sémantique et syntaxique dépendemment de la méthode de caractérisation choisie. Ainsi une mesure de similarité peut tenter d'exprimer des relations telles que :

- la similarité de deux histogrammes,
- la recherche d'un objet ou l'ensemble de plusieurs objets,
- les relations spatiales ou proportionnelles entre les régions de deux images,
- une combinaison pondérée de plusieurs caractéristiques visuelles, etc.

Pour l'implantation des programmes, nous utilisons l'interface de programmation *Qt API-C++* sous Linux avec l'environnement de développement graphique *Qt designer*. Dans l'annexe I, nous présentons les aspects fonctionnels de notre application.

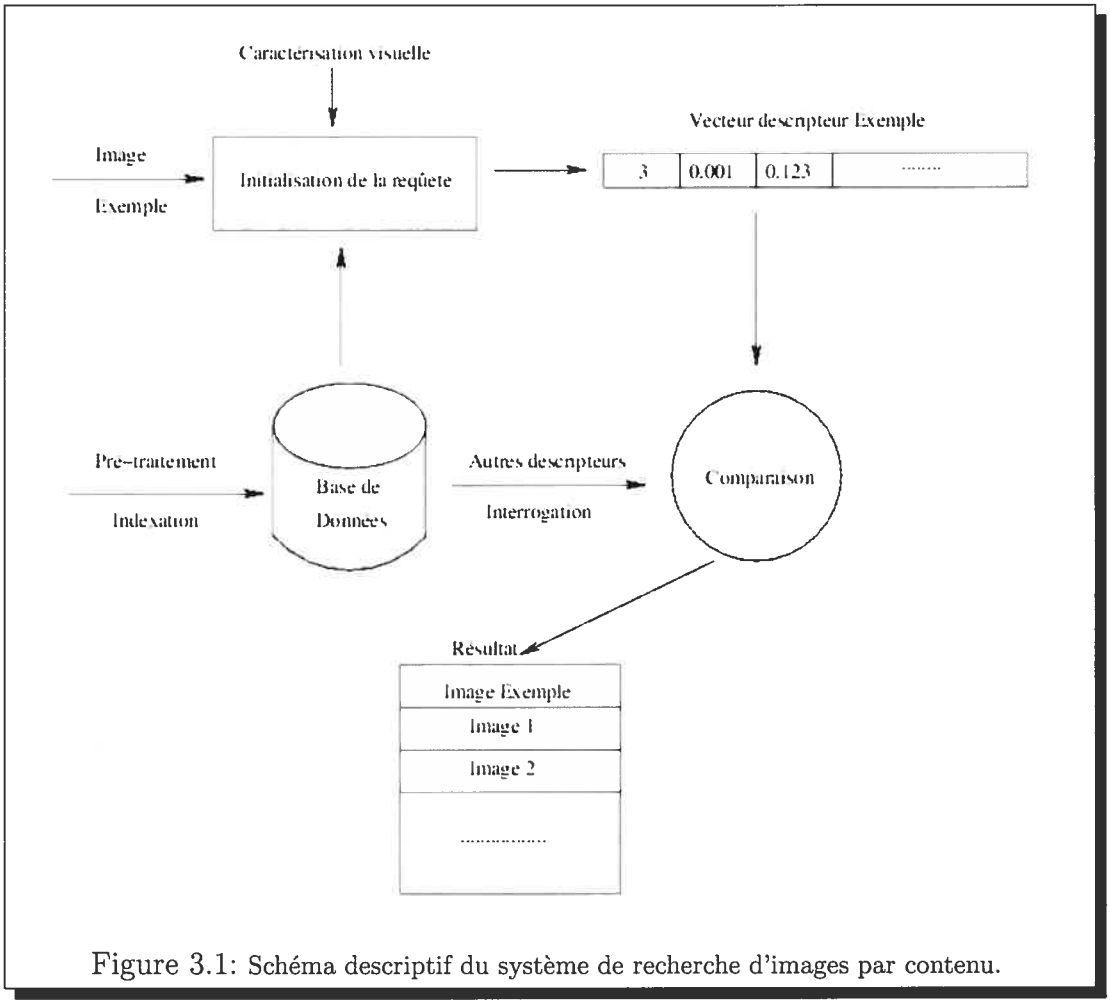


Figure 3.1: Schéma descriptif du système de recherche d'images par contenu.

3.2 Évaluation par la méthode de précision/rappel

En l'absence d'une plate-forme de tests (*benchmark*) et pour combler le manque de méthodes d'évaluation objectives, rationnelles et globales d'un système de recherche d'images, on utilise souvent un critère d'évaluation basé sur les notions de rappel et de précision pour un certain nombre d'images choisies au préalable. Soit n la taille de la collection utilisée (20000 pour les images *Corel* et 28300 pour les images *St. Andrews*). Soit I l'image considérée pour l'évaluation du système et $L = (I_0, I_1, \ldots, I_{n-1})$ la liste ordonnée des images retournées par le système en utilisant I comme image exemple. On demande alors à un sujet humain d'évaluer visuellement les images de la liste L de telle sorte à attribuer un facteur de perti-

nence ϕ_i à chaque image I_i ; en général, $\phi_i = 1$ si l'image I_i est pertinente et $\phi_i = 0$ sinon. Soit $L_k = (I_0, I_1, \ldots, I_{k-1})$ la liste des premières k images de L, on définit les entités suivantes :

- $A_k = \sum_{i=1}^{k} \phi_i$: nombre des images pertinentes de L_k,
- $B_k = \sum_{i=1}^{k} (1 - \phi_i)$: nombre des images non pertinentes de L_k,
- $C_k = \sum_{i=k+1}^{n} \phi_i$: nombre des images pertinentes qui ne figurent pas dans L_k,
- $D_k = \sum_{i=k+1}^{n} (1 - \phi_i)$: nombre des images non pertinentes qui ne figurent pas dans L_k.

Les mesures de précision P_k et de rappel R_k sont exprimées comme suit :

- $P_k = \frac{A_k}{A_k + B_k}$: précision de la recherche par rapport à L_k,
- $R_k = \frac{A_k}{A_k + C_k}$: proportion des images pertinentes retrouvées dans L_k.

Enfin, les valeurs de P_k sont portées sur un diagramme précision/rappel en fonction de R_k pour différentes valeurs de k.

3.3 Couleurs

En général, le descripteur associé à l'attribut couleur est l'histogramme des couleurs de l'image qui dépend de l'espace de représentation des couleurs et de la méthode de quantification des couleurs de cet espace. Bien que plusieurs travaux de recherche d'images par couleurs utilisent les espaces HSV, $L^*a^*b^*$ ou $L^*u^*v^*$, le choix d'un espace de couleurs reste encore un problème ouvert. L'espace HSV présente l'avantage de se rapprocher de la perception humaine des couleurs et les formules de conversion à partir de et vers l'espace RGB sont relativement simples par rapport à l'utilisation de l'espace $L^*a^*b^*$. Pour des fins de test et de comparaison, nous commençons par implémenter deux méthodes simples de quantification des couleurs que nous présentons dans la section 3.3.1.

3.3.1 Quantification régulière des espaces de couleurs RGB et HSV

Quantification de l'espace RGB :

Pour une image I dont les couleurs sont codées dans l'espace RGB ($I(i,j) \in$

$[0, 255]^3$), nous procédons à une quantification régulière du cube $[0, 255]^3$ en subdivisant chaque axe en 5 intervalles égaux ; on obtient ainsi un découpage uniforme du cube RGB en 125 sous-cubes. Toutes les couleurs se trouvant à l'intérieur d'un sous-cube sont associées à son centre (code de la quantification), ce qui produit un histogramme de couleurs RGB de 125 bâtonnets pour chaque image. Nous utilisons ensuite la distance euclidienne D_2 comme mesure de similarité entre les histogrammes.

Quantification de l'espace HSV :

Après avoir transformé les couleurs d'une image I dans l'espace HSV, nous quantifions l'espace cylindrique HSV en 166 régions conformément à la méthode proposée par Smith [1] et illustrée dans la figure 3.2 :

- La composante relative à la teinte (H) est subdivisée en 18 arcs de cercle (quantification circulaire de 20 degrés) du fait que c'est la composante la plus significative au sens de la perception de la couleur et de telle sorte à ce que les couleurs rouge, verte, bleu, jaune, magenta et cyan (bleu vert) soient représentées chacune par trois subdivisions.
- Les composantes S et V sont subdivisées en 3 intervalles égaux.
- Les niveaux de gris sont quantifiés en 4 régions pour tenir compte de la valeur indéfinie de H quand S=0.

Avec les paramètres choisis ci-dessus, nous nous référons à cette méthode par le terme : quantification 18xH+3xS+3xV+4 de l'espace HSV. Là aussi nous utilisons la distance D_2 pour la comparaison des histogrammes.

Pour une même image exemple, les figures 3.3 et 3.4 illustrent les 25 premières images retrouvées en utilisant respectivement la quantification RGB et la quantification HSV ; l'image exemple représente un lion (couleur visuelle dominante : marron) et un fond d'herbes (couleur visuelle dominante : vert). De même, la figure 3.5 montre les 6 premières images retrouvées pour 3 images exemples en utilisant les espaces RGB et HSV.

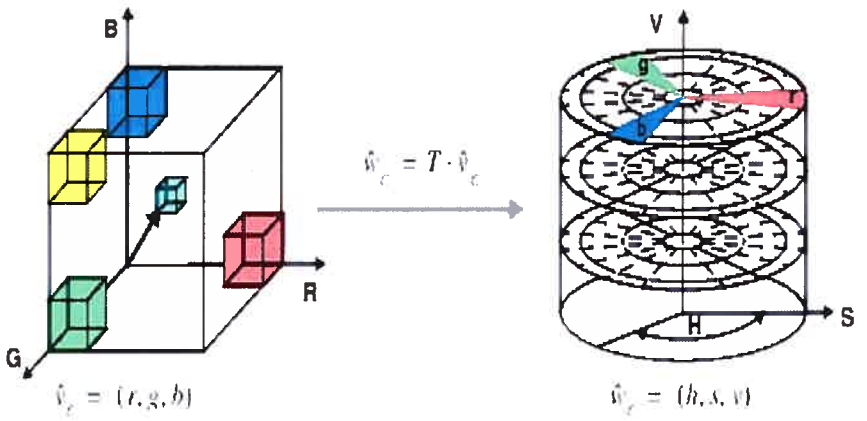


Figure 3.2: La transformation de l'espace RGB vers l'espace HSV et la quantification de ce dernier en 166 régions : 18 teintes (H) × 3 saturations (S) × 3 luminosité (V) + 4 niveaux de gris. (source : thèse de Smith [1])

On remarque que visuellement l'espace HSV donne de meilleurs résultats :

- Dans la figure 3.3, on constate la présence d'images contenant les couleurs rouge, bleu et violet tandis que celles de la figure 3.4 sont plus cohérentes au sens des couleurs dominantes de l'image exemple (marron et vert). La même remarque s'applique pour les exemples de la figure 3.5.
- En considérant l'évaluation subjective "présence d'un animal avec de l'herbe", la méthode de la quantification HSV retourne 10 images pertinentes sur les 25 premières images (figure 3.4), tandis que la méthode de la quantification RGB n'en retourne que 6 (figure 3.3).
- De même, on peut noter la performance de l'utilisation de l'espace HSV pour les exemples de la figure 3.5.

Pour ces deux méthodes, plus la quantification est fine, plus on pourrait s'attendre à de meilleurs résultats ; cependant, la taille croissante des histogrammes qui en résultera nécessitera plus d'espace de stockage et plus de temps de calcul pour la comparaison des histogrammes. Par ailleurs, le fait d'utiliser un seul histogramme global pour l'intégrité de l'image n'est pas suffisant pour tenir compte de la distribution des couleurs en régions.

Figure 3.3: Recherche des images similaires à l'image exemple 105072 (coin haut gauche) en utilisant la quantification en 125 régions de l'espace RGB. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images.

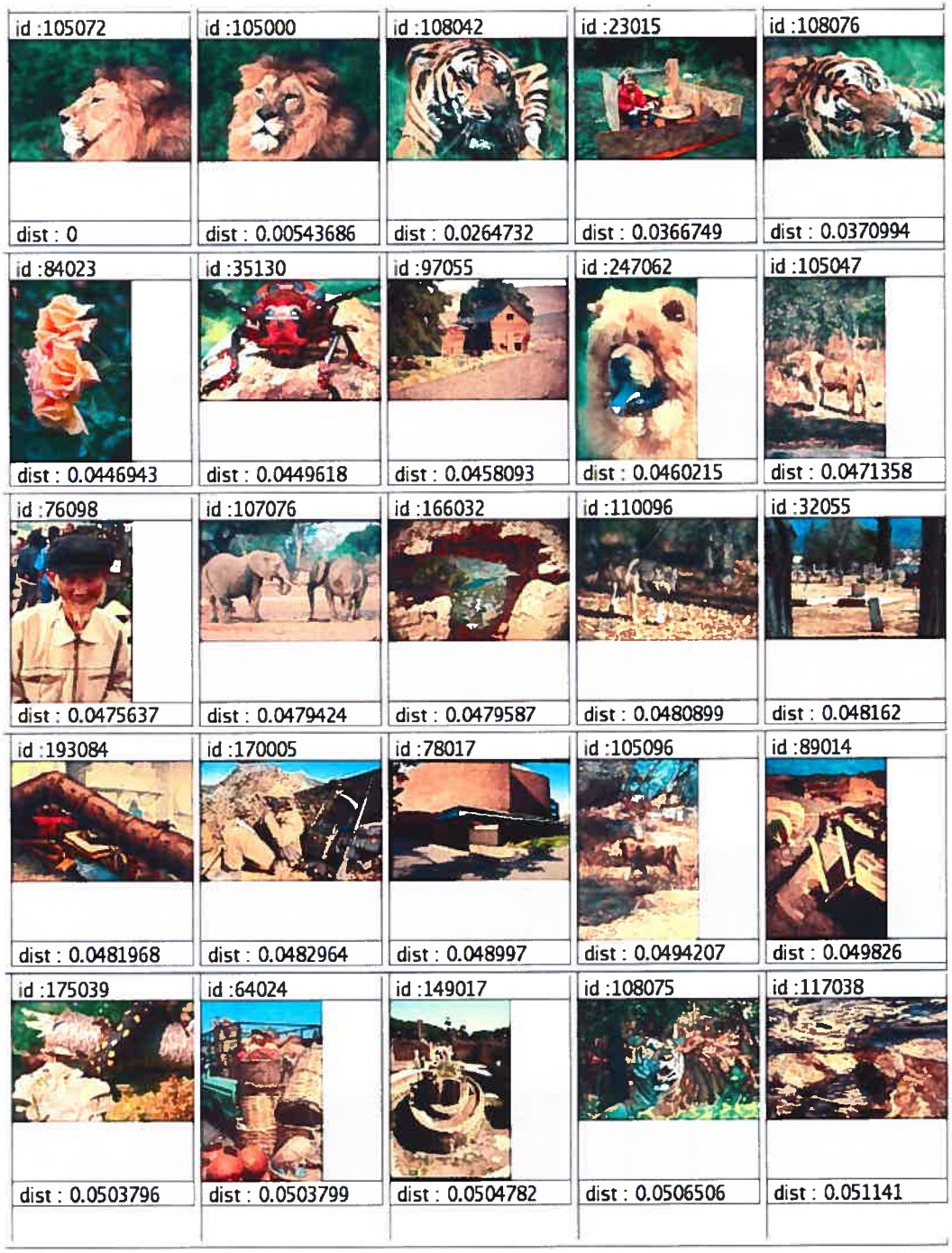


Figure 3.4: Recherche des images similaires à l'image exemple 105072 (coin haut gauche) en utilisant la quantification en 166 régions de l'espace HSV. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images.

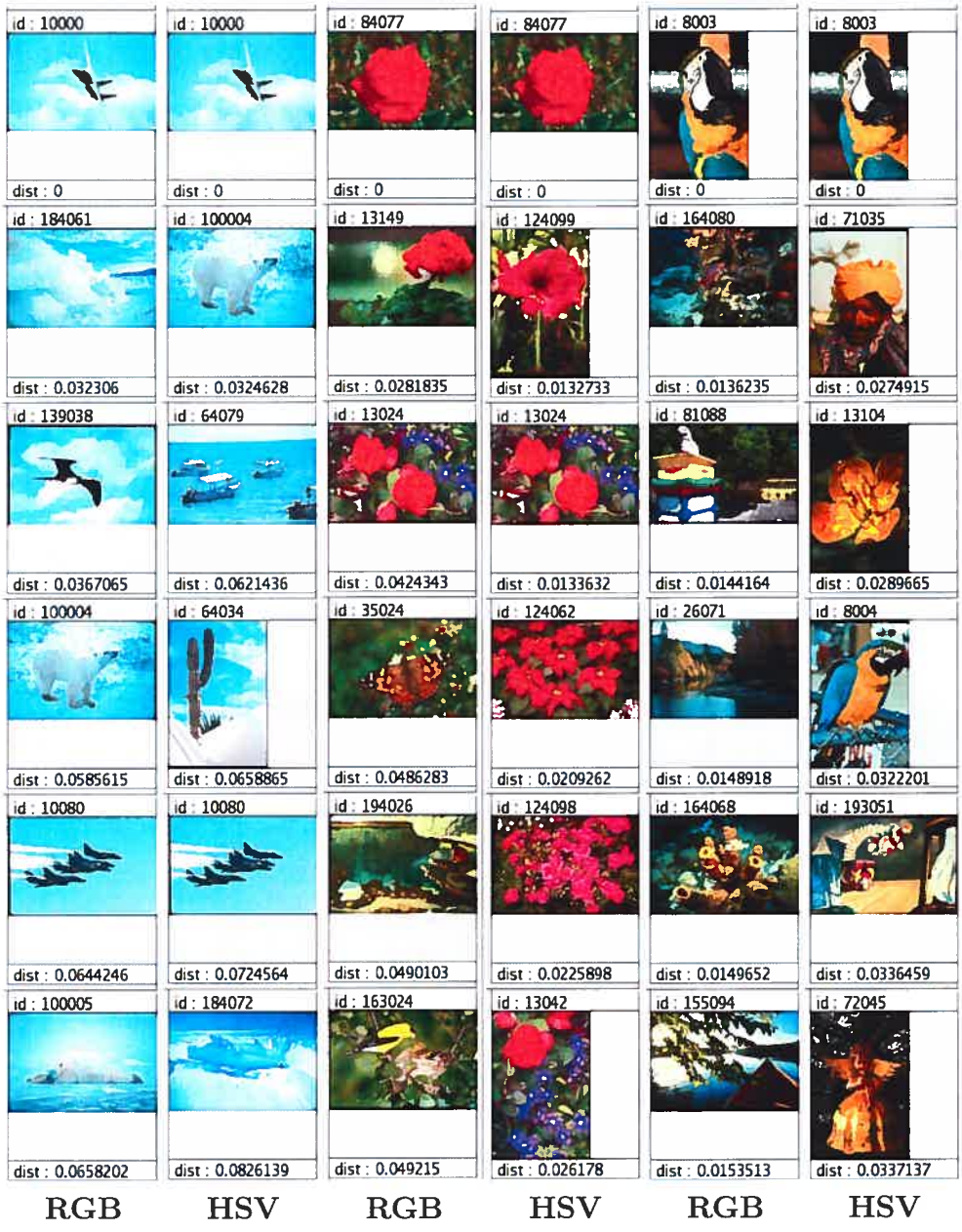


Figure 3.5: Chaque colonne d'images illustre les 6 premières images de la recherche par rapport à l'image exemple située en haut en utilisant la quantification RGB ou la quantification HSV.

3.3.2 Segmentation adaptative et distances inter-régions

La méthode que nous proposons peut être décrite sommairement en trois étapes :

1. Segmentation couleur des images en régions en utilisant l'espace couleur HSV.
2. Association de chaque image aux histogrammes couleurs de ses régions en quantifiant l'espace HSV en 54 régions.
3. Définition de deux mesures de similarité qui comparent les histogrammes des régions associés à une image exemple à ceux d'une image cible.

Étape 1 : Segmentation adaptative

Soit $I_{HSV} = \{c_{i,j} : i \in [0, W-1], j \in [0, H-1]\}$ l'ensemble des vecteurs 3D résultant de la transformation d'une image I de taille $W \times H$ dans l'espace HSV. Nous utilisons l'algorithme de groupement k-mean pour la quantification vectorielle de l'ensemble I_{HSV} pour segmenter l'image I en k régions ($k \in \{2, 3, 4\}$) au sens des couleurs HSV, ainsi nous définissons l'ensemble des codes $C = \left\{y_0^k, y_1^k, .., y_{k-1}^k\right\}$ associé au quantificateur $Q_{k-mean} : I_{HSV} \rightarrow C$ où y_i^k est le centroïde de $Q_{k-mean}^{-1}(y_i^k)$ au sens de la distance euclidienne D_2. Le quantificateur Q_{k-mean} définit alors une partition $\left\{R_i^k, i = 0, .., k-1\right\}$ de I_{HSV} où les R_i^k correspondent à la segmentation en régions de l'image I. La procédure k-mean est décrite par l'algorithme suivant :

1. Soit k le nombre de groupements de vecteurs (régions ou *clusters*) désiré pour le partitionnement de I_{HSV}.
2. Configuration d'entraînement :
   - initialiser aléatoirement l'ensemble des codes de la quantification $C = \{y_0^k, y_1^k, \ldots, y_{k-1}^k\}$ en puisant ses éléments dans I_{HSV}
   - $\forall v \in I_{HSV},\ Q_{k-mean}(v) = \arg\min_{y_i^k}(D_2(v, y_i^k))$ avec $i \in \{0, \ldots, k-1\}$
   - $\forall i \in \{0, \ldots, k-1\},\ y_i^k = centroïde(Q_{k-mean}^{-1}(y_i^k))$
3. Déplacement des vecteurs entre régions :

   $\forall v \in I_{HSV} / Q_{k-mean}(v) = y_c^k, n = \arg\min_i(D_2(v, y_i^k))$ avec $i \in \{0, .., k-1\}$

   si $c \neq n$, mettre à jour les centroïdes des régions R_c^k et R_n^k en déplaçant v de R_c^k vers R_n^k et $Q_{k-mean}(v) = y_n^k$
4. Répéter l'étape 3 jusqu'à convergence : aucun vecteur ne change de groupement

Soit $\Delta_{HSV} = \sqrt{5}$ la distance euclidienne maximale entre deux couleurs de l'espace cylindrique unitaire HSV. Pour déterminer le nombre de régions k_I associé à chaque image I, nous proposons une méthode de regroupement ("clustering") adaptative qui se base sur la distance minimale entre deux couleurs (codes de quantification) issues du groupement k-mean. L'algorithme utilisé est le suivant :

1. En prenant des valeurs décroissantes de k ($k = 4$, $k = 3$ et enfin $k = 2$), segmenter l'image I en k régions associées à l'ensemble des codes $C^k = \{y_0^k, y_1^k, \ldots, y_{k-1}^k\}$
2. Calculer la distance minimale entre les couleurs de C^k :

$$\delta^k = \min(D_2(y_i^k, y_j^k)) \text{ avec } (i,j) \in \{0,\ldots,k-1\}^2 \text{ et } i \neq j$$

3. Si $\delta^k > (0.1 \ \ \Delta_{HSV})$, $k_I = k$ et aller à l'étape 4, sinon décrémenter k et aller à l'étape 1
4. Calculer les histogrammes des k_I régions ($k_I \in \{1,\ldots,4\}$) et ne retenir que ceux qui représentent au moins une proportion de 5% de l'image.

Étape 2 : Histogrammes des régions

Une fois la segmentation d'une image I effectuée en k_I régions, nous calculons les histogrammes non normalisés h_i ($i = 0,\ldots,k_{I-1}$) des régions $R_0,\ldots,R_{k_I-1}$ ainsi obtenues. Pour cela, nous utilisons la quantification 6xH+3xS+3xV+4=54 moins fine qui consiste à découper le cercle associé à la composante H en 6 régions. Nous utilisons cette dernière pour des fins d'optimisation des temps de calcul et de stockage et pour mettre en valeur l'apport de la méthode de segmentation adaptative avec une distance inter-régions par rapport à l'utilisation d'une méthode de quantification plus fine avec une distance euclidienne. Ainsi le descripteur d'une image sera composé du nombre des régions issues de la segmentation adaptative et des histogrammes de ces régions. La figure 3.6 montre l'exemple de quelques images avec le résultat de leurs segmentations en régions ainsi que les histogrammes de ces régions obtenus par la quantification 6xH+3xS+3xV+4=54. Le fait qu'un histogramme peut contenir des bâtonnets espacés s'explique par :

- une région peut être composée de plusieurs couleurs comme pour la région de l'herbe (vert+marron) de l'image de la fleur.
- les indices de l'histogramme correspondent aux régions quantifiées de l'espace cylindrique HSV parcouru selon un ordre préférentiel des trois composantes

H, S et V, et de ce fait deux régions adjacentes de l'espace HSV peuvent correspondre à des indices non consécutifs de l'histogramme.

Le fait d'utiliser l'histogramme d'une région et non la couleur associée à son code de quantification permet de :

- Décrire la composition couleur de la région qui peut être vue comme une texture (exemple : herbe, ciel).
- Considérer chaque région comme un objet de l'image pour lequel on peut mesurer le degré de sa présence dans une image cible en utilisant la distance d'intersection $D_\cap$.
- Utiliser le nombre des régions comme un critère de filtrage pour la recherche d'images.
- Inclure une option qui permettrait à l'utilisateur de chercher une région segmentée dans les images de la collection.
- Penser à d'autres mesures qui utilisent les histogrammes comme entités homogènes en vue de leur comparaison.
- Utiliser la proportion des régions dans l'image comme critère supplémentaire dans la comparaison des images en considérant les histogrammes non normalisés.

Étape 3 : Comparaison des histogrammes des régions

Soient I_q une image exemple et I_t une image cible qui sont segmentées respectivement en k_q et k_t régions (k_q peut être différent de k_t). Soient $h_0^q, ..., h_{k_q-1}^q$ et $h_0^t, ..., h_{k_t-1}^t$ les histogrammes associés respectivement aux régions des images I_q et I_t. Soit h_t l'histogramme global non normalisé de l'image I_t. Pour l'image exemple de dimension $W_q \times H_q$, nous définissons les facteurs de pondération w_i ($i = 0, \ldots, k_q - 1$) comme la proportion de la région R_i dans l'image, c'est-à-dire

$$w_i = \frac{|h_q|}{W_q \times H_q}$$

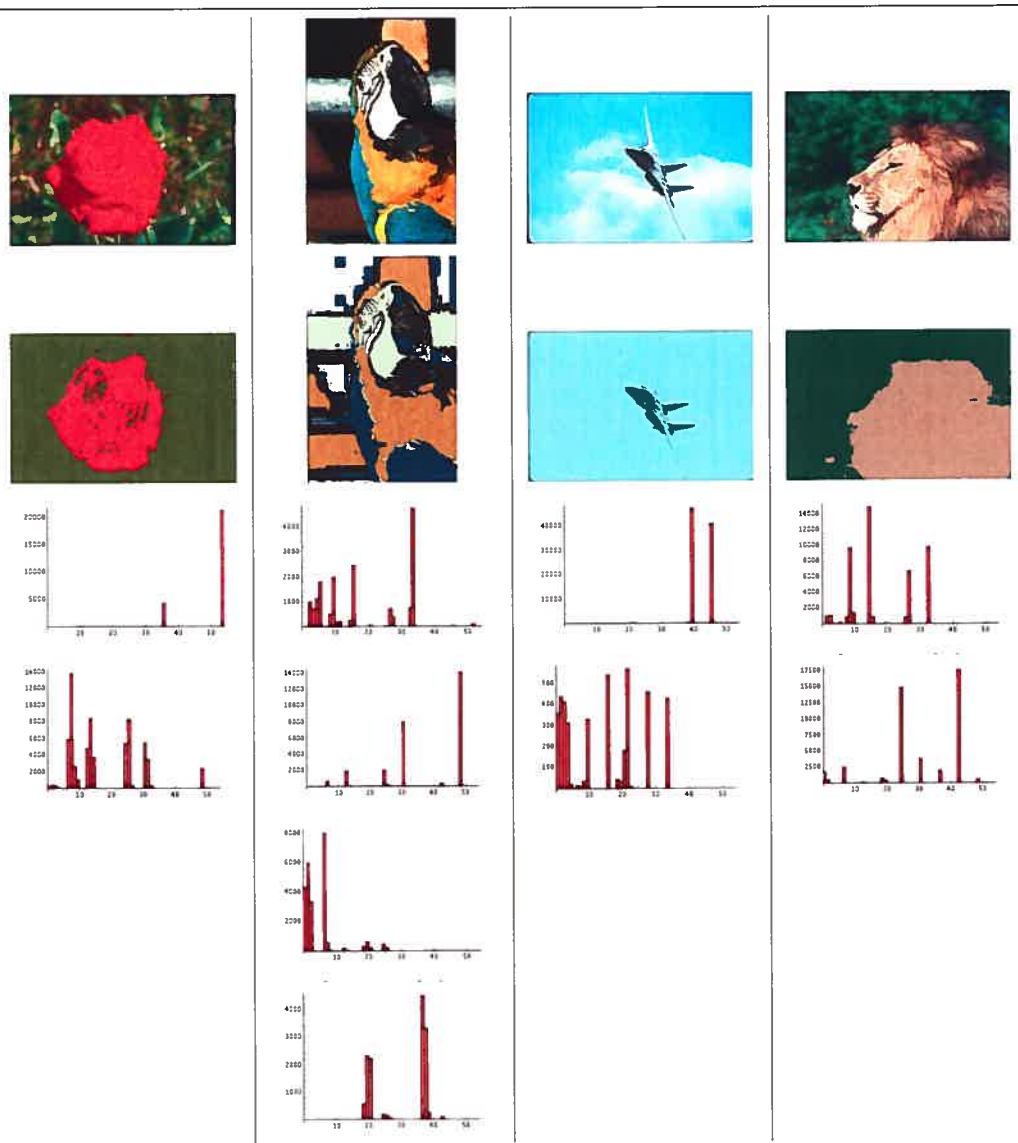


Figure 3.6: Chaque colonne contient l'image exemple, l'image segmentée en régions couleurs de l'espace HSV, les histogrammes 6xH+3xS+3xV+4 de chaque région. La segmentation de l'image de l'avion a été effectuée en 2 régions, mais nous excluons la région dont la proportion dans l'image est inférieure à 5% pour ne retenir qu'un seul histogramme.

où

$$|h_q| = \sum_{i=0}^{k_q-1} h_q(i).$$

Nous proposons deux mesures de similarité entre I_q et I_t :

1. La distance $D_{region\_a\_image}$ qui mesure le degré de présence des régions de l'image exemple dans l'image cible que nous définissons par

$$D_{region\_a\_image}(I_q, I_t) = \sum_{i=0}^{k_{I_q}-1} w_i \, D_\cap(h_i^q, h_t)$$

2. La distance $D_{region\_a\_region}$ qui utilise toutes les combinaisons possibles entre les régions de l'image exemple et celles de l'image cible pour retourner la mesure associée à la combinaison optimale et que nous définissons par

$$D_{region\_a\_region} = \begin{cases} \min_c \left(\sum_{i=0}^{k_{I_q}-1} D_1(h_i^q, h_{c(i)}^t) \right) & \text{si } |k_{I_q} - k_{I_t}| \leq 2 \\ \infty & \text{sinon} \end{cases}$$

où c représente les combinaisons possibles entre les ensembles $\{0, \ldots, k_t - 1\}$ et $\{0, \ldots, k_q - 1\}$. Par exemple pour les ensembles $\{0, 1\}$ et $\{0, 1\}$, c prendra deux formes : $c_0(0) = 0, c_0(1) = 1$ et $c_1(0) = 1, c_1(1) = 0$. Autrement dit, nous investissons toutes les façons possibles d'associer les régions des images I_q et I_t deux à deux en utilisant un filtrage qui consiste à éliminer les images dont le nombre des régions diffère de celui de l'image exemple par au moins 3.

Pour une même image, nous présentons dans l'annexe II les résultats de la recherche des images similaires par les méthodes suivantes :

- Quantification régulière de l'espace de couleurs RGB en 125 sous-cubes en utilisant la distance euclidienne D_2 (figure II.1),
- Quantification régulière de l'espace de couleurs HSV en 166 blocs en utilisant la distance euclidienne D_2 (figures II.2 et II.3),
- Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des

histogrammes des régions par la distance $D_{region\_a\_image}$ (figures II.4 et II.5),
- Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des histogrammes des régions par la distance $D_{region\_a\_region}$ (figures II.6 et II.7).

Une comparaison visuelle permet de rapporter les remarques suivantes :
- Comme pour les exemples précédents, l'espace HSV produit de meilleurs résultats que l'espace RGB.
- Sur les 50 premières images retournées par la méthode de quantification en 166 régions de l'espace HSV, certaines images contiennent des couleurs qui ne concordent pas avec l'image exemple (bleu, jaune et violet) et ce avec un degré beaucoup plus moindre comparativement aux méthodes qui utilisent les distances inter-régions.
- Bien que les méthodes qui utilisent les distances inter-régions se basent sur une quantification moins fine (54 blocs), les résultats sont aussi satisfaisants que l'utilisation de la quantification régulière de l'espace HSV en 166 régions.
- Sur les 50 premières images retournées, la distance $D_{region\_a\_region}$ se comporte mieux que la distance $D_{region\_a\_image}$. En fait, cette dernière utilise des facteurs de pondération qui dépendent de la superficie de chaque région dans l'image. D'autre part, la distance $D_{region\_a\_region}$ cherche à représenter la combinaison optimale entre les diverses régions des images à comparer.
- Les images non pertinentes de la recherche par la distance $D_{region\_a\_image}$ contiennent toutes un fond verdâtre du fait que la région dominante de l'image exemple est verdâtre. Nous aurions pu utiliser des facteurs de pondération égaux pour avoir un meilleur effet.

3.4 Les propriétés texturales de Tamura

Dans l'absence d'une définition précise, la texture peut être vue comme une propriété relative, entre autres, à la nature et à l'apparence d'un objet ou à la

disposition de plusieurs objets. Une définition plus formelle consiste à considérer une texture comme un ensemble de primitives (*"textons"*) qui sont caractérisées, entre autres, par leurs dimensions ou résolutions, leurs périodicités, leurs contrastes et leurs complexités. La texture est aussi un facteur primordial dans le processus humain de la reconnaissance des objets, qui est difficile à décrire par des mesures qualitatives, d'autant plus que les textures observées dépendent des conditions de lumière, de l'angle de vue et de la distance de prise de vue.

Pour l'analyse des textures, il existe plusieurs méthodes quantitatives qu'on peut classer grossièrement en deux catégories : les méthodes statistiques qui décrivent les relations entre un pixel et ses voisins et les méthodes structurelles qui décrivent les primitives qui composent la texture. Les méthodes de la première catégorie utilisent des mesures statistiques des niveaux de gris des pixels d'une image :

- Statistiques de premier ordre de la distribution des niveaux de gris : moyenne, variance, rapport signal-bruit (SNR : rapport de la moyenne sur l'écart-type), le *"kurtosis"* qui correspond au moment d'ordre 4 centré autour de la moyenne, etc.
- Statistiques de second ordre : entre autres, la méthode des matrices de co-occurrence et la méthode de différence des niveaux de gris (moyenne, variance, corrélation, entropie, contraste, moment angulaire, etc.).
- Statistiques d'ordres supérieurs qui étudient notamment la distribution des plages de niveaux de gris (régions ayant le même niveau de gris).

Dans le cadre des méthodes structurelles, nous adoptons l'approche proposée par Tamura, Mori et Yamawaki [8] pour caractériser les textures. En fait, ces attributs texturaux constituent une alternative intéressante à l'utilisation des méthodes statistiques du fait qu'ils caractérisent les textures par un nombre réduit de dimensions qui correspondent à la perception humaine des textures. Les dimensions introduites par Tamura *et al.* sont : le contraste, la granularité de la texture (*"Coarseness"*), la directivité (*"Directionality"*), la similitude des contours (*"line-likeness"*), la régularité (*"regularity"*) et la rugosité (*"roughness"*). Nous utilisons juste les propriétés de granularité et de directivité comme caractérisation de la

texture pour les deux raisons suivantes :

- Il semble qu'il y ait une corrélation entre les trois premières propriétés et les autres [15],
- Les images historiques de la collection *"St. Andrews"* présentent des contrastes très variés dépendemment de leurs origines (figure 3.7) et non de leurs contenus.

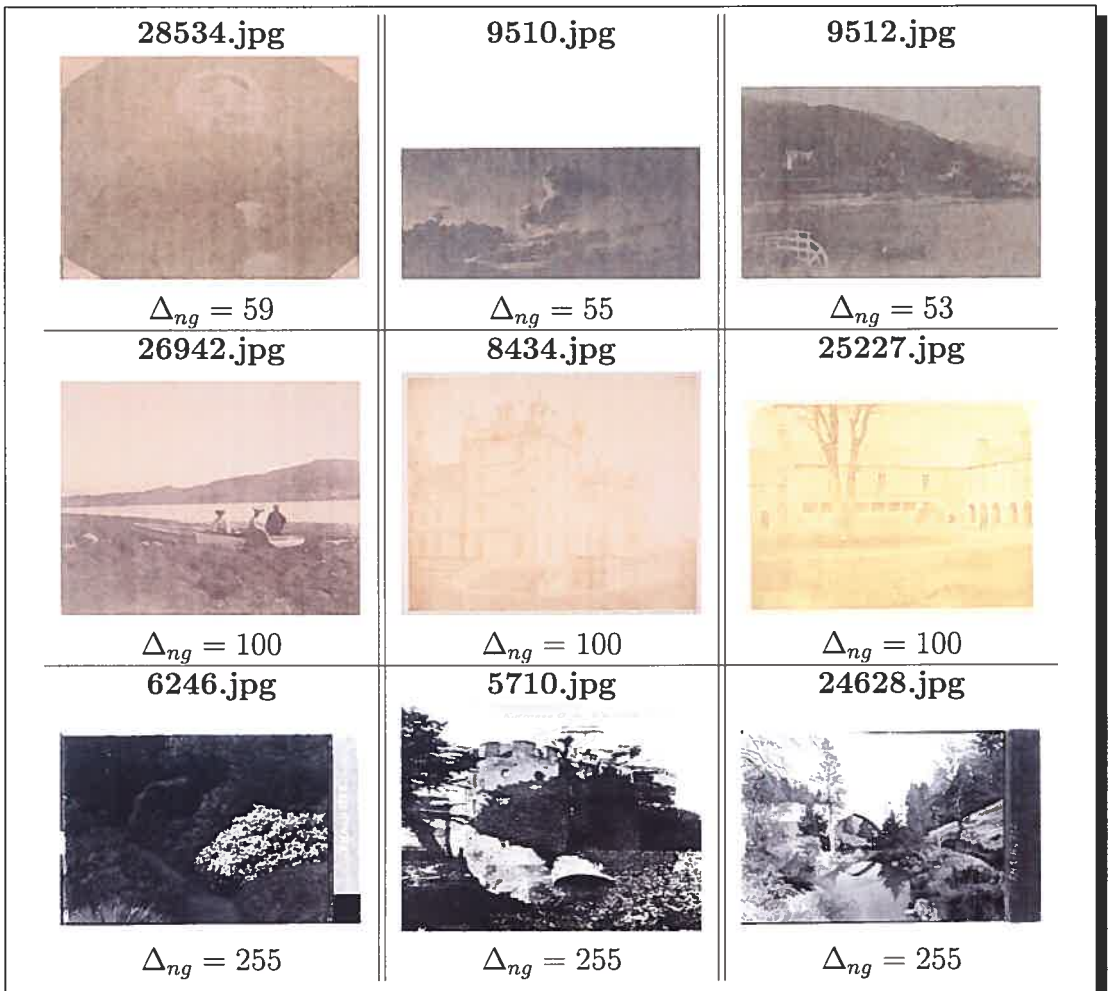


Figure 3.7: Exemple de quelques images de la collection *"St. Andrews"* avec des niveaux de contrastes très variés. La plage de concentrations des bâtonnets de l'histogramme des niveaux de gris est indiquée en dessous de chaque image.

Contraste

Le contraste d'une image est une propriété qui peut être définie comme étant le degré des variations locales des intensités des pixels et pour laquelle plusieurs mesures peuvent être utilisées (par exemple, à partir de la matrice de co-occurrence). Dans [8], le contraste est défini par :

$$F_{con} = \frac{\sigma}{(\alpha_4)^n}$$

où σ est l'écart-type de l'histogramme des niveaux de gris de l'image, α_4 est le *"kurtosis"* qui mesure la polarisation des régions noires et blanches dans l'image, et n un nombre positif défini à partir des expérimentations à $1/4$ pour une meilleure discrimination des textures. Le *"kurtosis"* est défini par

$$\alpha_4 = \frac{\mu_4}{\sigma^4}$$

où μ_4 est le moment central de degrés 4 de l'histogramme des niveaux de gris.

Nous commençons par transformer les images des deux collections en des images à niveaux de gris, ensuite nous ramenons les images de la collection *"St. Andrews"* à un niveau de contraste comparable en effectuant un étalage des niveaux de gris entre 0 et 255.

Granularité

La granularité est une notion qui est liée à la résolution ou à la dimension de la texture dans l'image : une image d'un immeuble prise de loin est moins granulaire que celle prise à une distance plus proche où l'apparence texturale des blocs est plus évidente. Aussi, la granularité est-elle liée à la taille des éléments qui forment la texture, et un estimateur de ce paramètre serait la résolution ou échelle qui décrit le mieux la taille des éléments de la texture dans l'image. L'algorithme utilisé dans la littérature [8] [16] pour le calcul de la granularité d'une image est le suivant

1. Pour chaque pixel de coordonnées (x, y) de l'image I, et pour chaque valeur de k (k prend ses valeurs dans $\{1, 2, \ldots, 6\}$), calculer la moyenne des niveaux de gris de ses voisins dans la fenêtre de taille $2^k \times 2^k$:

$$A_k(x,y) = \sum_{i=(x-2^{k-1})}^{(x+2^{k-1}-1)} \sum_{j=(y-2^{k-1})}^{(y+2^{k-1}-1)} \frac{I(i,j)}{2^{2k}}$$

où $I(i, j)$ est le niveau de gris [a] du pixel (i, j) de l'image I.

2. Pour chaque pixel, et pour les directions horizontales et verticales, calculer la différence entre les moyennes des fenêtres qui ne se chevauchent pas et qui se trouvent de part et d'autre du pixel. Les différences horizontales et verticales sont exprimées par :

$$\begin{aligned} E_{k,horizontal}(x,y) &= |A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)| \\ E_{k,vertical}(x,y) &= |A_k(x,y+2^{k-1}) - A_k(x,y-2^{k-1})| \end{aligned}$$

3. Pour chaque pixel, la valeur de k qui maximise $E_k(x, y)$, sans tenir compte de la direction, est considérée comme étant la meilleure description de la résolution de la texture au niveau de ce pixel, c'est-à-dire $S_{best}(x, y) = 2^k$. Comme descripteur, on peut considérer, soit la mesure scalaire de la granularité qui est la moyenne des S_{best} sur toute l'image, soit l'histogramme des S_{best} qui est plus précis pour la discrimination des textures.

[a] Composante Y de l'espace de couleurs YIQ qui est traduite dans l'interface de programmation Qt-API par la formulation $(r * 11 + g * 16 + b * 5)/32$ où r, g et b représentent les couleurs de l'espace RGB.

Le fait d'utiliser la distribution de S_{best} permet de prendre en compte l'existence de plusieurs résolutions de textures dans l'image. Par ailleurs, nous apportons une modification en deux points à cet algorithme :

1. Au lieu de considérer des fenêtres de taille $2^k \times 2^k$ avec $k = 1, \ldots, 6$ (c'est-à-dire 2, 4, 8, 16, 32 et 64), nous fixons ces tailles à 2, 8, 14, 20, 26, 32 et 38. Ceci permet de prendre compte des résolutions de textures d'une façon plus régulière en augmentatnt à chaque fois la résolution de la texture par 6 pixels. Ainsi, nous utilisons $k = 0, \ldots, 6$ comme indice d'un tableau à 7 dimensions qui stocke les tailles choisies.
2. Pour l'étape 3 de l'algorithme ci-dessus, nous utilisons un seuil T de telle sorte que si $E_{k,horizontal} < T$ et $E_{k,vertical} < T$ pour $k = 0, \ldots, 6$, alors $S_{best}(x, y) = 38$. En d'autres termes, si le pixel se trouve dans une région homogène et que la taille maximale utilisée n'est pas suffisante pour décrire la résolution de la texture en ce point, on prend la résolution maximale au lieu d'une valeur qui sera prise aléatoirement en fonction du bruit présent dans l'image.

Directionnalité

La directionnalité est une propriété texturale globale de l'image dans le sens où elle est reliée à la forme de l'élément de la texture (exemple : texture d'une image de briques) et à la disposition de ces éléments (horizontalement ou verticalement). Une texture de forte directionnalité se manifeste dans des images où les contours sont orientés selon quelques directions seulement (exemple : un mur de briques), et une texture à faible directionnalité se manifeste lorsqu'il n'y a pas une structure régulière des éléments de la texture (exemple : nuages). Cette propriété mesure la concentration des orientations des points de contours selon quelques directions.

La méthode utilisée dans [8], peut être décrite comme suit :

1. Calculer la direction et la magnitude des contours en utilisant le détecteur de contours Sobel [11].
2. Après quantification des directions, compter les pixels ayant une magnitude supérieure à un certain seuil pour construire l'histogramme $H(\phi)$ des directions.
3. Utiliser une mesure de similarité qui tient compte des creux d'un histogramme, c'est-à-dire de la distance entre ses bâtonnets-sommets.

Du fait qu'un histogramme $H(\phi)$ ayant quelques bâtonnets est lié à une texture de forte directionnalité (et inversement, un histogramme plat est lié à une texture de faible directionnalité), une mesure quantitative de l'élancement des bâtonnets d'un histogramme $H(\phi)$ est utilisée dans [8] comme descripteur de l'image

$$F_{dir} = 1 - rn_p \sum_p \sum_{\phi \in w_p} (\phi - \phi_p)^2 H(\phi)$$

où

- n_p est le nombre des bâtonnets-sommets de $H(\phi)$,
- ϕ_p est la position du $p^{\text{ième}}$ bâtonnet-sommet dans $H(\phi)$,
- pour chaque bâtonnet-sommet p, ω_p est l'ensemble des bâtonnets distribués dans son alentour,
- r est un facteur de normalisation lié à la finesse de la quantification de ϕ (nombre d'entrées de $H(\phi)$).

Pour l'implantation de notre système, nous effectuons une convolution de chaque image I avec les filtres de Sobel S_H et S_V

$$S_H = \begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \quad \text{et} \quad S_V = \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

pour obtenir deux images contours G_x et G_y. Nous calculons ensuite l'image gradient $\bigtriangledown$ par l'approximation suivante :

$$\bigtriangledown(i,j) = |G_x(i,j)| + |G_y(i,j)|, \forall (i,j) \in [0,W] \times [0,H].$$

Pour un pixel (i,j) dont le gradient est supérieur au seuil $T = 20$, nous calculons l'angle du contour par

$$\phi = \arctan\left(\frac{G_y(i,j)}{G_x(i,j)}\right) + \frac{\pi}{2}.$$

Finalement, nous quantifions l'ensemble des valeurs de ϕ (l'intervalle $[0, 2\pi[$) en 8 intervalles égaux. Dans la figure 3.8, nous montrons quelques images avec leurs histogrammes de granularité et de directionnalité.

Descripteur et mesure de similarité retenus

Pour la caractérisation de la texture, nous retenons la combinaison des deux histogrammes de la granularité et de la directionnalité comme descripteur de dimension $K = 7 + 8 = 15$. Soient h_q et h_t les descripteurs de deux images I_q et I_t, en considérant h_q et h_t comme des histogrammes, plusieurs mesures statistiques peuvent être utilisées comme distances de similarités [17] [18] [19], nous en citons quelques-unes avec leurs expressions empiriques :

1. Le test statistique χ^2 qui est exprimé par

$$D_{\chi^2}(h_q, h_t) = \sum_{k=1}^{K} \frac{(h_q(k) - \widehat{h}(k))^2}{\widehat{h}(k)} \tag{3.1}$$

   où $\widehat{h}(k) = [h_q(k) + h_t(k)]/2$.

2. La distance Kullback-Leibler-divergence (KL-divergence) qui est exprimée par

$$D_{KL}(h_q, h_t) = -\sum_{k=1}^{K} h_q(k) \ln \frac{h_t(k)}{h_q(k)}. \tag{3.2}$$

3. La distance empirique Jeffrey-divergence qui est exprimée par

$$D_{JD}(h_q, h_t) = \sum_{k=1}^{K} \left[h_q(k) \log \frac{h_q(k)}{\widehat{h}(k)} + h_t(k) \log \frac{h_t(k)}{\widehat{h}(k)} \right]. \tag{3.3}$$

   Comparativement à la distance Kullback-Leibler-divergence, la distance Jeffrey-divergence est symétrique, plus stable numériquement et plus robuste au bruit et à la taille des histogrammes [17].

4. La distance WMV (Weighted-Mean-Variance : Moyenne-Variance-Pondérées)

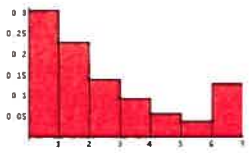
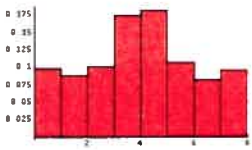
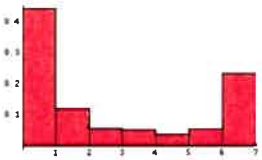
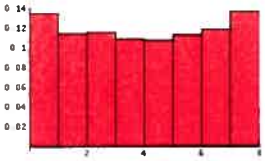
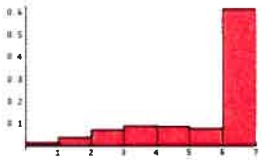
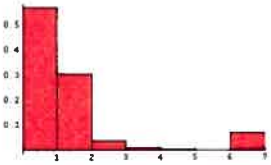
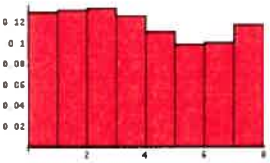
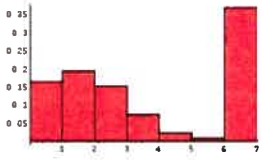
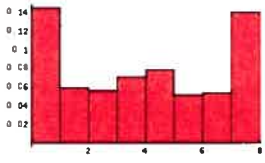
image	histogramme granularité	histogramme directivité
 St. Andrews 7753.jpg		
 St. Andrews 35.jpg		
 Corel 10000.jpg		
 Corel 11011.jpg		
 Corel 11016.jpg		

Figure 3.8: Exemples de quelques images des collections *St. Andrews* et *Corel* avec leurs histogrammes de granularité à 7 dimensions (7 résolutions de textures) et de directivité (8 directions).

qui est exprimée par

$$D_{WMV}(h_q, h_t) = \frac{|\mu_q - \mu_t|}{\sigma(\mu)} + \frac{|\sigma_q - \sigma_t|}{\sigma(\sigma)} \tag{3.4}$$

où μ_q et μ_t sont les moyennes de h_q et h_t, σ_q et σ_t sont les écarts-types de h_q et h_t, et $\sigma(.)$ est l'écart-type d'une entité par rapport à toutes les images de la collection.

5. La distance de Fisher qui est exprimée par

$$D_F(h_q, h_t) = \frac{(\mu_q - \mu_t)^2}{\sigma_q^2 + \sigma_t^2}. \tag{3.5}$$

Dans les expérimentations de Puzicha *et al.* [17] [18] , la distance de Jeffrey-divergence et le test χ^2 ont été préférés à d'autres mesures de similarité pour la discrimination des textures. Dans [19], des expérimentations sur 344 images tomographiques ont montré que la distance de Jeffrey-divergence donne de meilleurs résultats par rapport à 7 autres mesures de similarité. D'autre part, la distance de Jeffrey-divergence est très bien connue et utilisée dans le domaine de la recherche d'images pour l'attribut texture. Pour ces raisons, nous adoptons la distance de Jeffrey-divergence dans notre système. La figure 3.9 montre l'exemple d'une recherche d'images par contenu en utilisant notre descripteur basé sur les propriétés de Tamura *et al.* et la distance Jeffrey-divergence.

3.5 Caractérisation des contours par les ondelettes

Les ondelettes sont des fonctions qui ont l'allure de petites ondes ou ondelettes. La transformation d'une image par ondelettes est une méthode multi-résolutionnelle et hiérarchique pour la caractérisation et la décomposition du signal 2D de l'image dans les dimensions fréquentielles et spatiales (ou temporelles). Ainsi, l'image est transformée en un ensemble de coefficients correspondant à diverses échelles d'analyse. Elle constitue un moyen plus efficace d'analyse du contenu de l'image par rapport aux coefficients de la transformée de Fourier [11] qui ne traitent que de

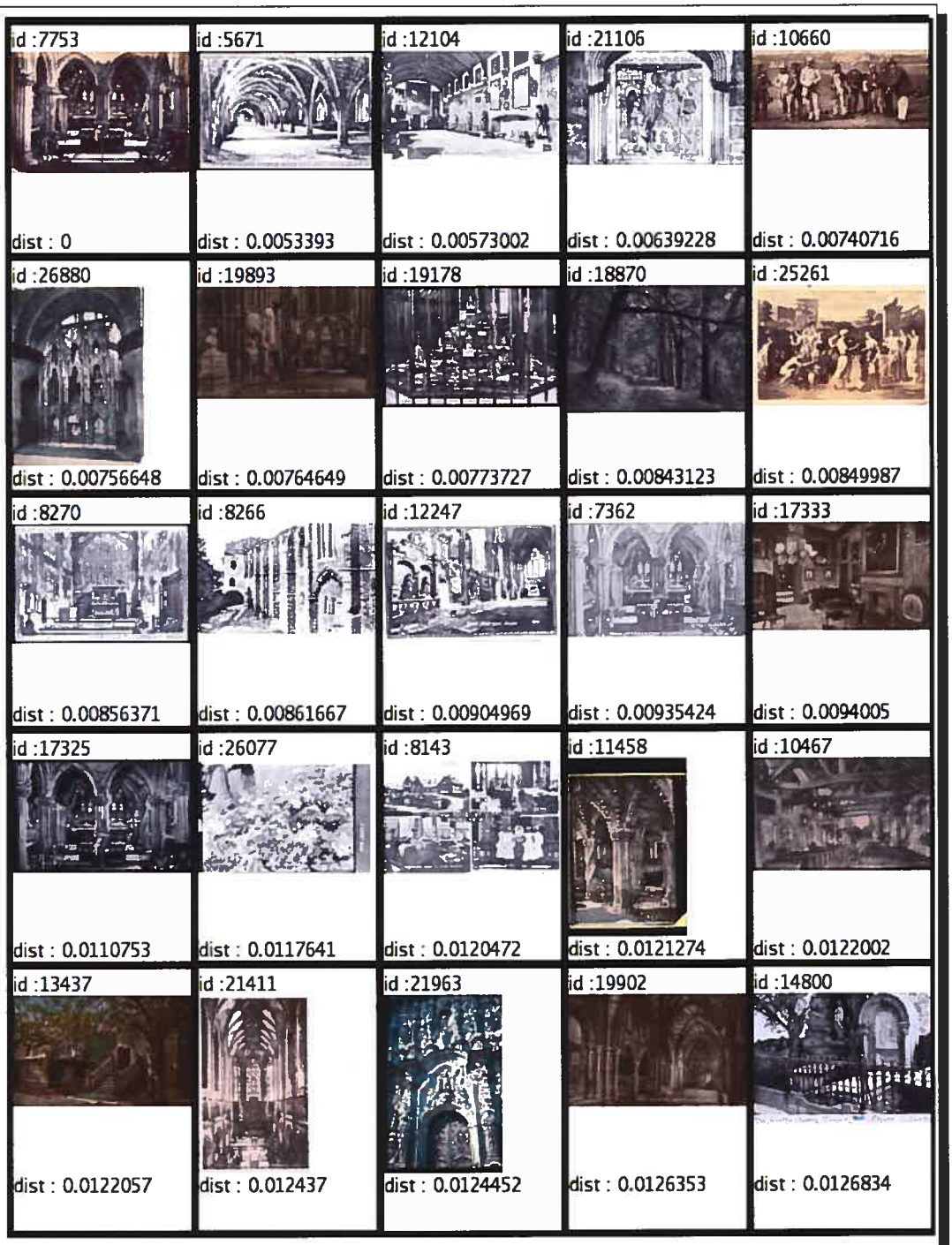


Figure 3.9: Recherche des images similaires à l'image exemple *7753.jpg* (coin haut gauche) en utilisant le descripteur composé des histogrammes de la granularité et de la directionnalité. La distance de similarité utilisée est la distance Jeffrey-divergence. La recherche est effectuée sur un ensemble de 28133 images.

l'aspect fréquentiel d'un signal en utilisant la décomposition par les fonctions sinus et cosinus. Les ondelettes sont très utilisées dans les systèmes de recherche d'images parce qu'elles permettent de décrire le contenu textural et la distribution des contours d'une image à plusieurs échelles (analyse multi-résolution).

Dans notre système, nous utilisons la transformation par les ondelettes de Haar [20] des niveaux de gris d'une image. Pour la tâche spécifique de recherche d'images, il paraît que l'utilisation d'autres types d'ondelettes (les ondelettes de Daubechies par exemple) n'apporte pas d'amélioration notable à la performance d'un système de recherche d'images [15].

Notions de base pour l'introduction des ondelettes

Soit $C[0,1]$ l'ensemble de toutes les fonctions continues sur l'intervalle $[0,1]$ et qui est un espace vectoriel,

- Le produit intérieur ("*inner product*") entre deux éléments f et g de $C[0,1]$ se définit par
$$< f|g >= \int_0^1 f(x)g(x)dx.$$
- Deux éléments u et v de $C[0,1]$ sont orthogonaux au sens du produit intérieur $< .|. >$ si $< u|v >= 0$.
- Un élément u de $C[0,1]$ est normalisé au sens du produit intérieur $< .|. >$ si $< u|u >= 1$.

Ondelettes 1D de Haar [21]

La transformation par les ondelettes d'un signal résulte en un ensemble de coefficients (appelés coefficients d'ondelettes ou détails). Pour illustrer la notion des détails d'un signal et la méthode de moyennage et différence utilisée pour les ondelettes de Haar, nous considérons l'exemple d'un signal 1D, soit $f_4 = [9 \quad 7 \quad 3 \quad 5]$. Le moyennage des éléments de f deux à deux donne $f_2 = [8 \quad 4]$. Pour pouvoir retrouver f_4 à partir de f_2, on a besoin des détails qui ont été perdus dans le moyennage, pour cela on va retenir $1 = 9 - 8 = 8 - 7$ pour retrouver les deux premières valeurs de f_4. De même, on retient $-1 = 3 - 4 = 4 - 5$ pour retrou-

ver les deux autres valeurs de f_4. En réitérant le même principe, on obtient la décomposition multi-résolutionnelle suivante

Signal	Résolution	Moyennes	Coefficients / Détails
f_4	4	[9 7 3 5]	
f_2	2	[8 4]	[1 -1]
f_1	1	[6]	[2]

qui donne la transformation suivante $T_{Haar}(f_4) = [6 \quad 2 \quad 1 \quad -1]$.

Un signal f peut être vu comme un ensemble de fonctions constantes par intervalles (fonctions en escalier) définies sur l'intervalle $[0, 1[$. Si f est composé de $n = 2^i$ éléments, il peut être considéré comme une fonction en escalier définie sur les n sous-intervalles réguliers de $[0, 1[$. L'ensemble de toutes les fonctions de ce type est noté V^i ($f \in V^i$) et constitue un sous-espace vectoriel de $C[0, 1]$. Il est facile de remarquer que les espaces V^j sont imbriqués au sens de l'inclusion (c'est à dire $V^0 \subset V^1 \subset V^2 \ldots$), ce qui est une condition nécessaire dans la théorie de la résolution multi-dimensionnelle.

Pour chaque espace V^j, on définit une base de fonctions

$$\phi_i^j(x) = \phi(2^j x - i), \quad i = 0, \ldots, 2^j - 1,$$

où

$$\phi(x) = \begin{cases} 1 & \text{si } 0 \leq x < 1 \\ 0 & \text{sinon} \end{cases}.$$

Ces fonctions correspondent à une dilatation et diverses translations de la fonction porte ϕ (appelée aussi fonction ouverture). En exemple, la figure 3.10 montre les fonctions de la base de V^2.

On définit ensuite W^j comme l'espace vectoriel de toutes les fonctions de V^{j+1} qui sont orthogonales à toutes les fonctions de V^j au sens du produit intérieur. Les ondelettes sont alors définies comme étant l'ensemble des fonctions $\Psi_i^j(x)$ qui englobent (*"span"*) W^j et qui permettent de représenter la partie des fonctions

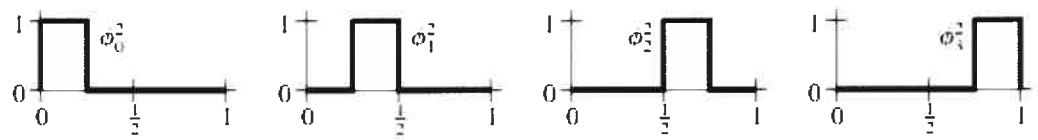


Figure 3.10: Les fonctions de la base de V^2 qui sont obtenues à partir de la fonction porte ϕ. (source : http ://diuf.unifr.ch/courses04-05/improc/Annexes/).

de V^{j+1} qui ne peut être représentée dans V^{j+1}. Les ondelettes correspondant à la fonction porte ϕ sont connues sous le nom des ondelettes de Haar et sont exprimées par

$$\Psi_i^j(x) = \Psi(2^j x - i), \quad i = 0, \ldots, 2^j - 1,$$

où

$$\Psi(x) = \begin{cases} 1 & \text{si } 0 \leq x < 1/2 \\ -1 & \text{si } 1/2 \leq x < 1 \\ 0 & \text{sinon} \end{cases}$$

est l'ondelette mère. La figure 3.11 montre les ondelettes de Haar de l'espace W^1.

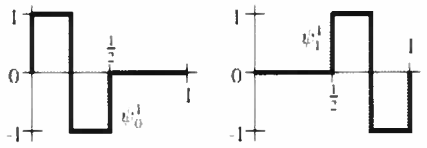


Figure 3.11: Les ondelettes de Haar relatives à l'espace W^1. (source : http ://diuf.unifr.ch/courses04-05/improc/Annexes/)

On peut rendre la base de Haar orthonormée en normalisant les fonctions $\phi_i^j(x)$ et $\Psi_i^j(x)$ par

$$\phi_i^j(x) = 2^{j/2}\phi(2^j x - i) \quad \text{et} \quad \Psi_i^j(x) = 2^{j/2}\Psi(2^j x - i).$$

Dans ce cas, les coefficients des ondelettes seront multipliés par les termes $2^{-j/2}$ relativement à leur indice j. Pour l'exemple de ce paragraphe, les coefficients nor-

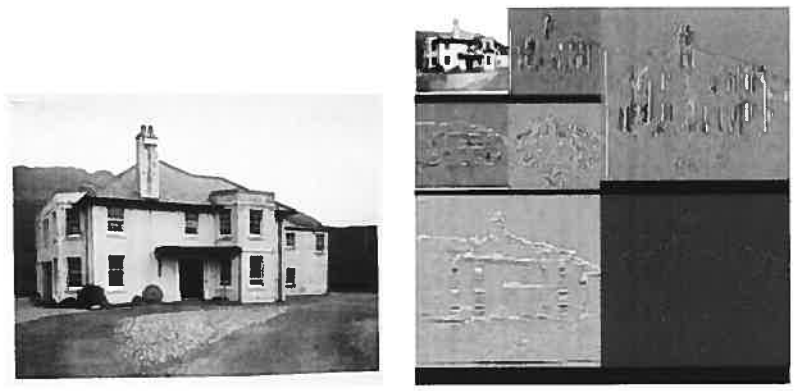
malisés deviendront $[6 \quad 2 \quad 1/\sqrt{2} \quad -1/\sqrt{2}]$.

Descripteur basé sur les ondelettes 2D de Haar

La décomposition standard d'un signal 2D par les ondelettes consiste à appliquer la transformée d'ondelettes 1D sur chaque ligne de pixels et d'appliquer ensuite une autre transformée d'ondelettes 1D sur les colonnes résultant de la première transformation. Pour une image de taille $n \times n$, cette procédure requiert $4(n^2 - n)$ opérations [22] . Pour les ondelettes de Haar, la décomposition peut s'effectuer aussi par un ensemble d'opérations de moyennage et de différence effectuées alternativement sur les lignes et les colonnes : On fait la moyenne et la différence sur chaque paire de lignes, puis on fait la même chose sur les colonnes. On répète la même opération sur la région qui contient les moyennes. Cette technique est plus efficace avec un nombre d'opérations requis de $\frac{8}{3}(n^2 - 1)$ [22]. Ceci revient en fait à décomposer le signal 2D de l'image en 4 sous-bandes (4 ensembles de coefficients d'ondelettes) qui sont notées par LL, LH, HL et HH (L : Low and H : High) en fonction de leurs caractéristiques fréquentielles ; dans le niveau suivant de la décomposition, on réitère la même procédure pour la bande LL qui n'est autre que le moyennage de l'image. Ainsi, pour un niveau de décomposition M, nous obtenons $3M + 1$ sous-bandes. Pour l'implantation de notre système, nous utilisons un niveau de décomposition de 3, et de ce fait nous obtenons 10 ensembles de coefficients d'ondelettes : $\{C_0, \ldots, C_9\}$.

Pour chaque C_i, nous calculons la moyenne μ_i et l'écart-type σ_n pour former un descripteur $\{\mu_{C_0}, \sigma_{C_0}, \ldots, \mu_{C_9}, \sigma_{C_9}\}$ de dimension 20 qui caractérise la distribution des contours de l'image à plusieurs niveaux. La figure 3.12 montre l'exemple d'une image où nous pouvons remarquer la présence des contours horizontaux, verticaux et diagonaux de part et d'autre de la deuxième diagonale de l'image.

Comme les éléments du descripteur ainsi définis sont hétérogènes (valeurs d'ordres différents), nous utilisons la distance D_{WMV} comme mesure de similarité entre deux images I_q et I_t :



(a) (b)

Figure 3.12: *a) L'image originale* 26737.JPG *de taille* 238 × 308. *b) Les coefficients d'ondelettes de Haar de l'image a) après recallage entre* 0 *et* 255.

$$D_{WMV}(I_q, I_t) = \sum_{n=0}^{9} \left(\left| \frac{\mu_{C_n^q} - \mu_{C_n^t}}{\sigma(\mu_n)} \right| + \left| \frac{\sigma_{C_n^q} - \sigma_{C_n^t}}{\sigma(\sigma_n)} \right| \right),$$

où $\{\mu_{C_0^q}, \sigma_{C_0^q}, \ldots, \mu_{C_9^q}, \sigma_{C_9^q}\}$ et $\{\mu_{C_0^t}, \sigma_{C_0^t}, \ldots, \mu_{C_9^t}, \sigma_{C_9^t}\}$ sont les descripteurs de I_q et I_t, et $\sigma(\mu_n)$ et $\sigma(\sigma_n)$ sont les écarts-types des composantes μ_n and $\sigma_n$par rapport à toute la base de données.

La figure 3.13 montre l'exemple d'une recherche d'images qui utilise les coefficients d'ondelettes comme descripteur du contenu avec la distance D_{WMV}.

3.6 Formes

L'extraction des formes des objets contenus dans une image est un processus qui dépend essentiellement du degré de séparation et de distinction des couleurs de chaque objet vis-à-vis des autres composantes de l'image ; par exemple, les contours dans une image d'un échiquier sont beaucoup plus simples à extraire correctement que ceux d'une image qui contient des objets avec une certaine texture (des nuages, des zèbres ou un paysage d'herbes et d'arbres). Il existe plusieurs méthodes d'extraction des contours qu'on peut paramétrer (un ou plusieurs seuils) en fonction de la nature de l'image afin d'arriver à des résultats satisfaisants. Comme les images

Figure 3.13: Recherche des images similaires à l'image exemple *10000.jpg* (coin haut gauche) en utilisant le descripteur des coefficients d'ondelettes. La distance de similarité utilisée est la distance D_{WMV}. La recherche est effectuée sur un ensemble de 20000 images.

des collections que nous utilisons dans ce projet sont de natures très diversifiées, nous proposons une approche de segmentation similaire à celle discutée dans la section 3.3.2 pour extraire les contours des formes d'une façon non supervisée et en se basant sur la distinction visuelle des objets par leurs couleurs. C'est en ce sens que nous utilisons l'espace de couleurs $L^*u^*v^*$ au lieu de l'espace HSV. Par ailleurs, la méthode que nous proposons permet l'extraction des contours connectés, chose qui n'est pas toujours facile avec les méthodes standards ; ceci revient en fait à caractériser une forme par son contour.

La méthode que nous proposons est donc axée autour de deux volets :

- Segmentation de l'image en des régions homogènes selon le critère de similarité visuelle des couleurs ; ce qui est assuré par l'utilisation de l'espace $L^*u^*v^*$ qui est perceptuellement uniforme.
- Extraction des contours des formes et leur caractérisation par un descripteur approprié.

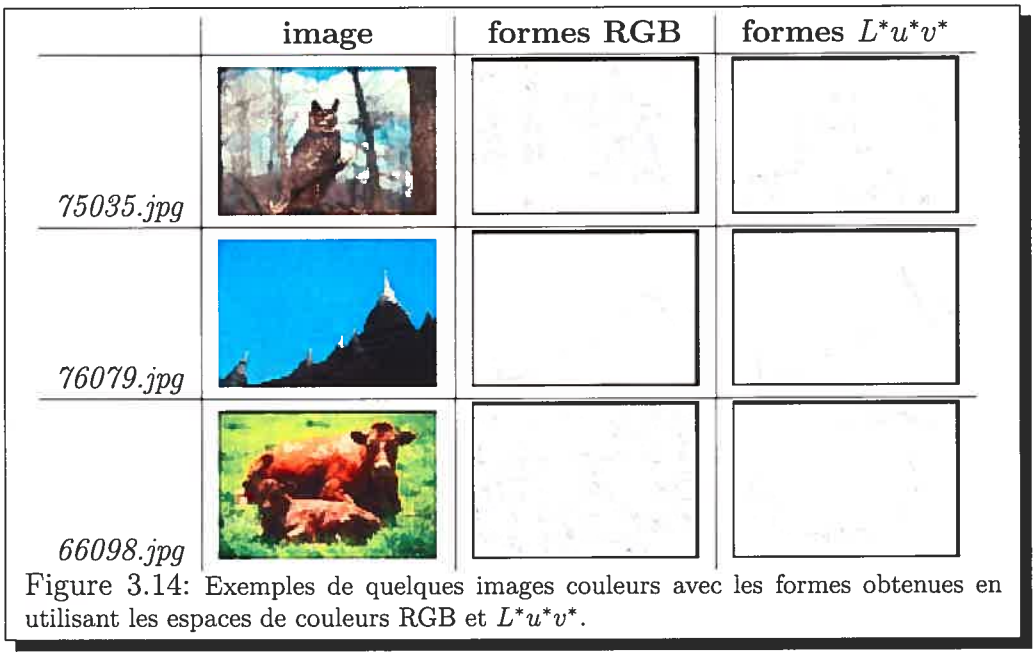


Figure 3.14: Exemples de quelques images couleurs avec les formes obtenues en utilisant les espaces de couleurs RGB et $L^*u^*v^*$.

3.6.1 Segmentation $L^*u^*v^*$ par voisinage

L'algorithme que nous proposons repose sur une technique utilisée dans la compression des images [23] qui consiste à considérer les vecteurs issus du voisinage des pixels pour la segmentation (*vectorisation*). Ainsi, pour une image $I(W, H)$, l'ensemble des vecteurs à quantifier n'est plus l'ensemble des vecteurs couleurs de l'image, soit $\{C_{L^*u^*v^*}(i,j), (i,j) \in [0, W-1] \times [0, H-1]\}$, mais l'ensemble des vecteurs

$$V(I) = \{V(i,j), (i,j) \in [w, W-w] \times [h, H-h]\}$$

avec

$$V(i,j) = (C_{L^*u^*v^*}(x,y) | (x,y) \in W(i,j))$$

et $W(i,j)$ est le voisinage du pixel (i,j) de dimension $w^2 \times h^2$. D'autre part, nous utilisons l'algorithme de groupement "Generalized LLoyd" [24] [25] qui est souvent utilisé dans le contexte de compression d'images et dont le code est fourni dans la bibliothèque *QccPack Library*[1].

Pour les images monochromes, nous ne retenons que la composante "niveau de gris" d'un pixel au lieu des trois composantes couleurs L^*, u^* et v^*. Pour l'implantation du système de recherche d'images, nous utilisons un voisinage de 4×4 $(w = h = 2)$; ainsi, les vecteurs $V(i,j)$ sont de dimension $16 \times 3 = 48$ pour les images couleurs et de dimension 16 pour les images monochromes. Le fait de considérer tout le voisinage d'un pixel pour le partitionnement réduit l'impact de la présence du bruit dans l'image sur la connectivité des contours, et ce à l'encontre des méthodes qui considèrent les pixels individuellement.

Une fois l'ensemble $V(I)$ partitionné en K régions, chaque pixel (i,j) est associé à une classe $C_{i,j} \in \{0, ..., K-1\}$ en fonction de la classe d'appartenance du vecteur correspondant $V(i,j)$. Les contours sont ensuite extraits par une simple détection

[1] http ://qccpack.sourceforge.net/

des bords de chaque classe de pixels. Pour les images couleurs, nous effectuons la segmentation pour deux valeurs de K, soit 2 et 3 régions. Pour les images à "niveaux de gris", nous effectuons une seule segmentation à 2 régions. Quand plusieurs segmentations sont effectuées, les images contours sont additionnées ce qui produit un effet de dédoublement des contours parce que la quantification est effectuée par rapport au vecteur voisinage au lieu du vecteur couleur du pixel (figure 3.15). Nous nous sommes limités à 2 régions dans l'extraction des contours pour ne pas encombrer l'ultime image des contours après addition des différents contours.

La figure 3.14 montre l'exemple de quelques images pour lesquelles les contours des formes ont été obtenus selon l'approche décrite ci-dessus ($K = 1$) en utilisant les espaces de couleurs HSV et $L^*u^*v^*$ de telle sorte à mettre l'emphase sur l'uniformité perceptuelle de l'espace $L^*u^*v^*$. La figure 3.15 montre l'exemple de quelques images avec les contours issus d'une segmentation en deux régions et l'addition des contours issus des segmentations en 2 et 3 régions. Dans ces exemples, on voit l'effet du dédoublement de contours qui nécessiterait éventuellement un traitement pour combiner les contours issus des différentes segmentations d'une façon plus homogène ; nous utilisons par la suite un descripteur qui tient compte de ce dédoublement.

3.6.2 Caractérisation des contours

Pour chaque pixel de l'image des contours, nous définissons une direction du contour en ce point (horizontale, verticale, première ou deuxième diagonale) en fonction de la disposition géométrique des pixels de contours de son voisinage. Par la suite, nous comptons le nombre de pixels qui sont associés à chaque direction pour produire un histogramme de 4 bâtonnets. Quand il se présente, le dédoublement des contours a pour effet d'amplifier certaines composantes de l'histogramme des directions. Par conséquent, et étant donnée une image exemple constituée de plusieurs régions, cette façon de faire permet de favoriser les images ayant plusieurs régions en premier lieu. En fin de compte, nous utilisons la distance euclidienne

Figure 3.15: Dans chaque ligne, une image exemple est présentée avec l'image de ses contours issus d'une segmentation en 2 régions et avec l'image contour résultant de l'addition des contours résultant des segmentations en 2 et 3 régions.

pour la comparaison des histogrammes lors de la phase de recherche d'images similaires à une image exemple donnée.

La figure 3.16 montre l'exemple d'une recherche d'images par contenu en utilisant la méthode de segmentation par voisinage 4×4 ; on peut remarquer visuellement la ressemblance de près de 14 images sur les 20 premières retournées à l'image exemple au sens des formes. Par contre, la position géométrique des formes n'est pas tenue en considération par cette méthode. Il est évident qu'il y a plusieurs aspects dont il faut tenir compte pour caractériser le plus fidèlement possible les formes d'une image, ce qui explique la diversité et parfois la difficulté des méthodes existantes.

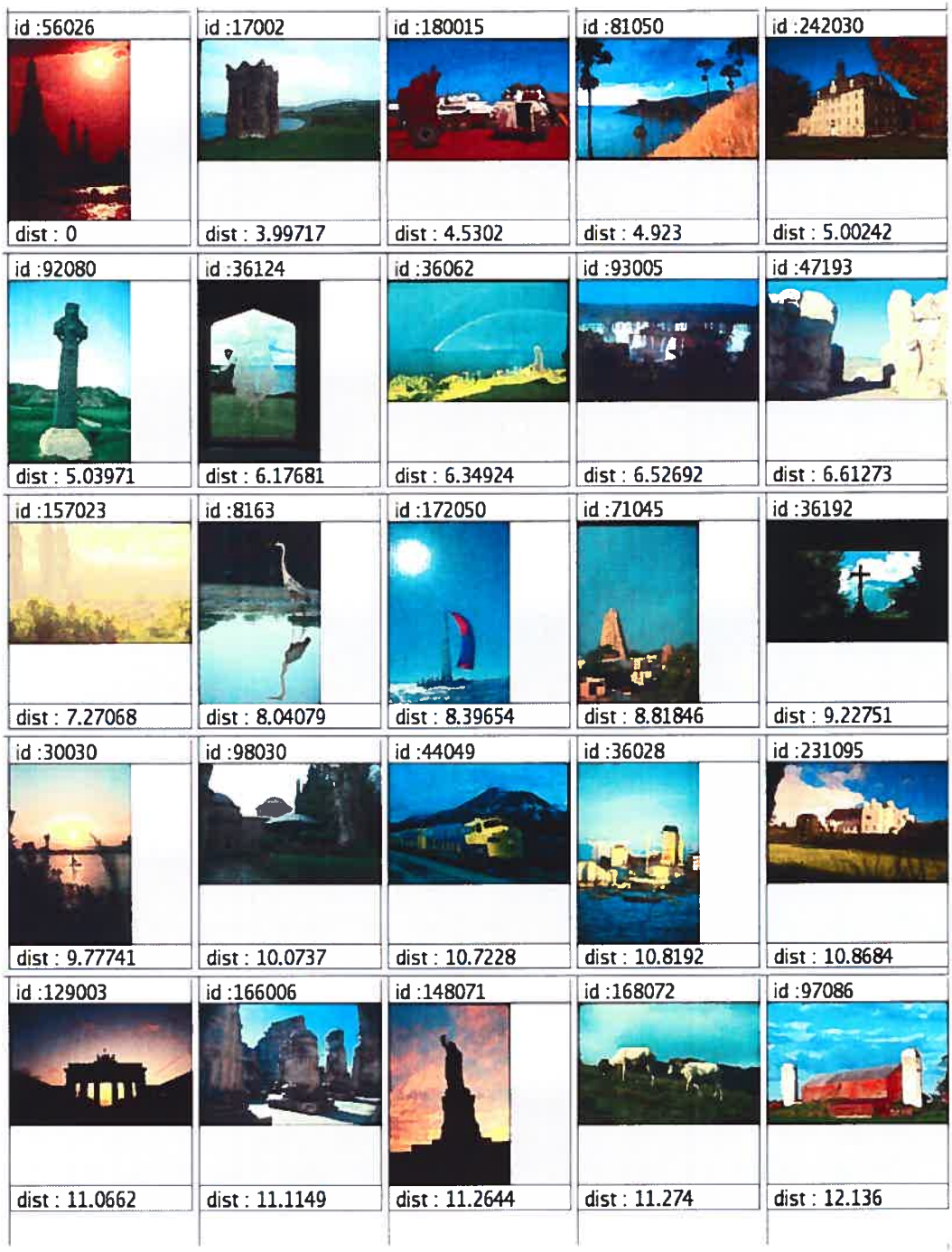


Figure 3.16: Recherche des image similaires à l'image exemple 56026 (coin haut gauche) en utilisant l'histogramme des 4 directions associées aux contours des formes obtenues par segmentation $L^*u^*v^*$ avec un voisinage de 4×4. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images.

CHAPITRE 4

ARTICLE : TOWARD CROSS-LANGUAGE AND CROSS-MEDIA IMAGE RETRIEVAL

Cet article [2] a été publié comme l'indique la référence bibliographique

C. Alvarez, A. Id-Oumohmed, M. Mignotte, and J.Y. Nie. Toward cross-language and cross-media image retrieval. In *Lecture notes in Computer science. Title : Multilingual information access for text, speech and images : 5th Workshop on Cross Language Evaluation Forum, CLEF 2004*, volume 3491, pages 676-688. Springer, September 2004.

Nous en avons fait une présentation orale dans l'atelier *ImageCLEF 2004* qui s'est tenu à l'université de Bath au Royaume Uni.

Préambule :

Dans cette rubrique, nous introduisons le principe et les notations concernant la procédure d'apprentissage automatique qui définit les caractéristiques visuelles des concepts. Nous présentons des résultats additionnels dans l'annexe III et nous en discutons dans la conclusion de ce mémoire.

Le processus d'apprentissage se fait par rapport à une base de données (collection). En considérant un concept (mot) donné w, nous dénotons par $\mathbf{I}_w$ l'ensemble de toutes les images de la collection qui sont annotées par le mot w. Comme chaque image de $\mathbf{I}_w$ est caractérisée par 3 vecteurs (descripteurs) relativement aux trois méthodes de caractéristiques visuelles qu'on cherche à modéliser (les contours, la texture et les formes), le concept w se trouve associé à 3 ensembles de vecteurs $D_{I_w}^{texture}$, $D_{I_w}^{edge}$ et $D_{I_w}^{shape}$. Nous procédons ensuite à un groupement (*clustering*) des

éléments de ces trois ensembles en K régions (*clusters*) tout en considérant plusieurs valeurs de K. Le centroïde de chaque région est un vecteur (descripteur virtuel) qui peut être utilisé pour classer les images selon la proximité de leurs descripteurs vis à vis de ce centroïde. Pour une valeur de K et une caractérisation visuelle (*class*) données, les centroïdes des K régions seront notés par $[D_{1,w}^{class}, ..., D_{k,w}^{class}]$ et seront utilisés comme des vecteurs prototypes. Pour chacun de ces vecteurs prototypes, nous introduisons ensuite la notion de topN ($N_{k,w}^{class}$) qui mesure le nombre d'images de $\mathbf{I}_w$ retrouvées parmi les N premières images de la recherche des images dont le descripteur est le plus similaire au vecteur prototype (descripteur virtuel) en question. Un simple seuillage sur $N_{k,w}^{class}$ permet alors de déterminer les caractéristiques visuelles les plus descriptives de w ainsi que les vecteurs prototypes associés. En fait, nous considérons la caractéristique visuelle qui maximise la mesure $N_{k,w}^{class}$ qui mesure, dans un certain sens, le degré d'indépendance des vecteurs descripteurs d'une région par rapport à tous les autres vecteurs descripteurs de la collection et ce relativement à la proximité de ces vecteurs par rapport au centroïde de la région. Ce principe est schématisé dans la figure 5.1 du chapitre 5.

Se basant sur la collection d'images *St. Andrews*, l'atelier *Workshop ImageCLEF2004 (Cross Language Evaluation Forum)* propose 25 requêtes textes (voir annexe III) pour lesquelles plusieurs groupes soumettent un classement d'images par requête. À cette étape, et dans le cadre de notre participation à cet atelier, nous nous sommes fixés les objectifs suivants :

- Définir une sémantique visuelle des concepts.
- Combiner les résultats de recherche issus de la recherche par texte (contribution du groupe RALI) et des résultats de la recherche par sémantique visuelle de telle sorte à étendre le sens informationnel des mots par leurs descriptions visuelles.

Abstract

This report describes the approach used in our participation of ImageCLEF. Our focus is on image retrieval using text, i.e., Cross-Media IR. To do this, we first determine the strong relationships between keywords and types of visual features. Then the subset of images retrieved by text retrieval is used as examples to match other images according to the most important types of features of the query words.

4.1 Introduction

The RALI group at University of Montreal has participated in several CLEF experiments on Cross-Language IR (CLIR). Our participation in this year's ImageCLEF experiments is to see how our approach can be extended to Cross-Media IR. Two research groups are involved in this task : one on image processing and the other on text retrieval. Our CLIR approach is similar to that used in our previous participation in CLEF, i.e., we use statistical translation models trained on parallel web pages for French to English translations. For the translation from other languages, we use bilingual dictionaries. Our focus is on image retrieval from text queries.

Different approaches have been used for image retrieval. 1) A user can submit a text query, and the system can search for images using image captions. 2) A user can submit an image query (using an example image - either selected from a database or drawn by the user). In this case, the system tries to determine the most similar images to the example image by comparing various visual features such as shape, texture, or color. 3) There is a third group of approaches which tries to assign some semantic meaning to images. This approach is often used to annotate images by concepts or by keywords [26]. Once images have been associated with different keywords, they can be retrieved for a textual query.

The above three approaches have their own advantages and weaknesses.

The first approach is indeed text retrieval. There is no particular image proces-

sing. The coverage of the retrieval is limited to images with captions.

The second approach does not require the images to be associated with captions. However, the user is required to provide an example image and a visual feature or a combination of some features to be used for image comparison. This is often difficult for a non-expert user.

The third approach, if successful, would allow us to automatically recognize the semantics of images, thus allow users to query images by keywords. However, the development up to now only allows us to annotate images according to some typical components or features. For example, according to a texture analysis, one can recognize a region of image as corresponding to a tiger because of the particular texture of tigers [14]. It is still impossible to recognize all the semantic meanings of images.

Some recent studies [27] have tried to automatically create associations between visual features and keywords. The basic idea is to use a set of annotated images as a set of learning examples, and to extract strong associations between annotation keywords and the visual features of the images. In our study, we initially tried to use a similar approach in ImageCLEF. That is, we wanted to extract strong relationships between the keywords in the captions and the visual features of the images. If such relationships could be created, then it would be possible to use them to retrieve non-annotated images by a textual query. In this case, the relationships play a role of translation between media. However, we discovered that this approach is extremely difficult in the context of ImageCLEF for several reasons :

1. The annotations (captions) of the images in the ImageCLEF corpus often contain keywords that are not strongly associated with particular visual features. They correspond to abstract concepts. Examples of such keywords are "Scotland", "north", and "tournament". Therefore, if we use the approach systematically, there will be many noisy relationships.
2. Even if there are some relationships between keywords and visual features, these relationships may be difficult to be extracted because there are a huge number of possible visual features. In fact, visual features are continuous.

Even if we use some discretization techniques, their number is still too high to be associated with some keywords. For example, for a set of images associated with the keyword "water", one would expect to extract strong relationships between the keyword and the color and texture features. However, "water" in images may only take up a small region of the image. There may be various other objects in the same images, making it difficult to automatically isolate the typical features for "water".

Due to these reasons, we take a more flexible approach. We also use the images with captions as a set of training examples, but we do not try to create relationships between keywords and particular visual features (such as a particular shade of blue for the word "water"). We only try to determine which type(s) of feature is (are) the most important for a keyword. For example, "water" may be associated with "texture" and "color". Only strong relationships are retained. During the retrieval process, a text query is first matched with a set of images using image captions. This is a text retrieval step. Then the retrieved images are used as examples to retrieve other images, which are similar according to the determined types of features associated with the query keywords. The whole process of our system is illustrated in figure 4.1.

In the following sections, we will first describe the image processing developed in this project. In particular, we will describe the way that relationships between keywords and visual features are extracted, as well as image retrieval with example images. In section 4.3, we will describe the CLIR approach used. In section 4.4, both approaches are combined to perform image retrieval. Section 4.5 will describe the experimental results and some conclusions.

Our approach is much less ambitious than that of [27], but it is more feasible in practice. In fact, in many cases, image captions contain abstract keywords that cannot be strongly associated with visual features, and even if they can, it is impossible to associate a single vector to a keyword. Our approach does not require determining such a single feature vector for a given keyword. It abandons the third approach mentioned earlier, but combines the first two families of approaches. The

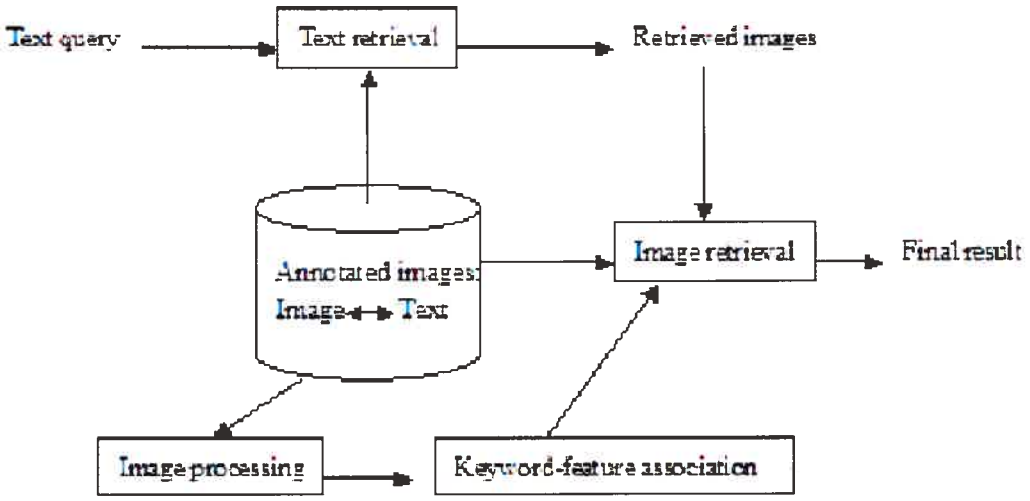


Figure 4.1: Workflow of image retrieval.

advantage of extracting keyword-feature associations is to avoid the burden of requiring the user to indicate the appropriate types of features to be used in image comparison.

4.2 Image processing-based learning procedure

The objective of the automatic image processing-based learning procedure that we propose in this section is twofold :

- First, this procedure aims at estimating the most discriminant type(s) of high-level visual features for each annotated keyword. In our application, we have considered the three fundamental visual characteristics ; namely, *texture* (including color information), *edge*, and *shape*. For example, the keyword "animal" could belong to the *shape* class since the measure using *shape* information will be the most discriminant to identify images with animals (but the more specific keywords "zebra" and "tiger" will more probably belong to the *edge* and *texture* classes respectively due to the characteristic coat of these animals).

  A discriminant measure, belonging to each of these classes of visual features

has then been defined. We have considered :

1. The mean and the standard deviation of the energy distribution in each of the sub-band of an Haar wavelet [20] decomposition of the image as discriminant measure of the *edge* class.
2. The coarseness measure proposed by Tamura *et al.* [8] as discriminant measure of the *texture* class.
3. The histogram of the edge orientation of the different shapes extracted from the input image (after a region-based segmentation) as discriminant measure of the *shape* class.

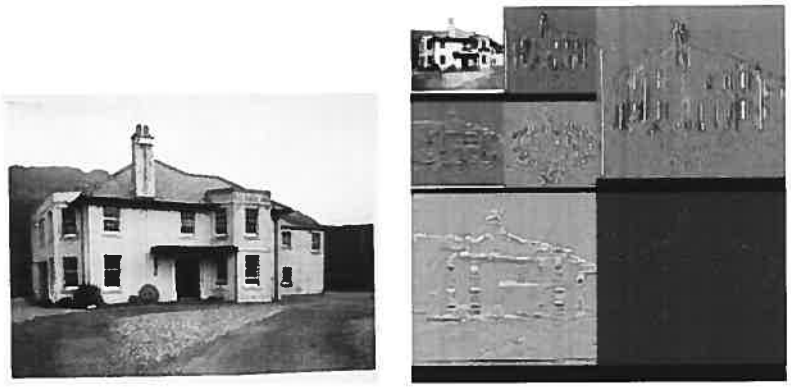
- The second objective is to identify a set of candidate images that are the most representative for each annotated keyword, in the sense of similarity distance combining one or several pre-estimated visual feature classes.

The type of high-level visual feature (along with its discriminant measure) and a set of candidate images along with its associated normalized similarity distance will be used with cross-language information, to refine the retrieval process.

4.2.1 Edge class and its measure

Wavelet-based measures have often been used in content-based image retrieval systems because of the appealing ability to describe the local texture and the distribution of the edges of a given image at multiple scales. In our application we use the Harr wavelet transform [20] for the luminance (i.e., grey-level) component of the image. There are several other wavelet transforms but the Haar wavelet transform has better localization properties and requires less computation compared to other wavelets (e.g., Daubechies' wavelets). The procedure of image decomposition into wavelets involves recursive numeric filtering. It is applied to the set of pixels of the digital image which is decomposed with a family of orthogonal basis functions obtained through translation and dilatation of a special function called *mother* wavelet. At each scale (or step) in the recursion, we obtain four sub-bands (or sets of wavelet coefficients), which we refer to as LL, LH, HL, and HH according to their

frequency characteristics (L : Low and H : High, see Figure 4.2). The LL sub-band is then decomposed into four sub-bands at the next scale decomposition. For each scale decomposition (three considered in our application), we compute the mean and the standard deviation of the energy distribution (i.e., the average and the square of each set of wavelet coefficients) in each of the sub-bands. This leads to a vector of 20 (i.e., $(2 \times 3 \times 3) + 2$) components or attributes which can be viewed as the descriptor (or the signature) of the *edge* information/characteristics of the image. For example, an image containing a zebra thus has high energy in the HL sub-band and low energy in the LH sub-band due to the vertical strips of the coat of this animal.



(a) (b)

Figure 4.2: a) The original image STAND03_1028/STAND03_26737_BIG.JPG of size 238×308. b) The Haar wavelets coefficients after the image a) is adjusted to size 256 × 256.

4.2.2 Texture class and its measure

Tamura *et al.* [8] have proposed to characterize image textures along the dimensions of contrast, directionality, coarseness, line-likeness, regularity, and roughness. These properties correspond to the human texture perception.

– Contrast is a scalar value related to the amount of local intensity variations present in an image and involves the standard deviation of the grey-level probability distribution.

– Directionality is a global texture property which is computed from the oriented edge histogram, obtained by an edge detector like the Sobel detector [28]. The directionality measures the sharpness of the peaks in this edge histogram.

– In this class of visual features, we have utilized only the coarseness property which yields a histogram of 6 bins, for the following reasons :

- The contrast is not very discriminant for textural retrieval,
- The edge information has been already treated in the wavelet and shape class,
- The line-likeness, regularity, and roughness properties are correlated to the coarseness, contrast, and directionality properties.

Coarseness refers to the size of the *texton*; i.e., the smallest unit of a texture. This measure thus depends on the resolution of the texture. With this measure, we can compute a histogram with 6 bins (i.e., a 6-component attribute vector) which will be used as the descriptor of the *texture* characteristics of a given image. The procedure for computing the coarseness histogram is outlined below.

1. At each pixel with coordinates (x, y) in the image, and for each value k (k taking its value in $\{1, 2, \ldots, 6\}$), we compute the average over its neighborhood of size $2^k \times 2^k$, i.e.,

$$A_k(x, y) = \sum_{i=(x-2^{k-1})}^{(x+2^{k-1}-1)} \sum_{j=(y-2^{k-1})}^{(y+2^{k-1}-1)} \frac{I(i, j)}{2^{2k}}$$

where $I(i, j)$ is the intensity pixel of the image at pixel (i, j).

2. At each pixel, and for the horizontal and vertical directions, we compute the differences between pairs of averages corresponding to pairs of non-overlapping neighborhoods just on opposite sides of the pixel. The horizontal and vertical differences are expressed as :

$$\begin{aligned} E_{k,horizontal} &= |A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y)| \\ E_{k,vertical} &= |A_k(y + 2^{k-1}, y) - A_k(y - 2^{k-1}, y)| \end{aligned}$$

3. At each pixel, the value of k that maximizes $E_k(i, j)$, in either direction (horizontal or vertical), is used to set the best size $S_{best}(i, j) = 2^k$. At this stage we can consider, as descriptor, the scalar measure of coarseness which is the average of S_{best} over the entire image, or consider, as in our application, the histogram (i.e., the empirical probability distribution) of S_{best} which is more precise for discrimination.

4.2.3 Shape class and its measure

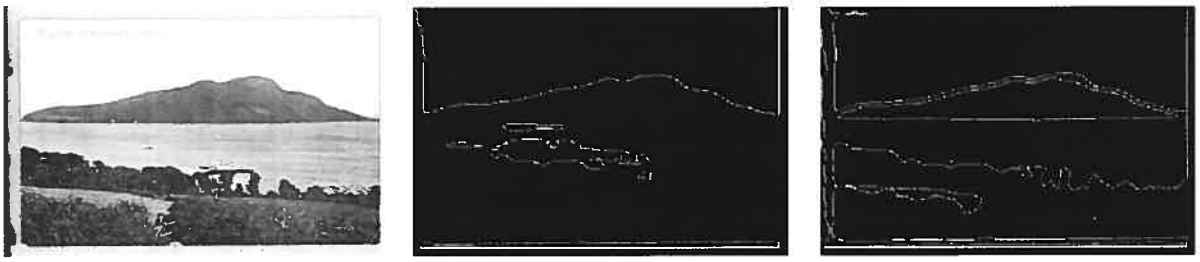
Description and interpretation of shapes contained in an input image remains a difficult task. Several methods use a contour detection of the images (such as the Canny or Sobel edge detectors) as a preliminary step in the shape extraction. But these methods remain dependent on certain parameters as thresholds (on the

magnitude of the image gradient).

In image compression, some approaches [23] use a vector quantization method on the set of vectors of dimension K^2 of grey-levels corresponding to $K \times K$ blocks extracted from the image. By using a clustering procedure into K classes, we can obtain an image with separate regions (a set of connected pixels belonging to a same class) from which we extract the contours of the different regions. These edges are connected and obtained without any parameter adjustment and the noise is taken into consideration in this procedure. Figure 4.3 shows an example of edge detection using three regions, i.e., three clusters in the vector quantization.

In our application, we use this strategy of edge detection and we use, as clustering procedure, the Generalized LLoyd [24] [25] (generally used in this context). In our implementation, we use the code provided from the QccPack Library[1].

For each edge pixel, we define a direction (horizontal, vertical, first or second diagonal) depending on the disposition of its neighboring edge pixels. For each direction we count the number of edge pixels associated with it, which yields a 4−bin histogram.



(a) (b) (c)

Figure 4.3: a) The original image STAND03_2093/STAND03_7363_BIG.JPG b) 4 × 4 pixel blocks and clustering result into 2 regions. c) 4 × 4 pixel blocks and clustering result into 3 regions.

[1]http ://qccpack.sourceforge.net/

4.2.4 The learning procedure

Given a training database, we first pre-compute and store *off-line* for each image its three descriptors (related to each of the three visual features). These sets of three vectors simplify the representation of each image, by giving maximal information about its content (according to each considered feature). We now define a similarity measure between two images given a visual feature class. This measure is simply the euclidean distance between two vectors.

The learning procedure which allows us to determine the type of high-level visual feature (and its measure) that is the most representative for each annotated keyword, is outlined below.

1. Let $\mathbf{I}_w$ be the set of all images I_w (each described by its three vectors or descriptors $[D_{I_w}^{texture}, D_{I_w}^{edge}, D_{I_w}^{shape}]$) in the training database that are annotated with the keyword w and $|\mathbf{I}_w|$, the number of images in $\mathbf{I}_w$.
2. For each CLASS { *Texture*, *Edge*, *Shape* }
   (a) We use a K-mean clustering procedure [28] (with a euclidean distance for the similarity measure) on the set of samples $D_{I_w}^{class}$.
   (b) This clustering allows us to approximate the distribution of the set of samples $D_{I_w}^{class}$ by K spherical distributions (with identical radius) and to give K prototype vectors $[D_{1,w}^{class}, ..., D_{k,w}^{class}]$ corresponding to the centers of these distributions. Several values of K are used to find the best clustering representation of $D_{I_w}^{class}$.
      i. For each PROTOTYPE VECTOR { $D_{1,w}^{class}, ..., D_{k,w}^{class}$ }
         - We search in the whole training database for the closest descriptors (or images) of $D_{k,w}^{class}$, according to the euclidean distance. Let $\mathbf{I}_{k,w}^{class}$ be this set of images.
         - We compute the number of the first top-level T samples of $\mathbf{I}_{k,w}^{class}$ also belonging to $\mathbf{I}_w$ (best results were obtained with $T = 10$). Let $N_{k,w}^{class}$ be this number.
3. We retain the CLASS(ES) and $\mathbf{I}_{k,w}^{class}$ for which we have $N_{k,w}^{class}$ above a given threshold ξ.
4. We normalize in $[0, 1]$ all the similarity distances of each sample of each selected set $\mathbf{I}_{k,w}^{class}$.
5. We combine the similarity distance measures of the selected sets $\mathbf{I}_{k,w}^{class}$, with an identical weighting factor, in order to find a final set of images i associated with each annotated keyword w. The similarity measures of these final images are then normalized, and the normalized similarity measure of an image i for the given word w is represented as $R_{cluster}(i, w)$ for retrieval, as described in section 4.4.

The first 24 images of the set of images associated to the word *garden* are shown in Figure 4.4. We can see that, even if most images are not annotated by the word *garden* (the word does not exist in any field of the text associated with the image), we can visually count about 9 images which are related to gardens from the 14 non-annotated images.

Figure 4.4: Results of learning procedure applied to the word "garden". Below each image, we can read its identifier key in the database and the score (similarity measure) obtained after normalization. The images which are not annotated by the word "garden" have their identifier key written in a gray box.

4.3 Cross-language text retrieval

4.3.1 Translation models

Two approaches are used for query translation, depending on the resources available for the different languages. For Spanish, Italian, German, Dutch, Swedish, and Finnish, FreeLang bilingual dictionaries [2] are used in a word-for-word translation approach. The foreign language words in the dictionaries are stemmed using Snowball stemmers[3], and the English words are left in their original form. The queries are also stemmed, and stop words are removed with a stoplist in the foreign language. The translated query consists of the set of all possible English word translations for each query term, each translated word having equal weight.

For French, a translation model trained on a web-aligned corpus is used [29]. The model associates a list of English words and their corresponding probabilities with a French word. As with the bilingual dictionaries, the French words are stemmed, and the English words are not. Word-for-word translation is done. For a given French root, all possible English translations are added to the translated query. The translation probabilities determine the weight of the word in the translated query. The term weights are represented implicitly by repeating a given translated word a number of times according to its translation probability. For French as well as for the other languages, the words in the translated query are stemmed using the Porter stemming algorithm.

This query translation approach was found to be optimal, using training data described in the following section. The parameters evaluated were :

- Whether to use a bilingual dictionary, or the translation model, for French.
- For a given query term, whether to pick just one translation from the dictionary or translation model, all translations, or in the case of the translation model, the first n probable translations.
- When to stem the English words : The English words could be stemmed

[2] http ://www.freelang.net
[3] http ://snowball.tartarus.org

in the dictionary, rather than after translation. This affects the number of times a particular word appears, and therefore its implicit weight, in the final translated query. Without stemming English words in the dictionary, multiple forms of a word may appear as a possible translation for a foreign language stem. After the translated query is stemmed, the English root appears several times.

4.3.2 CLIR process

For retrieval, the Okapi retrieval algorithm [30] is used, implemented by the Lemur Toolkit for Language Modeling and Information Retrieval. In particular, the BM 25 weighting function is used. The following parameters contribute to the relevance score of a document (an image annotation) for a query :

- BM 25 k1
- BM 25 b
- BM 25 k3
- FeedbackDocCount : the number of documents (image captions) to use for relevance feedback
- FeedbackTermCount : the number of terms to add to the expanded query
- qtf : the weight of the query terms added during relevance feedback.

The training data used to optimize each of these parameters, as well as the translation approaches described in section 4.3.1 was the TREC-6 AP89 document collection and 53 queries in English, French, Spanish, German, Italian, and Dutch. Since no training data was available for Finnish and Swedish, the average of the optimal values found for the other languages is used.

While the training collection, consisting of news articles about 200-400 words in length, is quite different from the test collection of image captions, the volume of the training data (163000 documents, 25 or 53 queries, depending on the language, and 9403 relevance assessments) is much greater than the training data provided from the image collection (5 queries, 167 relevance assessments).

Once the parameters for relevance feedback and the BM 25 weighting function are optimized with the training data, retrieval is performed on the test data, producing a list of images and their relevance scores, for each query. We annotate this image relevance score for a query, based on textual retrieval, as $R_{text}(i, q)$.

4.4 Combining text and images in image retrieval

4.4.1 The image relevance score based on clustering

The image analysis based on clustering, described in section 4.2.4, provides a list of relevant images i for a given word w, with a relevance score for each image, $R_{cluster}(i, w)$. The relevance score of an image for a query, based on clustering, is then a weighted sum of the relevance scores for that image for each query term :

$$R_{cluster}(i, q) = \sum_{w \in q} \lambda_w R_{cluster}(i, w) \tag{4.1}$$

In our approach, each word has the same weight, and the relevance score for the query is normalized, with $\lambda_w = \frac{1}{|q|}$, where $|q|$ is the number of words in the query.

4.4.2 Combining the five image relevance scores

We now have 5 lists of images for each query, with the following relevance scores :

- $R_{text}(i, q)$
- $R_{cluster}(i, q)$
- $R_{edge}(i, q)$: The similarity between the query image q and a collection image i, according to the wavelet measure described in section 4.2.1.
- $R_{texture}(i, q)$: The similarty according to the texture class measure from section 4.2.2.
- $R_{shape}(i, q)$: The similarity according to the shape class measure from section 4.2.3.

Each of these relevance scores contributes to a final relevance score as follows :

$$\begin{aligned} R(i,q) &= \lambda_{text} R_{text}(i,q) + \lambda_{cluster} R_{cluster}(i,q) \\ &+ \lambda_{edge} R_{edge}(i,q) + \lambda_{texture} R_{texture}(i,q) + \lambda_{shape} R_{shape}(i,q) \end{aligned} \quad (4.2)$$

The coefficients we chose for the contribution of each approach are as follows :

- $\lambda_{text} = 0.8$
- $\lambda_{cluster} = 0.1$
- $\lambda_{edge} = \lambda_{texture} = \lambda_{shape} = 0.033$

These values have been determined empirically using the training data provided in ImageCLEF.

4.4.3 Filtering the list of images based on location, photographer, and date

A final filtering is applied to the list of images for a given query. A "dictionary" of locations is extracted from the location field in the entire collection's annotations. Similarly, a "dictionary" of photographers is extracted. If a query contains a term in the location dictionary, then the location of a potential image, if it is known, must match this location. Otherwise, the image is removed from the list. The same approach is applied to the photographer. Similarly, if a date is specified in the query, then the date of the image, if it is known, must satisfy this date constraint.

4.5 Experimental results and conclusion

A preliminary analysis shows that our image retrieval works well. In particular, using the French queries, our system produced the best results among the participants. This may be related to two factors :

- The method of query translation used for these queries is reasonable. For French queries, we used a statistical translation model trained on parallel web pages. This translation model has produced good results in our previous CLIR

experiments.

- The method based on keyword-feature type association we used in these experiments may be effective. However, further analysis has to be carried out to confirm this.

For the experiments with other languages, our results are relatively good; they are often among the top results. However, the absolute MAP is lower than for the French queries.

CHAPITRE 5

ARTICLE : SEMANTIC-BASED CROSS-MEDIA IMAGE RETRIEVAL

Cet article [3] a été publié comme l'indique la référence bibliographique

A. Id-Oumohmed, M. Mignotte, and J.Y. Nie. Semantic-based cross-media image retrieval. In *ICAPR '05 : Proceedings of the Third International Conference on Advances in Pattern Recognition, ICAPR'05*, volume 3686(2), pages 414-423. Springer, August 2005.

Nous en avons fait une présentation orale dans la conférence *ICAPR 2005* qui s'est tenue à l'université de Bath au Royaume Uni.

Préambule :

Le présent travail est une extension du travail présenté dans le chapitre 4 que nous essayons d'améliorer. Nous utilisons les bases de données *Corel* et *St. Andrews* et nous appliquons le même principe de groupement avec les modifications suivantes :

- Nous étudions et comparons plus exhaustivement notre approche avec les méthodes existantes qui portent sur la sémantique des images et/ou des concepts.
- Pour chaque attribut visuel, le groupement de l'ensemble des vecteurs descripteurs est fait en fonction de la mesure de similarité utilisée avec cet attribut, et non en fonction de la distance euclidienne.
- Au lieu de caractériser un cluster uniquement par sa mesure *top20*, nous tenons compte de ses caractéristiques géométriques et nous introduisons la notion de pureté d'un cluster.

Abstract

In this paper, we propose a novel method for cross-media semantic-based information retrieval, which combines classical text-based and content-based image retrieval techniques. This semantic-based approach aims at determining the strong relationships between keywords (in the caption) and types of visual features associated with its typical images. These relationships are then used to retrieve images from a textual query. In particular, the association *keyword/visual feature* may allow us to retrieve non-annotated but similar images to those retrieved by a classical textual query. It can also be used for automatic image annotation. Our experiments on two different databases show that this approach is promising for cross-media retrieval.

5.1 Introduction

In general, a content-based image retrieval (CBIR) system tries to determine the most similar images to a given query image by using one or a combination of several low-level visual feature(s) such as color, texture, or shape. Depending on the content of each image, it is highly difficult to choose the appropriate feature(s) to use and eventually the manner to combine them. While users are mostly interested by the high-level (i.e., abstract) concepts present within an image query, the most similar images to this latter according to some low-level visual features can be non-relevant in the sense of semantics. This is known as the semantic gap. Usually, an annotated-based image retrieval (ABIR) system is based on a certain model representation of the concepts (words) associated to each image (document). Given a textual query, such a system scores and ranks images according to the importance of each word of text query to images. In this case, the search result is more limited to images that are really annotated by at least one of the words that form the textual query. In this work, we attempt to reach the same objective by finding non-annotated, but similar, images to those retrieved by a classical textual query.

To this end, and based on a training set of several images annotated by the same single word, we propose an unsupervised learning procedure which determines the most representative visual feature (visual semantic) of this word. Given an image query and the words of its caption, the user can choose the characterization of a certain word as a new search criterion.

5.1.1 Related Work

Organizing a set of images into clusters was used by Chen, Wang, and Krovetz [31] in their CBIR system (*CLUE*). Instead of sorting images by feature similarities with respect to a query image, the system retrieves image clusters. Especially, the user can navigate between queries according to each defined cluster (semantic *clue*). After the resemblance between the query image and target images are evaluated and sorted, a collection of target images that are "close" to the query image are selected as the neighborhood of the query image. The set of descriptor vectors of this collection is clustered into a dynamically-defined number of regions. This approach offers a different manner to present and visualize the most similar images to a given query image with an interesting interaction with the user.

Among the semantic-based approach, but only image content-based, different kinds of methods have already been investigated. We can cite, for example, the approach used in [32] which consists in grouping images into semantically meaningful categories. This system was applied on 6931 vacation photographs to obtain a classification such indoor/outdoor, city/landscape, etc. This classification is performed by a Bayesian classifier under the constraint that the test image does belong to one of the classes beforehand established by human subjects. We can also cite the approach used in [33] which clusters the image regions into 10 clusters (cloud, grass, etc.) and uses a probabilistic approach to define a semantic codebook of every cluster. Nevertheless, some recent studies [27] have tried to automatically create associations between visual features and keywords. The basic idea is to use a set of annotated images as a set of learning examples, and to extract strong associations between annotation keywords and the visual features of the images. In particular,

a segmentation algorithm, such as Blobword [34] or Normalized-cuts [35] is used to produce segmented regions, then for each region, feature information (color, texture, position, and shape) is computed. The set of computed features are clustered into regions which are called "blobs" and which define the vocabulary for the set of images. Finally images are annotated by the means of a cross-media relevance model.

Among the semantic-based approach trying to model the relationships between image features and associated text, we can cite the interesting work of Barnard et al. [36]. Their approach tries to provide a statistical joint distribution for associated words and features of each region of an image (image segments). After a training step which consists in estimating the parameters of a mixture of (Gaussian) distributions, a query search consists in computing the probability of each candidate image of emitting the query items. This method remains nevertheless highly dependent of the segmentation results and parameters associated to the segmentation (number of classes). Besides it is also highly dependent of the assumption that the cluster-conditional distribution of *index terms* (words or image segments) (i.e., the likelihood of this model) is unimodal and Gaussian. We can also cite the work of Wang et al. in [37] which try to address the challenging and -closely related problem- of automatic linguistic indexing of pictures. Association between an image and textual description of a concept is modeled via a likelihood given by a two-dimensional multi-resolution hidden markov model (HMM) whose parameters is learned in a training step. Once again, a query search consists in computing the likelihood of each candidate image for each pre-learned concept. As in applications, where this strategy is commonly used (e.g., handwritten text and speech recognition), this method remains highly dependent of the parameter estimation step of the HMM which is then used for the recognition step. In the case of 2D signal (i.e., image) this estimation may not efficiently model all the diversity of the different concepts and classes of images.

5.1.2 Our Approach

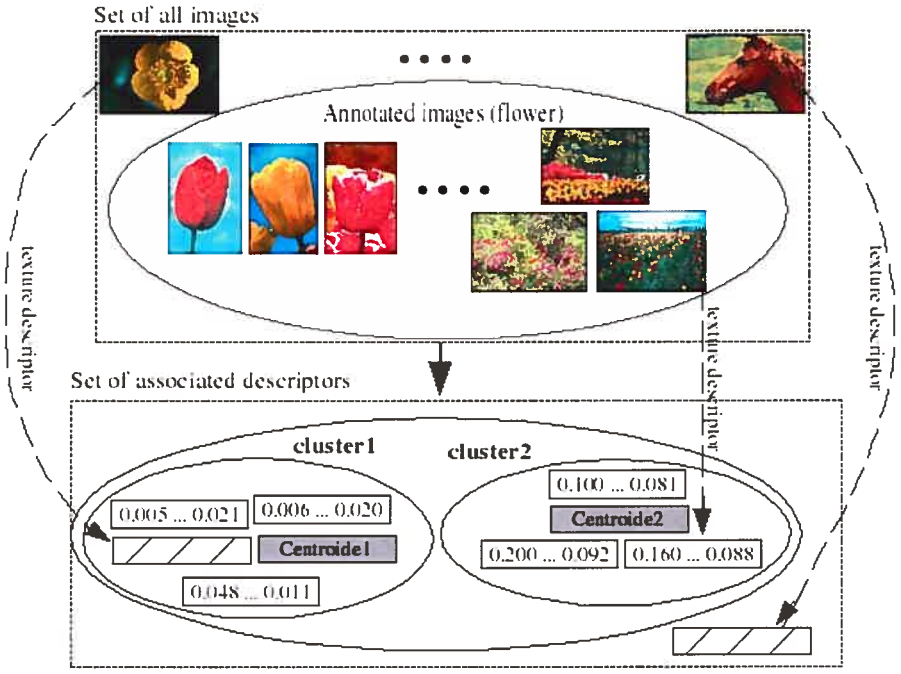


Figure 5.1: For each word, the training data is the set of corresponding annotated images which yield to three sets of descriptors (vectors) according to each high-level visual feature. Each set of descriptors is clustered in several regions. The figure shows an example of clustering in 2 regions for the set associated to the texture feature.

Instead of using pre-segmented image regions, described by multiple features (color, texture, shape, etc.), our approach uses the whole image content and tries to find out the most representative visual feature(s). Compared to [27], our approach has the advantage of not being dependent of a specific segmentation and can take into account relationships between regions (e.g., airplane-sky, animal-grass, boat-sea, etc.). Besides, some (key)words are best represented by one feature than by considering several features (e.g., sea with texture and cathedral with contours) which can introduce noise in the automatic retrieval model if they are not relevant. Our approach tries to identify such strong associations between words and visual features.

Our training data for each word is the set of images annotated by this word.

This dataset is exploited to obtain several sets of descriptor vectors according to the high-level visual features which will later be associated with the aforementioned (key)word. Each set of descriptors is then clustered by using several numbers of partitions (cf. Fig. 5.1 showing an example of clustering associated to a feature with respect to 2 partitions). This clustering allows our system to automatically estimate or capture the optimal number of partitions associated to the number of classes of images in the sense of their visual content (e.g., four types of mountains, six types of cars, etc.). Each cluster is then described by some statistical and spatial characterizations. We also describe the quality and the performance of a query based on the centroid feature (i.e., a model associated to a virtual image) of each cluster. According to some criteria on these descriptions, the keyword is associated with its most representative high-level visual feature, the number of regions used in the clustering, and the corresponding cluster centroid.

This unsupervised learning process also allows to propose a new image retrieval method by prompting the user to submit both a query image and a query keyword. To this end, the centroid of the cluster which contains the descriptor of the query image (and which can be viewed as the learned semantic concept of the keyword) can be exploited as a virtual image to perform the query. In particular, this visual semantic allows to retrieve similar images to the image query in the sense of the visual semantic of the given keyword.

5.1.3 Outline of the Paper

The reminder of the paper is organized as follows : In section 5.2, we will present the image processing techniques developed for this retrieval system; i.e., the considered visual features (texture, contours, and shape/color) as well as their corresponding similarity measures. In section 5.3, we will describe the way that relationships between keywords and visual features are extracted by the means of a learning procedure. In section 5.4, we will present some experimental results on the annotated *'St. Andrews University Library Photographic Collection'* and *Corel©* databases and we conclude.

5.2 Image Processing Retrieval Techniques

Edge, texture, and shape (including color) informations are important cues for pattern recognition and retrieval purposes in large image databases. In our approach, we have considered these cues as the three fundamental classes of visual characteristics, which we will call features in this paper. For each of the features, we consider a descriptor and an associated discriminant measure of similarity $S_{feature}$.

Edge Descriptor : Wavelet-based measures have often been used in content-based image retrieval (CBIR) systems because of their appealing ability to describe the local texture and the distribution of the edges of a given image at multiple scales. We use the Haar wavelet transform on the gray-level component of the image. The procedure of image decomposition into wavelets involves recursive numeric filtering. It is applied to the set of pixels of the digital image which is decomposed with a family of orthogonal basis functions obtained through translation and dilatation of a special function called *mother* wavelet. Three scales of transformations are considered here. For decomposition of each scale, we compute the mean and the standard deviation (μ_n and σ_n) of the energy distribution in each (of the $n = 10$) sub-band. This leads to an edge descriptor $\{\mu_{n=1}, \sigma_{n=1}, \ldots, \mu_{n=10}, \sigma_{n=10}\}$ of 20 components. For this descriptor, the similarity measure (S_{edge}) we use is the weighted-mean-variance distance.

Texture Descriptor : Tamura *et al.* [8] have proposed to characterize image texture along the dimensions of contrast, directionality, coarseness, line-likeness, regularity, and roughness. Coarseness refers to the average of the best representative sizes of the *textons* (i.e., texture resolution). To describe the texture feature, we use the coarseness and directionality histograms. We make two adjustments to the well known coarseness algorithm [8]. First, we set some predefined texture resolutions $\{2, 8, 14, 20, 26, 32, 38\}$ instead of $2^k \times 2^k$ with $k = 0, \ldots, 6$, then, we deal with homogeneous regions larger than the maximum of texture resolutions taken in account. After thresholding, the oriented edges are quantized into an 8-bin histogram. The similarity measure ($S_{texture}$) used is the Jeffrey divergence [18].

Shape and Color Descriptor : Extraction of shapes contained in an image remains a difficult task. Following [23], we first estimate a segmented image from which we extract the contours of different regions. The segmented image defines a set of connected pixels belonging to a same class. In this procedure, the noise is taken into consideration, edges are always connected, and the only parameter adjustment is the number of regions used in the segmentation procedure. Then, for each edge pixel, we define a direction (horizontal, vertical, first or second diagonal) depending on the disposition of its neighboring edge pixels and compute a 4-bin histogram. We complete this information by computing a 32-bin color histogram by using the HSV color space. The similarity measure S_{shape} used for this 36-bin histogram is the weighted-mean-variance distance.

5.3 Associating words with representative images and features

Given a set of training images with caption, we try to automatically determine one or several clusters of images representative for each word, together with the most discriminative feature(s), i.e. *texture*, *edge*, and *shape-color*. The principle is as follows : for each word, we try to group the images associated with it into several clusters (at different scales) according to each feature. Using one cluster as a visual query, if we can find many images annotated with the word among the most similar images according to the associated feature, then the cluster and the feature are considered to be characteristic for the word. In this way, each word can be associated with zero, one, or several clusters and features.

More precisely, let us define some notations : let $\mathbf{I}$ and $\mathbf{I}_w$ be respectively the set of all images in the training dataset and the set of all images that are annotated with the keyword w. $|.|$ designs the number of elements of a considered set : by applying the three visual features characterizations to $\mathbf{I}_w$, we obtain three sets of descriptors $\mathbf{D}_{I_w}^{texture}$, $\mathbf{D}_{I_w}^{edge}$ and $\mathbf{D}_{I_w}^{shape}$. We will use the notation $\mathbf{D}_{I_w}^{feature}$ to refer to each of these descriptors.

For a fixed number of regions (we consider 1, 2,..., 5 regions in our case), we use

the Generalized Loyd [25] algorithm to cluster each set $\mathbf{D}_{I_w}^{feature}$ in R partitions, thus, we obtain several ${}_c^R\mathbf{D}_{I_w}^{feature}$ clusters, where R denotes the number of partitions used in the clustering and c the c^{th} cluster in this R-clustering. The error-distance used in the clustering of $\mathbf{D}_{I_w}^{feature}$ is the similarity measure of the feature $S_{feature}$. For each value of R, this clustering allows us to approximate the distribution of the set of samples $\mathbf{D}_{I_w}^{feature}$ by R spherical distributions with identical radius. The centers (centroids) of these approximated spherical distributions are then considered as prototype vectors and are denoted by ${}_c^R P_{I_w}^{feature}$. Several values of R are used to take in account the fact that a given word may be associated to many image classes. For example, the word BOAT may be associated with images with small shape of boat in sea, or with a closer view of boat, and so on. For each cluster ${}_c^R\mathbf{D}_{I_w}^{feature}$, its associated centroid is used as a descriptor vector of a virtual image representative of the word. The virtual image will be used to query the whole training database $\mathbf{I}$ to get the closest descriptors (or images) according to the similarity measure associated to the feature $feature$. The training process is as follows :

- First, in order to associate each (key-)word w with the most discriminant class of visual characteristic $feature$, we use the following strategy : for each considered cluster ${}_c^R\mathbf{D}_{I_w}^{feature}$, we count the number of images annotated by the word w that are retrieved among the first X ($X = 20$ in our case) retrieved images for each $feature$. Let $topX^{feature}$ be this number. We count the sum of the $topX^{feature}$ resulting from the query by all corresponding prototype vectors. We then consider the class of visual feature for which this sum is maximal.

- Second, in order to define a set of prototype vectors associated to the pre-estimated class of visual feature, we adopt the following strategy : we characterize a given cluster ${}_c^R\mathbf{D}_{I_w}^{feature}$ by three measures : its proportion ρ within $\mathbf{I}_w$ (simply, $\rho = |{}_c^R\mathbf{D}_{I_w}^{feature}|/|\mathbf{I}_w|$), its standard deviation σ (computed according to the similarity measure of $feature$), and an empirical measure P which represents the number of images, not annotated by the word w, for which the distance between its descriptor vector and the prototype vector ${}_c^R P_{I_w}^{feature}$ is less than the pre-estimated standard deviation σ, namely

$$P = |\{I \notin \mathbf{I}_w \mid S_{feature}({}^{R}_{c}\mathbf{D}^{feature}_{\mathbf{I}_w}, {}^{R}_{c}P^{feature}_{\mathbf{I}_w}) < \sigma\}|/|I|$$

Once one feature or several weighted features are fixed, we choose representative prototype vectors regarding to P, their proportion and their standard deviation as follows : we use a first criterion to exclude prototype vectors for which $P > 0.05$ and $\rho < 0.05$. If there is no remaining prototype vector, then we ignore this criterion. The second criterion is to retain prototype vectors for which ρ/σ is greater than a threshold. The result of the training process is that a word may be associated with zero, one or several clusters of representative images, together with an associated feature to each cluster (i.e., vectors associated with high peak spherical distribution).

5.4 Experimental Results and Conclusion

The experimental results are based on the historical image database *'St. Andrews University Library Photographic Collection'* provided by *ImageCLEF 2004* [2]. This database contains 28133 images with caption. The caption text associated to each image contains around tens of (key)words. Our goal was to improve textual and multi-word queries by extending words to their associated visual features but our experiments in this context are extremely difficult due to the poor quality of the images of this database and also due to the presence of some (key)words used in the request with an abstract concept ("Scotland", "north", "tournament", etc.). For our experiments, we have also considered a set of 20000 images extracted from the Corel© database where each image is annotated by a few concrete and significant keywords. To test the relevance of our approach, we remove each word from the caption of 50% of associated images. We use these images as references and we try to see how our approach is able to retrieve these images with a query made of the removed word. We will emphasizes on two aspects of our results : the retrieved reference images and the non-annotated images retrieved but also related to the word in consideration.

Table 5.2 shows some words with the estimated weights for each class of visual features. Most associations have a significant meaning : animal is associated to shape and texture features, ocean is most described by shape (probably due to the presence of boats or due to the color component included with shape descriptor), tiger is described by texture and contours, zebra is associated to texture, etc. However, some words have almost the same weights for the three features, for example water, sky, garden and tree. This may be due to the high number of learning vectors. The word texture is strangely associated with shapes and contours. By choosing clusters with high value of P, we can guess to obtain more images that are not annotated by the word, but which are related to this word. In other hand, low values of this measure may yield to more images that are really annotated by the word ; this may be useful in the case of queries with multiple words, so to eventually improve the text retrieval result. Figure 5.3 shows three semantic query results for the words flower, canal, and grass : the algorithm described in 5.3 was used to produce these results. It shows also a query for word grass according to its second relevant feature. Even if the reference images were not retrieved successfully, we can see that most of images are related to the query word.

database	word	selected feature			Number of training vectors
		Feature 1	Feature 2	Feature 3	
	water	contours (74)	shape (65)	texture (61)	2550
	sky	contours (66)	texture (65)	shape (60)	2323
	tree	texture (85)	contours (79)	shape (72)	2242
C	people	contours (76)	texture (60)	shape (51)	1908
	grass	contours (35)	shape (28)	texture (27)	1061
	flower	shape (61)	contours (51)	texture (16)	934
O	wild	contours (17)	texture (15)	shape (15)	707
	bird	texture (24)	contours (12)	shape (9)	595
	plant	contours (13)	shape (10)	texture (8)	439
R	garden	texture (14)	contours (14)	shape (14)	301
	sunset	shape (19)	contours (15)	texture (8)	260
	ice	contours (8)	texture (6)	shape (5)	240
E	ocean	shape (44)	contours (26)	texture (15)	231
	animal	shape (11)	texture (7)	contours (3)	204
	ski	contours (4)	shape (1)	texture (0)	153
L	texture	shape (17)	contours (10)	texture (8)	126
	rural	contours (7)	texture (3)	shape (3)	124
	insect	contours (10)	shape (7)	texture (1)	123
	tiger	texture (14)	contours (10)	shape (9)	73
	zebra	texture (13)	contours (9)	shape (8)	26
	street	contours (119)	shape (101)	texture (96)	2348
St-	church	contours (57)	texture (48)	shape (48)	2721
	boat	texture (61)	shape (40)	contours (37)	1740
AND-	golfer	texture (18)	shape (14)	contours (10)	309
	canal	texture (3)	shape (3)	contours (2)	178
REW	swing	texture (8)	contours (1)	shape (1)	94

Figure 5.2: A list of concepts with their discriminative features ranked by the sum of $top20^{feature}$ over all the clusters of the feature (criterion used to choose the most discriminative feature or eventually to combine several features).

Corel word	top10	top20	top50	top100	ref10	ref20	ref50	ref100	vis20	vis40	vis60
flower (shape)	2	2	3	7	2	3	5	8	9	17	28
animal (shape)	1	1	2	3	0	0	0	0	6	9	16
birds (texture)	1	1	4	5	1	1	3	5	3	7	9
ice (contours)	0	0	0	1	0	0	0	1	0	0	0
grass (contours)	0	0	0	5	0	1	1	4	9	15	26

St-Andrew word	top10	top20	top50	top100	vis20	vis40	vis60
canal (texture)	0	1	1	2	10	17	29
street contours)	1	4	14	26	12	26	37
boat (texture)	1	4	8	10	4	9	12

Tableau 5.1: Some statistics about the top retrieved images for some words. topX is the number of images annotated by the word among the first X retrieved images. Identically, refX and visX are related respectively to reference images and visually accepted images (a subjective judgment).

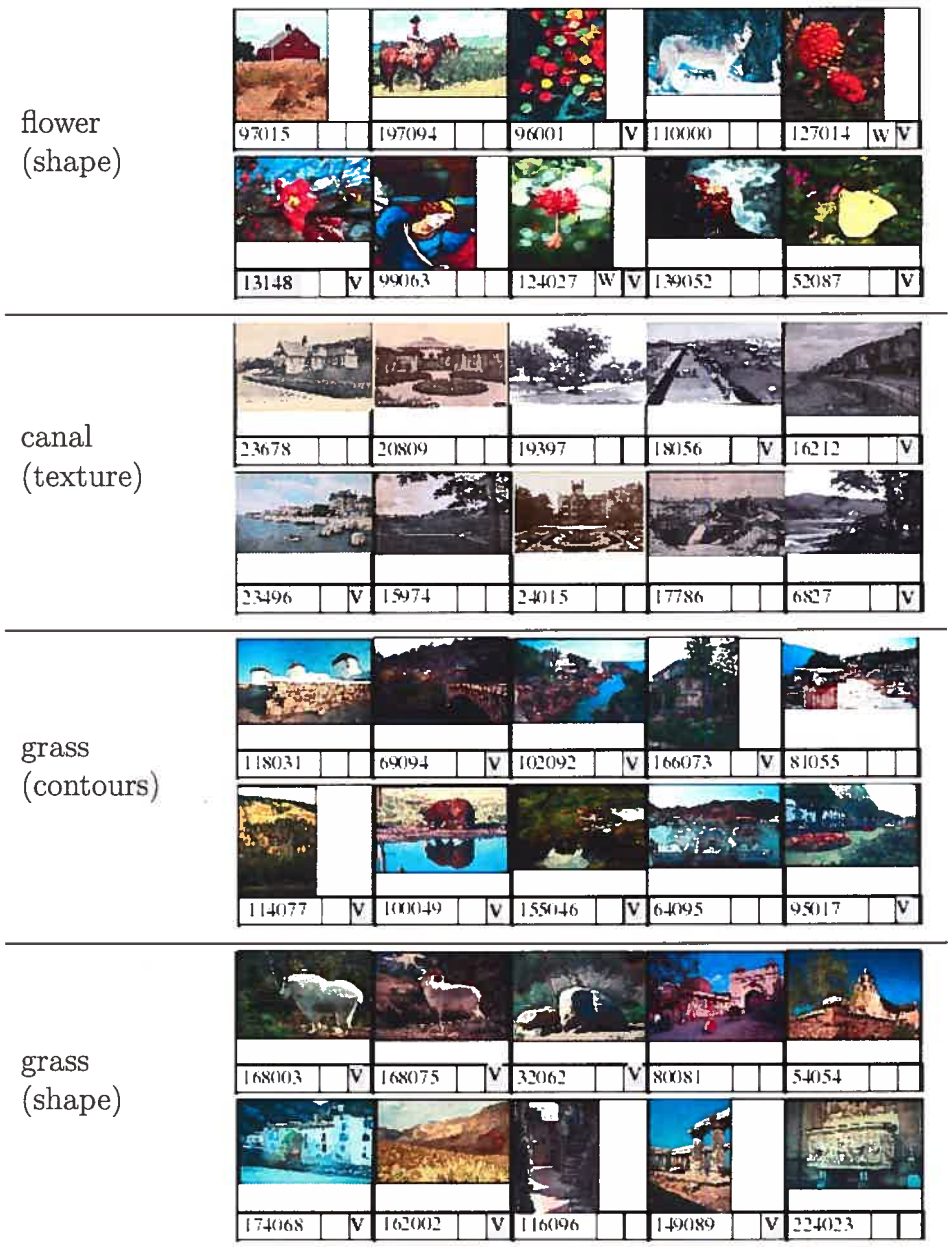


Figure 5.3: Semantic query results for concepts *flower* (shape), *canal* (texture), and *grass* (contours). The last query is made according to the best cluster of feature *shape*. The identification number is shown above each image. Annotated images are marked by a W box. Visually related images to the concept are marked by V box. Reference images have their identification number in a gray box.

CONCLUSION

Système de recherche d'images par contenu :

Bien que les résultats visuels concernant les méthodes de recherche d'images et les mesures de similarité associées soient satisfaisants, il n'en reste pas moins qu'il vaudrait mieux choisir quelques images de référence et comparer les résultats de recherche des méthodes que nous proposons avec d'autres méthodes préexistantes. En l'absence d'un support de test et de validation (*benchmark*) des systèmes de recherche d'images, ceci impliquerait la définition des images similaires à ces images références par un ou plusieurs sujets humains. Hormis le temps qui serait nécessaire pour cette opération (vue la taille des collections que nous utilisons), le processus d'identification des images similaires reste très subjectif ; par exemple, si on s'intéresse aux images similaires à une image de référence contenant une fleur rouge prise en gros plan, et qu'on veuille décider de la pertinence d'une image contenant une fleure jaune prise en gros plan et une autre image contenant un champ de petites fleurs, on peut considérer que ces images sont pertinentes selon leurs contenus (ce qu'un sujet humain ferait en supposant qu'il ne dispose pas d'informations sur les critères de recherche utilisés) comme on peut aussi les considérer comme non pertinentes vis-à-vis à une méthode de recherche d'images particulière comme les couleurs ou les contours. C'est dans ce sens que nous n'avons pas investi cette voie. En fait, il serait possible de consacrer des études approfondies à ce sujet.

Participation *ImageCLEF 2004* :

Nos premières expérimentations de recherche par sémantique visuelle sur certains concepts ayant un sens sémantique concret ont donné des résultats acceptables surtout au sens de retrouver des images visuellement attachées à un certain concept même si elles ne sont pas annotées avec ce concept. Pour chaque concept des requêtes de test (annexe III), nous disposons de trois listes d'images :

- Liste classée selon la pertinence du concept dans le texte annotatif des images (techniques de recherche d'information qui sont prises en charge par les

membres du groupe RALI).

- Liste des images selon leurs similitudes du point de vue des ondelettes (description multi-résolutionnelle de la distribution des contours et des textures) par rapport à l'image exemple de la requête. En fait, une image exemple a été fournie avec chaque requête pour encourager les participants au Workshop pour intégrer le contenu visuel de l'image avec la recherche par texte. D'autre part, nous utilisons les ondelettes parce que c'est un moyen assez robuste et reconnu pour caractériser des images monochromes.
- Liste classée des images selon leurs similitudes au vecteur prototype de la région (*cluster*) associé à la caractérisation visuelle la plus forte du concept.

La combinaison de ces trois listes est une tâche très difficile du fait que les mesures de classification sont de natures différentes et même après avoir normalisé toutes les listes à une moyenne nulle et une variance unitaire, les distributions de ces mesures entre 0 et 1 restent très hétérogènes. Nous avons essayé entre autres de ne normaliser que les 1000 premières images des listes issues de la recherche par contenu de l'image exemple et de l'apprentissage sémantique, en supposant que seules les premières images des 28133 images de la collection sont intéressantes. Enfin de compte, nous avons utilisé des facteurs de pondérations pour intégrer les différentes listes. Il faudrait travailler d'avantage sur ce point qui est très crucial pour la mise en valeur de l'apport de notre approche dans le raffinement et l'amélioration des recherches textuelles, surtout qu'une fois que l'apprentissage sémantique est effectué selon une langue d'annotation donnée, il devient indépendant de la langue dans laquelle sont exprimées les requêtes et peut servir de ce fait pour remplacer ou améliorer les requêtes exprimées dans d'autres langues.

Après la publication des résultats et des réponses escomptées aux requêtes du Workshop, nous avons effectué des expérimentations pour mesurer le degré d'apport de l'intégration des sémantiques visuelles des mots dans la recherche textuelle. Il s'est avéré qu'en général, l'ajout de ces requêtes dans le système de recherche textuel dégrade les performances du système sauf pour un nombre réduit de requêtes

où l'amélioration n'est pas très significative. À part les difficultés d'intégration des différentes listes citées ci-dessus, cette dégradation peut être due au fait que bon nombre des concepts des requêtes ont un sens abstrait (par exemple la première requête contient les mots abstraits *portrait, picture, minister, Thomas* et *Rodger* ainsi que le mot *church* qui signifie *cathédrale* ou *église*). Une suite logique des choses serait d'essayer de trouver une façon de distinguer les mots abstraits des mots concrets et de les inclure d'une manière adaptative et/ou supervisée dans les requêtes.

Dans la section III.3 de l'annexe III nous présentons une liste de mots avec les informations suivantes :

- *word* : le mot.
- *reqt* : la requête dans laquelle figure le mot *word*.
- *n_v* : le nombre d'images annotées par le mot *word* (taille de l'ensemble d'apprentissage).
- *feature* : la caractérisation visuelle.
- K : le nombre de régions utilisé dans le groupement ou *clustering*.
- i_K : l'indice de la région ou *cluster* (i_K $= 0, \ldots, K-1$).
- *top10* : le nombre d'images annotées par le mot *word* parmi les 10 premières images de la recherche relative au centroïde du cluster *i_K*.
- *top20* : le nombre d'images annotées par le mot *word* parmi les 20 premières images de la recherche relative au centroïde du cluster *i_K*.
- *q1000* : le nombre d'images de référence (images à retrouver pour une requête) parmi les 1000 premières images de la recherche relative au centroïde du cluster *i_K*.

Pour chaque mot, ces lignes d'information sont triées par ordre décroissant de la valeur de *top20*. Nous essayons à travers ces résultats de souligner quelques aspects décisionnels quant à la caractéristique visuelle à retenir pour un certain mot :

- Contrairement à ce qu'on peut croire que plus le cluster est petit (i.e. plus K est grand), plus la mesure *top20* tendrait à être plus grande, les meilleurs

clusters sont obtenus avec $K = 4$ ou $K = 3$.

- Des fois, il y plusieurs clusters (en égard à une même ou à plusieurs caractéristique(s) visuelle(s)) qui se classent en premier. La question se pose alors si on ne passe pas à côté du cluster le plus adapté pour caractériser le mot.
- La mesure de *top20* ne permet pas de distinguer si un mot a un sens concret ou abstrait (sensiblement les mêmes valeurs pour les mots *boat* et *north*).
- Le seuil choisi pour *topN* (*top10* ou *top20*) influencerait sur la décision à prendre pour certains mots comme *portrait*, en fait la corrélation entre les mesures *top10* et *top20* n'est pas idéale.
- Le nombre de clusters qui sont potentiellement intéressants varie d'un mot à un autre, et ce vraisemblablement en fonction de la taille de l'ensemble d'apprentissage *n_v*.

Annotation automatique des images :

L'approche que nous proposons peut être utilisée dans l'annotation automatique des images. Soit une image I à annoter (classifier) par rapport à une liste de concepts (classes) $W = \{w_1, \ldots, w_n\}$ associés aux caractérisations visuelles $F_W = \{(feature_1, descriptor_1), \ldots, (feature_n, descriptor_n)\}$. Comme l'image I est caractérisée par un certain nombre d'attributs visuels de type $(feature_I, descriptor_I)$, on pourrait s'intéresser à l'élément $(feature_i, descriptor_i)$ de F_W qui minimise la distance $D(descriptor_i, descriptor_I)$ telle que $descriptor_i$ et $descriptor_I$ soient liés au même attribut visuel et la distance D soit relative à la mesure de similarité de cet attribut visuel.

Pour un travail futur, il faudrait utiliser d'autres méthodes de recherche d'images qui soient plus efficaces et plus robustes, comme celles basées sur les caractérisations statistiques et probabilistiques des images. Il faudrait aussi essayer de trouver un modèle pour représenter les différentes techniques utilisées dans ce travail.

Annexe I

Interface et fonctionnalités du système de recherche d'images C-T-S IR

La figure I.1 montre un aperçu de l'interface d'interrogation des collections d'images et de mise en œuvre de notre approche de sémantique visuelle des mots que nous avons implémentée avec l'interface de programmation QT API sous Linux en utilisant C++ et MySQL. Les cadres en pointillés et les numéros associés ont été ajoutés pour des fins d'explication et de localisation des zones de la fenêtre de l'application. Nous baptisons cette dernière par : C-T-B IR (*"Content-Text-Semantic based Image Retrieval"*).

I.1 Zone 1

L'utilisateur commence par choisir l'une des deux collections d'images que nous utilisons.

I.2 Zone 2

L'utilisateur peut ouvrir une image exemple en la choisissant dans une fenêtre d'ouverture des fichiers ou en spécifiant son numéro d'identification.

I.3 Zone 3

L'interface affiche la liste des mots annotatifs des images de la collection choisie. L'utilisateur peut choisir un mot de cette liste pour effectuer une recherche d'images par texte. À titre d'indication, chaque mot de la liste déroulante est affiché avec le nombre d'occurrences où il apparaît dans la collection (c'est-à-dire le nombre d'images qui sont annotées avec ce mot) et son numéro d'ordre dans la liste. Le nombre d'occurrences d'un mot donne une idée sur la taille de l'ensemble d'apprentissage qui sert à définir ses caractéristiques visuelles.

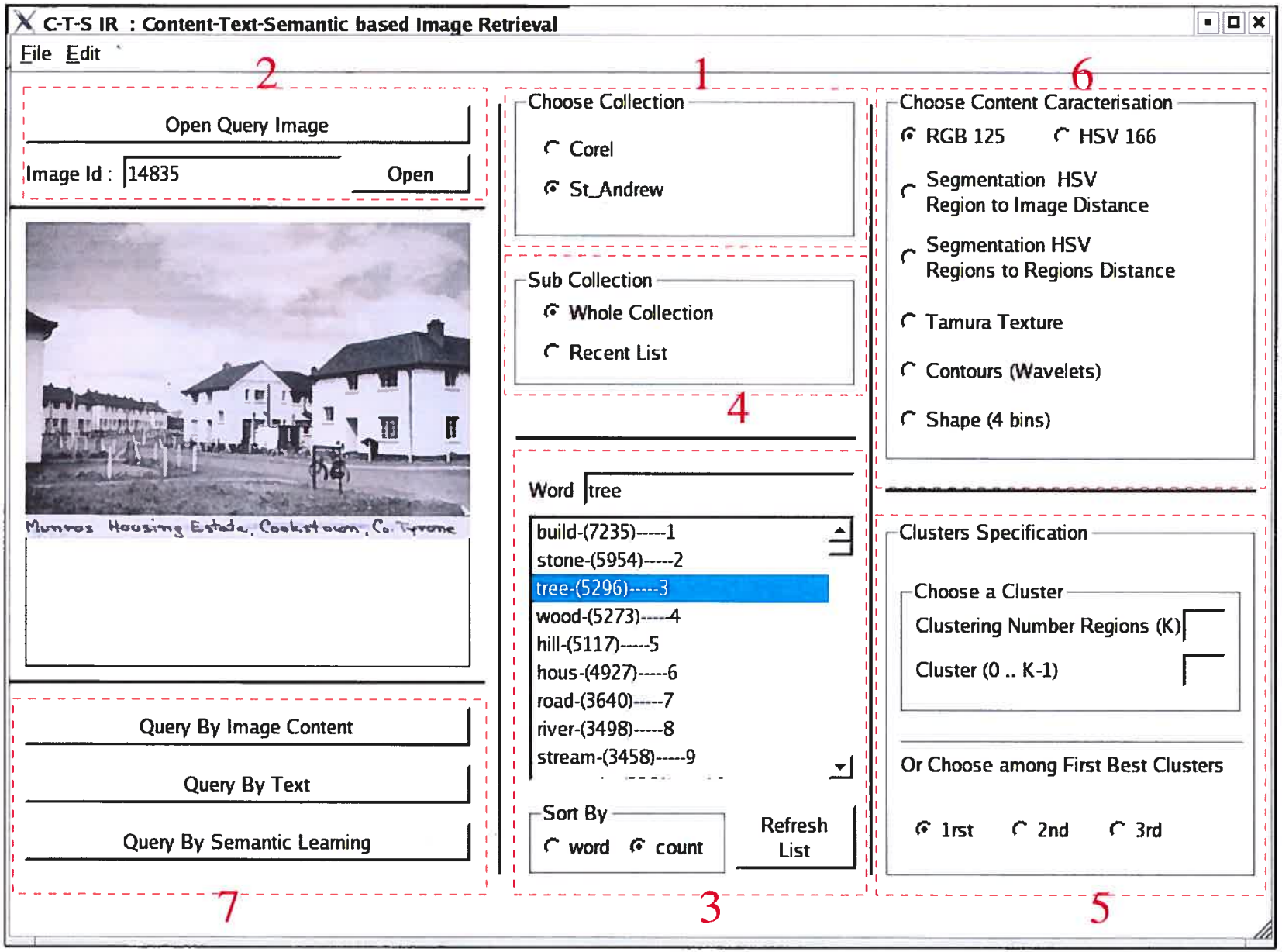


Figure I.1: Interface d'exploitation du système de recherche d'images C-T-S IR (*"Content-Text-Semantic based Image Retrieval"*).

I.4 Zone 4

Une fois qu'une recherche d'images par texte est effectuée, l'utilisateur peut choisir le sous-ensemble des images retournées pour effectuer la prochaine recherche au lieu de l'ensemble de toutes les images de la collection.

I.5 Zone 5

Pour effectuer une recherche d'images par sémantique visuelle pour un mot sélectionné, l'utilisateur peut choisir un groupement d'images (*"cluster"*) particulier, c'est à dire le nombre de régions utilisé dans le partitionnement des images en groupements et le numéro du groupement. Le programme propose aussi les trois meilleurs groupements au sens des critères définis dans [2] et [3].

I.6 Zone 6

Pour une recherche d'images par contenu, l'utilisateur choisit une méthode de caractérisation des images ou attribut. Chaque attribut étant associé avec une mesure de similarité appropriée.

I.7 Zone 7

Finalement, l'utilisateur choisit l'une des trois méthodes de recherche d'images :

- par contenu avec l'image exemple sélectionnée.
- par texte avec le mot sélectionné.
- par sémantique visuelle avec le mot sélectionné et le groupement spécifié.

Annexe II

Résultats comparatifs de 4 méthodes de recherche d'images par contenu couleur

II.1 Quantification régulière de l'espace de couleurs RGB en 125 sous-cubes en utilisant la distance euclidienne D_2

Figure II.1: Les 25 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la quantification en 125 régions de l'espace RGB. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images.

II.2 Quantification régulière de l'espace de couleurs HSV en 166 blocs en utilisant la distance euclidienne D_2

Figure II.2: Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la quantification en 166 régions de l'espace HSV. La distance de similarité utilisée est la distance euclidienne. La recherche est effectuée sur un ensemble de 20000 images.

(suite)

Figure II.3: Suite de la figure II.2.

II.3 Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des histogrammes des régions par la distance $D_{region\_a\_image}$

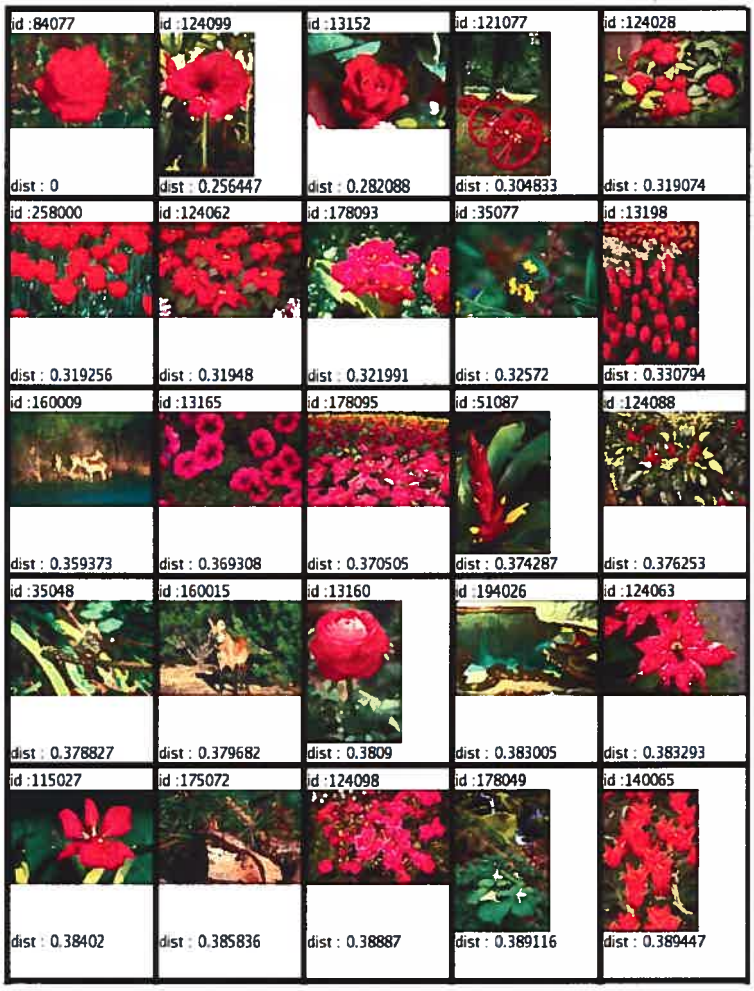


Figure II.4: Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la segmentation adaptative des couleurs HSV de l'image et la quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs. La distance de similarité utilisée est la distance $D_{region\_a\_image}$. La recherche est effectuée sur un ensemble de 20000 images.

(suite)

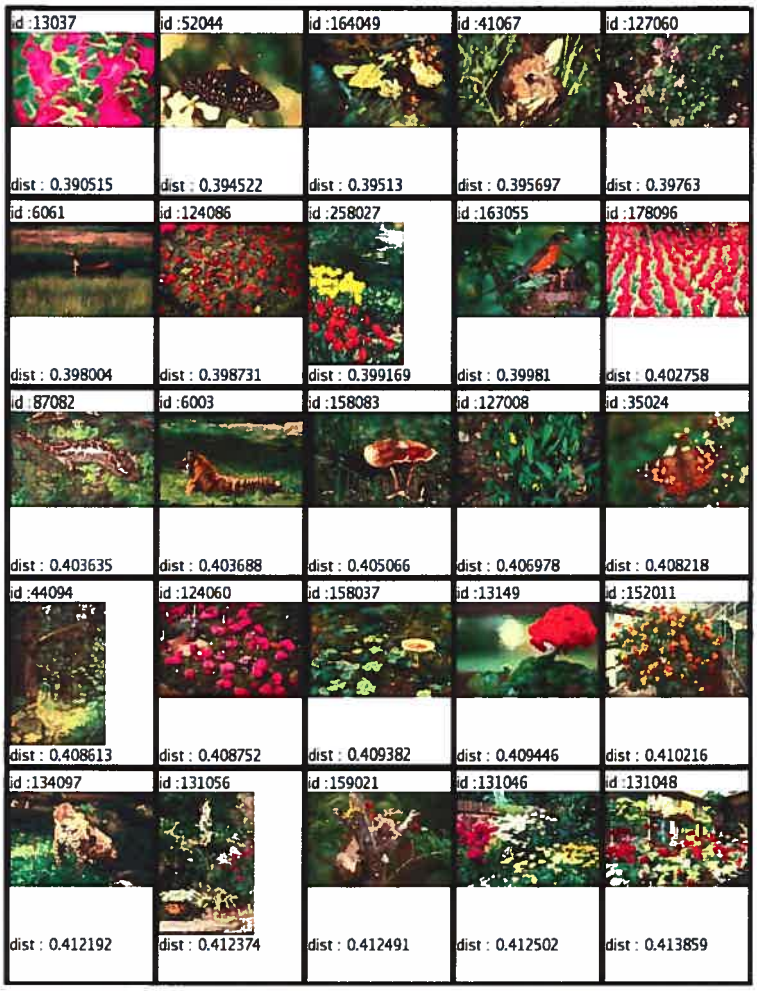


Figure II.5: Suite de la figure II.4.

II.4 Segmentation adaptative des couleurs HSV de l'image, quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs et comparaison des histogrammes des régions par la distance $D_{region\_a\_region}$

Figure II.6: Les 50 premières images similaires à l'image exemple 84077 (coin haut gauche) obtenues par la segmentation adaptative des couleurs HSV de l'image et la quantification régulière de l'espace de couleurs HSV de chaque région en 54 blocs. La distance de similarité utilisée est la distance $D_{region\_a\_region}$. La recherche est effectuée sur un ensemble de 20000 images.

(suite)

Figure II.7: Suite de la figure II.6.

Annexe III

Participation *ImageCLEF 2004*

III.1 Liste des 25 requêtes de référence

1. Portrait pictures of church ministers by Thomas Rodger.
2. Photos of Rome taken in April 1908.
3. Views of St. Andrews cathedral by John Fairweather.
4. Men in military uniform, George Middlemass Cowie.
5. Fishing vessels in Northern Ireland.
6. Views of scenery in British Columbia, Canada.
7. Exterior views of temples in Egypt.
8. College or university buildings, Cambridge.
9. Pictures of English lighthouses.
10. Busy street scenes in London.
11. Composite postcard views of Bute, Scotland.
12. Tay Bridge rail disaster, 1879.
13. The Open Championship golf tournament, St. Andrews 1939.
14. Elizabeth the Queen Mother visiting Crail Camp, 1954.
15. Bomb damage due to World War II.
16. Pictures of York Minster.
17. All views of North Street, St. Andrews.
18. Pictures of Edinburgh Castle taken before 1900.
19. People marching or parading.
20. River with a viaduct in background.
21. War memorials in the shape of a cross.
22. Pictures showing traditional Scottish dancers.
23. Photos of swans on a lake.
24. Golfers swinging their clubs.
25. Boats on a canal.

III.2 Classement des résultats soumis au *Workshop ImageCLEF2004*

	Group	Submission ID	MAP	%monolingual	Rank
Monolingual					
	Montreal	UmenTNFBTI	0.56	Na	5
Dutch					
	Montreal	UmnlTFBTI	0.4	68.27	7
Finnish					
	Montreal	UmfiTFBTI	0.23	40.02	1
French					
	Montreal	UmfrTFBTI	0.51	87.4	1
Italian					
	Montreal	UmitTFBTI	0.36	61.34	8
Spanish					
	Montreal	UmesRevTFBTI	0.45	76.82	7
Swedish					
	Montreal	UmsvTFBTI	0.34	57.98	1

Figure III.1: Aperçu du classement à l'issu de notre participation au *Workshop ImageCLEF2004* (source : Archives du *CLEF Forum2004* à l'adresse http ://clef.isti.cnr.it/).

III.3 Liste de quelques mots de la collection *St. Andrews* avec leurs sens sémantiques

word	feature	K	i_K	top10	topn20	q1000	n_v	reqt
boat	texture	4	2	4	7	3	1740	25
boat	texture	3	2	4	7	4	1740	25
boat	texture	2	2	4	7	4	1740	25
boat	texture	5	5	2	6	1	1740	25
boat	texture	5	2	5	6	4	1740	25
boat	texture	4	4	3	6	2	1740	25
boat	shape	5	1	5	5	1	1740	25
boat	texture	4	3	2	5	1	1740	25

word	feature	K	i_K	top10	top20	q1000	n_v	reqt
church	texture	4	2	2	9	0	2721	1
church	contours	5	3	5	7	0	2721	1
church	contours	5	5	3	6	0	2721	1
church	contours	5	1	4	6	0	2721	1
church	contours	2	1	2	5	0	2721	1
composit	contours	3	3	10	16	11	679	11
composit	texture	4	4	8	15	10	679	11
composit	texture	2	2	7	15	8	679	11
composit	contours	4	3	8	14	10	679	11
composit	contours	1	1	9	14	11	679	11
composit	shape	5	2	7	13	8	679	11
composit	shape	4	1	8	13	5	679	11
composit	contours	5	1	6	12	8	679	11
composit	texture	3	1	7	12	9	679	11
composit	contours	5	5	8	11	1	679	11
golfer	contours	5	4	4	7	20	309	24
north	texture	5	2	4	7	1	1676	17
north	texture	4	4	4	7	1	1676	17
north	contours	4	1	2	7	3	1676	17
north	contours	3	3	3	5	4	1676	17
north	shape	1	1	3	5	1	1676	17

word	feature	K	i_K	top10	top20	q1000	n_v	reqt
portrait	contours	4	4	4	10	11	669	1
portrait	contours	5	5	5	9	10	669	1
portrait	contours	5	2	4	8	9	669	1
portrait	texture	2	1	3	7	4	669	1
portrait	texture	5	1	3	6	7	669	1
portrait	contours	4	1	4	6	9	669	1
portrait	shape	3	1	1	6	8	669	1
rodger	shape	5	5	4	7	12	247	1
rodger	contours	5	5	4	6	5	247	1
rodger	contours	5	4	3	5	10	247	1
scene	texture	5	3	1	5	1	1273	10
scotland	texture	4	4	10	20	0	18874	11
scotland	texture	5	1	9	18	0	18874	11
scotland	shape	5	4	8	17	0	18874	11
scotland	contours	5	1	10	17	0	18874	11
scotland	texture	5	5	7	16	0	18874	11
scotland	contours	5	4	9	16	0	18874	11
scotland	texture	5	3	7	16	1	18874	11
scotland	texture	4	3	7	16	0	18874	11
scotland	shape	4	1	7	16	0	18874	11
scotland	texture	3	1	8	16	0	18874	11
scotland	contours	5	3	9	15	1	18874	11

scotland	texture	2	1	6	15	0	18874	11
scotland	contours	3	3	7	15	0	18874	11
scotland	contours	4	3	7	14	0	18874	11
scotland	contours	4	2	5	14	0	18874	11
scotland	shape	4	2	5	13	0	18874	11
scotland	shape	3	2	6	13	0	18874	11
scotland	contours	2	1	7	12	1	18874	11
scotland	contours	2	2	6	12	0	18874	11
scotland	texture	2	2	5	11	0	18874	11
scotland	texture	5	2	6	11	0	18874	11
scotland	contours	5	2	5	10	0	18874	11
scotland	contours	3	1	5	10	0	18874	11
street	contours	4	3	6	11	10	2348	17
street	texture	5	5	6	10	10	2348	17
street	texture	5	3	4	10	3	2348	17
street	contours	5	2	4	10	10	2348	17
street	contours	4	2	5	10	10	2348	17
street	contours	3	2	6	10	0	2348	17
street	contours	4	4	5	9	10	2348	17
street	texture	4	3	6	9	10	2348	17
street	contours	3	3	3	9	10	2348	17
street	contours	5	5	3	8	4	2348	17
street	contours	3	1	4	8	9	2348	17
street	contours	2	1	5	8	2	2348	17
street	texture	2	1	5	8	10	2348	17
thoma	shape	5	4	2	5	9	372	1

BIBLIOGRAPHIE

[1] J.R. Smith. *Integrated Spatial and Feature Image Systems : Retrieval Analysis and Compression.* PhD thesis, Graduate School of Arts and Sciences, Columbia University, New York, 1997.

[2] C. Alvarez, A. Id-Oumohmed, M. Mignotte, and J.-Y. Nie. Toward cross-language and cross-media image retrieval. In *Lecture notes in Computer science. Title : Multilingual information access for text, speech and images : 5th Workshop on Cross Language Evaluation Forum, CLEF 2004*, volume 3491, pages 676–688. Springer, September 2004.

[3] A. Id-Oumohmed, M. Mignotte, and J.-Y. Nie. Semantic-based cross-media image retrieval. In *ICAPR '05 : Proceedings of the Third International Conference on Advances in Pattern Recognition, ICAPR'05*, volume 3686(2), pages 414–423. Springer, August 2005.

[4] J.D. Foley, A. Van Dam, S.K. Feiner, and J.F. Hughes. *Introduction to Computer Graphics.* Addison Wesley, 1994.

[5] M.J. Swain and D.H. Ballard. Color indexing. In *International Journal on Computer Vision*, volume 7, pages 11–32, 1991.

[6] J.R. Smith. Query vector projection access method. In *IS&T/SPIE Symposium on Electronic Imaging : Science and Technoloy-Storage & Retrieval for Image and Video Databases VII*, volume 3656, pages 511–522, 1999.

[7] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project : Quering images by content using color, texture, and shape. In *Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases*, pages 173–187, February 1993.

[8] H. Tamura, S. Mori, and T. Yamawaki. Texture features corresponding to visual perception. *IEEE Transactions on Systems, Man, and Cybernetics*, 8 :460–473, 1978.

[9] V.E. Ogle and M. Stonebraker. Chabot : Retrieval from a relational database of images. *IEEE Computer*, 28(9) :40–48, Septembre 1995.

[10] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T.S. Huang. Supporting similarity queries in MARS. In *Proceedings of the 5th ACM International Multimedia Conference*, pages 403–413, Novembre 1997.

[11] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Prentice Hall, 2nd edition, 2002.

[12] C. Nastar, M. Mitschke, C. Meilhac, and N. Boujemaa. Surfimage : A flexible content-based image retrieval system. In *Proceedings of the ACM International Multimedia Conference*, pages 339–344, September 1998.

[13] N. Boujemaa, J. Fauqueur, M. Ferecatu, F. Fleuret, V. Gouet, B. Le Saux, and H. Sahbi. IKONA : Interactive generic and specific image retrieval. In *International workshop on Multimedia Content-Based Indexing and Retrieval*, 2001.

[14] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. Blobworld : A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems.* Springer, 1999.

[15] V. Castelli and L.D. Bergman. *Image Databases : Search and Retrieval of Digital Imagery.* John Wiley & Sons, Inc., 2002.

[16] S. Battiato, G. Gallo, and S. Nicotra. Perceptive visual texture classification and retrieval. In *12th International Conference on Image Analysis and Processing (ICIAP'03)*, pages 524–529, September 2003.

[17] J. Puzicha, T. Hofmann, and J.M. Buhmann. Non-parametric similarity measures for unsupervised texture segmentation and image retrieval. In *CVPR '97 : Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 267–272, Washington, DC, USA, 1997. IEEE Computer Society.

[18] J. Puzicha, J.M. Buhmann, Y. Rubner, and C. Tomasi. Empirical evaluation of dissimilarity measures for color and texture. In *ICCV '99 : Proceedings of*

the International Conference on Computer Vision-Volume 2, pages 1165–1173, Washington, DC, USA, 1999. IEEE Computer Society.

[19] A. Corboy, W. Tsang, D. Raicu, and J. Furst. Texture-based image retrieval for computerized tomography databases. In *The 18th IEEE International Symposium on Computer-Based Medical Systems(CBMS'05)*, June 2005.

[20] S.G. Mallat. A theory for multiresolution signal decomposition : The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11 :674–693, 1989.

[21] E.J. Stollnitz, T.D. DeRose, and D.H. Salesin. Wavelets for computer graphics : A primer, part 1. *IEEE Computer Graphics and Applications*, 15(3) :76–84, May 1995.

[22] E.J. Stollnitz, T.D. DeRose, and D.H. Salesin. Wavelets for computer graphics : A primer, part 2. *IEEE Computer Graphics and Applications*, 15(4) :75–85, July 1995.

[23] M. Goldberg, P. Boucher, and S. Shlien. Image compression using adaptive vector quantization. *IEEE Transactions on Communications*, COM-34 :180–187, 1986.

[24] S.P. Lloyd. Last square quantization in PCM's. *Bell Telephone Laboratories Paper*, 1957.

[25] Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, COM-28 :84–95, 1980.

[26] W. Liu, S. Dumais, Y. Sun, H. Zhang, M. Czerwinski, and B. Field. Semi-automatic image annotation, 2001.

[27] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *ACM SIGIR*, 2003.

[28] S. Banks. *Signal Processing Image Processing and Pattern Recognition.* Prentice Hall, 1990.

[29] J.-Y Nie and M. Simard. Using statistical translation models for bilingual ir. In *Revised Papers from the Second Workshop of the Cross-Language Evaluation*

Forum on Evaluation of Cross-Language Information Retrieval Systems, pages 137–150, 2001.

[30] S.E. Robertson, S. Walker, S. Jones, M.M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In *Proc. of the Third Text Retrieval Conference (TREC-3), NIST Special Publication 500-225*, 1995.

[31] Y. Chen, J.Z. Wang, and R. Krovetz. Content-based image retrieval by clustering. In *MIR '03 : Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, pages 193–200, New York, NY, USA, 2003. ACM Press.

[32] A. Vailaya, A.T. Figueiredo, A.K. Jain, and H.-J. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10 :117–130, 2001.

[33] W. Wang, Y. Song, and A. Zhang. Semantic-based image retrieval by region saliency. In *Int'l Conf. on Image and Video Retrieval*, July 2002.

[34] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik. Blobworld : A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*, pages 509–516. Springer, 1999.

[35] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 :888–905, 2000.

[36] K. Barnard, P. Duygulu, and D. Forsyth. Modeling the statistics of image features and associated text, 2002. (citeseer.csail.mit.edu/barnard02modeling.html).

[37] J. Li and J.Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9) :1075–1088, 2003.