



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS MOSTAGANEM

Faculté des Sciences Exactes et d'Informatique
Département de mathématiques et d'informatique
Filière Informatique

Thème :

Détection des cuts (coupures ou changement de plans) dans une image vidéo couleur : application d'un descripteur visuel

Etudiants :

- Benatia Kadda
- Chaouch Abdenour

Encadrant:

- Henni Karim Abdelkader

Deuxième Année Master Ingénierie des Systèmes d'Information
Année Universitaire 2013/ 2014

Remerciements

En premier lieu, nous tenons à remercier Dieu ALLAH qui nous a donné la force, la santé, le courage et la volonté pour achever ce mémoire et qui nous a facilité le chemin d'avoir nos besoins.

Nous souhaitons remercier nos encadrant Mr Henni Abdelkarim qui nous a encadré avec patience durant la réalisation de ce travail de fin d'études. Leurs conseils nous ont été bien utiles, notamment pour la rédaction de ce mémoire.

Un grand remerciement à l'ensemble des personnes qui ont participés de près ou de loin à l'élaboration de ce travail.

Nous exprimons également nos gratitudes aux membres du jury, qui nous ont honorés en acceptant de juger ce modeste travail.

Nous tenons à remercier l'ensemble du corps enseignants et les responsables de la Filière d'informatique de l'Université d'Abdelhamid ibn Badis.

Enfin, nous remercions nos famille pour leurs soutient durant nos parcours d'étude.

Dédicaces



Je dédie cette mémoire à ... ✍️

A ma très chère mère

Affable, honorable, aimable : Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

Ce travail est le fruit de tes sacrifices.

A mon très cher défunt Père

Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours eu pour vous pendant ta vie.

Vous resterez toujours mon idole.

Ce travail est le fruit de tes sacrifices.

A mon cher frère et mes chères sœurs.

Mon petit cher neveu Habib.

A mes proches amis.

Benatia Kadda

Dédicaces

Ce projet de fin d'étude fruit d'une association n'aurait pas touché son terme sans un travail assidu et acharné.

Je tiens à adresser mes dédicaces tout d'abord aux :

Mes chers parents

qui n'ont jamais cessé de m'aider et m'encourager durant tout mon parcours et qui m'ont permis d'être aujourd'hui ce que je suis,

A mon frère et ma sœur,

Mon partenaire BENATIA KADDA pour son dynamisme et sa patience,

Aux étudiants de la promotion de 2^{ème} année master

(Informatique ISI et SIG) 2013-2014.

Chaouch Abdenour

Résumé :

Actuellement les bases de données vidéo s'agrandissent d'une manière exponentielle, leurs indexations et recherches par le contenu reposent sur plusieurs techniques ; dans un premier temps il s'agit de segmenter la vidéo temporellement, ç-a-d il s'agit de récupérer les plans d'images constituant la vidéo, ceci se fait grâce aux algorithmes de détection des changements de plans (ou cuts en anglais), dans un deuxième temps définir une ou plusieurs image clef dans chaque plan puis constituer un résumé vidéo ; la phase de recherche est maintenant rendue possible ou faisant des requêtes sur ces résumé vidéos.

Mots clés : Indexation de vidéo par le contenu, cut, plan, image clef.

Sommaire

Résumé.....	IV
Sommaire	V
Liste des figures	VII
Introduction général.....	1

CHAPITRE I : Etat de l'art sur l'indexation et la recherche d'images et de vidéos par le contenu

1. Introduction.....	4
2. Indexation et recherche des images par le contenu	4
3. Quelques travaux concernant l'indexation et la recherche d'images par le contenu	4
4. Domaines d'applications.....	5
5. Les principaux systèmes existants.....	5
6. Indexation et la recherche de vidéos par le contenu	7
6.1. La nécessité d'avoir des systèmes de recherche pour les vidéos.....	7
7. Quelques travaux concernant l'indexation et la recherche de vidéo par le contenu	7
8. Domaines d'applications.....	8
9. Quelques systèmes de recherche de vidéos.....	9
10. Conclusion.....	10

CHAPITRE II : L'indexation des images et de vidéos par le contenu

1. Introduction.....	12
2. Architecture d'un système d'indexation et de recherche d'images par le contenu	12
3. Les types de recherches	13
4. Descripteurs.....	15
4.1. Les descripteurs de couleurs	15
4.1.1. L'histogramme de couleur	17
4.1.2. Descripteur par couleurs dominantes	19
4.2. Les descripteurs de textures	20
4.2.1. Matrice de cooccurrence	20
4.2.2. Les filtres de Gabor	21
4.2.3. Les ondelettes	21

4.3. Les descripteurs de formes	22
4.3.1. Descripteurs de Fourier	22
4.3.2. Les invariants de Hu.....	22
4.3.3. Moments de Zernike.....	22
4.3.4. Descripteur CSS.....	23
4.4. Les descripteurs de mouvements.....	23
4.4.1. Descripteur d'activité de mouvement	24
4.4.2. Descripteur de Trajectoire	24
4.4.3. Le descripteur de mouvement paramétrique	25
5. Mesures de similarité.....	25
6. Indexation et recherche de vidéo par le contenu	26
6.1. Découpage en images individuelles	28
6.2. Segmentation temporelle « plan »	28
6.2.1. La détection des cuts dans les vidéos	29
6.3. Sélection des images clefs	30
7. Conclusion	30

CHAPITRE III : Conception et mise en œuvre

1. Introduction.....	32
2. Les outils de développements de l'application	32
3. Notre approche	32
4. Algorithme	32
5. Interfaces.....	33
6. Résultat d'une expérimentation.....	38
6.1. Discussion.....	40
7. Conclusion	40
Conclusion générale.....	41
Bibliographie	42

Liste des figures

Figure 1 : Architecture d'un système d'indexation et de recherche d'images	13
Figure 2 : Recherche par mot clé sur Google	14
Figure 3 : Une requête par esquisse dans QBIC.....	14
Figure 4 : Recherche par l'exemple sur Google	15
Figure 5 : Espace de couleur RVB	16
Figure 6 : Espace de couleur HSV	16
Figure 7 : Les différents histogrammes d'une image couleur	18
Figure 8 : Exemple de texture	20
Figure 9 : Les mouvements courants de la caméra	23
Figure 10 : Phase d'indexation de vidéos.....	26
Figure 11 : Phase de recherche de vidéos	27
Figure 12 : Structure hiérarchique dans une vidéo.....	27
Figure 13 : Exemple d'un découpage de vidéo en images	28
Figure 14: Segmentation en plans de la vidéo	29
Figure 15 : Images clefs de la vidéo	30
Figure 16 : Interface d'accueil	33
Figure 17 : Ouvrir une vidéo.....	34
Figure 18 : Lecture de la vidéo.....	35
Figure 19 : Ensemble des images individuelles.....	35
Figure 20 : Fenêtre de calcul	36
Figure 21 : Détection des cuts	37
Figure 22 : Images clefs	38
Figure 23 : Diagramme des images clefs par rapport au seuil.....	39

Introduction générale :

Puisque l'informatique développe très rapidement et aussi l'utilisation des caméras numériques est si facile, il y a une explosion de la photographie numérique sur l'internet. Donc beaucoup d'immenses bases d'images numériques de milliers et même si dizaines de milliers d'images apparaissent. Les utilisateurs ont besoin d'outil qui peuvent gérer efficacement et accès rapidement ces bases d'images.

Donc par conséquence, un système d'indexation et de recherche d'images est nécessaire.

Ce domaine offre la possibilité aux utilisateurs d'accéder, d'interroger et d'exploiter directement ces bases d'images en utilisant leur contenu, ceci explique l'activité de recherche consacrée à ce domaine. La recherche d'images par le contenu consiste à caractériser le contenu visuel des images par des descripteurs visuels et d'effectuer des recherches par similarité visuelle à partir de ces descripteurs. Cette nouvelle technique permet de répondre à de nouveaux besoins dans le domaine de la recherche d'images.

Des techniques de recherche d'images ayant été développées à cet effet depuis la fin des années 70. Parmi ces approches on trouve la technique de recherche d'images à base de texte connue sous le nom « *Text-based Image Retrieval* » ou *TBIR* [1], qui est l'approche la plus ancienne utilisée jusqu'à nos jours.

aujourd'hui la recherche d'images migrent vers des techniques d'indexation automatique, ce thème de recherche est couvert par le mot *CBIR* pour "*Content Based Image Retrieval*" [2], ces techniques permettent de représenter chaque image de la base par un index automatique décrivant son contenu, ces index sont moins gourmands en mémoire, puis dans un second temps de faire la recherche à la base de ces index en introduisant une image exemple et le système retourne de la base toutes les images similaires à cette image.

Avec les progrès de l'informatique et des télécommunications les bases de vidéos sont apparues. Ces bases sont amenées à stocker des millions de vidéos, ce qui a engendré des nouveaux besoins dans plusieurs domaines.

Quelle que soit la base de données sur laquelle l'utilisateur travaille, ce dernier a toujours cherche d'avoir une information précise, la recherche d'une vidéo dans une base de plusieurs milliers de vidéos nécessite un temps considérable, pour accélérer le processus, le concept d'indexation est alors employé. La recherche de vidéos tente aujourd'hui d'extraire les informations directement des vidéos et d'une manière automatique. Ce type de travaux est couvert par le terme : "indexation de vidéos basée sur le contenu". Dans ce type de méthode

Introduction générale

deux étapes sont importantes : une première étape est la segmentation en groupes d'images appelés plans, la deuxième étape est de représenter chaque plan par une image clef. La collection de ces derniers forme le résumé vidéo.

L'intérêt du résumé de vidéo réside aussi dans la grande variété des applications qui en découlent comme la recherche, la classification et la navigation dans des bases de vidéos.

Dans le cadre de notre travail on a traité trois chapitres dont :

Chapitre 1 : on a étudié un état de l'art sur l'indexation et la recherche des images et de vidéos par le contenu dont les travaux et les systèmes existants.

Chapitre 2 : les processus et les techniques d'indexation et de recherche des images et de vidéos par le contenu, les descripteurs, les caractéristiques visuelles et les mesures de similarités.

Chapitre 3 : notre approche et on a créé une application qui détecte les cuts dans les images vidéo couleur.

CHAPITRE I :

*Etat de l'art sur l'indexation et la
recherche d'images et de vidéos par le
contenu.*

1. Introduction :

Dans ce chapitre, nous abordons un état de l'art sur l'indexation et la recherche des images et de vidéos par le contenu, les travaux existants, les domaines d'applications et les systèmes existants.

2. Indexation et recherche des images par le contenu :

L'indexation automatique des images par leur contenu permet la recherche d'images par le contenu (*CBIR*). Cette technique qui apparut dans les années 90 pallie les problèmes posés par la recherche textuelle qui a développé dans les années 70, et permet d'améliorer des applications interrompues et contribue aussi à faire émerger des nouvelles applications dans divers domaines.

La première utilisation du terme "recherche d'images par le contenu" dans la littérature a été faite par T. Kato en 1992 [3]. Il s'agissait de rechercher des images à l'aide des caractéristiques de bas niveau telles que la couleur, la texture et la forme. A partir de là, le terme a été utilisé pour décrire le processus de recherche d'images dans des bases de données multimédia.

3. Quelques travaux concernant l'indexation et la recherche d'images par le contenu :

Scott Cohen aborde le problème de la recherche d'images par le contenu avec une approche de distribution couleurs et de reconnaissance de contours d'objets en utilisant la distance *EMD* (*Earth Mover Distance*). Il a étudié également les requêtes partielles dans lesquelles on cherche des images par comparaison d'attributs des régions similaires. [4]

Jérôme Fournier a traité dans sa thèse « l'indexation d'image par le contenu et recherche interactive dans les bases généralistes », Cette thèse se focalise sur les aspects « image », à savoir sur le calcul et l'utilisation des caractéristiques primaires pour l'indexation et la recherche dans des grandes bases d'images généralistes, avec toutefois pour objectif la satisfaction des requêtes sémantiques. [5]

Julien Fauqueur étudie dans sa thèse « la recherche d'images par composition des catégories des régions issues de la segmentation couleurs des images ». [6]

Sid-Ahmed Berrani a utilisé les techniques d'indexation des données multidimensionnelles et une recherche approximative des plus proches voisins afin de déterminer les images les plus proches de l'image requête de l'utilisateur. [7]

Bedouhene Saïda a traité dans son mémoire « la Recherche d'images par le contenu », elle a présenté des méthodes d'analyse d'images permettant de décrire le contenu visuel. [8]

4. Domaines d'applications :

Les applications des systèmes de recherche d'images par le contenu sont variées. Citons les plus importantes :

- ✓ **Domaines judiciaires** : les services de police possèdent des grandes collections d'indices visuels (visages, empreintes,...etc.) exploitables par des systèmes de recherche d'images.
- ✓ **Domaines militaires** : bien que peu connues du grand public, sont sans doute les plus développées : reconnaissance d'engins ennemis via images radars, systèmes de guidage, identification des cibles via images satellites.
- ✓ **journalisme et la publicité** : les agences de journalisme ou de publicité maintiennent en effet des grosses bases d'images afin d'illustrer leurs articles ou supports publicitaires.
- ✓ **D'autres domaines incluent** : le diagnostic médical, les systèmes d'informations géographiques, la gestion d'œuvres d'art pour explorer et rechercher des peintures similaires, architecture pour retrouver des bâtiments ou des aménagements intérieurs.

5. Les principaux systèmes existants :

Jusqu'à maintenant, quelques systèmes d'indexation et de recherche d'images par le contenu ont été développés, nous citons quelques-uns dans ce qui suit :

QBIC :

Le logiciel QBIC (*Query By Image Content*) d'IBM, est le premier système commercial de recherche d'images par le contenu. Il supporte différents types des requêtes : par l'exemple, par croquis...etc. Les développeurs de ce logiciel rassemblaient un large panel des descripteurs d'images basé sur le contour, couleur et la texture. Le système utilise la moyenne pour caractériser la couleur dans les espaces *RGB*, *YIQ*, *Lab* et *Munsell*. La texture est représentée par une version améliorée des caractéristiques de *Tamura*. La forme est assurée

par des signatures classiques comme la circularité, la surface, l'excentricité et les moments invariants. La distance Euclidienne est utilisée pour comparer les images. [9]

Virage :

Virage est le moteur de recherche d'images développé par la société *Virage Inc.* Son objectif est de construire un environnement dédié à la recherche d'images, principalement composé de primitives. Similairement à QBIC, Virage propose des requêtes portant sur la couleur, la localisation des couleurs, la texture et la structure de l'image. L'interface de Virage offre la possibilité d'ajouter et de pondérer les différentes primitives ainsi que d'utiliser le bouclage de pertinence. L'avantage de Virage par rapport à QBIC est qu'il autorise une combinaison entre les différents modes de recherche.

L'utilisateur définit le poids qu'il veut attribuer à chaque mode. [10]

Photobook :

Photobook est un système d'indexation d'images développé par le *MIT Media Laboratory*. Ce système se base sur la couleur, la texture et la forme pour définir les signatures d'une image. La requête s'effectue classiquement par le choix d'une image candidate. La distance Euclidienne est utilisée pour mesurer la similarité. [11]

Blobworld :

Blobworld a pour but de retrouver, à partir d'une image requête, des régions similaires en couleur et texture, appelés *blobs*, dans les images de la base.

La couleur est décrite par un histogramme. Le contraste et l'anisotropie ont été utilisés pour représenter la texture. Pour caractériser la forme, Blobworld utilise la surface, l'orientation et l'excentricité. La requête consiste à sélectionner une région que l'utilisateur juge importante dans une image segmentée. La distance utilisée combine la distance quadratique (pour la couleur) et la distance euclidienne (pour la forme et la texture). [12]

SIMPLIcity :

visé à réduire le fossé sémantique dans les systèmes de recherche d'images par le contenu. Les images sont segmentées en régions, la caractérisation de chaque région est basée sur les ondelettes. Ainsi les images sont automatiquement triées suivant des critères sémantiques simples, ce qui permet ensuite d'accélérer et d'aider la recherche d'images similaires. [13]

IKONA :

IKONA est un système de recherche et de navigation interactive dans des grandes bases de données multimédia développées par l'équipe *IMEDIA* de *l'INRIA*, il est basé sur une architecture client/serveur. IKONA procède à l'extraction des descripteurs de la couleur à l'aide de l'histogramme de couleurs pondéré. L'extraction de la texture est effectuée en utilisant le spectre de Fourier. Pour la forme, il utilise l'histogramme d'orientation des contours. [14]

6. Indexation et la recherche de vidéos par le contenu :

6.1. La nécessité d'avoir des systèmes de recherche pour les vidéos :

Les systèmes de recherche de la vidéo basée sur le contenu visent à aider l'utilisateur à retrouver une séquence vidéo dans une base de données vidéo potentiellement grande. Trois cas principaux peuvent être distingués:

- L'utilisateur a en tête une séquence vidéo spécifique et sait qu'elle existe dans la base de données. L'utilisateur pourra décrire avec précision la séquence cible et peut voir dès le premier regard si une séquence suggérée correspond à son besoin.
- L'utilisateur a une vidéo spécifique en tête et ne sait pas si elle existe dans la base de données. Le problème ici est de fournir un outil précis de recherche de sorte que l'utilisateur puisse rapidement prendre une décision si la vidéo cible est dans La BD ou non.
- L'utilisateur recherche une vidéo simplement en se rapportant à son thème ou à un certain événement qui se produit dans la séquence vidéo.

Exemple : tous les buts marqués pendant un match de football.

7. Quelques travaux concernant l'indexation et la recherche de vidéos par le contenu :

Didier Zugaj et Patrick Bouthemy ont traité « la détection des volets pour l'indexation de vidéos par le contenu », ils ont intéressés au partitionnement temporel de vidéo, qui est une étape nécessaire dans l'analyse du contenu d'un document audiovisuel en vue de son indexation et de son exploitation. [15]

Itheri Yahiaoui a traité « la construction automatique de résumé vidéo », il a proposé une nouvelle approche de construction et d'évaluation automatique de résumé vidéo. Cette

approche est basée sur un principe qu'il a dénommé « Principe de Reconnaissance Maximale ». [16]

Fabrice Souvannavong a étudié « l'indexation et la recherche de plans vidéo par le contenu sémantique ». [17]

Mickaël Guironnet a traité dans sa thèse « les méthodes de résumé de vidéo à partir d'informations bas niveau, du mouvement de caméra ». [18]

Mohamed Amine Bourenane a développé « un outil d'indexation et la recherche de vidéos personnelles par le contenu », il a étudié les caractéristiques existantes pour avoir un meilleur résultat lors de la recherche et il a appliqué une nouvelle caractéristique à l'indexation de la vidéo, en occurrence l'histogramme de la couleur aux alentours des points de contour. [19]

Ben Abdelali, Krifa, Touil, Mtibaa et Bourenane ont traité « l'indexation de la vidéo par le contenu et les besoins en architectures matérielles dédiées », ils ont fait une étude du domaine de l'indexation de la vidéo dans le but d'examiner l'impact des nouvelles exigences dans ce domaine sur le choix des méthodes de conception et des solutions technologiques pouvant être utilisées pour la mise en œuvre d'architectures matérielles dédiées pour ce type d'application. [20]

Adrien Brilhault a étudié « l'indexation et la recherche par le contenu de documents vidéos », il a traité l'indexation sémantique de documents vidéos dont la reconnaissance des concepts de haut niveau par des méthodes de classification basées sur un apprentissage supervisé, il est focalisé sur la création des méthodes automatiques de sélection et de fusion des différentes sources d'informations dans la chaîne de traitement de la vidéo. [21]

8. Domaines d'applications :

✓ **La télédiffusion :**

En général, les chaînes de télévision gardent un archive complet de leur diffusion et avec le temps cet archive devient gigantesque. Les journalistes utilisent ce dernier pour préparer des nouveaux documentaires ou émissions en cherchant dans les archives les anciens reportages liés au thème voulu. Un système de recherche de la vidéo facilitera énormément leur tâche.

✓ **La sécurité :**

La vidéosurveillance est devenue un outil important utilisé pour assurer la sécurité dans les villes, aéroports, métros,...etc. Après un acte criminel, les services de sécurité ont souvent besoin de visionner les séquences de vidéosurveillance pour pouvoir identifier les suspects.

La rapidité de l'extraction des données.

✓ **La formation à distance :**

De nos jours, les avancées dans les télécommunications et les technologies multimédias ont permis la formation multimédia aux étudiants éloignés, via Internet. Par conséquent, la vidéo est largement utilisée dans la formation à distance.

✓ **L'archivage des vidéos :**

Ces dernières années, il y a eu l'apparition d'un nombre important des sites Internet qui offre le partage de la vidéo en ligne, par exemple le site *www.YouTube.com*.

✓ **Les applications médicales :**

La vidéo dans le domaine médical est essentiellement utilisée pour des usages pédagogiques, par exemple: pour enregistrer les opérations cliniques, pour présenter les détails des maladies, leurs symptômes, les comparaisons et les chirurgies. Un outil de recherche de vidéos permettrait à ceux qui enseignent la médecine d'économiser beaucoup de temps, et aidera à perfectionner ceux qui sont dans les endroits isolés.

9. Quelques systèmes de recherche de vidéos :

Nous allons donner quelques détails sur les systèmes ou prototypes existants destinés à la recherche de vidéos.

Informedia :

Le développement du projet Informedia a commencé en 1994, à l'université de Carnegie Mellon à Pittsburgh aux États-Unis. Le but du projet étant l'extraction du contenu visuel et audio de la vidéo pour permettre une recherche par le contenu. [22]

Fischlar :

Ce système a été développé au centre de traitement de vidéo numérique de l'université de Dublin en Irlande [23]. Ses développeurs se sont intéressés principalement au développement des techniques de navigation basées sur le contenu dans les bases de données de vidéos de

bulletins d'informations. Cela comprend la recherche, la navigation, le filtrage, la lecture et le résumé de la vidéo.

IBase :

Ce système a été développé au Collège Imperial de Londres [24]. L'objectif global du projet est de faciliter la navigation, la recherche et la visualisation de vidéos d'une base de données de bulletins d'informations. Pour ce faire, les développeurs de ce système ont extrait des caractéristiques de couleur et de texture pour faire de la recherche par l'exemple.

10. Conclusion :

Plusieurs travaux ont été réalisés et développés à propos de l'indexation et la recherche d'images et de vidéos par le contenu qui viennent d'être utilisés par plusieurs domaines comme on a vu précédemment, ces systèmes sont toujours en développement car les besoins des utilisateurs dans ce domaine cessent de s'accroître.

CHAPITRE II :

*L'indexation des images et des vidéos par
le contenu.*

1. Introduction :

Nous allons détailler dans ce chapitre les processus et les techniques d'indexation et de recherche des images et de vidéos par le contenu, les descripteurs, les caractéristiques visuels et les mesures de similarités.

2. Architecture d'un système d'indexation et de recherche d'images par le contenu :

Deux aspects indissociables coexistent dans ce système, l'indexation et la recherche. Le premier concerne le mode de représentation informatique des images et le second concerne l'utilisation de cette représentation dans le but de la recherche. L'architecture classique d'un système d'indexation et de recherche d'images par le contenu présentée en figure 1, se décompose en deux phases de traitement :

- **Une phase d'indexation** ou phase hors ligne : est une étape de caractérisation où les attributs sont automatiquement extraits à partir des images de la base, et stockés dans un vecteur numérique appelé *descripteur visuel*. Ensuite, ces caractéristiques sont stockées dans une base de données.
- **Une phase de recherche** ou phase en ligne : dans cette étape l'utilisateur introduit au système une image requête qui sera représentée par son vecteur caractéristique, il calcule la distance entre son vecteur caractéristique et celui des images de la base d'images en utilisant une mesure de similarité. Le résultat correspond à une liste d'images similaires à la requête de l'utilisateur.

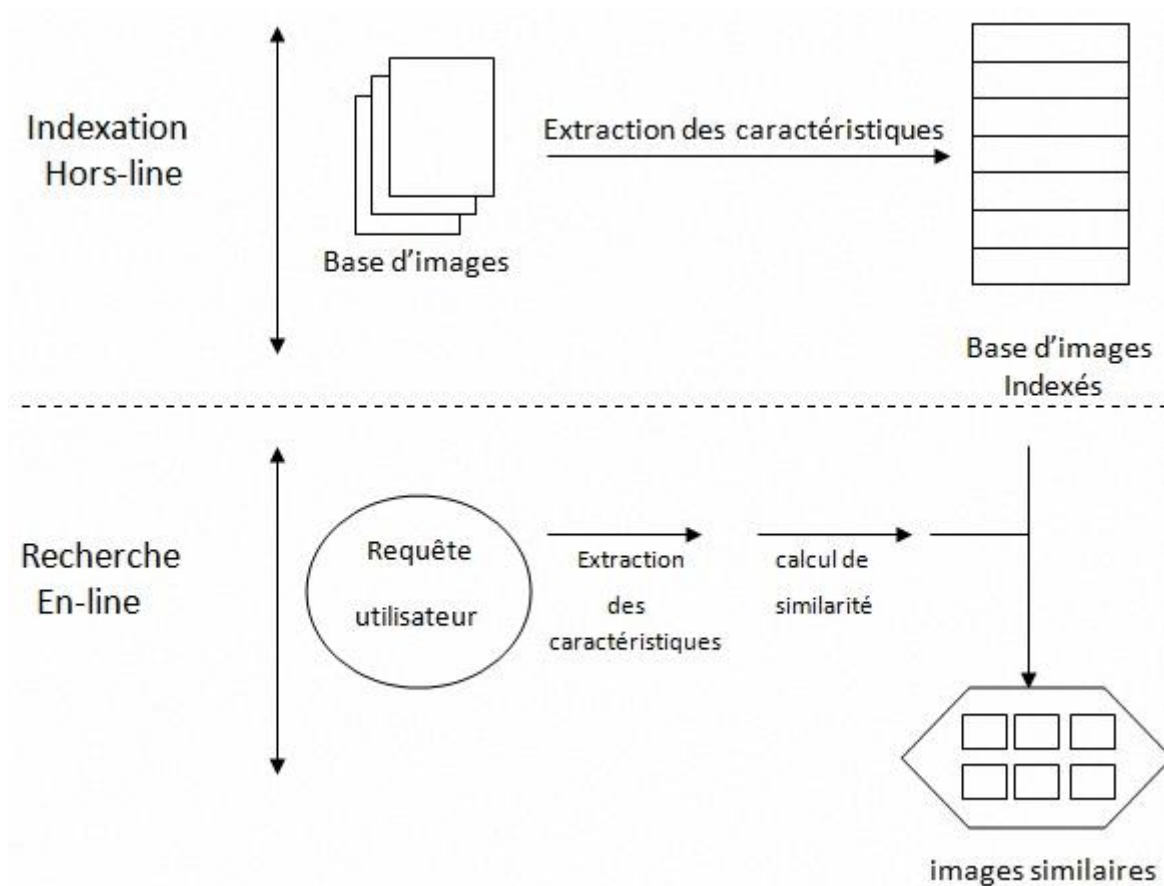


Figure 1 : Architecture d'un système d'indexation et de recherche d'images.

3. Les types de recherches :

Dans le domaine de la recherche d'images nous pouvons distinguer trois types de recherches :

➤ **Requête par mots clés :**

Le système extrait de la base les images dont leurs noms correspondent aux mots clés introduits au système. Afin de faire ce travail le système se base sur l'annotation textuelle manuelle des images.

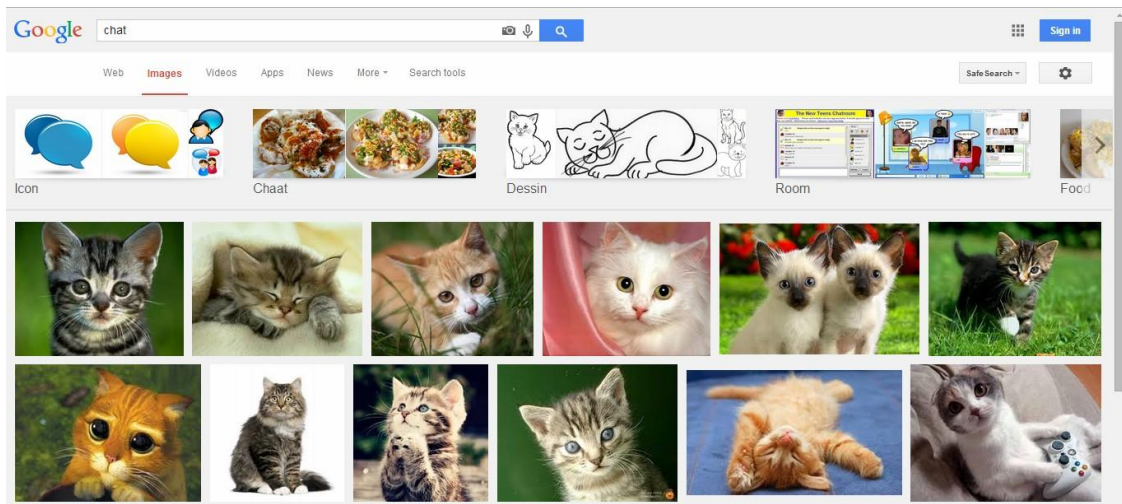


Figure 2 : Recherche par mot clé sur Google.

➤ **Requête par esquisse :**

Dans ce cas, le système fournit à l'utilisateur des outils lui permettant de constituer une esquisse (dessin) correspondant à ses besoins. L'esquisse fournie sera utilisée comme exemple pour la recherche. L'esquisse peut être une ébauche de forme ou contour d'une image entière ou une ébauche des couleurs ou textures des régions d'une image. L'utilisateur choisira, en fonction de la base d'images utilisée, de ses besoins et préférences, l'une ou l'autre de ces représentations. Cette technique présente l'inconvénient majeur qu'il est parfois difficile pour l'utilisateur de fournir une esquisse, malgré les outils qui lui sont fournis.

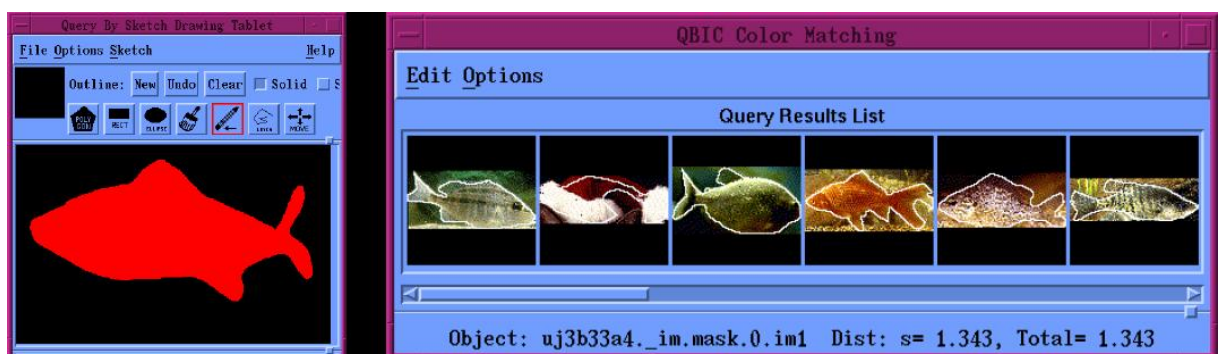


Figure 3 : Une requête par esquisse dans QBIC.

➤ **Requête par l'exemple :**

Pour les systèmes de recherche d'images à base d'exemples, l'utilisateur utilise une image (ou une partie d'image) qu'il considère similaire aux images qu'il recherche. Cette image est appelée image exemple ou requête.

Cette technique est simple et ne nécessite pas de connaissances approfondies pour manipuler le système.

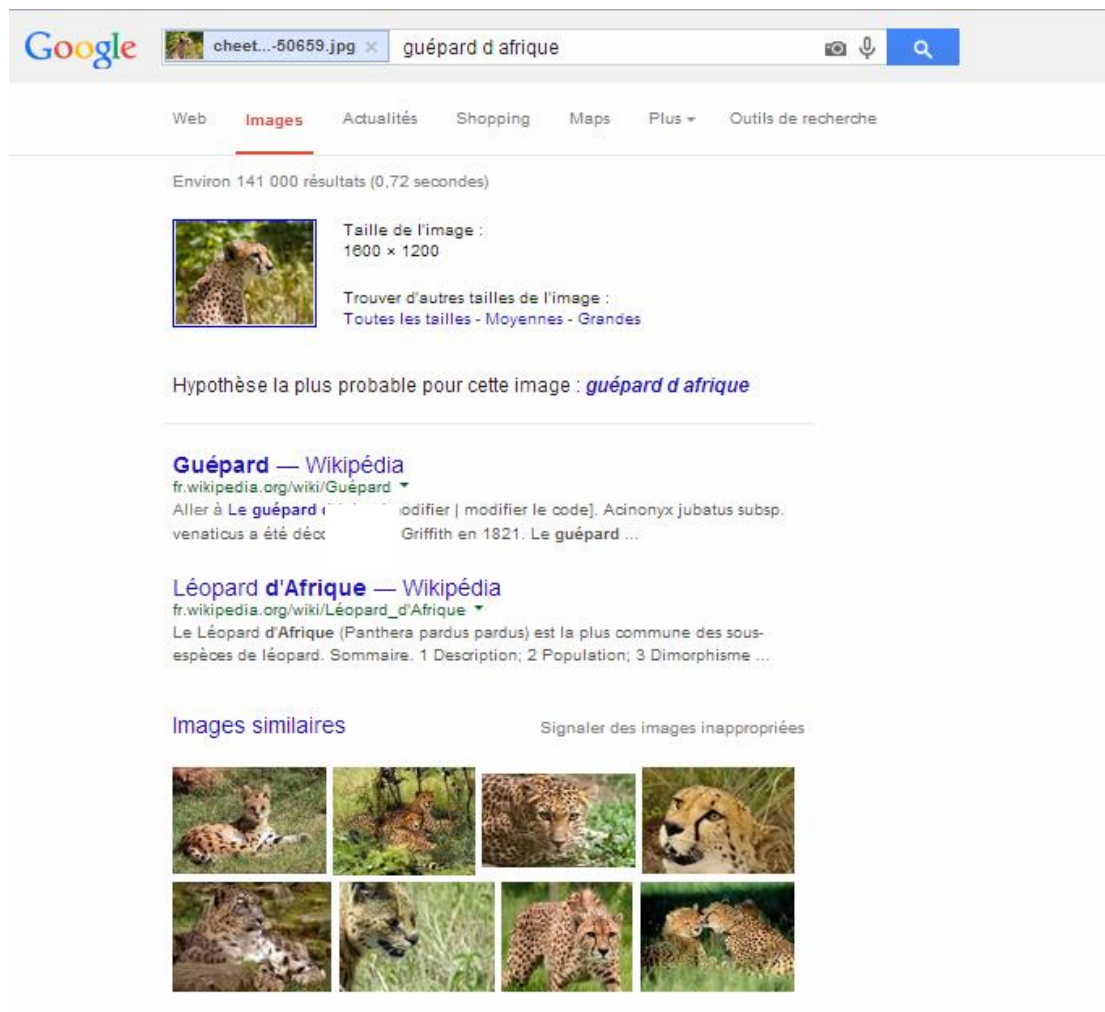


Figure 4 : Recherche par l'exemple sur Google.

4. Descripteurs :

Un descripteur est une description numérique du contenu. Il doit être compact et précis, de plus une distance ou une mesure de similarité doit permettre de comparer les descripteurs entre eux.

Nous nous intéressons dans ce manuscrit uniquement aux descripteurs visuels.

4.1. Les descripteurs de couleurs :

La couleur est le descripteur visuel le plus employé, certainement car c'est le plus perceptuel. Les grands problèmes soulevés par le choix de bons descripteurs de couleurs sont l'identification de l'espace de couleur le plus discriminant. Il existe plusieurs espaces

colorimétriques qui ont chacun certaines caractéristiques intéressantes, nous citons ci-dessous les espaces les plus utilisés:

➤ **L'espace RVB :**

C'est l'espace de couleur de base. La représentation des couleurs dans cet espace donne un cube appelé « *cube de Maxwell* ». Le système de couleur RGB est un système de couleur additif, c.-à-d. que les couleurs sont obtenues par le mélange des trois couleurs de base qui sont le rouge, le bleu et le vert. [25]

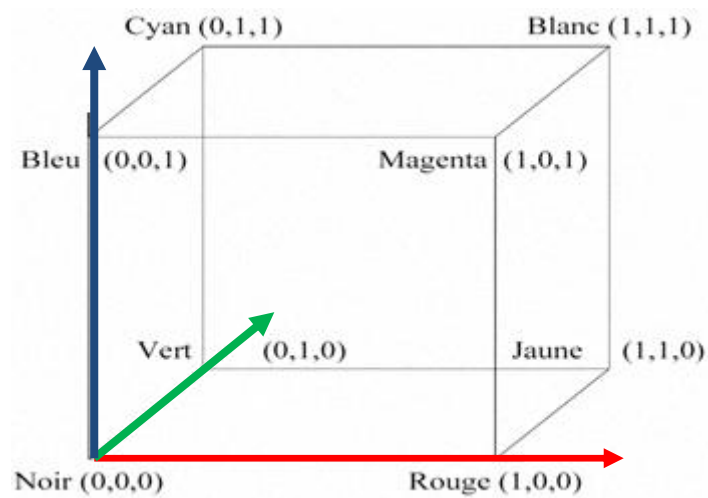


Figure 5 : Espace de couleur RVB.

➤ **L'espace HSV :**

C'est un espace dérivé de l'espace RGB, Il a été formellement décrit en 1978 par Alvy Ray Smith [26]. Les couleurs dans cet espace sont représentées selon des notions de teinte (*Hue*), de pureté (*Saturation*) et de luminosité (*Value*). La teinte caractérise la couleur. Sa valeur varie entre 0 et 360°. La saturation caractérise la pureté de la couleur, sa valeur varie entre 0 et 100 %. La luminosité correspond à la brillance perçue de la couleur. Elle varie entre 0 et 100 %.

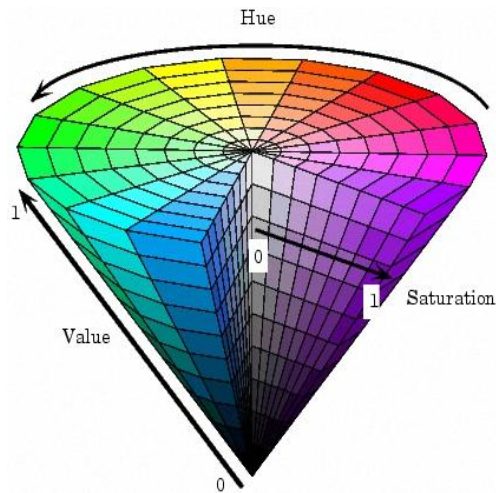


Figure 6 : Espace de couleur HSV.

4.1.1. L'histogramme de couleur :

L'histogramme de couleur est une représentation de la distribution des couleurs dans une image. Il est produit en découpant d'abord les bandes de l'espace de couleur utilisé dans un certain nombre de cases, puis en comptant le nombre de pixels dans chaque case. Formellement, l'histogramme de couleurs est défini comme suit:

$$h_{A, B, C}[a, b, c] = N. Prob\{A = a, B = b, C = c\}$$

Où A , B et C représente les bandes de couleur dans l'espace de couleur choisie (RGB, HSV, ...etc.), et N est le nombre de pixels dans l'image.

L'histogramme de couleur est couramment utilisé dans la recherche d'images et de vidéos par le contenu en raison de ses nombreux avantages. Micheal Swain et Dana Ballard [27] sont parmi les premiers à l'avoir utilisé en recherche d'images, en 1991.

Parmi les avantages de son utilisation, nous pouvons citer les suivants:

- ✓ L'extraction de l'histogramme est facile et rapide.
- ✓ Il est invariant à plusieurs transformations comme la translation, la rotation, le changement d'échelle et le point de vue de l'image.
- ✓ Généralement, il représente bien le contenu de l'image.
- ✓ Différentes mesures de similarité peuvent y être appliquées.

La figure 7 illustre un exemple d'une image et ses différents histogrammes de couleur, C-à-d. un histogramme pour chaque bande de couleur.

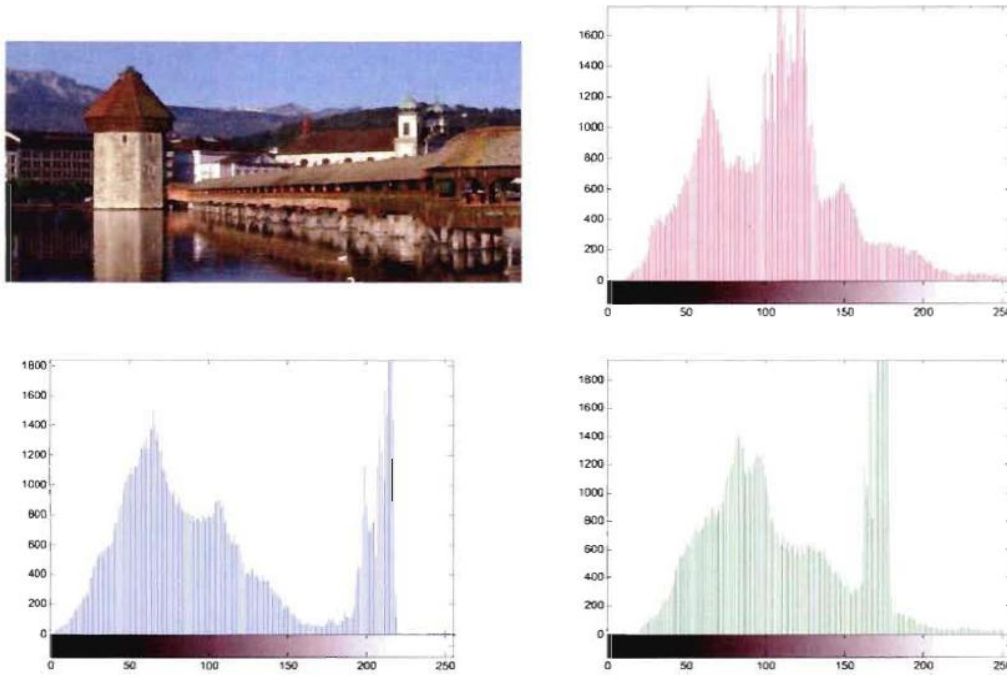


Figure 7: Les différents histogrammes d'une image couleur.

Il existe plusieurs formules pour calculer la distance entre les histogrammes, nous pouvons citer :

❖ **La distance Euclidienne :**

La distance Euclidienne entre deux histogrammes de couleur h et g est calculée pour chaque bande de couleur comme suit :

$$Dist(h, g) = \sqrt{\sum_{k=1}^n (h(k) - g(k))^2} \quad [28]$$

Où :

k est le nombre de cases de l'histogramme.

Cette mesure compare les cases identiques dans les histogrammes respectifs, où toutes les cases contribuent à parts égales à la distance.

❖ **La distance de Mahalanobis :**

La distance de Mahalanobis entre deux histogrammes de couleur h et g est calculée pour chaque bande de couleur comme suit :

$$D(h, g) = \sqrt{(h - g)^T P^{-1} (h - g)} \quad [29]$$

Où :

P est la matrice de covariance.

Cette mesure prend en compte la corrélation entre les deux histogrammes, et elle est invariante au changement d'échelle.

❖ **L'intersection d'histogramme :**

La distance entre deux histogrammes de couleur h et g est calculée pour chaque bande de couleur comme suit :

$$D(h, g) = \frac{\sum_A \sum_B \sum_C \min(h(a,b,c), g(a,b,c))}{\min(|h|, |g|)} \quad [30]$$

Où :

$|h|$ et $|g|$ sont les magnitudes des histogrammes, lesquelles sont égales au nombre d'échantillons.

a , b et c représentent les bandes de couleur.

4.1.2. Descripteur par couleurs dominantes :

Ce descripteur fournit une description compacte des couleurs représentatives dans une image.

Il est défini par :

$$F = \{(c_i, p_i, v_i, s)\}, i = (1, 2, \dots, N)$$

Où :

N : désigne le nombre de couleurs dominantes.

c_i : la valeur de la i^{eme} couleur dominante.

p_i : un poids exprimant sa fréquence relative d'apparition dans l'image et v_i sa variance.

s : La cohérence spatiale représente l'homogénéité spatiale des couleurs dominantes dans l'image.

Pour le calcul des couleurs dominantes, il faut choisir un espace de couleurs uniforme au niveau de la perception humaine.

4.2. Les descripteurs de textures :

La texture est la répétition d'éléments de base construits à partir des pixels qui respectent un certain ordre. Le sable, l'eau, l'herbe, la peau sont autant d'exemples de textures. L'aléatoire joue un rôle particulier dans les textures.

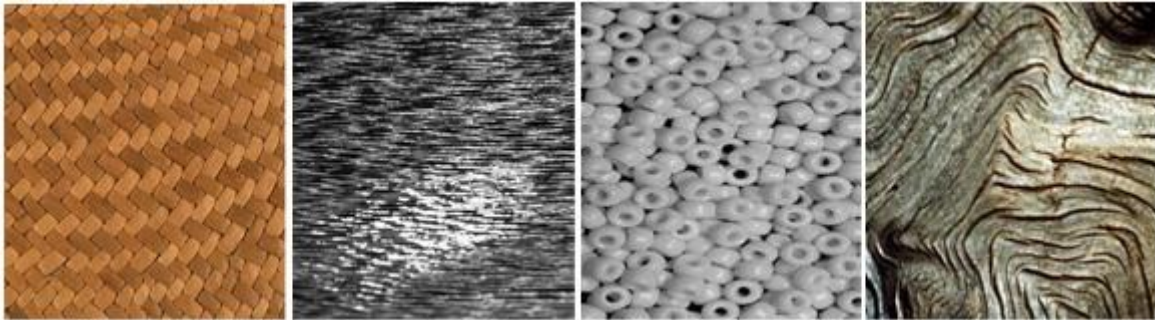


Figure 8 : Exemple de texture.

4.2.1. Matrice de cooccurrence :

Dans les années 70 Haralick et Al [31] ont proposé une des premières méthodes de caractérisation de texture baptisée *matrice de cooccurrence*. Cette approche consiste à explorer les dépendances spatiales des textures en construisant d'abord une matrice de cooccurrence basée sur l'orientation et la distance entre les pixels de l'image.

Pour mesurer la similarité entre l'image requête et les images de la base, la formule suivante a été utilisée :

$$Dist(im_{Req}, im_{Cour}) = \frac{1}{m} (\sum_{i=1}^m |param_i(im_{Req}) - param_i(im_{Cour})|^2)^{1/2} \quad [32]$$

Où :

im_{Req} est une image requête.

im_{Cour} est une image de la base.

m représente le nombre total de paramètres (moyenne, énergie, entropie, ...etc.).

Il y'a 14 paramètres de caractéristiques de texture, nous citons parmi eux :

$$\text{Moyenne} = \sum_j \sum_i p(i, j)$$

$$\text{Variance}_i = \sum_j \sum_j (i - \text{Moyenne})^2 p(i, j)$$

$$\text{Energie} = \sum_j \sum_i (p(i, j))^2$$

$$\text{Inertie} = \sum_j \sum_i (i - j)^2 p(i, j)$$

$$\text{Entropie} = \sum_j \sum_i p(i, j) \log(p(i, j))$$

Où :

(i,j) : coordonnées dans la matrice.

P(i,j) : valeurs normalisées de la matrice.

4.2.2. Les filtres de Gabor :

Les filtres de Gabor sont largement utilisés en indexation, pour la description de la texture. Ils permettent une bonne résolution temporelle à haute fréquence et une bonne résolution harmonique sans grande précision temporelle à basse fréquence.

Le filtre de Gabor est donné par l'expression suivante :

$$H(u, v) = \exp[-2\pi^2(\sigma_u^2(u - u_0)^2 + \sigma_v^2(v - v_0)^2)] \quad [33]$$

Où :

(u₀, v₀) : paramètre de position.

σ_u², σ_v² : paramètre d'étalement.

4.2.3. Les ondelettes :

La transformée en ondelettes consiste à extraire des caractéristiques de texture, et pour cela elle décompose un signal en utilisant une famille de fonctions $\varphi_{m,n}(x)$ obtenues par la translation et la dilatation d'une fonction mère $\varphi(x)$.

Les ondelettes sont générées à partir de cette fonction $\varphi(x)$.

$$\varphi_{m,n}(t) = \frac{1}{\sqrt{m}} \varphi\left(\frac{t-n}{m}\right) \quad m > 0, n \in R \quad [34]$$

Où :

m est le facteur d'échelle (dilatation).

n est le facteur de translation.

$\frac{1}{\sqrt{m}}$ est un facteur de normalisation à travers les différentes échelles.

4.3. Les descripteurs de formes :

L'utilisation de descripteurs de forme (*shape*) n'a de sens que sur une image segmentée. Pour extraire les descripteurs d'une forme, la première chose à faire est de définir sa fonction caractéristique. En général, elle est représentée sous la forme d'un masque dans lequel chaque pixel est représenté par le numéro de la région à laquelle il appartient. C'est à partir de cette fonction, que sont calculés la plupart des descripteurs de formes, soit à partir de la région entière, soit à partir des contours seulement.

4.3.1. Descripteurs de Fourier :

Les Descripteurs de Fourier font partie des descripteurs les plus populaires pour les applications de reconnaissance des formes et de recherche d'images. Ils ont souvent été utilisés par leur simplicité et leurs bonnes performances en termes de reconnaissance [35] et facilitent l'étape d'appariement. De plus, ils permettent de décrire la forme de l'objet à différents niveaux de détails.

4.3.2. Les invariants de Hu :

Hu [36] a proposé un ensemble de sept moments invariants aux translations, rotations et changement d'échelle. Ils sont très utilisés dans la littérature pour la description de formes en vue d'une classification ou d'une indexation.

Pour extraire l'information pertinente de la forme, les moments invariants de Hu sont utilisés. Ces moments permettent de décrire la forme à l'aide des propriétés statistiques, pour le calcul de deux formes d'images différentes la formule est la suivantes :

$$Dist(V_{req}, V_{cour}) = (\sum_{i=1}^7 |V(i)_{req} - V(i)_{cour}|^2)^{1/2} \quad [37]$$

4.3.3. Moments de Zernike :

Ce type des moments a été initialement introduit par Teague [38] et qui sont construits à partir de polynômes complexes et forment un ensemble orthogonal complet définie sur le disque unité. Ils sont invariants par rotation et changements d'échelles et présentent des propriétés intéressantes en termes de résistance aux bruits, efficacité informative et possibilité de reconstruction.

Les moments orthogonaux de Zernike d'ordre p sont définis de la manière suivante :

$$A_{m,n} = \frac{m+1}{\pi} \iint I(x,y) [V_{m,n}(x,y)] dx dy \quad [39]$$

Où :

m et n définissent l'ordre du moment.

$I(x, y)$ le niveau de gris d'un pixel de l'image I sur laquelle on calcule le moment.

4.3.4. Descripteur CSS :

CSS (Curvature Scale Space) est une méthode orienté contour. Ce descripteur a été proposé initialement par Mokhtarian, Abassi et Kitler [40]. Le CSS est un descripteur multi-échelle, est basé sur la description des parties concaves d'un objet. Invariant à l'échelle, à la translation et à la rotation. Ce descripteur fournit de très bons résultats pour comparer des objets un à un.

4.4. Les descripteurs de mouvements :

La vidéo est constituée d'une séquence d'images. Toutefois, l'extraction des caractéristiques visuelles que nous avons décrites dans les paragraphes précédents n'est pas suffisante pour indexer la vidéo. Cela est dû à la nécessité de décrire le contenu dynamique à l'intérieur de la vidéo. Ce dynamisme est le résultat des changements qui se produisent à travers le temps à l'intérieur du contenu de la vidéo. Ces changements sont le résultat du mouvement de la caméra, du mouvement des objets à l'intérieur de la vidéo, ou d'une combinaison des deux. Les mouvements de la caméra les plus courants sont: la translation, la rotation et le zoom. La figure 9 présente une illustration de ces différents types de mouvement de la caméra. [41]

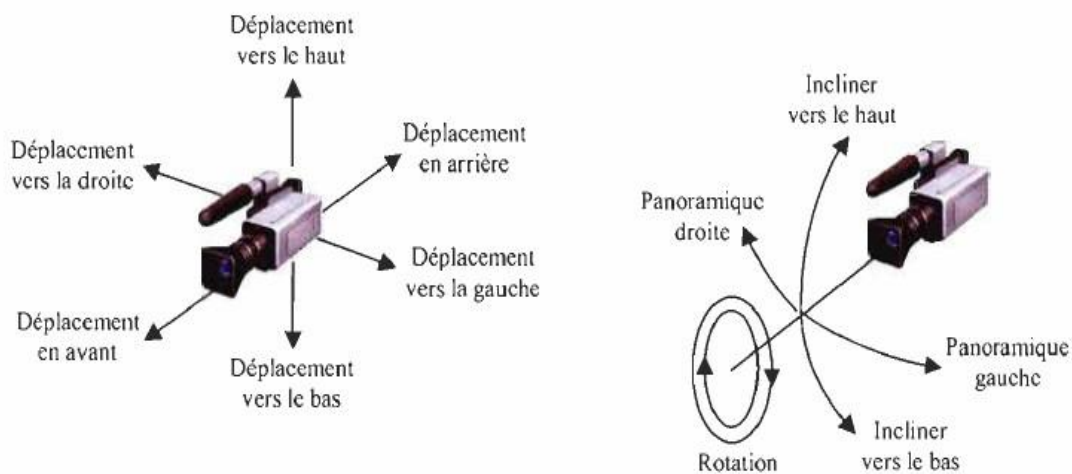


Figure 9 : Les mouvements courants de la caméra.

4.4.1. Descripteur d'activité de mouvement : (Motion Activity Descriptor)

Ce descripteur (*Descripteur promu par Mitsubishi Electric Research Laboratories (MERL), Murray Hills (Etats-Unis).*) [41] offre une mesure de la quantité du mouvement présent dans les séquences vidéos.

Il exploite les vecteurs de compensation de mouvement associés aux *macroblochs* MPEG (blocs carrés de dimension 16×16 pixels), codés en mode prédictif. L'avantage réside ici dans le fait que ces vecteurs sont déjà inclus dans les flux MPEG-1, 2 et 4 et peuvent être déterminés sans nécessiter un décodage complet de la vidéo. Quant aux *macroblochs* codés en mode *intra*, qui n'ont pas de vecteur mouvement associé, leur vitesse est considérée comme nulle.

4.4.2. Descripteur de Trajectoire : (Motion Trajectory Descriptor)

Les trajectoires (*Descripteur promu par Laboratoires d'Electronique Philips (France).*) [42] fournissent une représentation simplifiée du mouvement d'un objet dans une séquence vidéo, à partir de points clé correspondant aux positions d'un point d'intérêt de l'objet (le plus souvent, son centre de gravité) à différents instants temporels.

Dans MPEG-7, les trajectoires considérées peuvent être aussi bien 2D que 3D.

Une trajectoire est représentée par un ensemble ordonné de N quadruplets de nombres réels $(x_i, y_i, z_i, t_i)_{i=1..N}$.

Où (x_i, y_i, z_i) désignent les coordonnées du point d'intérêt de l'objet dans un repère cartésien prédéfini, à l'instant t_i .

Pour déterminer les positions de l'objet à l'intérieur d'un intervalle temporel (t, t_{i+1}) , un mécanisme d'interpolation est défini, s'appuyant sur une modélisation physique élémentaire, décrite par les équations suivantes :

$$\forall t \in (t, t_{i+1}), \begin{cases} x(t) = x_i + v_{ix}(t - t_i) + \frac{1}{2}a_{ix}(t - t_i)^2 \\ y(t) = y_i + v_{iy}(t - t_i) + \frac{1}{2}a_{iy}(t - t_i)^2 \\ z(t) = z_i + v_{iz}(t - t_i) + \frac{1}{2}a_{iz}(t - t_i)^2 \end{cases}$$

Où (v_{ix}, v_{iy}, v_{iz}) et (a_{ix}, a_{iy}, a_{iz}) désignent respectivement le vecteur vitesse et accélération à t_i . Ces positions doivent satisfaire des contraintes de continuité aux bords, exprimées comme suit :

$$x(t_{i+1}) = x_{i+1}, y(t_{i+1}) = y_{i+1}, z(t_{i+1}) = z_{i+1}.$$

4.4.3. Le descripteur de mouvement paramétrique : (Parametric Motion Descriptor)

Dans MPEG-7, le descripteur de mouvement paramétrique (*Descripteur promu par Institut National des Télécommunications (France), Université de Rochester (Etats-Unis) et Heinrich Hertz Institut (Allemagne).*) vient compléter les descriptions élémentaires de mouvement fournies par les trajectoires, qui se révèlent être souvent insuffisantes.

Le principe consiste à représenter le mouvement d'un objet entre deux trames successives par un modèle de transformation géométrique paramétrique, quadratique, projective, affine, affine simplifié ou constant.

Observons en outre, que ces modèles sont associés à des objets de forme arbitraire, définis en toute généralité comme des régions spatio-temporelles (ensemble de pixels de l'image, sur un intervalle de temps), ce qui offre un cadre unifié pour représenter les mouvements d'objet et les mouvements globaux.

5. Mesures de similarité :

Une fois ces caractéristiques extraites, la comparaison consiste généralement à définir diverses distances entre ces caractéristiques, et de définir une mesure de similarité globale entre deux images. On peut alors ordonner les images de la base suivant leur score, et présenter le résultat à l'utilisateur, les images de plus grand score étant considérées comme les plus similaires.

Remarque :

Deux images sont similaires si leurs distances est inférieure ou égale à un seuil fourni. Dans le cas contraire, nous dirons qu'ils sont différents. La notion de similarité joue un rôle important dans l'analyse des propriétés.

❖ Le seuil :

Après le calcul des distances, le résultat est une valeur comprise entre 0 et 1 (entre 0% et 100%). Pour dire que deux images sont similaires c'est en fonction de seuil.

6. Indexation et recherche de vidéos par le contenu :

Pour faire de la recherche dans une BD de vidéos, il faut en premier lieu indexer son contenu. Pour ce faire, il faut extraire, représenter et organiser efficacement le contenu de ces vidéos. La figure 10 donne une vue d'ensemble du processus de prétraitement des vidéos afin de les indexer. Ce prétraitement inclut le découpage de vidéos en images, l'extraction des caractéristiques de bas niveau, et puis vient le calcul de la mesure de similarité afin d'extraire les images clefs pour les utiliser dans la partie de recherche.

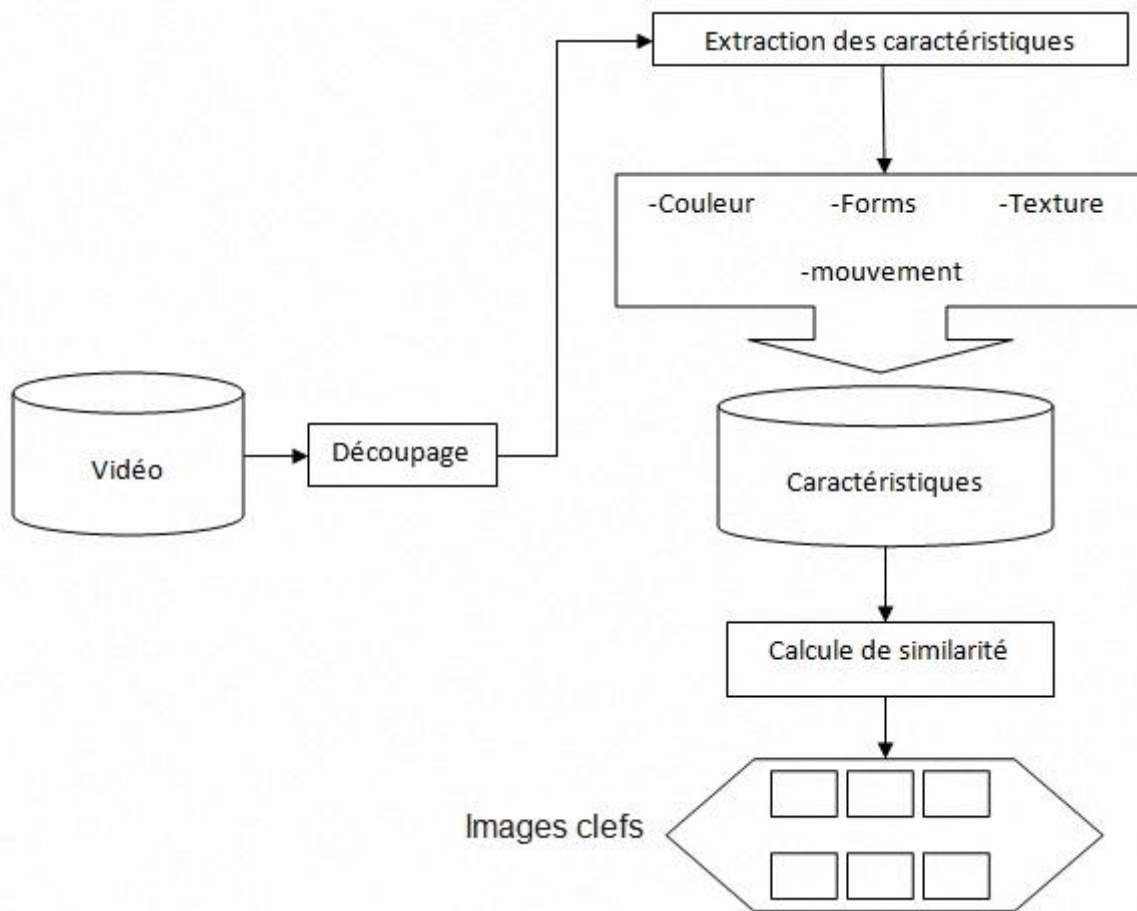


Figure 10 : Phase d'indexation de vidéos.

Une fois le pré traitement terminé, nous pouvons entamer le processus de recherche. Ce processus est illustré dans la figure 11.

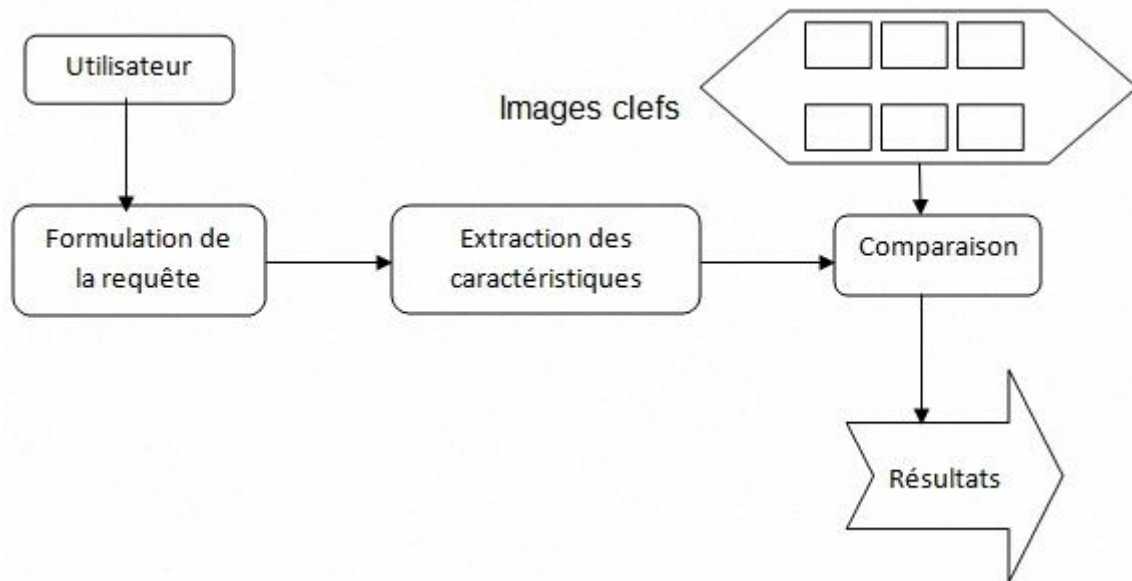


Figure 11 : Phase de recherche de vidéos.

En premier lieu, l'utilisateur doit formuler une requête qui peut inclure une vidéo. Le système analyse cette requête pour calculer les caractéristiques nécessaires, puis fait une comparaison avec les vidéos existantes dans la BD à partir de leurs images clés. Finalement, le système retourne à l'utilisateur les vidéos qui ressemblent le plus à sa requête.

La démarche est généralement suivante :

- Découpage en images individuelles.
- Segmentation en plans.
- Sélection de l'image clef.

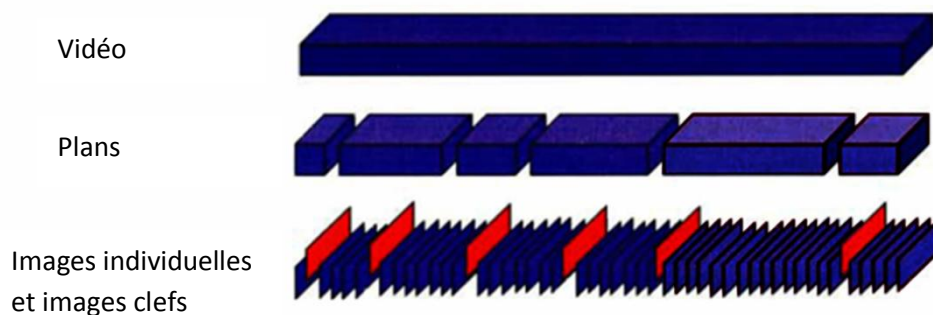


Figure 12 : Structure hiérarchique dans une vidéo.

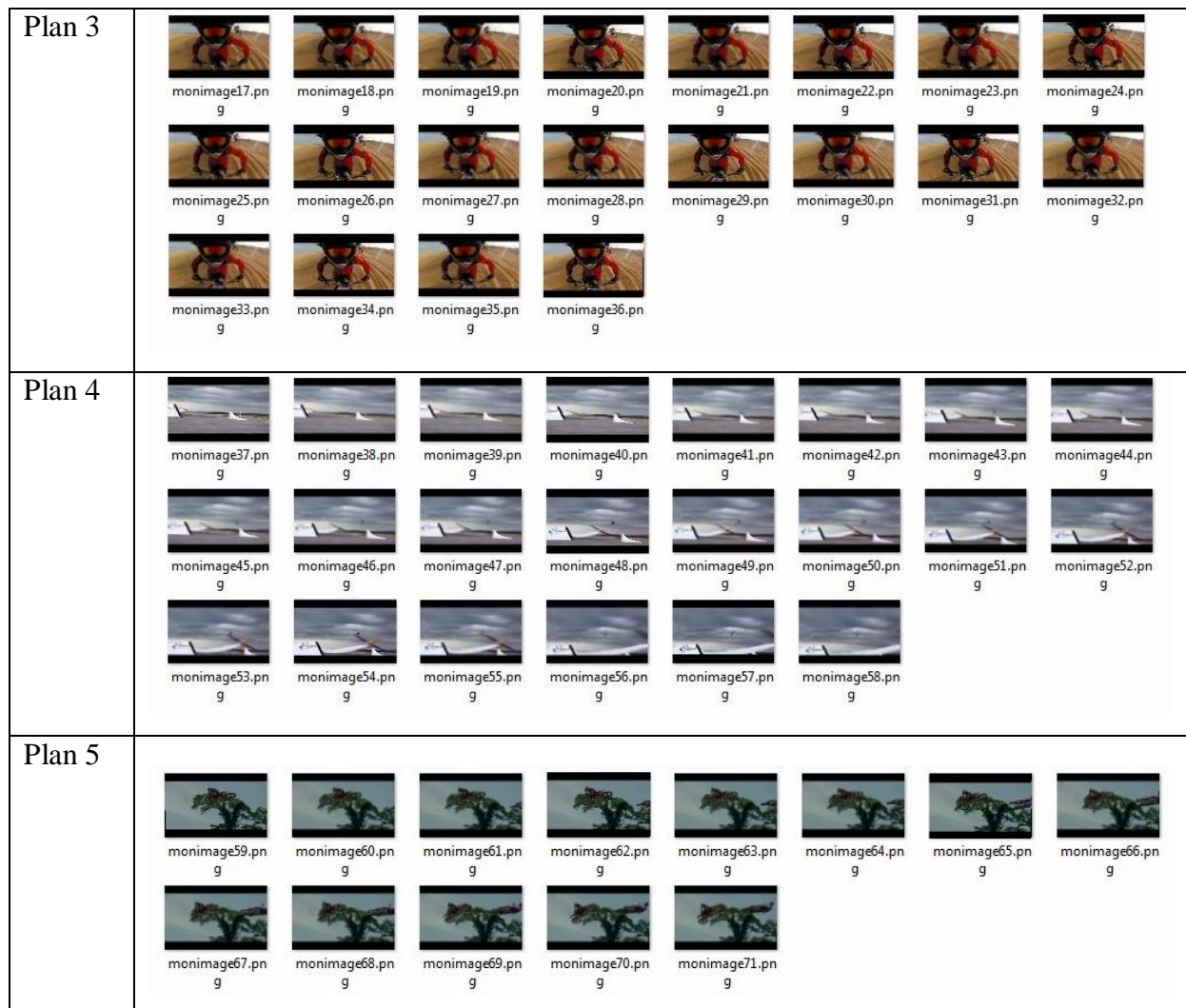


Figure 14: Segmentation en plans de la vidéo.

6.2.1. La détection des cuts dans les vidéos :

Cette section résume les approches existantes pour la détection des cuts dans les vidéos, la détection doit produire un certain nombre de plans tels que:

- toutes les images dans le même plan exposent des caractéristiques semblables.
- les images appartenant à différents plans ont des caractéristiques différentes.

Dans ce qui suit, nous allons présenter les différentes approches utilisées pour la détection des cuts dans les vidéos.

- Les méthodes basées sur les différences pixel à pixel détectent un changement de plan en calculant une différence entre les pixels de l'image à l'instant t et ceux de l'image à l'instant $t+1$. Si le nombre de pixels différents est supérieur à un seuil, alors on considère que l'on est en présence d'un "cut".

- Les méthodes à base sur l'histogramme comparent deux images successives en s'appuyant sur leurs histogrammes respectifs. Une différence des deux histogrammes est calculée et comparée à un seuil. Un cut est détecté si la différence obtenue est supérieure au seuil.
- Les méthodes basées sur une estimation du mouvement utilisent l'information de mouvement comme critère principal pour la détection des cuts. Les mouvements sont estimés pour chaque pixel d'une image obtenue à l'instant t , et sont comparés avec ceux de l'image correspondant à l'instant $t+1$. Un nombre trop important de mouvements incohérents entre les deux images successives implique alors la détection d'un cut.
- Les méthodes basées sur les blocs sont des méthodes intermédiaires entre les méthodes basées pixels (locales) et les méthodes basées histogrammes (globales).

6.3. Sélection des images clefs :

Nous devons extraire dans cette étape les caractéristiques visuelles de chaque plan, ces caractéristiques sont définies dans une ou plusieurs images appelées "images clefs".



Figure 15 : Images clefs de la vidéo.

L'image clef de notre approche est la première image de chaque plan.

Les images clefs sont les images les plus riches en informations par rapport aux autres images, l'ensemble de ces images forme ce que l'on appelle un : "résumé vidéo".

7. Conclusion :

Dans l'indexation de vidéos par le contenu on va manipuler les images, ç-à-d on va les indexer suivant leurs contenu par des descripteurs qui font l'extraction de leurs caractéristiques et puis la comparaison entre eux afin de déterminer les images clefs, l'ensemble de ces images présente le résumé vidéo.

CHAPITRE III :

Conception et mise en œuvre.

1. Introduction :

Dans ce chapitre, nous allons présenter la partie mise en œuvre. On va décrire les outils de développement employé pour réaliser notre système, nous décrivons notre approche, notre algorithme, les outils de développement employé pour réaliser notre système, ainsi que les différentes interfaces de l'application.

2. Les outils de développements de l'application :

➤ Langage de programmation :

JAVA : C'est un langage de programmation informatique orienté objet, moderne, très utilisé notamment par un grand nombre de programmeurs professionnels, développé par *Sun Microsystems* (aujourd'hui racheté par *Oracle*).

Avec le langage *Java*, on peut développer, des applications Desktop, développer des applets pour les sites web, des applications pour téléphone mobile.

➤ L'environnement de développement :

NetBeans IDE est un environnement de développement intégré (*EDI*), En plus de *Java*, *NetBeans* s'adapte aux langages de programmation

(*Python, C, C++, JavaScript, XML, Ruby, PHP* et *HTML*), aux outils et aux ressources qu'on dispose. Le programme détecte automatiquement la présence de *Java, JDK, SOA, Ruby, MySQL, ...*etc. sur le système, ainsi que les serveurs *Apache* ou *GlassFish*, pour fournir les plugins nécessaires. On peut ainsi créer facilement des applications Web, des portails d'entreprise, des logiciels multiplateformes sous *Java*, des logiciels pour mobiles, ...etc.

3. Notre approche :

Le but de ce mémoire est de construire un système capable de détecter les cuts dans une image vidéo couleur pour atteindre un résumé vidéo tel que les images clefs.

4. Algorithme :

1. Charger une vidéo ;
2. Lire la vidéo //au choix d'utilisateur ;
3. Découper la vidéo en images ;
4. Stocker les images extraites dans un dossier spécifié ;

5. Calculer le vecteur de description de chaque image ;
6. Lire seuil ;
7. image référence := la première image;
8. Calculer la distance entre l'image référence et les autres images;
9. Lorsque la distance dépasse le seuil, il y a un cut ;
10. Regrouper les images similaires dans un dossier // plan ;
11. Image référence := image de cut;
12. Répéter les étapes 8 jusqu'à atteindre le nombre d'images dans la vidéo ;
13. Afficher les images clefs // les premières images de chaque plan ;

5. Interfaces :

Lors de l'exécution de l'application, l'interface suivante apparaît :



Figure 16 : Interface d'accueil.

Nous avons définis quatre boutons dont :

- ❖ **Le premier bouton c'est « Introduire vidéo »** : ce bouton nous conduira vers le chemin pour charger une vidéo comme le montre la figure suivante :

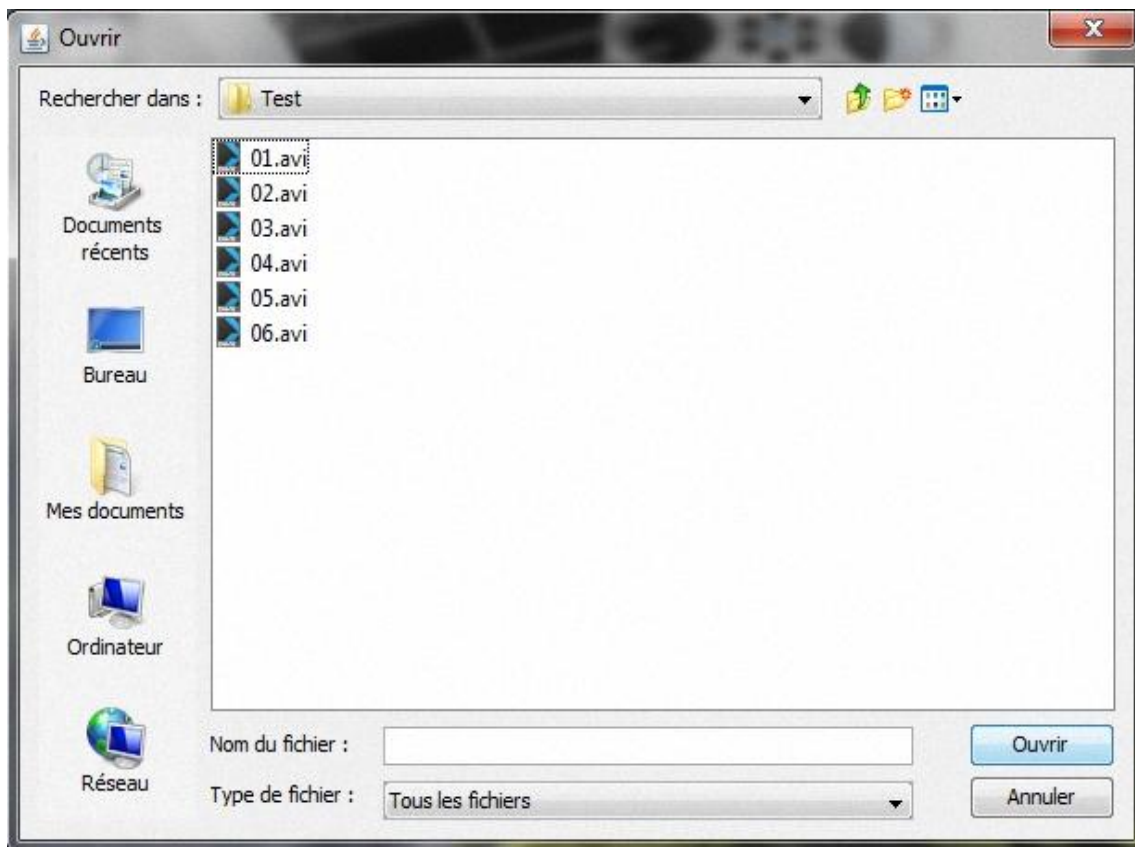


Figure 17 : Ouvrir une vidéo.

- ❖ **Le deuxième bouton c'est « Play »** : ce bouton revient au choix d'utilisateur pour voir le contenu de la vidéo déjà chargé en étape1, comme le montre la figure suivante :



Figure 18 : Lecture de la vidéo.

- ❖ **Le troisième bouton c'est « Découpage en images »** : ce bouton permet de récupérer les images individuelles et les stocks dans un dossier spécifié comme le montre la figure suivante :

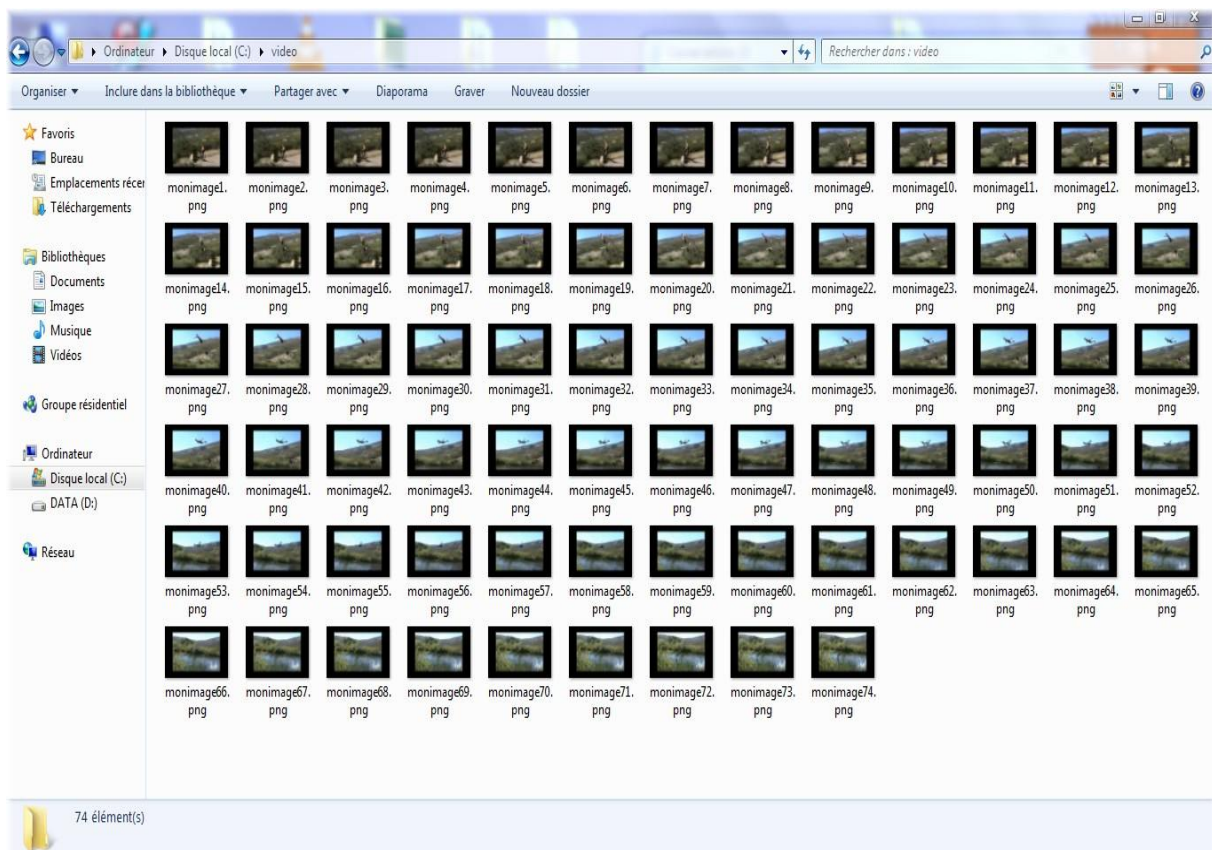


Figure 19 : Ensemble des images individuelles.

- ❖ **Le quatrième bouton « images clefs »** : ce bouton fait appel à une fenêtre dénommée « détection des cuts » comme le montre la figure suivante :

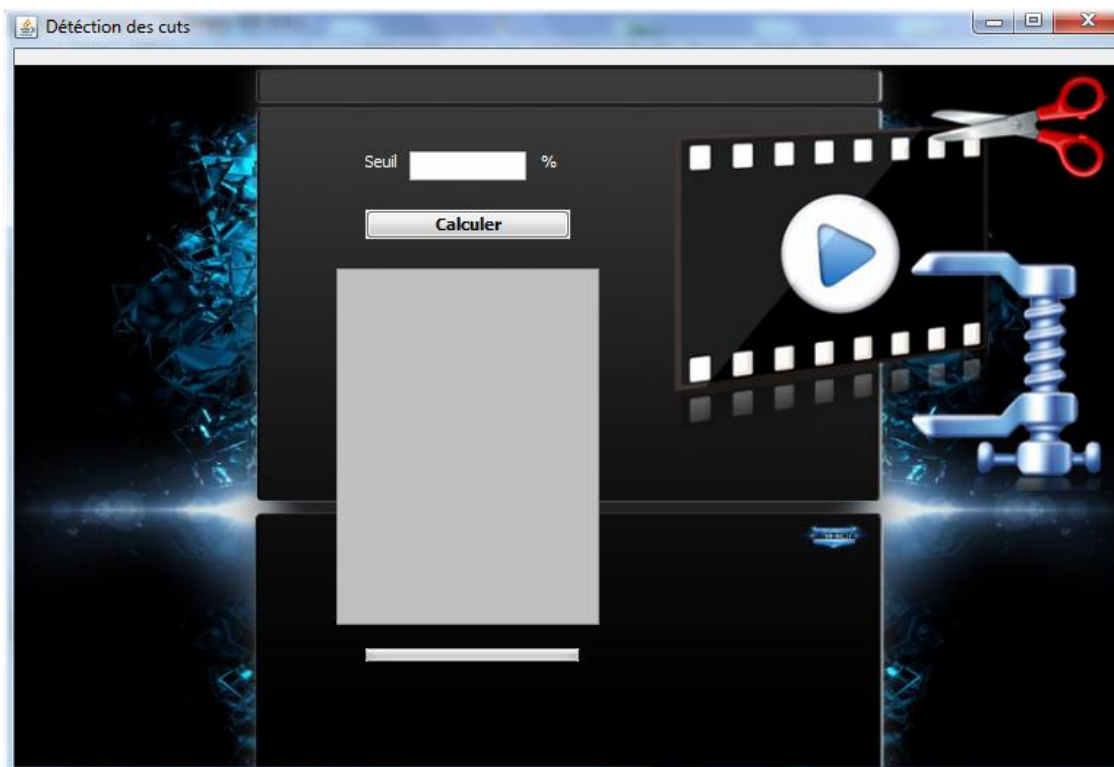


Figure 20 : Fenêtre de calcul.

Cette fenêtre est la responsable de tous les calculs nécessaires pour la détection des cuts. Dans un premier lieu l'utilisateur va introduire le seuil dont la valeur et au maximum 100 %.

Cliquer sur le bouton « calculer » l'application permettra de :

- **L'extraction des caractéristiques de chaque image** : nous avons choisi une méthode d'extraction des caractéristiques.

Les caractéristiques de notre test seront de 25 RVB triple correspondant à la moyenne des valeurs RVB sur les 25 régions.

Chaque image sera représentée par un vecteur de caractéristique de 25 x 3.

- **Le calcul de la distance et la similarité** : pour calculer la mesure de similarité entre deux images A et B nous avons pris chacun des 25 régions, calculer la distance euclidienne entre les régions et s'accumuler.

La formule qu'on a utilisée :

$$D(I_1, I_2) = \sqrt{\sum_{j=1}^n (r_{1j} - r_{2j})^2 + \sum_{j=1}^n (v_{1j} - v_{2j})^2 + \sum_{i=1}^n (b_{1j} - b_{2j})^2}$$

Où :

I_i : représente l'image i .

R_{ij} : la couleur rouge dans la région j de l'image i .

V_{ij} : la couleur vert dans la région j de l'image i .

B_{ij} : la couleur bleu dans la région j de l'image i .

Si la distance entre deux images dépasse le seuil prédéfini, l'application définit un cut, stocker les images similaires dans un plan t_i et marquer le temps de l'image où le cut a eu lieu, comme le montre la figure suivante :

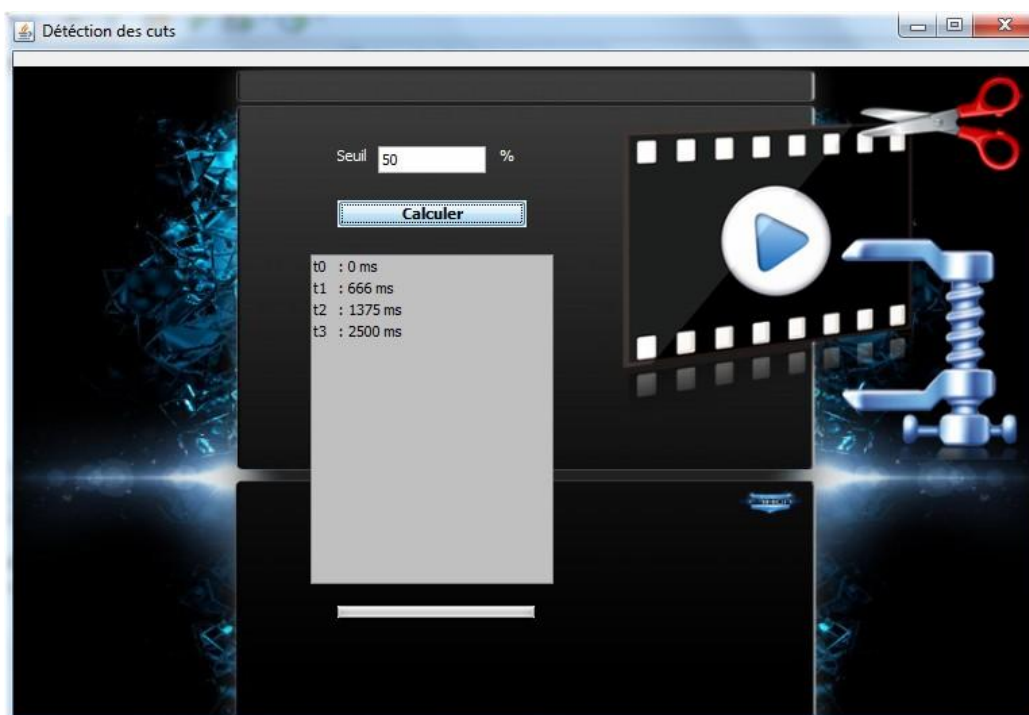


Figure 21 : Détection des cuts.

($t_0, t_1, t_2 \dots$) : représentent les temps où des cuts ont eu lieu.

Le temps affiché c'est en milliseconde.

- **L'affichage des images clefs** : afficher les premières images de chaque plan, comme le montre la figure suivante :



Figure 22 : Images clefs.

6. Résultat d'une expérimentation :

Nous avons évalué des tests sur une séquence vidéo couleur non compressée au format AVI ayant les caractéristiques suivantes :

- Résolution : 320 x 200.
- Durée : 3 secondes.
- Nombre d'image : 74.

N-test	Seuil	Nombre des images clefs	Temps des cuts (milliseconde)
1	20	14	<ul style="list-style-type: none"> • 166 ms • 333 ms • 666 ms • 833 ms • 958 ms • 1333 ms • 1833 ms • 1958 ms • 2333 ms • 2458 ms • 2541 ms • 2708 ms

			<ul style="list-style-type: none"> • 3041 ms
2	30	9	<ul style="list-style-type: none"> • 333 ms • 666 ms • 916 ms • 1291 ms • 1958 ms • 2375 ms • 2541 ms • 2750 ms
3	50	4	<ul style="list-style-type: none"> • 666 ms • 1375 ms • 2500 ms
4	70	2	<ul style="list-style-type: none"> • 958 ms
5	100	1	<ul style="list-style-type: none"> • /

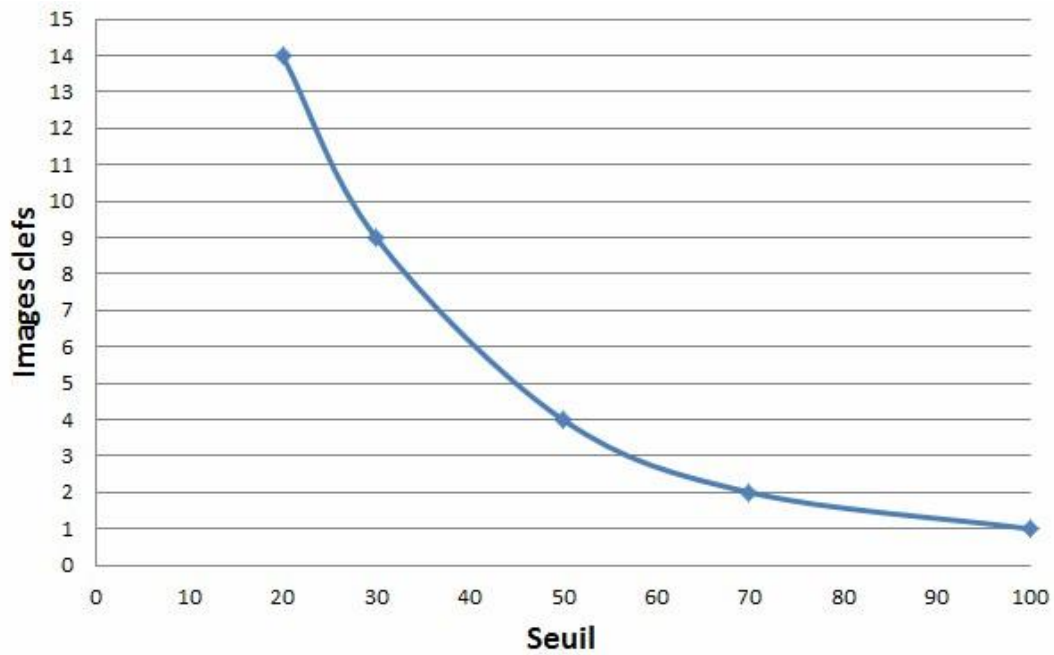


Figure 23 : Diagramme des images clefs par rapport au seuil.

6.1. Discussion :

Lors des résultats du tableau ci-dessus qui représente les différents tests sur des valeurs différents du seuil, nous avons remarqué que:

Avec un petit seuil on aura plusieurs cut et par conséquent le nombre d'images clefs est grand car un petit changement du contenu d'une image à une autre provoque une nouvelle image clef, mais plus le seuil est grand, moins il y a de chances de tomber sur une image qui varie beaucoup du contenu au point de définir un nouveau plan ce qui amène à avoir moins d'images clefs.

7. Conclusion :

Au cours ce chapitre nous avons détaillé notre application selon l'algorithme qu'on a défini, chaque interface a son propre rôle, le descripteur de statistique de couleur qu'on a utilisé pour l'extraction des caractéristiques de l'image ainsi que la distance employé pour comparer les images entre elles, le but majeur de cette application est de détecter les cuts et de construire le résumé vidéo.

Conclusion générale :

Dans ce mémoire, nous avons fait une étude générale sur l'indexation et la recherche d'images et de vidéos par le contenu pour qu'on puisse trouver une solution à notre problème qui est « la détection des cuts dans une image vidéo couleur » alors on a employé le processus suivant :

Dans un premier temps, on a étudié un état de l'art qui regroupe quelques travaux déjà existant dont chacun est le sujet qu'il a traité, les domaines d'application qui nécessitent l'indexation et la recherche, les systèmes existant et chacun son objectif et ces techniques d'indexation et de recherche, juste après, on a détaillé le processus de l'indexation d'images et de vidéos par le contenu, pour arriver à indexer une vidéo par son contenu il faut un descripteur (dans notre cas on est intéressé par les descripteur de bas niveau tel que : la couleur) et une mesure de similarité, le rôle de descripteur est l'extraction des caractéristiques, il existe plusieurs descripteur dont chacun son aspect qu'il va traiter, et la mesure de similarité c'est pour dire que tel image est similaire a tels images.

D'après l'étude qu'on a fait on est arrivé à créer une application qui est capable de détecter les cuts (changement de plans) et construire un résumé vidéo en se basant sur un descripteur de statistiques des couleurs et une distance euclidienne pour la mesure de similarité.

L'indexation et la recherche par le contenu est un domaine qui est en cours de développement, c'est vrai que les systèmes actuelle peuvent satisfaire les besoins des utilisateurs, que ça soit pour l'utilisation personnelle, ou l'utilisation professionnelle, mais sont toujours loin d'atteindre un niveau de perfection car la technologie est toujours en évolution par conséquent il y'aura des nouveaux besoins.

Bibliographie

- [1] **H. V. Dai.** « Association texte and images pour l’indexation et la recherche d’images », Rapport final, Juillet 2009.
- [2] http://en.wikipedia.org/wiki/Content-based_image_retrieval.
- [3] **T. Kato, K. Hirata.** « Query by visual example in content-based image retrieval», Proc. EDB192. Lecture Notes in computer Science, p.56-71,1992.
- [4] **S. Cohen.**«Finding color and shape patterns in images.Université Stanford». Thèse de Doctorat, Juin 1999.
- [5] **J.Fournier.** « Indexation d’images par le contenu et recherche interactive dans les bases généralistes », Université de Cergy-Pontoise, Thèse de Doctorat, Octobre 2002.
- [6] **J. Fauqueur.** « Contributions pour la Recherche d’Images par Composantes Visuelles », Université de Versailles Saint-Quentin en Yvelines, Thèse de Doctorat, Novembre 2003.
- [7] **S. A. Berrani.** « Recherche approximative de plus proches voisins avec contrôle probabiliste de la précision : application à la recherche d’images par le contenu», Université Rennes 1, Thèse de Doctorat, Février 2004.
- [8] **B. Sida.** « Recherche d’images par le contenu», Université Mouloud Mammeri, Tizi-ouzou, Mémoire de magister, 2011.
- [9] **QBIC** (IBM, 1995), <http://www.qbic.almaden.ibm.com>.
- [10] **Virage** (Virage Technologies, 1996), <http://www.virage.com>.
- [11] **Photobook** (Massachussets Institute of Technology, 1994), <http://vismod.media.mit.edu/vismod/demos/photobook>.
- [12] **Blobworld** (Université de Californie – Berkeley, 1999), <http://www.eecs.berkeley.edu/Pubs/TechRpts/1999/5567.html>.
- [13] **SIMPLIcity** (Stanford University, 2001), <http://infolab.stanford.edu/~wangz/project/imsearch/SIMPLIcity/TPAMI>.

- [14] **Ikona** (INRIA Rocquencourt – IMEDIA, 2001),
<http://www-rocq.inria.fr/imedia/ikona>.
- [15] **D. Zugaj, P.Bouthemy.** « Détection de volets pour l'indexation de vidéo par le contenu », Universitaire de Beaulieu, France. Septembre 1999.
- [16] **I. Yahiaoui.** « Construction automatique de résumés vidéos Proposition d'une méthode générique d'évaluation », Télécom Paris, Thèse de Doctorat, Octobre 2003.
- [17] **F. Souvannavong.** « Indexation et recherche de plans vidéo par le contenu sémantique », Ecole nationale supérieure des télécommunications, Paris, Thèse de Doctorat, juin 2005.
- [18] **M. Guironnet.** « Méthodes de résumé de vidéo à partir d'informations bas niveau, du mouvement de caméra ou de l'attention visuelle », Université Joseph Fourier, Grenoble 1, Thèse de Doctorat, Octobre 2006.
- [19] **M. Bourenane.** « Un outil pour l'indexation des vidéos personnelles par le contenu », Université du Québec à Trois-Rivières, Mémoire, Août 2009.
- [20] **A. Ben Abdelali, M. Krifa, L. Touil, A. Mtiba, E. Bourenane.** « Indexation de la vidéo par le contenu et besoin en architectures matérielles dédiées », Université de Bourgogne Dijon, France, 2009.
- [21] **A. Brillhault.** « Indexation et recherche par le contenu de documents vidéos », Ingénierie et Maintenance Aéronautique, centre d'information, Grenoble.
- [22] **H. Wactlar, T. Kanade, Smith, M. A., Stevens, S. M.** « Intelligent access to digital video: The informedia project », IEEE Computer, mai 1996.
- [23] **A. F. Smeaton.** « The Fischlar Digital Library: Networked Access to a Video Archive of TV News », Terena Networking Conference 2002, Limerick, Ireland, 3-6 June 2004.
- [24] **Imperial College London,** Multimedia and Information Systems: Research. Récupéré le 29 juin 2009 de <http://mmis.doc.ic.ac.uk/research.html>.
- [25] **A. Hafiane.** « Caractérisation de textures et segmentation pour la recherche d'images par le contenu », Université de Paris-Sud XI, Thèse de Doctorat, Décembre 2005.
- [26] **Smith, J. R., Chang, S. F.** « Automated image retrieval using color and texture », Technical Report CU/CTR 408-95-14, Columbia University, July 1995.

- [27] **D. Heesch, P. Howarth, J. Magalhães, A. May, M. Pickering, A. Yavlinski, S. Rüger.** « Video Retrieval using Search and Browsing », Gaithersburg, Maryland, 15-19 November 2004.
- [28] **N. Idrissi.** « La navigation dans les bases d'images : prise en compte des attributs de texture », Université de Nantes, Thèse de Doctorat, Octobre 2008.
- [29] **J. Canny.** « A computational approach to edge detection », IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 8, n°6 :679 – 698, Novembre 1986.
- [30] **B. S. Manjunath, W. Ma.** « Texture features for browsing and retrieval of image data », IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI Special issue on Digital Libraries), 18(8): 837-42, pages 18, 1996.
- [31] **R. M. Haralick, K. Shanmugan, I. Dinstein.** « Textural features for image classification », IEEE Trans. Syst. Man. Cybern. SMC-3(6): 610–621. 1973.
- [32] **O. Aude, A. Torralba.** « Modeling the shape of the scene : a holistic representation of the spatial envelope », International Journal of Computer Vision, 42(3):145_175, 2001.
- [33] **M. Tuceryan, A. K. Jain.** « Texture analysis », The Handbook of Pattern Recognition and Computer Vision (Edition 2), pages 207-248, 1998.
- [34] **G. Lu, J. Phillips.** « Using perceptually weighted histograms for color-based image retrieval ». Proceedings of the 4th International Conference on Signal Processing. pp.1150–1153.
- [35] **D. ZHANG, G. LU.** « Study and evaluation of different Fourier methods for image retrieval », Image Vision Computing, vol. 23, 2005.
- [36] **M. K. Hu.** « Visual Pattern Recognition by moment invariants ». IRE Transaction on Information Theory, Volume 8, n°2:179–187, 1962.
- [37] **M. K. Mandal, T. Aboulnasr, S. Panchanathan.** « Illumination invariant image indexing using moments and wavelets », Journal of electronic Imaging, 7(2) : 282– 293, 1998.
- [38] **M. R. Teague.** « Image analysis via the General Theory of moments », Applied optics, vol. 19, n° 8 (1980), pp. 1353-1356, 1980.

- [39] **P. Gros.** « De l'appariement à l'indexation des couleurs », Institut National Polytechnique Bretagne, Thèse de Doctorat 2008.
- [40] **F. Mokhtarian, S. Abbasi, J. Kittler.** « Efficient and robust retrieval by shape content through curvature scale spac », Proc. International Workshop on Image Databases and MultiMedia Search, Amsterdam, The Netherlands, pp 35-42, 1996.
- [41] **A. Divakaran, H. Ito, H. Sun, T. Poon.** « Scene change detection and feature extraction for indexing MPEG-2 and MPEG-4 sequences», IEEE Trans. Circuits Syst. Video Technology , October 1998.
- [42] **S. Jeannin, B. Mory.** « Video Motion Representation for Improved Content Access», *IEEE Trans.on Consumer Electronics*, Vol. 46, No. 3, pp. 645-655, Août 2000.