

Faculté des Sciences Exactes & de l'Informatique
Département de Mathématiques et d'Informatique
Filière Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : Ingénierie des Systèmes d'Information

**Recherche et indexation automatique des vidéos
par le contenu, Application d'un vecteur de
Descripteurs Visuels.**

Présenté par :

Medjahed Soufiane

Belarbi Mohammed Amin

Encadré par :

Henni Karim Abdelkader

Année Universitaire 2012/ 2013

Remerciement

Avant tout nous tenons à remercier le grand "DIEU" de nous avoir donné le courage et la volonté qui nous ont permis de réaliser ce travail.

Nous tenons vivement à remercier toutes les personnes qui ont participées de près ou de loin à l'élaboration de ce travail, en particulier notre encadreur Monsieur Henni Karim Abdelkader d'avoir accepté de nous prendre en charge dans l'accomplissement de ce projet qui est un sujet d'actualité, pour ses conseils et toutes ses recommandations. Merci pour vos précieux aides, encouragements et éclaircissements.

Nous présentons nos respects et nos remerciements aux membres du jury qui nous ont fait l'honneur d'accepté d'évaluer notre travail.

Nous adressons une pensée particulièrement affective à nos amis, qui ont rendu agréables nos longues années d'études.

Nous tenons aussi à remercier chaleureusement l'ensemble des enseignements du département d'Informatique de l'Université de ABDELHAMID IBN BADIS MOSTAGANEM.

Dédicaces

Je dédie ce travail fruit, de plusieurs années de réflexions :

A ma très chère mère et mon joyau père qui m'ont aidé par

leurs invocations ainsi, que leurs bénédictions et leur soutien

durant toute ma vie.

A tous mes frères (Mohammed, Saada, et Boubakra) et ma

sœur (Sabih).

A tous la famille Medjahed.

A tous mes amis (Sidou, Amine Ouldali, Amine Belouza et sa

femme, Rim, Zoubida, Abdelhamid Derrax)

A tous mes amis qui m'ont donnée le courage (Farouk, Oussama,

Nadia, Samia, Fatima, Asma, Sarah)

A tous les étudiants de la promotion master 2012 du département

d'informatique.

Softane

Dédicaces

Je dédie ce travail fruit, de plusieurs années de réflexions :

A ma très chère mère et mon joyau père qui m'ont aidé par

Leurs invocations ainsi, que leurs bénédictions et leur soutien

durant toute ma vie.

A toutes ma famille (Abdelakder et sa femme Imen, Abdelkhalek

et sa femme Samira, Houria et son époux Laid, Mansouria et son

époux Hadj, Elalia et Fatima)

A tous mes proches (Soumia Boudour, Lila, Wafae)

A tous mes amis (Abdelaaziz Chelili, Tijour Ahmed, Hagani

Zakaria, Abdelhamid Derrar)

A tous les étudiants de la promotion master 2012 du département

d'informatique.

Amin Mohammed

Résumé

L'indexation des vidéos par le contenu représente un grand intérêt dans le domaine de la recherche intelligente des vidéos. Les caractéristiques visuelles telles que la couleur fait partie des composantes les plus pertinentes.

Nous proposons dans ce mémoire des notions de base sur l'indexation par le contenu, ainsi notre approche et les démarches mise en œuvre dans notre application.

Notre projet consiste à réaliser un système reposant sur les deux phases ; l'indexation et la recherche.

Mots clés : indexation, indexation d'image par le contenu, indexation de vidéo par le contenu, traitement d'image, segmentation vidéo, mesure de similarité.

Abstract

Indexing video by content represents a great interest in the field of intelligent search of videos. The visual characteristics such as color are one of the most relevant components.

We are proposing in this memoir the basics of indexing by content and our approach and approaches implemented in our application.

Our project is about realizing a system based on two phases: indexing and search.

Keyword: indexing, indexing image by content, indexing videos by content, image processing, video segmentation, similarity measure.

SOMMAIRE

RESUME	V
ABSTRACT	V
SOMMAIRE.....	VI
LISTES DES FIGURES	VIII
INTRODUCTION GENERALE	1
MOTIVATIONS ET OBJECTIFS :.....	2
CHAPITRE 1 : ETAT DE L'ART SUR LES SYSTEMES D'INDEXATION ET DE RECHERCHE D'IMAGE ET DE VIDEO PAR LE CONTENU	4
1. INTRODUCTION :	4
2. INDEXATION ET RECHERCHE D'IMAGES :	4
3. INDEXATION ET RECHERCHE DE VIDEOS :	7
4. LES SYSTEMES DE RECHERCHE ET D'INDEXATION D'IMAGES PAR LE CONTENU :	8
5. LES SYSTEMES DE RECHERCHE ET D'INDEXATION DE VIDEOS PAR LE CONTENU :	9
6. CONCLUSION :	10
CHAPITRE 2 : L'INDEXATION	11
1. INTRODUCTION :	11
2. INDEXATION ET RECHERCHE :	11
3. INDEXATION DES IMAGES :	12
3.1. <i>Image numérique</i> :	12
3.2. <i>Les descripteurs</i> :	15
3.2.1. <i>Couleur</i> :	15
• <i>Histogramme</i> :	17
• <i>Histogramme Couleur</i> :	17
3.2.2. <i>La texture</i> :	19
3.2.3. <i>La forme</i> :	19
3.3. <i>Les mesures de similarité</i> :	20
3.3.1. <i>Définition</i> :	20
3.4. <i>Les Distances</i> :	20
3.5. <i>Le seuil</i> :	21
4. INDEXATION DES VIDEOS	21
4.2. <i>Un document vidéo</i> :	21
4.3. <i>Indexation de bas niveau</i> :	22
4.4. <i>Segmentation Temporelle</i> :	25
5. RESUME :	27
6. CONCLUSION :	27
CHAPITRE 3 : CONCEPTION ET MISE EN ŒUVRE	28
1. INTRODUCTION :	28
2. OUTIL ET ENVIRONNEMENT :	28
2.1. <i>Visual studio 2010</i> :	28
2.2. <i>C++</i> :	28
2.3. <i>My sql server</i> :	28
2.4. <i>OpenCV</i> :	29
3. NOTRE APPROCHE :	29
3.1. <i>Le codage</i> :	29
3.1.1. <i>Algorithme de codage</i> :	29
3.1.2. <i>Extraction des images individuelles</i> :	30
3.1.3. <i>Enregistrement dans la base de données</i> :	31

3.2.	<i>La recherche :</i>	31
3.2.1.	<i>Algorithme de recherche :</i>	31
3.2.2.	<i>Calcul de l'histogramme couleur et l'histogramme normalisé de l'image requête :</i>	32
3.2.3.	<i>Calcul de distance de similarité entre l'histogramme de la requête et ceux qui sont stocker dans la base :</i>	32
3.2.4.	<i>Récupération des vidéos résultats :</i>	33
3.3.	<i>Analyse :</i>	34
4.	APPLICATION :	34
4.1.	<i>Interface utilisateur (Première fenêtre) :</i>	34
4.1.1.	<i>L'onglet codage (partie indexation) :</i>	35
4.1.2.	<i>L'onglet Recherche :</i>	38
4.1.3.	<i>L'onglet aide : un petit site avec quelques informations sur l'application. Même l'application est disponible sur le site ou c'est possible de la télécharger. Le site :http://jamesbond20.wix.com/video-indexer-2</i>	41
5.	MESURES POUR EVALUER UN SYSTEME	41
6.	CONCLUSION :	42
	CONCLUSION GENERALE	43
	BIBLIOGRAPHIE	44

Listes des figures

Figure 1: Comparaison de quelque algorithme utilisé pour l'extraction de caractéristiques et de son application de recherche. [20]	8
Figure 2: Model général d'un système d'indexation et de recherche d'information.....	12
Figure 3: L'architecture d'un système d'indexation et de recherche d'images	14
Figure 4: La même image en RGB, YIQ, YUV, CMY.....	16
Figure 5: La même image en HSV, et HLS.....	16
Figure 6: Exemple d'histogramme.....	17
Figure 7: Problème d'histogramme au contraste et luminosité.....	18
Figure 8: Exemple de texture	19
Figure 9: Architecture d'un système d'indexation et de recherche de vidéos par le contenu	22
Figure 10: Découpage en série d'image.....	23
Figure 11: Segmentation temporelle (en plan)	24
Figure 12: Résumé vidéo (image clé de chaque plan)	25
Figure 13: Illustration de la segmentation temporelle	25
Figure 14: Instruction qui fait l'extraction des images individuelles.	30
Figure 15: l'histogramme normalisé d'une image.	30
Figure 16: Enregistrement des images et les histogrammes.....	31
Figure 17: la partie connexion avec la base de données.....	31
Figure 18: Calcul de l'histogramme et le normaliser	32
Figure 19: Calcul de la distance	33
Figure 20: la partie récupération	33
Figure 21: Première interface de l'application	34
Figure 22: la fonction du bouton Ouvrir.....	35
Figure 23: Dossier contenant les vidéos est chargé avec succès	36
Figure 24: l'interface lors du codage	37
Figure 25: Enregistrement des images clés dans le dossier images	37
Figure 26: La fenêtre information.	38
Figure 27: L'interface de l'onglet Recherche.	39
Figure 28: Les distances utilisées.	40
Figure 29: Le paramètre du seuil.....	40
Figure 30: l'application pendant le traitement de la recherche.	41

Introduction Générale

Durant ces dernières années, le développement des moyens de communication et de l'audiovisuelles sont en pleine expansion, avec l'explosion d'Internet et aussi le développement à grande échelle de la photographie numérique, il n'est pas rare d'avoir des bases de donnée multimédia contenant plusieurs milliers voir même plusieurs dizaines de milliers d'images , vidéos et sons, que ce soit des bases ciblées pour un domaine d'activité professionnelle (médical, sécurité ,journalisme, tourisme, éducation, musées, ...) ou tout simplement pour les particuliers qui accumulent des données personnelle comme : souvenirs, voyages , famille , événements ,collection de films , ...).

Un problème s'impose : le volume des données multimédia augmente d'une façon exponentielle, les applications qui ont pour but d'explorer et exploiter ces données, n'ont pas encore atteint un niveau de satisfaction. En effet, les applications qui visent la segmentation et l'indexation des séquences vidéo pour identifier les éléments de leurs contenus souffrent de plusieurs insuffisances, surtout lors de l'extraction des caractéristiques de ces derniers.

L'Indexation dans les bases de données est une étape importante et nécessaire pour faciliter la recherche et l'interrogation, dans le but de minimiser et réduire le temps réponse afin d'améliorer et augmenter la performance d'un système de recherche d'information.

L'interrogation de bases d'images par le contenu est notoirement onéreuse, ce type d'interrogation a été rendu possible par le mariage de techniques « bases de données » et « traitement d'image ». Les derniers développements en bases de données visent à accélérer les recherches dans des index multidimensionnels, ceux en images visent à améliorer les capacités de reconnaissances automatiques. Or, les techniques modernes de traitement d'image ont pour conséquences de changer fondamentalement la manière dont la reconnaissance s'effectue : il est maintenant nécessaire d'interroger de nombreuses fois la base puis de synthétiser les résultats avant de pouvoir retourner une réponse, et non plus de se contenter d'une seule et unique interrogation comme par le passé.

Motivations et objectifs :

Trois tâches sont essentielles pour la gestion de documents multimédias [17] : les numériser, les stocker et les retrouver. Les avancées technologiques ont permis aux professionnels et aux particuliers de numériser et de stocker de nombreux documents qui ne se limitent plus au texte mais qui incluent à présent la photo et la vidéo.

Le moindre accessoire (appareils photos, caméras, téléphones portables, etc ...) est maintenant capable d'acquérir de petites séquences d'images de notre quotidien. Et le tout peut facilement être partagé au travers du réseau Internet. Cependant retrouver un document voulu dans une multitude de documents multimédias n'est pas toujours efficace et parfois impossible.

Pour retrouver efficacement un document, deux phases sont importantes : la phase d'indexation et la phase de recherche. La phase d'indexation vise à calculer des descripteurs pertinents des documents multimédias et à créer des index à partir des descripteurs extraits. Une fois l'indexation effectuée, la phase de recherche consiste à mettre en correspondance la requête et les informations indexées et à retourner les résultats retrouvés à l'utilisateur. En général, le terme de "recherche d'information" contient lui-même implicitement les deux phases. La définition générale de la recherche d'information se trouve dans [18] :

"Multimedia information retrieval (MIR) is about the search for knowledge in all its forms, everywhere."

Une approche de recherche d'information est efficace si elle est capable de retrouver des informations appropriées à la requête de l'utilisateur dans un temps acceptable. La recherche d'information a naturellement commencé avec le texte, puis continué avec l'image et la vidéo. Les recherches de textes et d'images deviennent matures avec certains résultats obtenus [19]. La recherche de vidéos cependant devient active depuis seulement 10 ans. Les approches proposées au tout début étaient simplement de considérer une vidéo comme une

séquence d'images. C'est pourquoi les approches dédiées à la recherche d'images ont été appliquées à celle de vidéos.

La recherche de vidéo peut être classifiée par le type de vidéos : les vidéos professionnelles télédiffusées (films, émissions), les vidéos de sport, les vidéosurveillances et les vidéos domestiques. La recherche de vidéo a commencé avec les vidéos professionnelles et continué avec les vidéos de sport et les vidéos de vidéosurveillance. L'augmentation du nombre de chaînes de télévision donne de grandes bases d'émissions stockées. La recherche de vidéos télédiffusées a été développée pour pouvoir répondre au besoin de retrouver des informations d'intérêt dans les émissions. L'analyse de vidéos de sport associant deux applications, le résumé de vidéos et l'indexation et la recherche de vidéos, a reçu beaucoup d'attention. Les recherches de vidéos domestiques et de vidéosurveillance ont émergé ces dernières années.

Notre objectif est de mettre en fonction un système capable de récupérer les vidéos en basant sur le contenu.

Chapitre 1 : Etat de l'art sur les systèmes d'indexation et de recherche d'image et de vidéo par le contenu

1. Introduction :

La majorité des applications qui existent, que ça soit pour les chaînes télévisées, télésurveillance, recherche sur l'internet ou autre, se font en général en se basant sur une annotation textuelle ou des éléments de texte qu'on peut rattacher aux données.

Cette annotation, Souvent elle se fait manuellement par l'être humain, l'archivage des images et des séquences vidéos, à l'aide des mots clefs, en outre c'est une tâche longue et répétitive pour l'humain, surtout avec la quantité volumineuse qui ne cesse pas de s'augmenter avec le temps.

Dans ce chapitre on va dresser un panorama des approches dédiées à l'indexation et à la recherche d'images et de vidéos. Nous présentons tout d'abord l'indexation et la recherche d'images et ensuite celles de vidéos.

2. Indexation et recherche d'images :

L'indexation et la recherche d'images deviennent un domaine très actif depuis 1994. Selon [19], le nombre de publications dans ce domaine est environ 1000 publications chaque année. Nous présentons dans cet état de l'art trois aspects importants : la recherche locale, l'ontologie et le retour de pertinence.

2.1. Recherche locale vs recherche globale :

Les premières approches proposées pour l'indexation et la recherche d'images ont utilisé les descripteurs globaux d'images. Les descripteurs globaux tels que l'histogramme des couleurs sont calculées avec la participation de tous les pixels dans une image. La disposition des pixels n'est donc pas prise en compte. Evidemment, de telles approches ne sont pas appropriées pour la recherche d'images ayant une seule ou quelques parties similaires à la requête. [16]

La méthode de segmentation peut être simple ou complexe. L'objectif de cette segmentation est de décomposer une image en quelques régions. Dans le cas idéal, chaque région correspond à un objet réel. Le noyau des approches d'indexation et de recherche d'images au niveau régions est la mise en correspondance entre images. Cette mise en correspondance doit tenir compte de deux caractéristiques. (1) Le nombre de régions varie d'une image à l'autre. (2) La segmentation n'est pas parfaite, une région peut être appariée à plus d'une région.

2.2. Retour de pertinence

Les résultats de recherche d'information ne sont pas toujours parfaits en raison du fossé sémantique que nous expliquons dans le chapitre 1. La méthode pouvant combler ce fossé doit permettre de communiquer avec l'utilisateur. La recherche d'information interactive est une recherche d'information qui permet à l'utilisateur de faire un retour de pertinence. Le retour de pertinence est un processus qui consiste à apprendre à partir des retours de l'utilisateur et à trouver de nouveaux résultats pour répondre à l'utilisateur en se basant sur la connaissance apprise. L'approche permettant de faire un retour de pertinence comprend deux processus : un apprenant et un sélectionneur.

L'apprenant décide sur quelle partie de retours il va apprendre et comment il apprend. Le sélectionneur effectue la recherche en utilisant la connaissance apprise par l'apprenant et rend les nouveaux résultats à l'utilisateur. L'utilisateur démarre le processus de recherche en exprimant sa requête. Le système effectue la mise en correspondance entre les éléments indexés et la requête.

La liste des résultats ordonnés de manière décroissante par leurs similarités avec la requête est rendue à l'utilisateur. Si l'utilisateur est satisfait par la réponse, il le processus de recherche (sans avoir démarré le retour de pertinence). Sinon le retour de pertinence démarre. Il demande à l'utilisateur d'annoter les résultats obtenus comme exemples positifs et/ou exemples négatifs. L'apprenant est activé pour apprendre l'intention de l'utilisateur à partir de ses retours. Le sélectionneur vise à effectuer une nouvelle recherche et rendre de nouveaux résultats à l'utilisateur.

Nous présentons quelques définitions que nous utilisons pour le retour de pertinence

- Une itération de recherche comporte un appel de l'apprenant et du sélectionneur.
- La session de recherche d'un utilisateur commence lorsque l'utilisateur formule sa requête et se termine au moment où il est satisfait par la réponse de recherche. Une session de recherche peut comporter plusieurs itérations de recherche ;
- Le retour de pertinence est à court terme si son sélectionneur n'emploie que ce que l'apprenant a appris lors de la session en cours pour trouver de nouveaux résultats ; Le retour de pertinence est à long terme si son sélectionneur emploie ce que l'apprenant a appris lors de la session en cours et d'autres sessions de recherche pour trouver de nouveaux résultats. Notons que les sessions de recherche peuvent être effectuées par des utilisateurs différents ;
- Le retour de pertinence est dit avec exemples positifs si son apprenant n'utilise que les exemples positifs ;
- Le retour de pertinence est dit avec exemples positifs et négatifs si son apprenant utilise les exemples positifs et négatifs.

Le retour de pertinence se base sur quatre hypothèses générales introduites par Crucianu et al. [14] pour le retour de pertinence en indexation et recherche d'images :

- Il est possible de distinguer les objets pertinents et non pertinents en se basant sur leurs descripteurs ;
- Il existe un lien entre les espaces de descripteurs sur lesquels les objets sont représentés et les caractéristiques des objets que l'utilisateur veut chercher ;
- Le nombre d'objets pertinents d'une requête est relativement petit par rapport au nombre des objets dans la base.
- L'utilisateur est volontaire pour juger les résultats.

3. Indexation et recherche de vidéos :

3.1. Introduction :

Malgré de nombreux efforts de recherche, les caractéristiques de bas niveau existantes ne sont pas encore assez puissantes pour représenter le contenu d'image d'index. Certaines fonctionnalités peuvent atteindre relativement bonne performance, mais leurs dimensions de fond sont généralement trop élevées, ou la mise en œuvre de l'algorithme est difficile. L'extraction de caractéristiques est une étape cruciale dans le système de récupération de décrire la vidéo avec un nombre minimum de descripteurs. Les caractéristiques visuelles de base de la trame d'index incluent la couleur et la texture.

3.2. Quelques importantes approches dans le domaine :

Récupération de la vidéo basés sur les requêtes textuels : récupération de la vidéo basés sur des requêtes textuelles : a présenté une approche qui permet une recherche basée sur l'information textuelle présent dans la vidéo, les régions de l'information textuelle sont en retrait dans les images de la vidéo, est ensuite annoté avec le texte. Le contenu présent dans les images. . [20]

Automatique recherche par le contenu et la classification sémantique de contenu vidéo : a présenté un cadre d'apprentissage où la construction d'un index vidéo de haut niveau est visualisée par la synthèse de l'ensemble de ses fonctionnalités élémentaires. Cela se fait par l'intermédiaire du *support vecteur machines* (SVM). Les machines à vecteurs de soutien associer chaque ensemble de points de données dans l'espace de caractéristiques multidimensionnel à l'une des classes en cours de formation. . [20]

L'objectif n'est pas un recensement exhaustif ou à l'exhaustivité de la recherche dans ce domaine, mais citer et analyser qualitativement des approches telles qu'elles ont évolué dans ce domaine.

Algorithme	Approche	Caractéristiques	Retrieval application
Classification de couleur Texture	Segment d'image en région	Couleur, texture traits	importance perceptuelle
Contenu multimodal Basé sur les méthodes de Navigation et de recherche	P2P récupération systèmes	Key Frame extraction, forme Caractéristique	Basé sur l'objet
Résumés sémantiquement significatives	Séquence vidéo analyse pour identifier les points de vue pertinents de la caméra, et suit les mouvements de billes	Basé sur la Scénarisation, et matrice de cooccurrence	Sport Vidéo
Reconnaissant l'objet dans Séquences vidéo	classification séquentielle méthodes de blob	Filtre de Kalman appliqué sur blob	Objet de l'image en fonction

Figure 1: Comparaison de quelque algorithme utilisé pour l'extraction de caractéristiques et de son application de recherche. [20]

4. Les Systèmes de recherche et d'indexation d'images par le contenu :

Depuis les années 90, plusieurs systèmes d'indexation et de recherche d'image par contenu on était proposé, parmi les principaux systèmes existants nous pouvons citer :

- **Blobworld** (Université de Californie – Berkeley 1995) : Les caractéristiques utilisées pour l'interrogation sont la couleur, la texture, l'emplacement et la forme des régions (blobs) et de l'arrière-plan. La couleur est décrite par un histogramme de 218 bacs de couleur dans les coordonnées Lab-espace. La texture est représentée par contraste moyen et l'anisotropie sur la région, comme la coordonnée 2D. La forme est représentée par (approximatif) zone, l'excentricité et l'orientation. [05]

- Le fameux système **QBIC (Query By Image Content)** : a été développé par IBM en 1995 au centre de recherche Almaden d'IBM, il permet à l'utilisateur de faire la requête par le contenu visuelle d'image exemple. [01]
- **Photobook** par Pentland et al. (1996) dans the Mit Media Lab: Photobook inclue FourEyes, un agent d'apprentissage interactif qui sélectionne et combine des modèles basés sur des exemples tirés de l'utilisateur. Cela rend Photobook différent par rapport à d'autre outil comme SPEQ, Virage, SWIM, et CORE, qui aide à la recherche sur les différentes fonctionnalités, mais offrent peu d'aide en fait choisir un pour une tâche donnée. Foureyes, en revanche, permet aux utilisateurs de s'adresser directement à leur intention. [05]
- **CIRES (Content Based Image Retrieval System)**: il est basé sur le contenu du système de récupération d'images basé sur une combinaison de principes de vision de haut niveau et de niveau inférieur. Niveau supérieur d'analyse utilise les principes de perception organisation, d'inférence et le groupement d'extraire des informations décrivant le contenu sémantique structurale d'une image. Niveau inférieur analyse utilise un modèle énergétique canal pour décrire la texture d'image et utilise des techniques d'histogrammes couleur. Les filtres de Gabor sont utilisés pour extraire les énergies fractionnaires dans différents canaux de fréquence spatiale. Le système est capable de gérer les requêtes allant de scènes d'objets purement naturels tels que la végétation, les arbres, le ciel, etc... à des images contenant des objets visibles structurels tels que des bâtiments, des tours, des ponts, etc. [11]
- **RetrievalWare** par James (1993), **SIMPLIcity** (Stanford University), Chabot par Ogle et Stonebraker (1995), **VisualSEEk** par Smith et Chang (1996), **ARS** (Multimedia Analysis and Retrieval System), par Mehrotra et al. (1997) et bien d'autre systèmes.

5. Les Systèmes de recherche et d'indexation de vidéos par le contenu :

- **Blinkx** : est l'un des plus grands moteurs de recherche vidéo sur le web. Il totalise plus de 35 millions d'heures de vidéos en libre provenant de Google Vidéo, Youtube, MySpace, Metacafé et pleins d'autres.
source : <http://www.blinkx.com>

- **Totalvod** : est un moteur de recherche vidéo recensant les séries, les émissions TV, les journaux télévisés, les reportages provenant des grands sites médias français. source : <http://www.totalvod.com>
- **Fooooo** : est un moteur de recherche vidéo recensant plus de 300 millions de vidéos sur 120 sites de partage de vidéos. L'interface est disponible en plusieurs langues dont le français. Très complet. source : <http://fooooo.com>

6. Conclusion :

Plusieurs travaux ont été menés dans ce domaine, plusieurs aspects, méthodologies et techniques sont employés et se montrent en concurrence pour améliorer l'efficacité de la recherche, pour réduire le temps, et avoir des bons résultats.

Chapitre 2 : l'indexation

1. Introduction :

Dans ce chapitre, on va passer par quelques notions de base, en parlant de la recherche, l'indexation des images et des vidéos, et citant quelques descripteurs utilisés dans les domaines de la reconnaissance par le contenu.

2. Indexation et recherche :

Le but de l'indexation des données en général, que ce soit des données textuelles ou audiovisuelles (image, sons et vidéos) est de faciliter l'interrogation qui se fait à travers des requêtes de plusieurs types qu'on peut citer les plus connues :

- a) **Parcours au hasard** : La base est parcourue aléatoirement jusqu'à ce que l'utilisateur trouve la donnée qui l'intéresse.
- b) **Navigation par catégorie** : Les données de la base sont classées par catégories. L'utilisateur peut donc directement choisir la catégorie dans laquelle il pense pouvoir la trouver.
- c) **Recherche par mots clés** : L'utilisateur entre des mots censés représenter la donnée recherchée. Il dispose souvent d'une série de termes prédéfinis pour formuler sa requête.
- d) **Recherche par l'exemple** : Lors d'une telle requête, l'utilisateur fournit une donnée exemple et le logiciel recherche dans la base les données qui ressemblent à nos données exemple.

Le schéma ci-dessous montre bien, comment une requête interroge des documents indexés dans un système d'information [03].

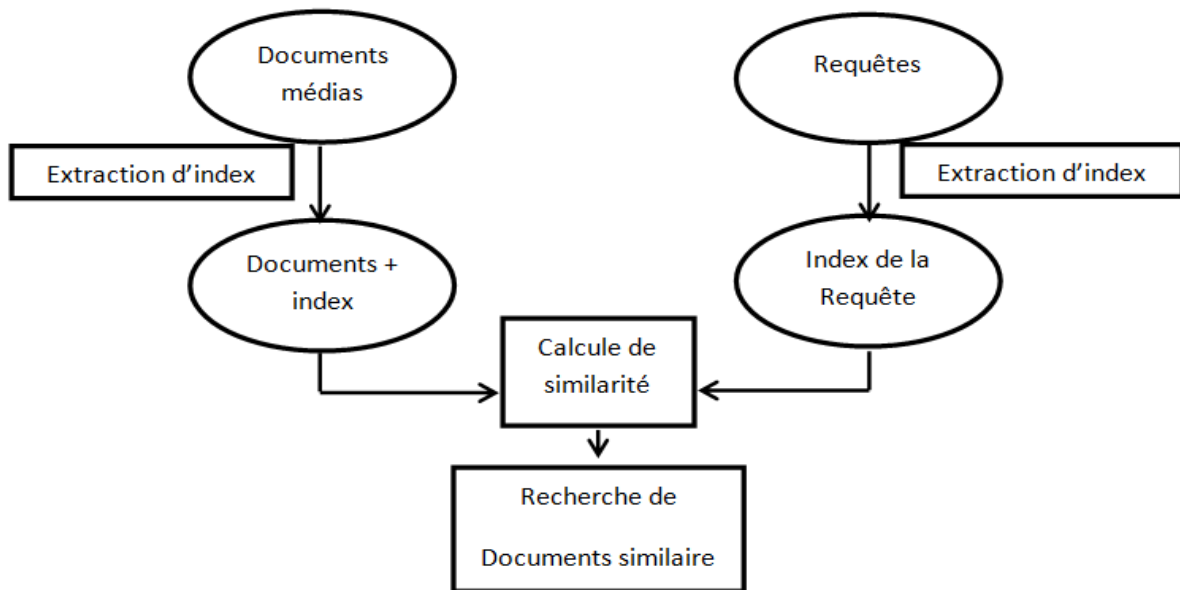


Figure 2: Model général d'un système d'indexation et de recherche d'information

3. Indexation des Images :

3.1. Image numérique :

Une image numérique est constituée d'un ensemble de points appelés **pixels** (abréviation de Picture Element) pour former une image. Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image. [26]

Ces pixels sont enregistrés dans des fichiers, sous plusieurs formats, on distingue en fonction de leur type de compression :

a) **Les fichiers non compressés** : format de fichier dit « non destructifs », ils enregistrent chaque pixel de l'image, en utilisant beaucoup d'espace mémoire. de part leur poids est élevé, pas très adapté pour le web mais elles sont utilisées lorsqu'on a besoin de garder la totalité des informations de l'image, pour retravailler dessus :

- **PSD** : format natif de Photoshop, la couleur peut être codée sur 8 ,12 , 16 ,24 ou 32 bits en noir et blanc ,RVB et CMJN.
- **BMP** : format natif de Windows, il permet d'enregistré des images bit map en 8,12et 24 bits en mode RVB.

Encore les fichiers de format **.TIFF** ; pour stocker des images de haute qualité, et **.RAW** format brut qui code les images avec le maximum d'information.

b) Les fichiers compressés : ce sont les formats de fichiers dit « destructif », il permet selon un algorithme particulier de gagner plus ou moins de mémoire, en supprimant certaines informations peu ou non perceptibles par l'œil humain. Souvent pour l'archivage ou la visualisation dans le web.

- **JPG :** Norme de compression pour les images fixes ; Elle donne la possibilité de sélectionner le taux de compression en fonction du niveau de restitution recherché (qualité réglable sur une échelle de 0 à 12). Elle supprime les informations redondantes et les détails fins. Fonctionne en 8 bit/pixel en RVB ou CMJN.
- **GIF :** C'est un format léger qui peut également contenir des animations. Une image GIF ne peut contenir que 2, 4, 8, 16, 32, 64, 128 ou 256 couleurs parmi 16,8 millions dans sa palette en mode RVB. Elle supporte également une couleur de transparence.
- **PNG :** il permet de stocker des images en noir et blanc (jusqu'à 16 bits par pixels), en

Couleurs réelles (True color, jusqu'à 48 bits par pixels) ainsi que des images indexées, faisant usage d'une palette de 256 couleurs. Il offre enfin une couche alpha de 256 niveaux pour la transparence.

L'indexation d'image est rendue possible, grâce à la fusion de la base de données et le traitement d'image, la recherche textuelle est dépassée, les utilisateurs s'intéressent beaucoup plus à une recherche qui vise directement le contenu.

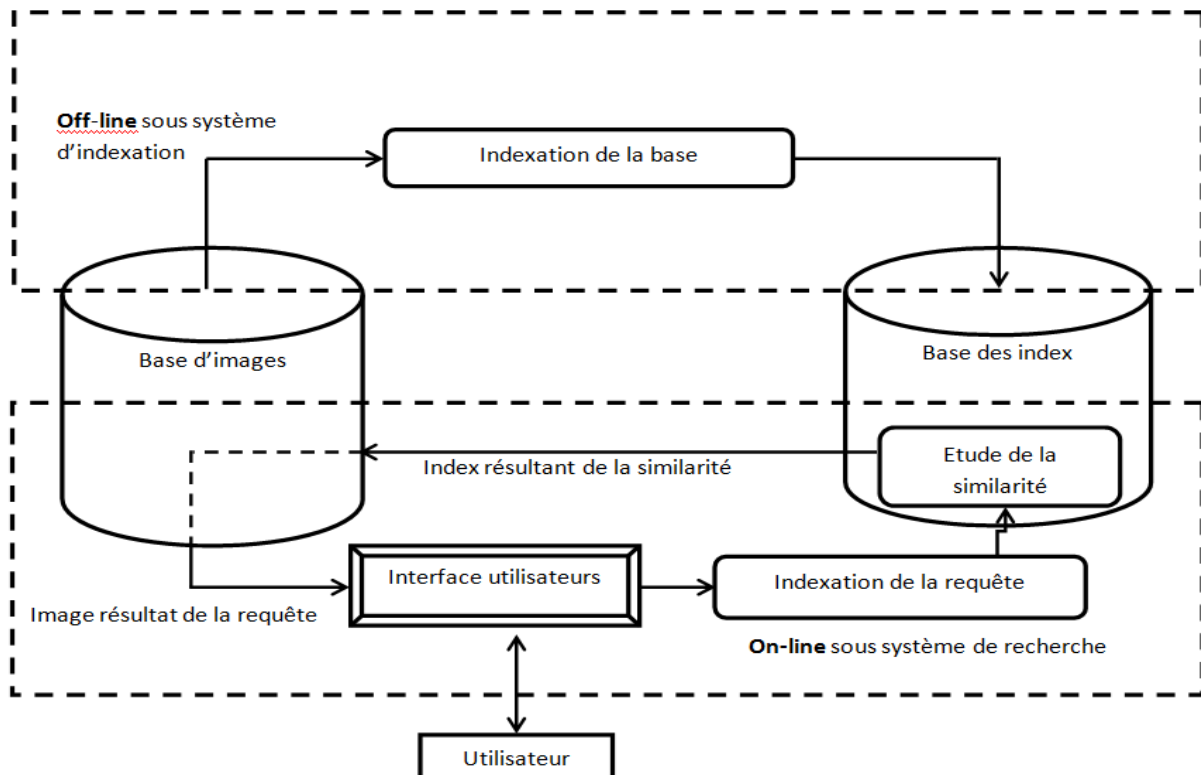


Figure 3: L'architecture d'un système d'indexation et de recherche d'images

La figure 3 montre un exemple d'un système qui s'exécute en deux étapes : l'étape d'indexation et l'étape de recherche.

Dans l'étape d'indexation, des caractéristiques sont automatiquement extraites à partir de la base d'image et stockées dans un vecteur numérique appelé en chaque descripteur visuel. Grâce aux techniques de la base de données, on peut stocker ces caractéristiques et les récupérer rapidement et efficacement.

Dans l'étape de recherche, le système prend une ou des requêtes de l'utilisateur et lui donne le résultat qui correspond à une liste d'images ordonnées en fonction de la similarité entre leur descripteur visuel et celui de l'image requête en utilisant une mesure de distance. Néanmoins, Les applications citées en avant ont bien montré leur efficacité dans ce domaine, plusieurs descripteurs et des mesures de similarités font actuellement l'objet de plusieurs systèmes d'indexation multimédia.

3.2. Les 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 et indirectement le contenu qu'ils représentent.

Nous nous intéressons dans ce manuscrit uniquement aux descripteurs visuels.

Dans les descripteurs visuels on peut distinguer :

3.2.1. Couleur :

C'est parce que la couleur est la plus perceptuel, certainement c'est le descripteur visuel le plus employé, Les grands problèmes soulevés par le choix de bons descripteurs de couleurs sont l'identification de l'espace de couleur le plus discriminant, la prise en compte des problèmes d'invariance aux conditions d'illumination et de prise de vue, la combinaison avec des descripteurs complémentaires (textures, formes...). Il existe plusieurs espaces colorimétriques qui ont chacun certaines caractéristiques intéressantes.

Un choix colorimétrique orienté matériel ; L'espace **RVB** est très simple à utiliser, car c'est celui employé par de nombreux appareils de capture d'images qui effectuent leurs échanges d'informations uniquement en utilisant les triplets (R, V, B) Cependant, ces trois composantes sont fortement corrélées (par exemple, si l'on diminue la composante verte, la teinte paraît plus rouge), l'espace RVB est sensible aux changements d'illumination, et ne correspond pas au processus de perception humaine.[04]

Autre choix matériels :

- **YIQ:** (Y représente la luminance, I la phase et Q la quadrature) Recodage du RGB par NTSC (National Television Standards Committee)
- **YUV:** (Y, représente la luminance et les deux autres, U et V, représentent la chrominance),utilisé dans le système de diffusion télévisuellePAL et SECAM des standards de la télévision couleur.
- **CMY** (cyan, magenta, yellow) **et** **CMYK**(cyan, magenta, yellow, key) (imprimantes couleurs à impact, et à jet d'encre)

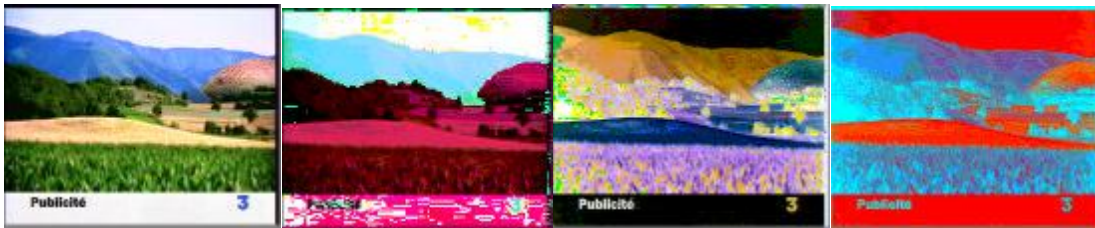


Figure 4: La même image en RGB, YIQ, YUV, CMY.

L'espace **HSV** (*Hue-Saturation-Value*) (aussi connu sous le nom de système de cône hexagonal) sépare les informations relatives à la teinte (*Hue*), la saturation (*Saturation*) et l'intensité (*Value*). Cet espace est plus intuitif à utiliser car il correspond à la façon dont nous percevons les couleurs. La teinte décrit la couleur (rouge, vert...), la saturation décrit l'intensité de la couleur, et la valeur décrit la luminosité de la couleur. La composante H de l'espace HSV offre une certaine invariance. Une étude récente compare six espaces colorimétriques et montre que l'espace HSV est le plus efficace pour la recherche d'images par le contenu, cependant cet espace n'est pas perpétuellement uniforme. [08]

Aussi HLS (*hue, saturation, and lightness*) est synonyme de la teinte, la saturation et la luminosité.

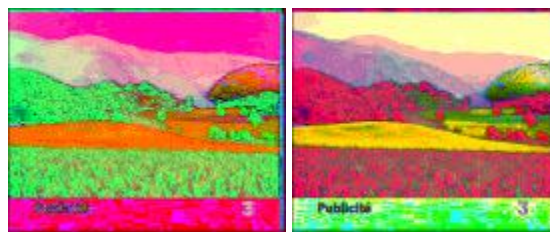


Figure 5: La même image en HSV, et HLS

Un espace colorimétrique est dite perpétuellement uniforme s'il vérifie les deux critères suivant :

- La distance $\delta(c1, c2)$ entre deux couleurs $c1$ et $c2$ est correcte si seulement si cette distance se rapproche de la différence perçue par l'œil humain ;

- La distance $\delta(c1, c2) = n * \delta'(ci, c2)$ est correcte, si et seulement si l'œil humain perçoit la couleur $c1n$ fois plus éloignée de la couleur ci que la couleur $c2$.

- **Histogramme :**

Une technique très utilisée pour la couleur et l'intersection d'histogrammes (Swain, 1991). Les histogrammes sont faciles et rapides à calculer, robustes à la rotation et à la translation.

L'histogramme d'une image est la fonction qui associe à chaque valeur d'intensité le nombre de pixels dans l'image ayant cette valeur.

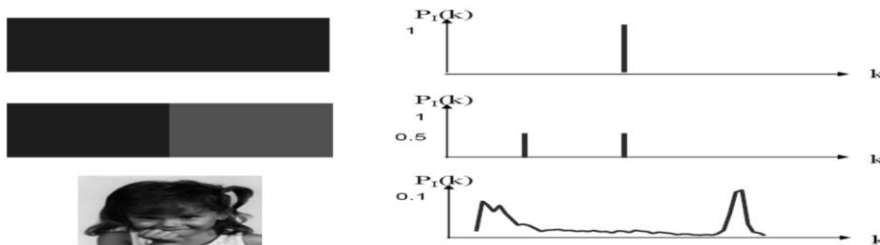


Figure 6: Exemple d'histogramme

- **Histogramme Couleur :**

Consiste à calculer la répartition statistique des couleurs présentes dans l'image. Pour une image I de taille $M \times N$, l'histogramme $h(c)$ est défini comme :

$$h(c) = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \delta(f(i, j) - c) \forall c \in C$$

- C : L'espace de couleurs possible
- c : La fréquence de l'apparition de la couleur c dans l'image
- δ : Le symbole de Kronecker. $\delta(x ; y) = 1$ si $x = y$, et 0 sinon.

Ce pendant l'histogramme pose des problèmes selon (Gong, 1998), sa grande taille cause un manque d'efficacité et la difficulté de créer une indexation rapide, l'absence de l'information spatiales sur les positions des couleurs, sensible au changement de luminosité et contraste. Aussi ils sont inutilisables pour la comparaison partielle des images.



Figure 7: Problème d'histogramme au contraste et luminosité

Mais ces problèmes ont été corrigé par la suite : enrichir ces histogrammes par des informations concernant la répartition géométrique des couleurs dans l'image, proposé par Pass et al [10], l'astuce c'est de subdivisé l'image et calculé leurs histogrammes, deux méthodes de cumul d'histogrammes locaux sont étudiées ici :

- Histogramme cumulé multiplicatif est définit par :

$$h(c) = \prod_{i=1}^N (1 + h_i(c)) \forall c \in \mathcal{C}, \sum_{i=1}^N h_i(c) = 1$$

- N est le nombre de fenêtres
- h_i est histogramme de la i eme fenêtre
- $1+h_i(c)$ terme de pondération, une valeur élevée de pondération veut dire :

Une grande présence de la couleur c dans la i eme fenêtre et par conséquent, Une forte contribution de c dans l'histogramme cumulé multiplicatif.

- Histogramme cumulé additif est définit par :

$$h(c) = \sum_{i=1}^N g(h_i(c)) \forall c \in \mathcal{C}$$

- g fonction de pondération de la présence local des couleurs.
- $g(h_i(c))$ est grand, implique l'importance de la contribution de c dans l'histogramme.
- une étude portant sur le choix de g dans [11], on choisi $g(x) = x^{\frac{1}{2}}$

3.2.2. La texture :

Dans le domaine de traitement d'images, la texture est une autre primitive visuelle, qui est étudiée depuis une vingtaine d'années et plusieurs techniques ont été développées pour l'analyser.

Il n'existe pas une définition satisfaisante de la texture, mais on va dire : C'est un champ de l'image qui apparaît comme un domaine cohérent et homogène.

Analyser une texture signifie chercher et quantifier les corrélations et relations entre les niveaux de gris des pixels en fonction de la distance qui les sépare.

Le traitement des textures se fait par une décomposition de la totalité en plusieurs élémentaires, pas mal de fonctions qui traitent cela, sous différents niveaux [07].

Le bas niveau tel que Fourier (décomposition à l'aide de sinus et cosinus), Ondelettes (utilisation de fonctions «localisées» (espace de Besov)), Autres espaces fonctionnels (Sobolev,...) etc.

Et le haut niveau, algorithme de Y.Meyer et l'approche de JF.ajol, etc...

Les caractéristiques de texture : la notion d'échelle et la répétition de motifs.

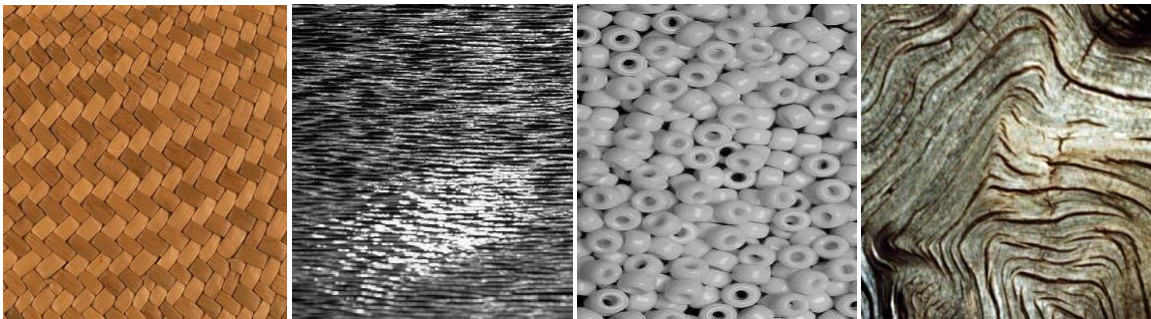


Figure 8: Exemple de texture

3.2.3. La forme :

La forme est une autre caractéristique visuelle importante, qui est toutefois considéré comme une tâche difficile d'être entièrement automatisé. Pour extraire des caractéristiques de forme d'un objet visuel ou d'une région, d'une part, les techniques de segmentation d'images sont nécessaires, puis une description géométrique des objets segmentés ou les régions sont appliquées. Dans de nombreux cas, surtout quand une détection précise est nécessaire, une intervention humaine est nécessaire. Un descripteur de forme vise à quantifier la forme d'un

objet dans le chemin de la perception humaine. Il existe plusieurs techniques pour caractériser les formes, comme périmètre, surface, zone de délimitation, la coque convexe, etc.[21].

Après avoir mentionné les descripteurs visuels, et on se basant sur l'histogramme couleur, on va citer dans ce qui vient, trois distances de similarité qu'on va utiliser dans notre approche.

3.3. Les mesures de similarité :

3.3.1. Définition :

En mathématique et en informatique la similarité est un critère important pour l'identification de sous-groupe dans un groupe d'objets, de valeurs (numériques ou non), de données (connus ou reconnus) dans un « *espace* » ou système.

La Mesure de similarité joue un rôle important dans la récupération. Une trame de la requête est accordée à un système qui récupère les vidéos similaires de la base de données. La distance (métrique) peut être qualifiée comme mesure de similarité, qui est la clef de composant dans le contenu en fonction de récupération de la vidéo. Dans l'extraction conventionnelle, les distances euclidiennes entre la base de données et la requête sont calculées et utilisées pour le classement. La trame de la requête est plus proche de la trame de la base de données, si la distance est plus petite. Si X et Y sont des vecteurs de caractéristique 2D de trame d'index de la base de données et la trame de requête, respectivement. [26]

Pour comparer deux histogrammes, nous devons d'abord choisir une métrique (d (H1, H2)) :

3.4. Les Distances :

- **Chi-squarre** (Chi-carré) : est l'une des mesures de distance qui peuvent être utilisés pour trouver la dissemblance entre deux histogrammes. Motivé par le fait que la discrimination de texture par le système de vision humain est basé sur les statistiques de second ordre.[22]

$$d(H_1, H_2) = \sum_I \frac{(H_1(I) - H_2(I))^2}{H_{12}(I)} \quad [24]$$

- **Corrélation** : Corrélation est souvent utilisée comme un outil descriptif dans la recherche non expérimentale. On dit que deux mesures sont corrélées s'ils ont quelque chose en commun. L'intensité de la corrélation est exprimée par un nombre appelé coefficient de corrélation qui est presque toujours désigné par la

lettre. Bien que généralement appelé le coefficient de Pearson de corrélation, il a d'abord été introduite par Galton (dans un article célèbre publié en 1886) et plus tard formalisée par Karl Pearson (1896), puis par Fisher (1935)

$$d(H_1, H_2) = \frac{\sum_I (H_1(I) - \bar{H}_1)(H_2(I) - \bar{H}_2)}{\sqrt{\sum_I (H_1(I) - \bar{H}_1)^2 (H_2(I) - \bar{H}_2)^2}} [24]$$

Ou

$$\bar{H}_k = \frac{1}{N} \sum_J H_k(J) [24]$$

N : nombre des éléments de l'histogramme

- **Bhattacharya** : est une méthode graphique pour la résolution d'une distribution en composantes gaussiennes.[23] ,elle a été développée par C.G. Bhattacharya en 1967 , Il était "un biologiste des pêches" de l'Inde ,Il a développé une technique pour la résolution de la distribution des «longueurs forkal de poissons (pagres)» en «groupes d'âge» ,Il est décrit que "Poissons groupes d'âge» - peuvent être estimées à partir de leur forkal longueur. La technique a depuis été appliquée par quelques individus dans le cadre d'un laboratoire clinique - pour la création d'intervalles de référence. .[23]

$$d(H_1, H_2) = \sqrt{1 - \frac{1}{\sqrt{H_1 H_2} N^2} \sum_I \sqrt{H_1(I) \cdot H_2(I)}} [24]$$

3.5. Le seuil :

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

4. Indexation des vidéos

4.2. Un document vidéo :

Un document vidéo est défini comme une combinaison de différents flux de données, où visuelle et audio sont les deux principaux flux de données. Le flux visuel se compose d'une séquence d'images qui sont animées, en général, dans 24 à 30 images par seconde. Le flux audio est un mélange de différents sons (par exemple, mono, stéréo), qui

sont, en général, un échantillonnage de 1600 à 48000 k hertz. Un troisième flux qui peut être associé ; est le texte, Il peut apparaître dans des documents vidéo pour aider les utilisateurs comprendre la présentation audiovisuelle.[21]

L'objectif d'un système d'indexation de vidéo est de permettre à un utilisateur d'effectuer une recherche dans un ensemble de vidéos. La recherche est conduite sur des critères facile à obtenir comme le réalisateur, le type d'émission, les acteurs ou le thème, mais il est extrêmement plus compliqué ou long d'obtenir une description précise du contenu des vidéos. Une indexation vidéo par le contenu est schématisée dans la figure suivante :

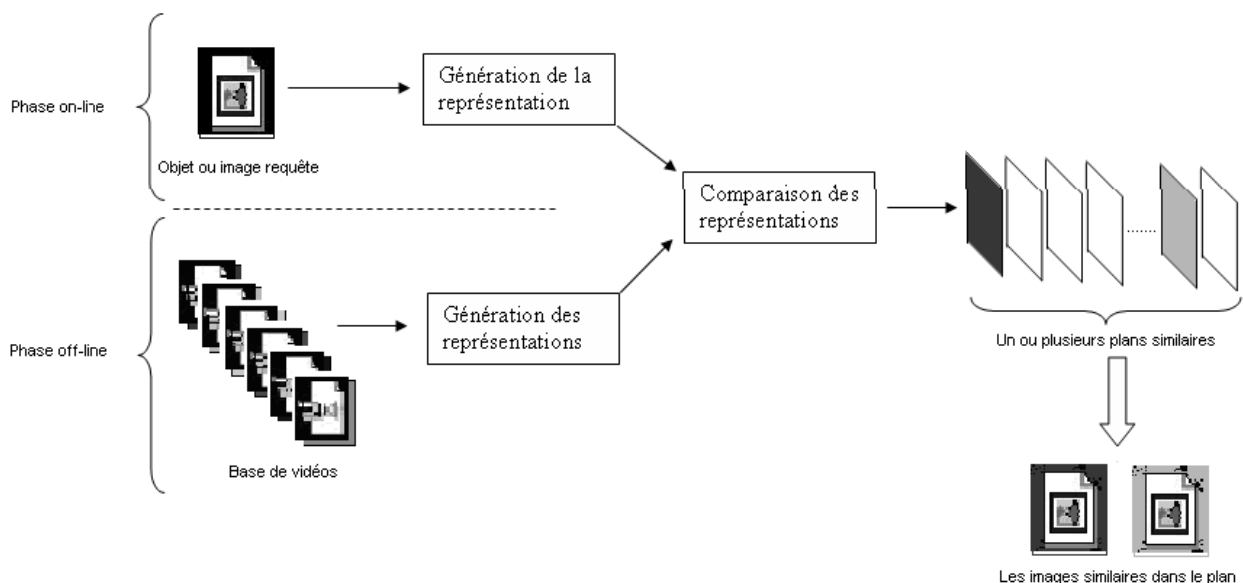


Figure 9: Architecture d'un système d'indexation et de recherche de vidéos par le contenu

4.3. Indexation de bas niveau :

Des descripteurs de bas niveau, tels que la couleur, la texture, la forme, et le mouvement, peuvent être associés aux plans ou aux objets. La couleur des images choisies peut être décrite par l'histogramme de couleur ou par les couleurs dominantes [06]. Les paramètres de mouvement de camera et le taux d'activité décrivent le mouvement au niveau du plan [06]. Le mouvement des objets peut être décrit par des trajectoires. Les sommaires d'images clés et les sommaires de segments importants sont généralement employés dans des applications commerciales. Les images clés, qui se rapportent à une ou plusieurs images représentatives dans un plan, fournissent une représentation compacte.

Plusieurs méthodes existent pour choisir automatiquement les images clés par l'analyse de caractéristiques de bas niveau [06].

Le but c'est de reprendre au mieux les caractéristiques essentielles de ces vidéos dans un volume minimal, pour extraire les images clef on doit passer par les étapes suivantes :

- Découpage en images : il s'agit dans cette étape de découper une vidéo en une série d'images individuelles, ces images nous permettront de définir par la suite des plans d'images.



Figure 10: Découpage en série d'image

- Application d'un descripteur visuel des images (défini précédemment).
- Calculer une mesure de similarité correspondante au descripteur appliqué dans l'étape précédente.
- Segmentation temporelle : plusieurs techniques ont été proposées pour segmenter une vidéo en plusieurs unités de base appelées "plans", nous pouvons citer :
 - Différence d'histogrammes
 - Différence de blocs

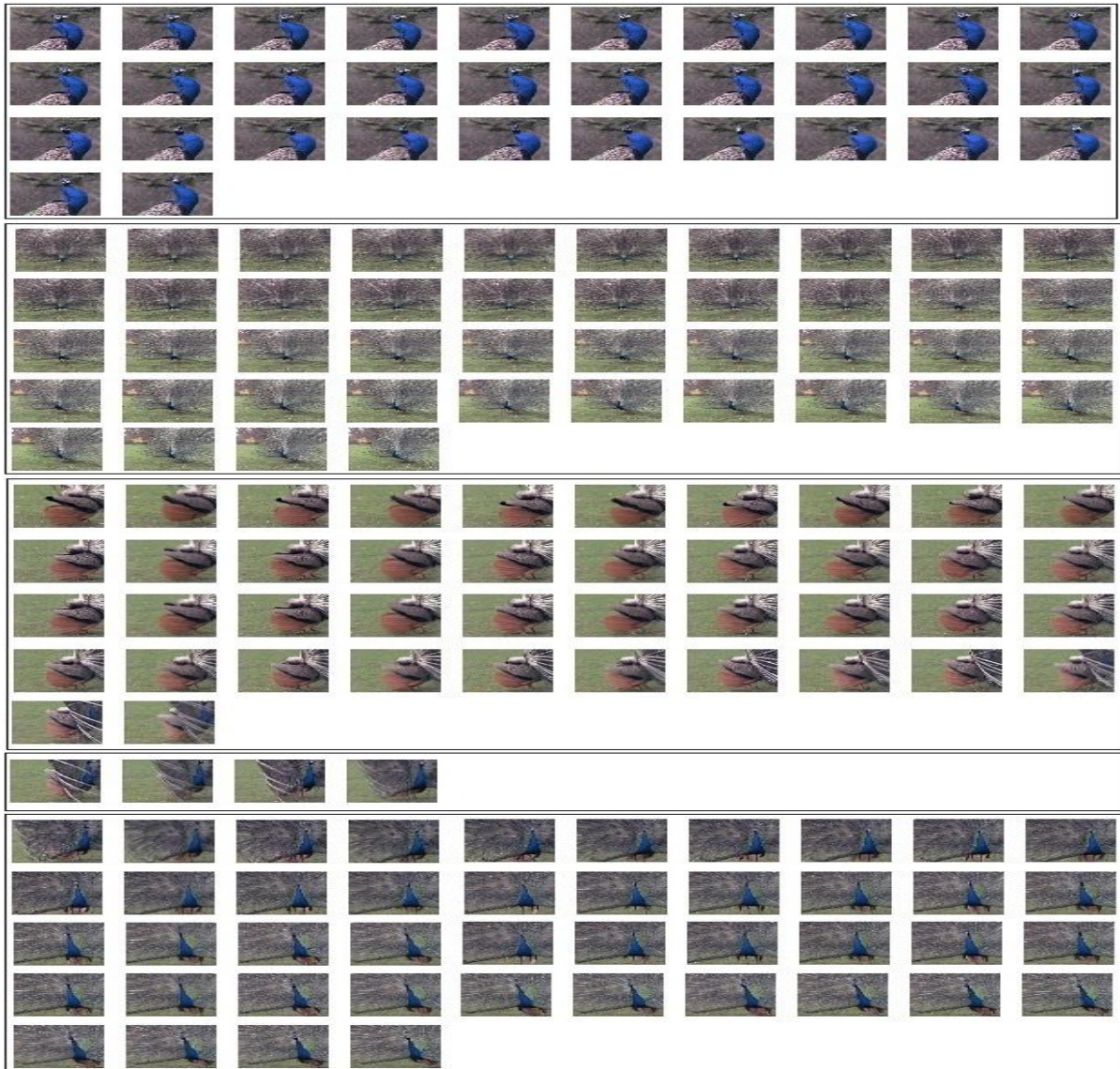


Figure 11: Segmentation temporelle (en plan)

- Sélection de l'image clef : nous devons extraire dans cette étape les caractéristiques visuelles de chaque plan, ces caractéristiques sont définies dans une ou plusieurs images appelées "images clefs". L'image clef de notre approche est la première image de chaque plan car ce dernier contient les images similaires à cette image.



Figure 12: Résumé vidéo (image clé 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 "résumé vidéo".

4.4. Segmentation Temporelle :

La segmentation temporelle est importante puisqu'elle permet de grandement limiter la quantité de données à traiter en se concentrant uniquement sur les images les plus pertinentes. D'autre part elle offre une décomposition structurée de la vidéo qui peut servir pour l'indexation.

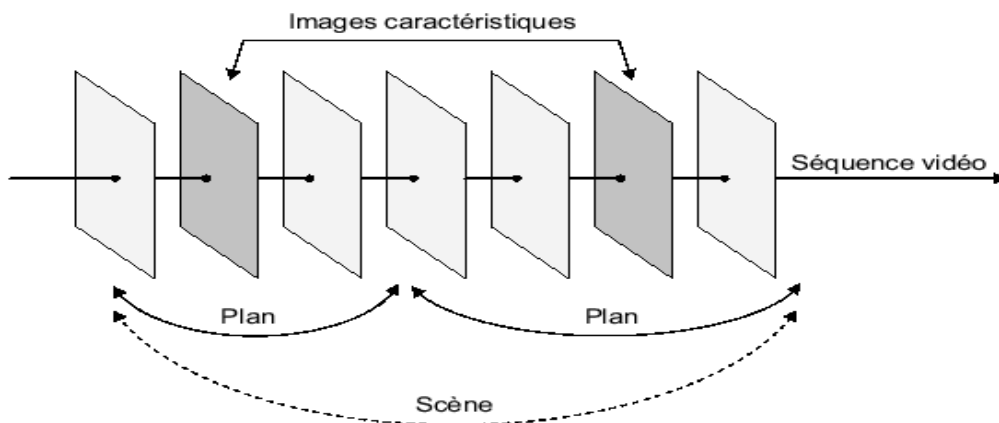


Figure 13: Illustration de la segmentation temporelle

L'unité la plus courte après l'image est le plan qui délimite deux prises de vue. La segmentation en plan peut paraître simple mais de nombreux artifices (la coupure, le fondu et la dissolution) sont utilisés pour effectuer les transitions entre les plans. Ils rendent cette tâche beaucoup moins évidente. Nous trouvons ensuite la scène qui est composée de plans étroitement liés (un dialogue, un reportage, la météo, ...). La scène est une structure sémantique c'est à dire qu'elle associe un signifié (un concept) à un signifiant (l'ensemble des plans). Malheureusement le problème de la segmentation en scènes n'est actuellement pas

correctement résolu dans le cas général et la plupart des méthodes d'indexation et de recherche de la vidéo reposent uniquement sur le découpage en plans. La segmentation temporelle est souvent associée à la sélection d'images représentatives des segments trouvés. La suite de l'indexation peut alors être conduite sur les images représentatives en conservant la plus grande partie de l'information.

La segmentation en plans est la technique de segmentation temporelle des enregistrements vidéo la plus répandue et la plus utilisée. Les méthodes de détection de changements de plan localisent les images, à travers lesquelles de grandes différences sont observées dans un certain espace de caractéristiques. L'espace de caractéristiques se compose habituellement d'une combinaison de couleur et de mouvement. Les changements de plan peuvent être instantanés (cuts) ou apparaître sur plusieurs images, appelées les effets de transition progressifs, tels que les fondus et les volets. Il est plus facile de détecter des cuts que des effets progressifs. La méthode la plus simple pour la détection des cuts est d'analyser les variations d'intensité des pixels entre les images successives. Si un nombre prédéterminé de pixels montre des différences plus grandes que la valeur du seuil, alors une occurrence d'un cut peut être déclarée.

Une approche légèrement différente consiste à diviser chaque image en blocs rectangulaires, à opérer des évaluations statistiques dans chaque bloc indépendamment, et à vérifier alors que le nombre de blocs qui ont globalement été modifiés est supérieur à un seuil

Les deux approches peuvent être sensibles au bruit et à la compression. Cependant, il existe de nombreuses solutions qui s'appliquent à la vidéo de manière générique avec une précision plus qu'acceptable.

La micro-segmentation est une segmentation temporelle à une échelle encore plus petite que celle du plan. Elle est basée sur la segmentation en événements, en mouvements de caméra, en entrée-sortie d'objets ou de personnages. Par opposition, la macro segmentation effectue une segmentation qui se rapproche de la composition sémantique des documents (segmentation en séquences, en chapitres, en programmes).

5. Résumé :

Une première étape consiste à définir les caractéristiques requises en fonction du problème. Les différentes caractéristiques pouvant être extraites afin de décrire au mieux le contenu, peuvent être regroupées principalement dans trois catégories primaires : les caractéristiques de couleur, texture et structure. Nous retrouvons alors les histogrammes de couleur, les corrélogrammes de couleur, les filtres de Gabor, les matrices de cooccurrence, la décomposition de Wold, etc... La deuxième étape consiste à définir les régions de l'image dans lesquelles les données vont être extraites. Selon l'application, il sera suffisant de décrire l'image comme une unique entité. D'autres applications plus complexes peuvent nécessiter une description locale du contenu faisant appel à une détection des régions ou des objets. Des compromis doivent être établis entre l'objectif de l'application, la complexité de la représentation et les temps de calcul requis pour le traitement des données.

6. Conclusion :

L'idée de l'indexation et la recherche de vidéo par le contenu, est la succession de l'indexation image, des contraintes de plus qu'il faut prendre en considération, l'espace mémoire et le temps.

Chapitre 3 : Conception et mise en œuvre

1. Introduction :

Dans ce chapitre, nous allons présenter la partie mise en œuvre. Nous allons décrire successivement notre approche, les environnements matériels et logiciel utilisés ainsi que la plateforme développée.

2. Outil et environnement :

2.1. Visual studio 2010 :

Depuis les premières versions de Visual Studio, Microsoft tente de répondre aux différents besoins des professionnels de l'informatique en proposant différentes versions de l'IDE. Visual Studio n'est évidemment pas uniquement un éditeur de code destiné aux développeurs, c'est une plateforme où les différents intervenants (développeurs, chefs de projets, testeurs, etc.) en charge d'un projet logiciel vont pouvoir collaborer tout au long du développement d'un projet. [25]

2.2. C++ :

Le C++ est un langage de programmation qui permet la programmation sous de multiples paradigmes comme la programmation procédurale, la programmation orientée objet et la programmation générique. C++ est actuellement le 2eme langage le plus utilise dans le monde. Le langage C++ n'appartient a personne et par conséquent n'importe qui peut l'utiliser sans avoir besoin d'une autorisation ou obligation de payer pour avoir le droit d'utilisation. Nous avons choisi ce langage parce qu'il est très rapide en exécution (assez bas niveau) et portable.

2.3. My sql server :

MySQL est la base de données open source la plus populaire au monde. Bien qu'elle soit avant tout connue pour son utilisation par des sociétés Web, telles que Google, Facebook

et Yahoo!, MySQL est également une base de données embarquée très populaire. Plus de 3000 éditeurs de logiciels et fabricants de matériel lui font confiance, parmi lesquels sept des dix plus grandes entreprises logicielles au monde.

2.4. OpenCV :

OpenCV est une bibliothèque libre de vision par ordinateur. Cette bibliothèque est écrite en C et C++ et peut être utilisée sous Linux, Windows et Mac OS X. Des interfaces ont été développées pour Python, Ruby, Matlab et autre langage. Open CV est orienté vers des applications en temps réel. Un des buts d'OpenCV est d'aider les gens à construire rapidement des applications sophistiquées de vision à l'aide d'infrastructure simple de vision par ordinateur. La bibliothèque d'OpenCV contient près de 500 fonctions. Il est possible grâce à la « licence de code ouvert » de réaliser un produit commercial en utilisant tout ou partie d'OpenCV. Il n'est pas obligatoire de montrer le code du produit et les améliorations réalisées au domaine public.

3. Notre approche :

Comme l'indexation comporte deux phases, en-ligne pour la requête et off-ligne pour l'indexation, on a pensé à appliquer un traitement qui se fait avant dans une étape précédente, qu'on a appelé par la suite le codage.

3.1. Le codage :

On résume cette phase en trois étapes, segmenter les vidéos de la base en images individuels, calculé leurs histogrammes, puis enregistrer ces données extraites dans une base de données.

Comme illustre notre algorithme de codage (pseudo langage) :

3.1.1. Algorithme de codage :

Début

Lire Nbvdiéos ;

Pour i=1 a Nbvidéos faire

{

Lire vidéo (i) // vidéo de type (.avi)

Lire Nbimage // nombre de trames pour la vidéo (i)

Pour j=1 a Nbimage faire

{


```

Lire image(j) // de type (.jpeg)
Calculer son histogramme (Hr,Hg,Hb)
    Calculer son histogramme (Hr,Hg,Hb)
    Indexer l'histogramme sous format Xml et le stocker dans un fichier
    Stocker l'image(j) dans un fichier
    Stocker dans une base de données le chemin de l'image, le nom de la
vidéo (i) et le fichier Xml
}
}
Fin de faire
Fin

```

3.1.2. Extraction des images individuelles :

```

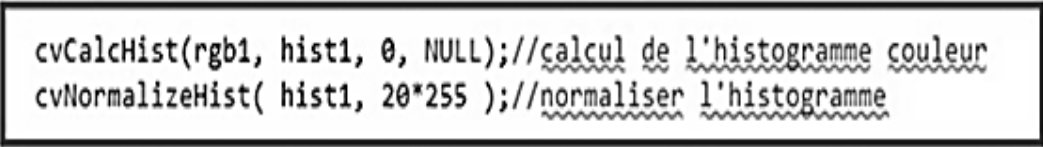
{

String^ name1="img"+j+".jpg";
String^ name2="img"+j+".xml";
String ^dest1=System::IO::Path::Combine(folder1,name1);
String ^dest2=System::IO::Path::Combine(folder2,name2);
const char* fileName1 =(char*)(void*)Marshal::StringToHGlobalAnsi(dest1);

```

Figure 14: Instruction qui fait l'extraction des images individuelles.

Grace à cette fonction de la bibliothèque open cv, l'extraction des frames (images) est facile, Les images individuelles seront ensuite stockées dans un dossier.

```

cvCvtColor(IplImage(img1, rgb1[2], rgb1[1], rgb1[0], NULL);
hist1 = cvCreateHist(3, taillehist, CV_HIST_ARRAY, NULL, 1);

cvSave(fileName2,hist1->bins); //enregistrer l'histogramme dans le dossier XML
cvSaveImage(fileName1,img1); //Enregistrer l'image dans le dossier Images

```

Figure 15: l'histogramme normalisé d'une image.

3.1.3. Enregistrement dans la base de données :

```
cvCalcHist(img01, hist1, 0, NULL); //calcul de l'histogramme couleur
cvNormalizeHist( hist1, 20*255 ); //normaliser l'histogramme
```

```
cvSave(fileName2, hist1->bins); //enregistrer l'histogramme dans le dossier XML
cvSaveImage(fileName1, img1); //Enregistrer l'image dans le dossier Images
```

```
command = gcnew MySqlCommand("INSERT INTO data(dossier,video,image,histogramme) VALUES
```

Figure 16: Enregistrement des images et les histogrammes

Il s'agit d'enregistrer les images individuelles dans un dossier « Image » et les histogrammes dans un autre dossier « XML », Ensuite enregistrer dans la base de données le chemin de chaque vidéo et ses images individuelles et le chemin de l'histogramme qui correspond à chaque image individuelle.

```
//enregistement dans la base de donnée
command = gcnew MySqlCommand("INSERT INTO data(dossier,video,image,histogramme) VALUES('+file333+', '+file3+'
connection->Open();
Reader = command->ExecuteReader();
connection->Close(); //effectuer une requête pour enregistrer le chemin de la vidéo ainsi les images individuels
```

Figure 17: la partie connexion avec la base de données.

3.2. La recherche :

C'est la deuxième partie de notre approche, la partie où on répond à une requête image. Donc notre recherche sera de la manière suivante :

3.2.1. Algorithme de recherche :

Début

{

```

Lire l'image requête
Lire le seuil
Lire le type de distance choisie
Calculer l'histogramme de l'image requête (H1r, H1g, H1b)
Normaliser l'histogramme (H1r, H1g, H1b)
Lire le contenu de la base de données
Tant qu'il existe des enregistrements dans la base de données faire
Tant i=1 au dernier enregistrement dans la base de données faire
{
    Lire l'ième histogramme de l'enregistrement (H2r, H2g, H12)
    Récupérer toutes les vidéos dont la distance d (h1, h2) < seuil
}
Fin de faire
}

```

3.2.2. Calcul de l'histogramme couleur et l'histogramme normalisé de l'image requête :

```

rgb1[0] = cvCreateImage(cvGetSize(img1), IPL_DEPTH_8U, 1); //définir l'espace RBV
rgb1[1] = cvCreateImage(cvGetSize(img1), IPL_DEPTH_8U, 1); //définir l'espace RBV
rgb1[2] = cvCreateImage(cvGetSize(img1), IPL_DEPTH_8U, 1); //définir l'espace RBV
cvCvtPixToPlane(img1, rgb1[2], rgb1[1], rgb1[0], NULL);
hist1 = cvCreateHist(3, taillehist, CV_HIST_ARRAY, NULL, 1); // créer la variable qui va contenir l'histogramme couleur
cvCalcHist(rgb1, hist1, 0, NULL); //calculer l'histogramme couleur de l'image requête
cvNormalizeHist( hist1, 20*255 ); // normaliser l'histogramme

```

1

Figure 18: Calcul de l'histogramme et le normaliser

Dans cette phase l'ors du chargement de l'image requête l'histogramme couleur sera calculé et normalisé.

3.2.3. Calcul de distance de similarité entre l'histogramme de la requête et ceux qui sont stocker dans la base :

C'est là où la phase de recherche intervient ; il s'agit de comparer l'histogramme de l'image requête avec ceux qui sont stockés dans la base en utilisant une de trois distance (batcharya, chi-sqaure, correlation) et selon le seuil définit par l'utilisateur.

```
String^ histo=Reader->GetString("histogramme");// récupérer le chemin de l'histogramme
hist2->bins=bins2;//récupérer l'hitogramme apartir du cemin de la base
double i = cvCompareHist( hist1,hist2,type);//comparer l'hitogramme de l'image requête et l'hitogramme cou
```

Figure 19: Calcul de la distance

3.2.4. Récupération des vidéos résultats :

Après avoir calculé la distance de similarité on récupère les vidéo résultats(les vidéos similaires).les vidéos similaires dépendent du seuil choisit par l'utilisateur.

```
if(i<=seuil)
{
    j++;//compteur pour conaitre le nombre de vidéo similaires
    String^video=Reader->GetString("video");//récupérer vidéo similaires
    CvCapture *avi=cvCaptureFromFile((char*)(void*)Marshal::StringToHGlobalAnsi(video));
    int frameNumbers = (int) cvGetCaptureProperty(avi, CV_CAP_PROP_FRAME_COUNT);//calculer le nombre d'ima
    int fps = (int) cvGetCaptureProperty(avi, CV_CAP_PROP_FPS);//calculer la fréquence d'image
    int videoLength = frameNumbers / fps;//calculer la longueur de la vidéo
    cmdDataBase1=gcnew MySqlCommand("insert into open1 values("+j+", "+System::IO::Path::GetFileName(vide
    connection1->Open();
    Reader1=cmdDataBase1->ExecuteReader();//insérer la vidéo similaire dans la base de données
    connection1->Close();
    cvReleaseCapture(&avi);
}
```

Figure 20: la partie récupération

3.3. Analyse :

Coût en mémoire vive Rom vs coût Ram+ le temps :

Cette méthode est couteuse au niveau du mémoire du système (disque dur) lors de la phase de codage par exemple une vidéo de longueur 40 min avec 25 trame/s on aura 60000 images et si la taille de chaque image est environ 20 ko on aura un espace de 1go.

Temps de recherche rapide car les calculs qui sont fait dans la partie codage se font une seule fois seulement.

4. Application :

4.1. Interface utilisateur (Première fenêtre) :

Dans La première fenêtre de l'application, trois onglets sont disponible, **Codage**, **Recherche** et **Aide** (quelques informations utile), les deux premiers comportent des fonctionnalités importantes et nécessaires pour la bonne manipulation de l'application, successivement et dans l'ordre.



Figure 21: Première interface de l'application

4.1.1. L'onglet codage (partie indexation) :

Cette partie se compose de :

- **Nouveau** : ce bouton permet la réinitialisation de l'application, une remise à zéro, et revenir à la première interface.
- **Ouvrir** : ouvre une boîte de dialogue afin de charger un dossier contenant des vidéos de type **.AVI**

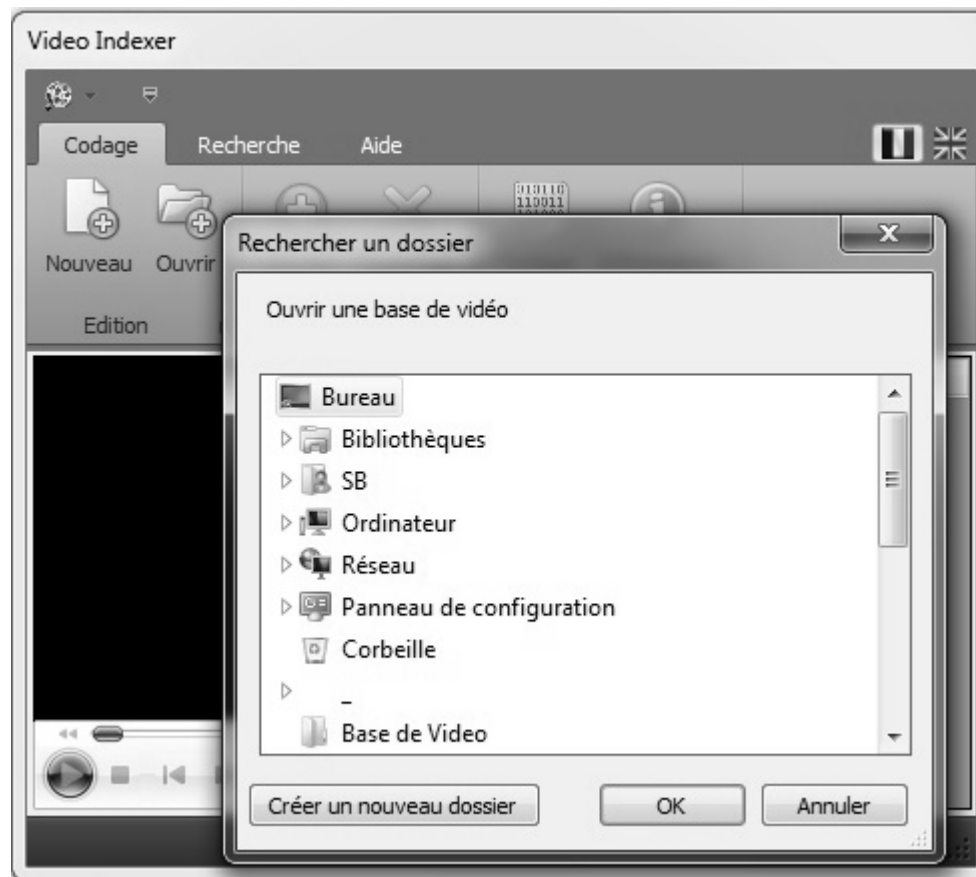


Figure 22: la fonction du bouton Ouvrir

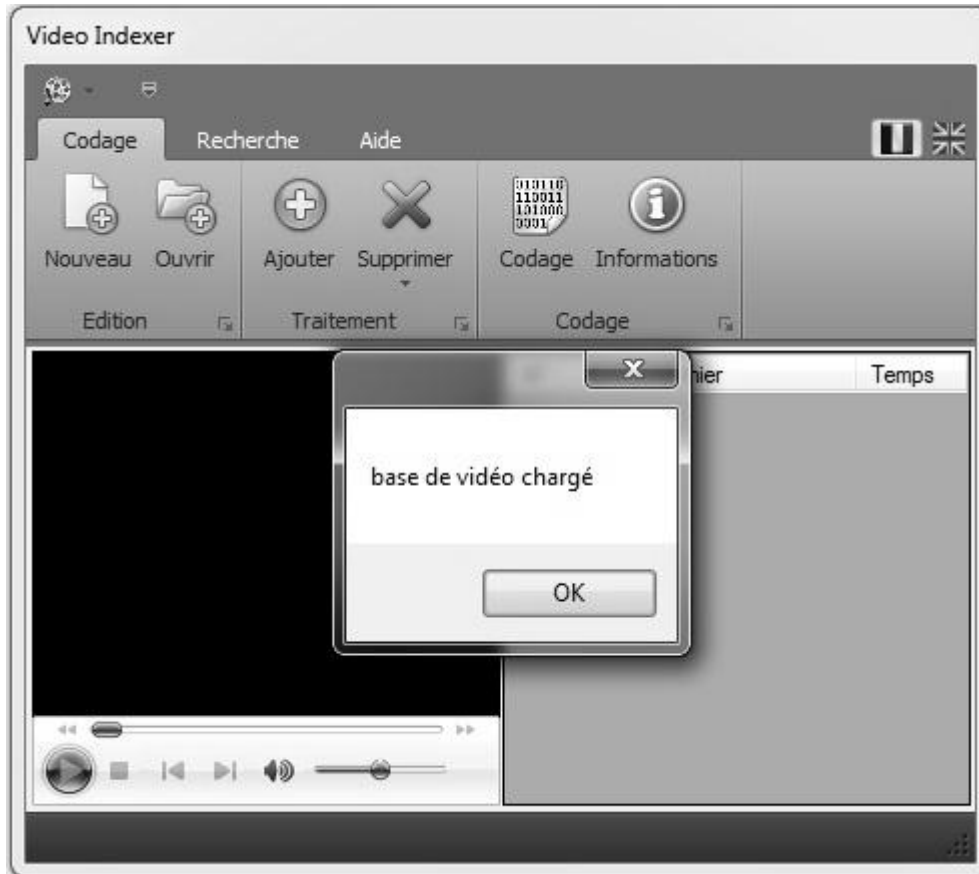


Figure 23: Dossier contenant les vidéos est chargé avec succès

- **Ajouter** : importe une vidéo dans le dossier déjà ouvert. Copy la vidéo dans ce dossier.
- **Supprimer** : supprime une vidéo sélectionnée en plus de son codage de la base de données.
- **Codage** : c'est là que le traitement de l'indexation commence ; la segmentation de la vidéo en série d'images, l'enregistrement de leur histogrammes dans la base.



Figure 24: l'interface lors du codage

En même temps pendant le codage, les images individuelles sont enregistrées dans le dossier Images.

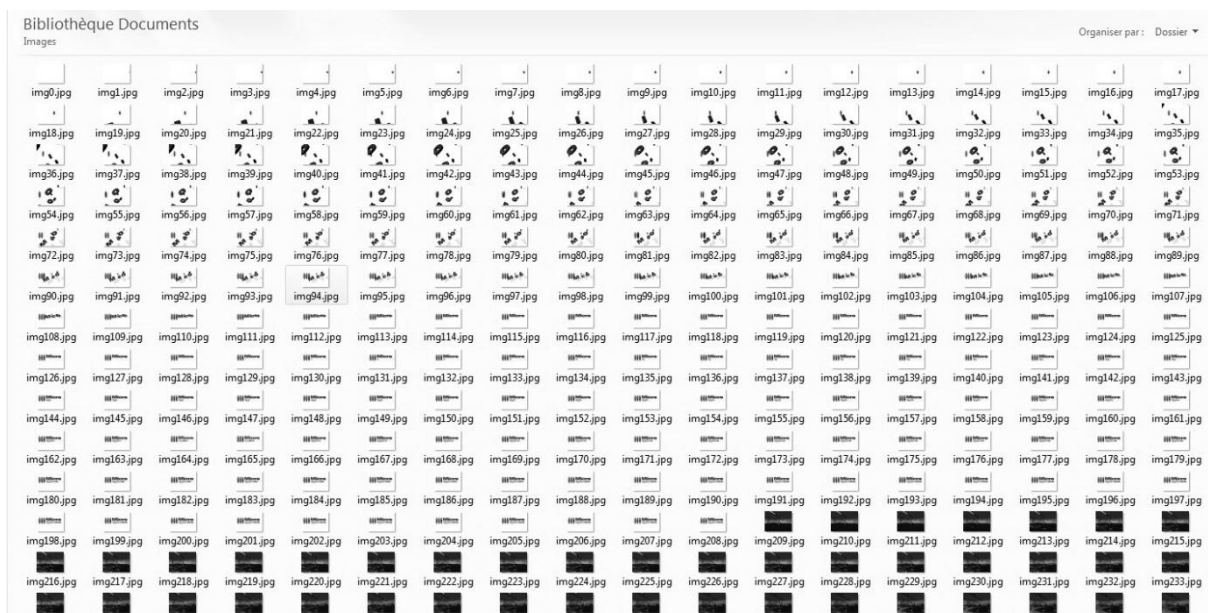


Figure 25: Enregistrement des images clés dans le dossier images

- **Information :** ce bouton permet d'avoir quelques informations sur la vidéo tel que nom, taille, longueur etc...



Figure 26: La fenêtre information.

4.1.2. L'onglet Recherche :

Cette partie représente le deuxième aspect de l'application, elle permet la recherche par le contenu, grâce à l'image requête, elle se compose de : nouveau, ouvrir, image, distance, seuil et recherche

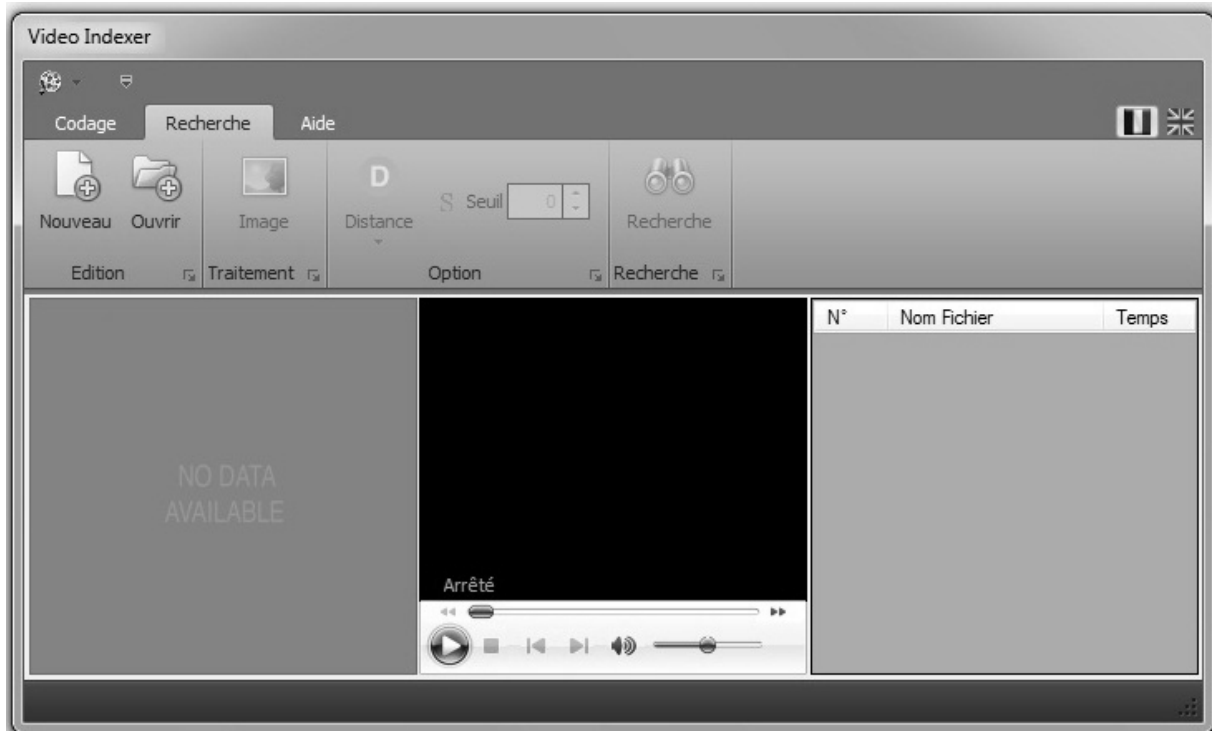


Figure 27: L'interface de l'onglet Recherche.

- **Nouveau** : ce bouton permet la réinitialisation de l'application, une remise à zéro, et revenir à la première interface.
- **Ouvrir** : ouvre une boîte de dialogue afin de charger un dossier contenant des vidéos de type **.AVI**
- **Image** : charger une image (image requête).
- **Distance** : choix de la distance utilisé pour la recherche.

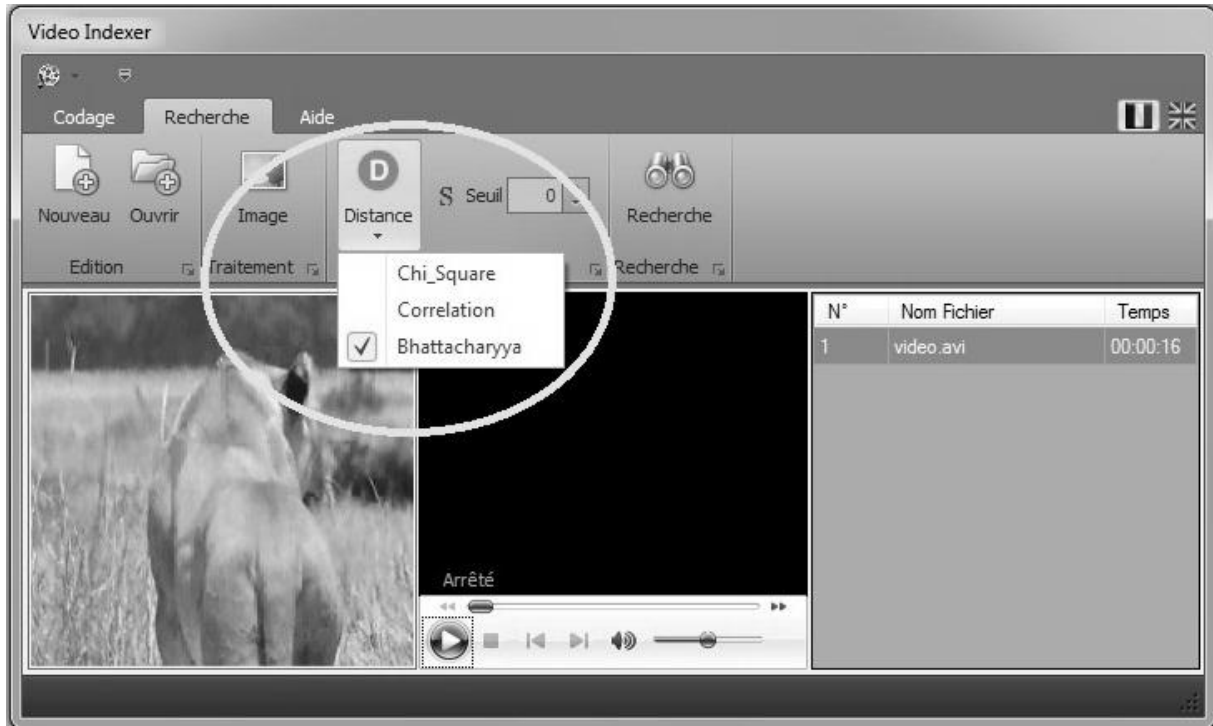


Figure 28: Les distances utilisées.

- **Seuil** : paramétrer le seuil de similarité.



Figure 29: Le paramètre du seuil.

- **Recherche** : lancer la recherche dans l'ensemble des vidéos.



Figure 30: l'application pendant le traitement de la recherche.

- 4.1.3. **L'onglet aide** : un petit site avec quelques informations sur l'application. Même l'application est disponible sur le site ou c'est possible de la télécharger. Le site : <http://jamesbond20.wix.com/video-indexer-2>

5. Mesures pour évaluer un Système

Avant l'exécution d'un système de recherche d'information, on exige la nécessité de la mesure de performance. Les mesures les plus courantes sont souvent le temps de réponses et l'espace utilisé, plus le temps de réponse est court, plus l'espace est petit, plus le système est considéré comme bon, mais dans le cas d'un système d'information plus de ces deux mesures on s'intéresse aux réponses pertinentes du système, deux mesures plus courantes dans cette section : le rappel et la précision (en anglais : Recall and Precision).

- **Le rappel** :

Le rappel est le rapport entre le nombre d'images pertinentes dans l'ensemble des images trouvées et le nombre d'images pertinentes dans la base d'images.

$$\text{Rappel} = \frac{|\text{Ra}|}{|\text{R}|}$$

- **La précision :**

La précision est le rapport entre le nombre d'images pertinentes dans l'ensemble des images trouvées et le nombre d'images trouvées.

$$\textit{Précision} = \frac{|Ra|}{|A|}$$

- R : l'ensemble des images pertinentes dans la base d'images utilisée pour évaluer.
- |R| : le nombre d'images pertinentes dans la base d'images.
- A : l'ensemble des réponses.
- |A| : le nombre d'images dans l'ensemble des réponses.
- |Ra| : le nombre d'images pertinentes dans l'ensemble des réponses.

6. Conclusion :

Dans ce chapitre nous avons présenté le prototype que nous avons implémenté avec la stratégie de segmentation temporelle exposé dans les chapitres précédents. Nous avons commencé par la description de l'environnement de développement. Puis nous avons présenté ces deux algorithmes de base de notre approche, et quelques exemples de codes qui ont servi à réaliser les formules proposées et ainsi notre prototype. Nous avons montré et commenté les résultats d'expérimentation, obtenus par les différentes stratégies de calcul de similarité. Nous avons constaté que ces résultats garantissent favorablement l'efficacité de notre approche et qu'un bon processus d'indexation vidéo est celui qui permet de trouver les correspondances les plus pertinentes avec le moindre nombre de calculs possibles, ainsi un temps réduit.

CONCLUSION GENERALE

Pour un système de recherche de vidéos intelligent, le contenu est un caractère très important, plusieurs approches ont été développées et mises en œuvre dans l'indexation et la recherche par le contenu.

Les résultats obtenus à partir de l'application, nous ont permis de juger et dire que l'histogramme est efficace pour la comparaison globale du contenu de l'image, simple et le temps de réponse est assez rapide, Ainsi de valider notre approche.

L'indexation et la recherche par le contenu est un domaine en cours de développement, même si les applications proposées sont loin d'atteindre un certain niveau de perfection, les résultats obtenus promettent un succès et des retours qui peuvent satisfaire le besoin des utilisateurs, que ce soit pour l'utilisation personnelle, ou l'utilisation professionnelle.

BIBLIOGRAPHIE

- [01] **A. Segonds ,directeur de recherche A. Gunthert**, « Indexation visuelle et recherche d'images sur le Web : Enjeux et problèmes», Ecole des hautes études en sciences sociales ,2009 ,p111 (.pdf)
- [02] **M. Campedel** , « Indexation des images »,école nationale supérieure des télécommunications, mars 2005 (.pdf)
- [03] **J.Noël Gouyet** , «Analyser-Indexer » ,DUNOD- Gestion médias numériques ,2006 . (.pdf)
- [04] **J. Marc OGIER** « Indexation, Segmentation et Recherche par le contenu » , Laboratoire L3i ,Université de La Rochelle ,DEC 2003 (.pdf)
- [05] **B.Saida** , mémoire de magister ,Thème « Recherche d'images par le contenu » ,Faculté de génie électrique et informatique , Département automatique , 2010 (.pdf)
- [06] **Siba Haidar**, « Comparaison des Documents Audiovisuels »UNIVERSITE TOULOUSE III - PAUL SABATIER,le 20 septembre 2005.
- [07]**Bertrand Collin, Jérôme Gilles, Emmanuel d'Angelo**, « traitement d'images » ,DGA/DET/CEP/ASC/GIP ,Jeudi 27 septembre 2007.
- [10] **G. Pass, R. Zahib**,« Histogram refinement for content-based image retrieval», IEEE Workshoon Application of Computer Vision, pp. 96–102, 1996
- [12] **Joo-Hwee Lim, Qi Tian et Philippe Mulhem**.« Home Photo Content Modeling for Personalized Event-Based Retrieval » . IEEE MultiMedia, vol. 10,no. 4, pages 28_37, 2003.

- [13] **Nicolas Maillot et Monique Thonnat.** « A Weakly Supervised Approach for Semantic Image Indexing and Retrieval ». In International Conference on Image and Video Retrieval (CIVR'05), volume 3568/2005, pages 629_638, 19-22 Aug 2005.
- [14] **Michel Crucianu, Marin Ferecatu et Nozha Boujemaa.** « Relevance feedback for image retrieval : a short review . In In State of the Art in Audiovisual Content-Based Retrieval, Information Universal Access and Interaction including Datamodels and Languages » ,DELOS2 Report, 2004.
- [15] **Vasileios Mezaris, Ioannis Kompatsiaris et Michael G. Strintzis.** « Region-Based Image Retrieval Using an Object Ontology and Relevance Feedback ». EURASIP Journal on Applied Signal Processing, vol. 6, pages 886_901, 2004.
- [16] **A. Pentland, R. Picard et S. Sclaro_.** « Photobook : Content-based manipulation of image databases. Proc. SPIE Storage and Retrieval for Image and Video Databases II », pages 34_47, 1994.
- [17] **Lawrence A. Rowe et Ramesh Jain.** « ACM SIGMM retreat report on future directions in multimedia research ». ACM Trans. Multimedia Comput. Commun. Appl., vol. 1, no. 1, pages 3_13, 2005.
- [18] **Micheal S. Lew, Nicu Sebe et Ramesh Jain.** « Content-Based Multimedia Information Retrieval : State of the Art and Challenges ». ACM Transactions on Multimedia Computing, Communications and Applications, vol. 2, no. 1, pages 1_19, February 2006
- [19] **Ritendra Datta, Dhiraj Joshi, Jia Li et James Z. Wang.** « Image Retrieval: ideas, In_uences, and Trends of the New Age» . ACM Computing Surveys, vol. 40, no. 2, pages 1_60, 2008.
- [20] **B V Patel et B B Meshram,** CONTENT BASED VIDEO RETRIEVAL SYSTEMS”, International Journal of UbiComp (IJU), Vol.3, No.2, April 2012

- [21] **Bahjat Safadi**, « indexation sémantique des images et des vidéos par apprentissage actif », Laboratoire d'Informatique de Grenoble, l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique (MSTII), 17 septembre 2012
- [22] **Dr. P. Nagabhushan, Dr. N. U. Bhajantri, V. Asha**, Department of Computer Applications New Horizon College of Engineering Bangalore, Karnataka, INDIA, 2010.
- [23] **John Calleja**, « Bhattacharya Analysis » Melbourne Pathology Services AACB SES: Sydney, 18th Jun 2010.
- [26] Raphaël Isdant, « Traitement numérique de l'image », 2009

Webographie :

- [08] **Lionel BRUNEL**, [http://www.brunel.nom.fr/these-lionel-brunel/5_132_espace_hsi_hsv_hls.html] vu le 09/01/2013
- [09] **Sabrina Tolari** [<http://www-ia.lip6.fr/~tollaris/ARTICLES/THESE/index.html>] vu le 09/01/2013
- [11] <http://amazon.ece.utexas.edu/~qazim/reserch.html> vue le 11/01/2013
- [24] **opencv, dev team**. Histogram comparaison, [http://docs.opencv.org/doc/tutorials/imgproc/histograms/histogram_comparison/histogram_comparison.html], vu le 27/05/2013
- [25] Developpez.com « http://dotnet.developpez.com/articles/nouveautes-visual-studio-2010/?page=page_2 » vu le 25/05/2013.