UNIVERSITE DE NICE-SOPHIA ANTIPOLIS

ECOLE DOCTORALE STIC

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

T H E S E

pour obtenir le titre de

Docteur en Sciences

de l'Université de Nice-Sophia Antipolis

Mention : automatique, traitement du signal et des images

présentée et soutenue par

Emilie DUMONT

SIMILARITE DES SEQUENCES VIDEO : APPLICATION AUX RUSHES

Thèse dirigée par *Bernard MERIALDO*

soutenue le *4 février 2009*

Jury :

Mme Benois-Pineau	Professeur	Rapporteur
M. Quafafou	Professeur	Rapporteur
M. Glotin	Maitre de conférence	Examinateur
M. Lambert	Professeur	Examinateur
M. Mérialdo	Professeur	Examinateur

Mis en page avec la classe thloria.

Table des matières

Introduction générale

Motivations

La diffusion et la possibilité d'accès à des bases de données vidéo deviennent des réalités de plus en plus évidentes. A cela, il faut ajouter des actions continues d'archivage (agences de presse, INA, musées, surveillance, etc). Malheureusement, si les bases de données existantes sont à la fois nombreuses et volumineuses, il est le plus souvent difficile d'identifier des informations pertinentes vis-à-vis d'une requête, ou d'accéder à des informations particulières, et donc d'exploiter ces bases (localement ou à distance) avec efficacité.

Afin de faciliter la recherche et la navigation dans une masse toujours croissante de vidéos, nous nous sommes intéressés au problème du développement d'outils adaptés à la construction de résumés automatiques et à la structuration sémantique des documents audiovisuels par le contenu en se basant en particulier sur une mesure de similarité du contenu visuel des séquences vidéo.

Contributions

Le travail de recherche effectué se fonde sur la campagne d'évaluation internationale "TREC Video Retrieval Evaluation" [1], et en particulier sur la tâche consacrée à l'exploitation des rushes vidéo qui débuta en 2006.

Les rushes d'un film sont constitués de documents originaux (bobines de film, bandes sons, cassettes vidéo) produits au tournage et issus de la caméra et de l'appareil d'enregistrement sonore. Ils sont donc constitués de beaucoup de séquences "outils" telles que les mires ou les claps, ou encore des séquences dites "poubelles" telles que des plans de couleurs uniformes (noirs, gris, bleu ...). De plus, toutes ces séquences sont temporellement très redondantes, c'est-à-dire, que certains plans peuvent durer plusieurs minutes pendant lesquelles, visuellement, rien ne se passe. Le chapitre **Prétraitement vidéo** explique nos méthodes de détection des séquences outils et poubelles, ainsi que notre accélération dynamique permettant de réduire la redondance temporelle.

Nous proposons un système pour l'exploration des rushes dans le chapitre **Dictionnaire visuel**. C'est un système de recherche de plans vidéo basé sur une adaptation des méthodes de recherche de documents textuels ([1], [2]).

[1] http ://www-nlpir.nist.gov/projects/trecvid/

Par la suite, dans le chapitre **Construction de résumé vidéo de rushes**, nous proposons une mesure du contenu sémantique d'une séquence vidéo ([3], [4]), ainsi qu'une méthode d'alignement des séquences vidéo permettant une structuration des vidéos en scènes. Ces méthodes ont été appliquées pour la construction de résumés vidéo. En parallèle, une méthode collaborative a été développée. L'idée fut d'utiliser les domaines de recherche de différents laboratoires (image, son, mouvement) pour construire des résumés vidéo ([6], [8], [9]) ; cette méthode est décrite dans le chapitre **Une approche collaborative**.

Le dernier chapitre de cette thèse, **Evaluation automatique**, décrit notre méthode d'évaluation automatique des résumés vidéo. La campagne d'évaluation TRECVID permet, une fois par an, une évaluation manuelle des résumés, et donc une comparaison des différents systèmes proposés. Cependant, il est difficile de développer un système sans moyen d'évaluation. Pour cette raison, une automatisation du système d'évaluation manuelle a été développée ([5]).

Bibliographie

[1] Benmokhtar Rachid, Dumont Emilie, Mérialdo Bernard and Huet, Benoit Eurecom in TrecVid 2006 : high level features extractions and rushes study *TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation, November 2006, Gaithersburg, USA*

[2] Dumont Emilie and Mérialdo Bernard Video search using a visual dictionary *CBMI 2007, 5th International Workshop on Content-Based Multimedia Indexing, June 25-27, 2007, Bordeaux, France*

[3] Dumont Emilie and Mérialdo Bernard Split-screen dynamically accelerated video summaries *TVS'07, TRECVID Workshop on Video Summarization at ACM Multimedia, September 24-29, 2007, Augsburg, Germany*

[4] Dumont Emilie and Mérialdo Bernard Redundancy removing and event clustering for video summarization *WIAMIS 2008, 9th International Workshop on Image Analysis for Multimedia Interative Services, May 7-9, 2008, Klagenfurt, Austria*

[5] Dumont Emilie and Mérialdo Bernard Automatic evaluation method for rushes summarization : experimentation and analysis *CBMI 2008, 6th International Workshop on Content-Based Multimedia Indexing, June 18-20, 2008, London, UK*

[6] Bailer Werner, Dumont Emilie, Essid Slim and Mérialdo Bernard A collaborative approach to automatic rushes video summarization *1st IEEE ICIP Workshop on Multimedia Information Retrieval : New Trends and Challenges, October 12, 2008, San Diego, USA*

[7] Dumont Emilie and Mérialdo Bernard Sequence Alignment for Redundancy Removal in video Rushes Summarization *TVS'08, TRECVID Workshop on Video Summarization at ACM Multimedia, October, 2008, Vancouver, Canada*

[8] Dumont Emilie, Mérialdo Bernard, Essid Slim Balier Werner, Rehatschek Herwig, Byrne Daragh, Bredin Hervé, O'Connor Noel E., Jone Gareth J.F., Smeaton Alan F., Haller Martin, Krutz Andreas, Sikora Thomas and Piatrik Tomas Rushes Video Summarization Using a Collaborative Approach *TVS'08, TRECVID Workshop on Video Summarization at ACM Multimedia, October, 2008, Vancouver, Canada*

[9] Dumont Emilie, Mérialdo Bernard, Essid Slim Balier Werner, Byrne Daragh, Bredin Hervé, Jone Gareth J.F., Smeaton Alan F., Haller Martin, Krutz Andreas, Sikora Thomas and Piatrik Tomas A collaborative approach to video summarization *SAMT 2008, 3rd International Conference on Semantic and Digital Media Technologies, December 3-5, 2008, Koblenz, Germany*

[10] Wilkins Peter, Adamek Tomasz, Ferguson Paul, Hughes Mark, Jones Gareth J F, Keenan Gordon, McGuinness Kevin, Malobabic Jovanka, O'Connor Noel E, Sadlier David,

Smeaton Alan F, Benmokhtar Rachid, Dumont Emilie, Huet Benoit, Mérialdo Bernard, Spyrou Evaggelos, Koumoulos George, Avrithis Yannis, Moerzinger R, Schallauer P, Bailer W, Zhang Qianni, Piatrik Tomas, Ch, ramouli Krishna, Izquierdo Ebroul, Goldmann Lutz, Haller Martin, Sikora Thomas, Praks Pavel, Urban Jana, Hilaire Xavier and Jose Joemon M K-Space at TRECVid 2006 *TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation, November 2006, Gaithersburg, USA*

Première partie

Introduction

Introduction

Dans ce premier chapitre du manuscrit, nous allons développer le contexte et la motivation de cette thèse qui sont entièrement liés à la campagne d'évaluation internationale TRECVID. Le but de cette campagne est d'encourager la recherche dans le domaine de la recherche d'informations. Nous allons définir clairement un des challenges de cette campagne : les résumés de rushes. Les rushes d'un film sont constitués des documents originaux (bobines de film, bandes sons, ...) produits au tournage et issus de la caméra et de l'appareil d'enregistrement sonore. Ce sont les documents uniques, bruts, qui seront utilisés au montage et en postproduction. Dans un deuxième temps, nous présenterons un état de l'art des sujets traités tout au long de cette thèse.

1 La campagne d'évaluation TRECVID

La campagne d'évaluation TRECVID [2] a pour objectif d'encourager la recherche dans le domaine de la recherche d'informations en mettant à disposition, pour les centres de recherche participants, des grandes bases de données, ainsi qu'une procédure d'évaluation afin que les organismes intéressés puissent comparer leur résultats. Cette campagne débuta en 2001 et ne cesse d'évoluer depuis ; elle est subventionnée par NIST [3] avec le soutien des pouvoirs publics américains. Cette campagne d'évaluation s'intéresse à plusieurs tâches qui évoluent en fonction des besoins de la recherche. En 2008, les tâches proposées étaient :

- Détection d'évènements pour les vidéos surveillances
- Extraction de caractéristiques de haut niveau
- Recherche de caractéristiques
- Résumé vidéo de rushes
- Détection de copie de contenu vidéo

Cette thèse se concentre sur la tâche de construction de résumés automatiques appliqués aux rushes. Une vidéo décrivant cette tâche est disponible[4].

[2] TREC Video Retrieval Evaluation : *http ://www-nlpir.nist.gov/projects/trecvid/*
[3] National Institute of Standards and Technology - *http ://www.nist.gov/*
[4] TRECVid BBC Rushes Summarization 2008 : http ://www.youtube.com/watch ?v=C02XBwKT3jU

2 Les résumés automatiques de rushes

2.1 Réalisation d'un film

Depuis une idée originale, du tournage à la distribution, un film implique nombre d'acteurs : techniciens, artistes, diffuseurs... Il peut s'étendre de plusieurs semaines à plusieurs mois. La réalisation d'un film peut être découpée en cinq étapes. Dans un premier temps, le développement d'un script est conçu par un scénariste. Ensuite, la préproduction se met en place pour préparer le tournage avec la conception d'un dossier de production. Puis vient la production proprement dite, durant laquelle le réalisateur tourne son film aux côtés de techniciens et d'artistes. Les éclairages sont mis en place, les acteurs maquillés et costumés. Ils répètent alors leur texte sous la direction du réalisateur, qui leur indique les mouvements à effectuer, corrige leur intonation... Enfin, le tournage peut commencer. Chaque scène est tournée en plusieurs prises et chaque prise est identifiée grâce à un clap, ce qui permettra au monteur de repérer les bons plans. C'est au réalisateur de décider si la prise est bonne, ou, au contraire, s'il faut la refaire. Par sécurité, les prises bonnes sont doublées. L'ensemble de ces prises de vue constitue les «rushes». Enfin, la postproduction permet le montage du film et l'ajout de la bande sonore ainsi que des effets spéciaux. Le processus se termine avec la distribution lorsque le film bénéficie de publicités et de copies favorisant sa diffusion.

2.2 Les rushes vidéo

Les rushes sont constitués de scènes diverses de la vie de tous les jours, certains acteurs apparaissent très régulièrement dans différentes scènes ou situations, avec par exemple différents vêtements. D'autres personnages ne peuvent apparaître qu'une seule fois. Le montage n'ayant pas été effectué, le son aussi est original : les bruits de l'environnement, la voix du réalisateur ou encore celles de l'équipe de tournage sont présents sur les bandes audio. Visuellement, beaucoup de redondances sont présentes sous diverses formes : les scènes sont refaites un grand nombre de fois, soit à l'identique, soit avec quelques changements dans le texte, le jeu des acteurs, l'angle de la caméra ...

Actuellement, le contenu des rushes n'est exploité que par une équipe alors qu'il pourrait être réutilisé. Ils sont aujourd'hui largement inexploités : il faut de 20 à 40 heures de bande vidéo pour une heure dans la vidéo finale. L'idée de résumer des rushes vidéo pourrait significativement contribuer à un nouveau management des rushes et à une meilleure solution d'exploitation de ceux-ci.

2.3 L'exploration de rushes

La tâche consacrée aux rushes vidéo fut lancée en 2006 comme une tâche pilote. Durant cette première année, elle fut très libre : chaque participant devait développer des outils adaptés aux données vidéo hautement redondantes. Ces outils devaient prendre en entrée les vidéos et effectuer les analyses nécessaires. Les outils proposés devaient au minimum enlever la redondance (autant que possible) et proposer une méthode d'organisation du matériel non redondant. Chaque participant devait aussi adapter une méthode d'évaluation automatique de leurs outils. Le but de cette première année était, grâce aux différentes approches, de programmer la tâche des années futures. Les discussions entre organisateurs, participants et membres de la BBC ont pu aboutir à la tâche définitive.

2.4 Les systèmes de résumés

En 2007 et 2008, le travail demandé aux participants fut le développement d'un système générique de création de résumés vidéo de rushes : étant donné un rush vidéo, créer automatiquement un résumé en compressant la vidéo initiale tout en enlevant la redondance et les séquences parasites. Le résumé sera construit dans le but de maximiser le contenu visuel d'une façon efficace dans la reconnaissance des principaux objets et événements présents dans la vidéo initiale. La durée du résumé vidéo ne devra pas dépasser 4% en 2007, puis 2% en 2008 de la durée de la vidéo initiale, c'est-à-dire que en moyenne une résumé doit durer 32 secondes. Les résumés doivent être présentés sous forme d'une vidéo au format MPEG-1 avec les mêmes caractéristiques que la vidéo initiale.

2.5 La base de données vidéo

Les vidéos proposées pour la tâche de résumé automatique de rushes dans la campagne d'évaluation TRECVID 2008 sont constituées uniquement de vidéos dites rushes provenant des tournages de séries pour la BBC : une série sur la Grèce antique, une série de détective contemporain, une série sur les services d'urgences, un drame policier, ainsi que divers programmes télévisés ; c'est-à-dire 42 vidéos pour la phase de développement des systèmes, et 39 vidéos pour la phase de test. Ces vidéos ont été choisies de manière aléatoire parmi l'intégralité des vidéos initiales. Les vidéos de test ont une durée variant de 9.8 minutes à 36.9 minutes, pour une durée moyenne de 26.6 minutes. Une vérité terrain est disponible pour les vidéos de développement, mais aussi pour les vidéos de test dès la phase d'évaluation terminée.

2.6 La méthode d'évaluation

2.6.1 Construction de la vérité terrain

La tâche du créateur de la vérité terrain est de visionner la vidéo, choisir les éléments d'histoire, et les identifier soit par une objet (animé ou inanimé), soit par un évènement (c'est-à-dire un ou plusieurs objets impliqués dans une action). Le nombre de séquences est défini en fonction de la vidéo et varie donc en fonction de celle-ci. Par leur nature, les rushes contiennent beaucoup de scènes répétitives : idéalement, le résumé ne doit contenir qu'une seule

version des scènes ; mais certaines fois, une improvisation ou une erreur peuvent avoir un grand intérêt et doivent donc être présente dans le résumé, donc dans la liste des éléments d'histoire.

Une séquence correctement choisie ne doit pas appartenir à plusieurs scènes et doit pouvoir rendre compte, par une seule description, des multiples prises de celle-ci. Par contre, si une prise redondante comporte une particularité semblant intéressante, alors une nouvelle séquence doit être correctement identifiée de manière à ne pas confondre les prises. La liste des séquences ne doit pas inclure les mires ou les claps. Chaque description d'une séquence doit avoir l'une des formes suivantes :

- un objet / des objets
- objet(s) + événement
- objet(s) + mouvement de caméra / style de prise de vue
- objet(s) + événement + mouvement de caméra / style de prise de vue

En 2008, la vérité terrain à été faite par cinq personnes retraitées ayant des connaissances en informatique. Au total, ils ont effectué un travail de 110 heures, chaque personne a annoté huit vidéos. Ensuite, les listes proposées ont été vérifiée afin de normaliser les annotations effectuées : certains détails ont été supprimés, les ambigüités ont été enlevées, et les expressions ont été raccourcies.

2.6.2 L'évaluation manuelle

Les résumés ont été évalués de manière manuelle, donc subjectivement : la fraction des éléments d'histoire présents dans le résumé, la redondance présente dans le résumé, le contenu inutile et la qualité visuelle du résumé. Pour l'année 2008, les résumés vidéo ont été évalués par dix étudiants diplômés de Dublin City University. Chaque résumé à été évalué par trois personnes différentes. Les résultats obtenus pour chaque résumé sont la moyenne des ces trois évaluations.

Un évaluateur se voit confier un résumé et une liste chronologique de, au maximum, douze éléments d'histoire choisis aléatoirement parmi l'annotation complète de la vidéo qui, en moyenne, contient vingt et une séquences. L'évaluateur visionne le résumé une seule fois avec une fenêtre du logiciel mplayer de taille 125mm $*$ 102 mm à une fréquence de 25 images par seconde et en utilisant uniquement les fonctions "lecture" et "pause". Il coche les séquences de la liste qu'il a remarquées. Ce processus permet d'évaluer le pourcentage de séquences importantes présentes dans le résumé.

Ensuite, l'évaluateur juge la qualité du résumé en termes de satisfaction visuelle, de qualité des séquences sélectionnées et de quantité de redondances présentes dans le résumé. La qualité est évaluée en attribuant une note variant de 1 à 5.

Enfin, les critères de qualité d'un résumé retenus par cette campagne d'évaluation sont les suivants :

- Pourcentage des éléments d'histoire présents dans le résumé
- Présence de séquences poubelle et outil
- Présence de redondances
- Qualité visuelle du résumé

– Temps passé à juger le résumé
– Durée du résumé par rapport au 2% de la vidéo initiale
– Temps pour la création des résumés

3 Etat de l'art

Après avoir défini les notions fondamentales nécessaires à la compréhension du travail présenté dans ce manuscrit, nous allons nous attarder sur les différentes idées qui ont été proposées pour l'exploitation des rushes. Durant l'approche pilote de la campagne TRECVID, 12 groupes ont présentés des méthodes, mais seulement quelques unes ont correctement répondu à la tâche demandant une évaluation. Puis, nous examinerons les méthodes de résumés vidéos.

3.1 Flux vidéo et compression vidéo

Un flux vidéo est composé d'une succession d'images, 25 par seconde en Europe (30 par seconde aux USA), composant l'illusion du mouvement. Chaque image est décomposée en lignes horizontales, chaque ligne pouvant être considérée comme une succession de points. La lecture et la restitution d'une image s'effectue donc séquentiellement ligne par ligne comme un texte écrit : de gauche à droite puis de haut en bas.
Les séquences vidéo contiennent une très grande redondance statistique, aussi bien dans le domaine temporel que dans le domaine spatial. La propriété statistique fondamentale sur laquelle les techniques de compression se fondent, est la corrélation entre pixels. Cette corrélation est à la fois spatiale, les pixels adjacents de l'image courante sont similaires, et temporelle, les pixels des images passées et futures sont aussi très proches du pixel courant.
Les algorithmes de compression vidéo de type MPEG utilisent une transformation appelée DCT (pour Discrete Cosine Transform), sur des blocs de 8x8 pixels, pour analyser efficacement les corrélations spatiales entre pixels voisins de la même image.
MPEG-1 [Le Gall 1991] est la première norme audio et vidéo utilisé pour les Vidéo CDs. Ce format offre une résolution à l'écran de 352 × 240 pixels à 30 images par seconde ou de 352 × 288 à 25 images par seconde avec un débit d'environ 1,5 Mbit/s. Elle comprend le populaire format audio MPEG-1 partie 3 audio couche 3 (MP3). MPEG-2 est la norme applicable au codage de l'audio et la vidéo, ainsi que leur transport pour la télévision numérique : télévision numérique par satellite, télévision numérique par câble, télévision numérique terrestre, et (avec quelques restrictions) pour les vidéodisques DVD ou SVCD. C'est notamment le format utilisé jusqu'à présent pour la TV sur ADSL. Les débits habituels sont de 2 à 6 Mbit/s pour la résolution standard (SD), et de 15 à 20 Mbit/s pour la haute résolution (HD).

3.2 Image 2D et codage informatique

Dans le cas des images à deux dimensions (le plus courant), les points sont appelés pixels. D'un point de vue mathématique, on considère l'image comme une fonction de $\mathbb{R} * \mathbb{R}$ dans $\mathbb{R}$ où le couplet d'entrée est considéré comme une position spatiale, le singleton de sortie comme un codage.
Il existe plusieurs modes de codage informatique des couleurs, le plus utilisé pour le maniement des images est l'espace colorimétrique Rouge, Vert, Bleu (RVB ou RGB - Red Green Blue). Cet espace est basé sur une synthèse additive des couleurs, c'est-à-dire que le mélange des trois composantes R, G, et B à leur valeur maximum donne du blanc, à l'instar de la lumière. Le mélange de ces trois couleurs dans des proportions diverses permet de reproduire à l'écran un part importante du spectre visible, sans avoir à spécifier une multitude de fréquences lumineuses. Il existe d'autres modes de représentation des couleurs : Cyan, Magenta, Jaune, Noir (CMJN ou CMYK) utilisé principalement pour l'impression, et basé sur une synthèse soustractive des couleurs ; Teinte, Saturation, Luminance (TSL ou HSL), où la couleur est codée suivant le cercle des couleurs ; base de couleur optimale YUV, Y représentant la luminance, U et V deux chrominances orthogonales. Il existe aussi un codage en niveau de gris ; on ne code plus que le niveau de l'intensité lumineuse, généralement sur un octet (256 valeurs). Par convention, la valeur zéro représente le noir (intensité lumineuse nulle) et la valeur 255 le blanc (intensité lumineuse maximale).

3.3 Caractéristiques des images et des vidéos

Une caractéristique peut se définir par une partie «intéressante» d'une image ou d'une vidéo, et est utilisée comme point de départ d'un grand nombre d'algorithmes des domaines de l'analyse d'images ou vidéos. Les caractéristiques étant utilisées comme point de départ et principal primitif pour les algorithmes subséquents, la qualité d'un algorithme dépend de la qualité du choix des caractéristiques.

La détection de caractéristiques est une opération de traitement d'image de bas niveau. Elle est d'ordinaire exécutée comme la première opération sur une image : comme beaucoup d'algorithmes informatiques utilisent la détection de caractéristique en tant qu'étape initiale, un très grand nombre de détecteurs de caractéristique ont été développés. Ceux-ci varient largement dans les types de caractéristiques détectés, la complexité computationnelle et la répétabilité.

3.3.1 Histogramme de couleur

Un histogramme de couleur est une représentation statistique d'une image dérivée de la densité de probabilité de la distribution des couleurs des pixels de l'image. L'idée a été proposée par Micheal Swain et Dana Ballard en 1991 [Swain 1991]. Les histogrammes de couleurs peuvent être construits dans plusieurs plages de couleurs, RGB, HSV, LUV ou toute autre plage de couleurs de toute dimension. Un histogramme de couleurs est produit en découpant d'abord les couleurs de l'image dans un certain nombre de cases, puis en comptant le pourcentage du nombre de pixels dans chaque case. Ceci fournit une vue d'ensemble bien plus compacte des données dans une image plus intéressante que de connaitre la valeur exacte de chaque pixel. La figure 3.1 montre des histogrammes de couleurs pour la même image : à droite, histogramme

global à l'image ; à gauche, histogramme par régions. L'histogramme de couleurs d'une image est invariable selon la translation ou de la rotation de l'axe de vue, et varie progressivement selon l'angle de vue. L'inconvénient principal de cet histogramme est que la représentation obtenue dépend seulement de la couleur de l'objet étudié.

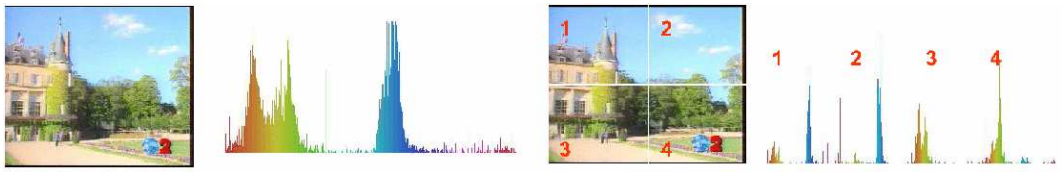


FIG. 3.1 – Exemples d'histogrammes dans l'espace de couleur HSV

3.3.2 Contour

Le but de la détection de contours est de repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. La détection des contours d'une image réduit de manière significative la quantité de données et élimine les informations qu'on peut juger moins pertinentes, tout en préservant les propriétés structurelles importantes de l'image. Il existe un grand nombre de méthodes de détection de l'image mais la plupart d'entre elles peuvent être regroupées en deux catégories.

- La première recherche les extremums de la dérivée première, en général les maximums locaux de l'intensité du gradient. Par exemple, le filtre de Prewitt [Prewitt 1970] introduit un flou, chacune des deux matrices étant le produit du filtre dérivation dans la direction considérée par un filtre de flou rectangulaire selon l'autre direction. Le filtre Soble [Sobel 1968] améliore la technique précédente en remplaçant le filtre rectangulaire par un filtre triangulaire. Ou encore, le filtre de Canny [Canny 1986] est un filtre de Sobel précédé par un lissage gaussien et suivi par un seuillage. Ce filtre est conçu pour être optimal, au sens de trois critères.

- La seconde recherche les annulations de la dérivée seconde, en général les annulations du laplacien ou d'une expression différentielle non-linéaire. Par exemple le filtre de Marr-Hildreth [Marr 1980] effectue le calcul du laplacien précédé par un lissage gaussien avec deux variances ajustables pour filtrer les hautes fréquences.

La figure 3.2 montre des exemples de détection de contour pour la même image : à gauche, l'image originale, puis la détection de contour par l'algorithme de Prewitt, suivi de la détection de Soble, et enfin la détection de Marr-Hildreth.

3.3.3 Point d'intérêt

Un point d'intérêt est un point dans l'image qui peut être caractérisé en général comme suit : il a une définition mathématique claire et bien fondée, il a une position bien définie dans l'image, la structure locale autour du point d'intérêt est riche en termes de contenus d'information locaux, il est insensible en présence de déformations et changement de luminosité, ce qui est quelques fois réduit à la une stabilité face aux transformations affines, aux changements

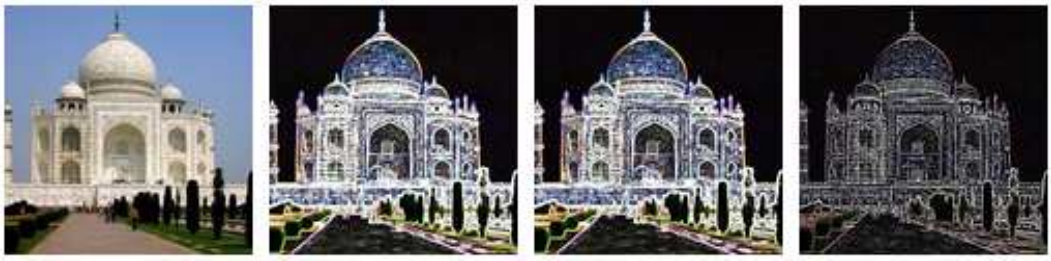


FIG. 3.2 – Exemples de détection de contour

d'échelle, aux rotations et/ou aux translations.
Le premier détecteur de point d'intérêt fut développé par Moravec [Moravec 1980] qui considère le voisinage d'un pixel et détermine les changements moyens d'intensité dans le voisinage, l'un des détecteurs de points clés le plus populaire est celui de Harris [Harris 1988] qui est une amélioration des détecteurs de Moravec.

FIG. 3.3 – Exemples de détection de point clés par le détecteur de Harris.

3.3.4 Mouvement

Le mouvement est une information riche qui renseigne sur l'activité d'une séquence vidéo et celles de ses objets. L'ensemble des vecteurs mouvement des points ou de régions est appelé flux optique. De nombreuses techniques ont été développées à partir des années 1980 [Barron 1992, Quénot 1996].
Une première méthode intéressante pour l'estimation des mouvements est décrite dans [Tan 1995]. Le modèle du mouvement de la caméra est décrit par un facteur zoom noté s qui est le rapport des longueurs focales de la caméra entre deux images, l'angle α du panoramique

horizontal qui est l'angle de rotation autour de l'axe Y, l'angle β du panoramique vertical qui est l'angle de rotation autour de l'axe X, l'angle γ de rotation qui est l'angle de rotation autour de l'axe Z et le vecteur translation $t = (t_x, t_y, t_z)^T$.
Les auteurs de [Tan 2000] expliquent comment détecter les mouvements de caméra en utilisant directement l'information disponible dans le flux MPEG. Ce qui améliore le temps de calcul ainsi que l'espace mémoire. Pour cela, ils modélisent le mouvement par un modéle affine à trois paramètres car ils supposent que les effets de distorsions sont minimaux, et que la caméra ne peut pas effectuer une rotation autour de l'axe de l'objectif de la caméra. Ces trois paramètres représentent le facteur zoom, panoramique horizontal et le panoramique vertical. La calcul de ces paramètres est simple : il suffit d'effectuer une minimisation aux moindres carrés.
Dans [Wang 1999], Wang et Huang utilisent aussi les informations du flux mpeg directement. Le mouvement est modélisé par un modéle affine à quatre paramétres représentant les facteurs zoom, panaoramique horizontal, panoramique vertical et rotation. En supposant que l'erreur suit une loi Gaussienne, les paramétres sont estimés par une minimisation aux moindres carrés. Cette méthode est applicable en temps réel et n'utilise pas beaucoup d'espace mémoire.
De nombreuses techniques continuent a être développées [Bhaskar 2001, Durik 2001, Del Bimbo 1995]. La méthode la plus utilisée reste [Horn 1980] pour estimer un champ de vitesse à partir de séquences d'images, elle repose sur une équation de conservation du niveau de gris. Cette unique équation n'est pas suffisante pour calculer la vitesse apparente bi-dimensionnelle : une équation supplémentaire est nécessaire pour fermer le système. Celle-ci est généralement fournie par une hypothèse de régularité sur le champ de vecteurs vitesse.

3.3.5 Activité entre deux images

L'activité d'une séquence vidéo est une caractéristique visuelle subjective de l'intensité du mouvement perçu. Pour cette raison, de nombreuses techniques de calcul de l'activité sont basées sur le mouvement. La méthode la plus couramment utilisée est le descripteur de MPEG-7 [MPEG-7 2002]. MPEG-7 définie un descripteur d'activité du mouvement qui tente de capturer « le rythme du mouvement dans la séquence, comme perçu par le téléspectateur ». L'intensité de l'activité de ce descripteur est définie comme l'écart type de la magnitude de vecteurs de mouvement de MPEG normalisés et quantifiés sur cinq niveaux. Une combinaison de caractéristiques visuelles et audios [Pfeioeer 1996], un calcul sur la magnitude des vecteurs mouvements [Wolf 1996], la distance de la tangente entre deux images successives [Vasconcelos 1997], ou encore la moyenne, la variance, et la médiane des vecteurs magnitudes [Peker 2003] sont utilisés pour déterminer le niveau d'activité au sens du mouvement dans les séquences vidéos.
La méthode que nous utilisons se base sur l'étude présentée dans [Oh 2002], qui utilise une différence de couleur pixel à pixel entre deux images successives pour déterminer l'activité d'une séquence vidéo. C'est-à-dire que l'activité d'une image est le pourcentage de pixels, après quantification, ayant changé de couleurs par rapport à l'image précédente.

3.4 Opérations sur les caractéristiques

3.4.1 Entropie de Shannon

L'entropie de Shannon est une fonction mathématique qui, intuitivement, correspond à la quantité d'information contenue ou délivrée par une source. Cette source peut être une langue, un signal électrique, ou un fichier informatique quelconque. La définition de l'entropie

d'une source, selon Shannon, est telle que plus la source est redondante, moins elle contient d'information au sens de Shannon.
L'entropie de Shannon d'une image I contenant n couleurs de pixels possibles, $1..n$, est définie comme suit :

$$Ent(I) = -\mathbf{E}[\log_2 p(i)] = \sum_{i=1}^{n} p(i) \log_2 \left(\frac{1}{p(i)}\right) = -\sum_{i=1}^{n} p(i) \log_2 p(i).$$

où $\mathbf{E}$ désigne l'espérance mathématique. L'entropie ainsi définie vérifie les propriétés suivantes :
- elle est positive ou nulle
- elle est nulle pour une image contenant une seule couleur de pixel
- elle est maximale pour une distribution uniforme des couleurs de pixel dans l'image
- elle augmente avec le nombre de couleurs possibles
- elle est continue : une faible modification de la répartition des couleurs la modifie faiblement.

3.4.2 Distance vectorielle

Soit deux histogrammes de couleurs $H1 : (x1, x2, ..., xn)$ et $H2 : (y1, y2, ..., yn)$, on exprime les différentes distances ainsi :

- 1-distance ou distance de Manhattan : $\sum_{i=1}^{n} |x_i - y_i|$
- 2-distance ou distance euclidienne : $\sqrt{\sum_{i=1}^{n} |x_i - y_i|^2}$, c'est la distance la plus intuitive.
- p-distance ou distance de Minkowski : $\sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p}$
- ∞-distance $\lim_{p\to\infty} \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} = \sup_i |x_i - y_i|$, elle est rarement utilisée en dehors des cas $p = 1$, 2 ou ∞.
- Bhattacharyya : $\sum_{i=1}^{n} \sqrt{x_i y_i}$

3.5 Segmentation temporelle

Les documents vidéo sont hiérarchiquement structurés en scènes, plans et images. Les plans sont définis comme des séquences continues d'images prises sans arrêter la caméra. Une scène est définie comme une suite de plans contigus qui sont sémantiquement reliés. Le but de le segmentation en plans est de trouver une méthode de détection automatique des plans dans la vidéo. Pour cela, il faut identifier les effets de transitions.

Dans la littérature, les méthodes les plus classiques se regroupent en trois catégories : différences pixel à pixel, comparaison d'histogrammes, estimation de mouvements :

- Le principe des méthodes pixel à pixel est de calculer le nombre de pixels différents entre deux images successives. Un changement de plan à lieu si ce nombre est supérieur à un seuil fixé. L'inconvénient principal de ces méthodes est la complexité des algorithmes, aussi bien la complexité en temps que la complexité en espace. De plus, les méthodes basées sur ce principe ne sont pas robustes au bruit et aux forts mouvements.

- Les méthodes à base d'histogramme utilisent le même principe que la méthode décrite précédemment : si deux images successives ont des différences sur leur histogramme supérieures à un seuil fixé, alors un changement de plan à lieu. L'inconvénient est qu'un

changement de plan peut avoir lieu entre deux images, mais qu'il ne soit pas détecté du fait de la similarité des histogrammes des deux images.

- La dernière méthode fréquemment utilisée consiste à estimer les mouvements pour chaque pixel d'une image et de les comparer à l'image successive. Un changement de plan est détecté si le nombre d'incohérence entre les deux images est supérieur au seuil préalablement fixé. L'inconvénient de cette méthode est qu'elle ne peut pas être utilisée en temps réel.

Les méthodes exposées ne sont pas toujours efficaces pour détecter les transitions progressives. Pour résoudre ce problème, des techniques basées sur la détection et/ou le suivi d'objets ont été proposées [Heng 2003]. L'idée générale est que le suivi d'un objet indique une continuité, et que la perte de suivi, peut indiquer une transition. D'autres proposent de modéliser spécifiquement le comportement de chaque type de transition progressive (fondu au noir, fondu enchaîné, volet...) par des méthodes heuristiques et des techniques de double seuillage [Truong 2000], ou un réseau de neurones [Lienhart 2001]. Les fondus enchaînés sont particulièrement difficiles à détecter, et certains travaux se concentrent uniquement sur cette tâche [Lienhart 2001]. D'autres se concentrent sur les volets, notamment parce que c'est une technique très utilisée à la télévision [Wu 1998]. Un autre problème majeur est celui des changements brutaux d'illumination, flashs, spots, apparition/disparition du soleil... Des méthodes spécifiques ont été développées pour diminuer les fausses alarmes liées à ces évènements, en s'aidant de la détection de contours [Heng 1999] ou d'un post-processing [Truong 2000].

3.6 Classification automatique

La classification automatique consiste à regrouper divers objets (les individus) en sous-ensembles d'objets (les classes). Elle peut être :

- supervisée : les classes sont connues à priori, elles ont en général une sémantique associée
- non-supervisée : les classes sont fondées sur la structure des objets, la sémantique associée aux classes est plus difficile à déterminer

Dans les deux cas, on a besoin de définir la notion de distance entre deux classes : le critère d'agrégation.

3.6.1 Classification non-supervisée

La classification non-supervisée est utilisée lorsque qu'on possède des objets qui ne sont pas classés et dont on ne connaît pas de classification. A la fin du processus de classification non-supervisée, les objets doivent appartenir à l'une des classes générées par la classification. Deux catégories de classification non-supervisées sont distinguée : hiérarchique et non-hiérarchique.

Classification hiérarchique

Dans la classification hiérarchique (CH), les sous-ensembles créés sont emboîtés de manière hiérarchique les uns dans les autres. On distingue la CH descendante (ou divisive) qui part de l'ensemble de tous les individus et les fractionne en un certain nombre de sous-ensembles, chaque sous-ensemble étant alors fractionné en un certain nombre de sous-ensembles, et ainsi de suite. Et la CH ascendante (ou agglomérative) qui part des individus seuls que l'on regroupe en sous-

ensembles, qui sont à leur tour regroupés, et ainsi de suite. Pour déterminer quelles classes on va fusionner, on utilise le critère d'agrégation.

Classification non-hiérarchique

Dans la classification non-hiérarchique, les individus ne sont pas structurés de manière hiérarchique. Si chaque individu ne fait partie que d'un sous-ensemble, on parle de partition. Si chaque individu peut appartenir à plusieurs groupes, avec la probabilité p_i d'appartenir au groupe i, alors on parle de recouvrement.

Critère d'agrégation

Le critère d'agrégation permet de comparer les classes deux à deux pour sélectionner les classes les plus similaires selon un certain critère. Les critères les plus classiques sont le plus proche voisin, le diamètre maximum, la distance moyenne et la distance entre les centres de gravités.

- Lien simple ou plus proche voisin. La distance entre la classe C_p et la classe C_q est la plus petite distance entre un élément de C_p et un élément de C_q.

$$D(C_p, C_q) = \min\{dist(i,j); i \in C_p, j \in C_q\}$$

- Lien complet ou diamètre maximum. La distance entre la classe C_p et la classe C_q est la plus grande distance entre un élément de C_p et un élément de C_q.

$$D(C_p, C_q) = \max\{dist(i,j); i \in C_p, j \in C_q\}$$

- Lien moyen de groupe ou distance moyenne. La distance entre la classe C_p et la classe C_q est la moyenne des distances entre les éléments de C_p et les éléments de C_q.

$$D(C_p, C_q) = \frac{\sum_{i,j}\{dist(i,j); i \in C_p, j \in C_q\}}{Card(C_p) \times Card(C_q)}$$

- Distance des centroïdes. Si G_p est le centre de gravité de la classe C_p et si G_q est le centre de gravité de la classe C_q alors la distance entre la classe C_p et la classe C_q est la distance entre leurs centres de gravités.

$$D(C_p, C_q) = dist(G_p, G_q)$$

3.6.2 Classification supervisée

L'apprentissage supervisé est une technique d'apprentissage automatique où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des exemples de cas déjà traités. Le but de la méthode d'apprentissage supervisé est alors d'utiliser cette base d'apprentissage afin de déterminer une représentation compacte de f notée g et appelée fonction de prédiction, qui à une nouvelle entrée x associe une sortie g(x). Le but d'un algorithme d'apprentissage supervisé est donc de généraliser pour des entrées inconnues ce qu'il a pu « apprendre » grâce aux données déjà traitées par des experts, ceci de façon « raisonnable ». Les méthodes les plus classiques sont les moindres carrés, les k plus proches voisins, les arbres de décision, les réseaux de neurones, les machines à vecteurs de support.

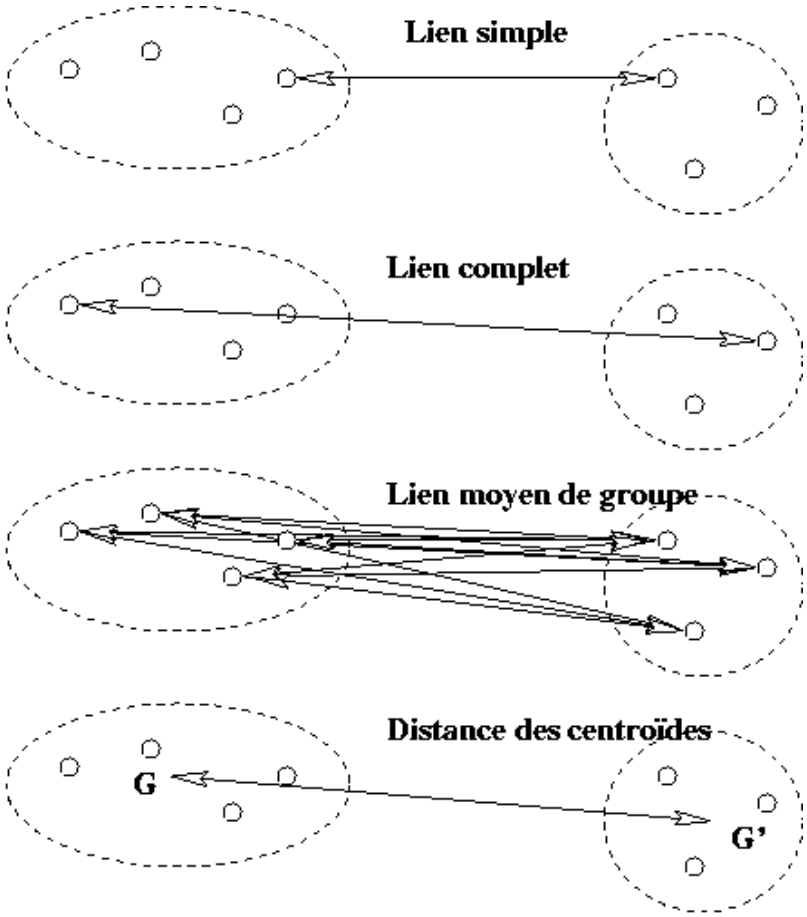


FIG. 3.4 – Illustration des critères d'agrégation.

Les machines à vecteurs de support

Les SVMs sont des classifieurs qui reposent sur deux idées clés qui permettent de traiter des problèmes de discrimination non-linéaire, et de reformuler le problème de classement comme un problème d'optimisation quadratique. La première idée clé est la notion de marge maximale. La marge est la distance entre la frontière de séparation et les échantillons les plus proches. Afin de pouvoir traiter des cas où les données ne sont pas linéairement séparables, la deuxième idée clé des SVMs est de transformer l'espace de représentation des données d'entrées en un espace de plus grande dimension (possiblement de dimension infinie), dans lequel il est probable qu'il existe une séparatrice linéaire.

3.6.3 Evaluation

Classification supervisée

Pour tester la qualité d'une procédure de classification supervisée, on sépare aléatoirement les éléments classés entre une base de référence(R) et une base de test(T). Ensuite, on détermine la procédure de classification C^f à partir des exemples de la base de référence. Puis, on utilise C^f pour retrouver la classe des éléments de la base de test.

La matrice de confusion, dans la terminologie de la classification, est un outil servant à mesurer la qualité d'un système. Chaque colonne de la matrice représente le nombre d'occurrences d'une classe estimée, tandis que chaque ligne représente le nombre d'occurrences d'une classe réelle. Un des intérêts de la matrice de confusion est qu'elle montre rapidement si le système parvient à classifier correctement.

	Positif réel	Négatif réel
Positif prédit	TP (vrai positif)	FP (faux positif)
Négatif prédit	FN (faux négatif)	TN (vrai positif)

TAB. 3.1 – Matrice de confusion

Les mesures de classifications [Makhoul 1999] les plus répandues sont calculées à partir de cette matrice :

- la précision = $\frac{TP}{TP+FP}$ qui représente la proportion de données correctement classées parmi celles retournées.
- le rappel $\frac{TP}{TP+FN}$ qui représente la proportion de données correctement classées restent les plus utilisés.

Mais il existe aussi l'exactitude = $\frac{TP+TN}{TP+FN+FP+TN}$, la sensibilité = $\frac{TP}{TP+FN}$, la spécificité = $\frac{TN}{FP+TN}$ ou encore la F-mesure = $2 * \frac{rappel*precision}{rappel+precision}$. La courbe rappel - précision 3.5 permet d'évaluer l'impact d'un seuil : la précision se trouve en abscisse, et le rappel en ordonnée.

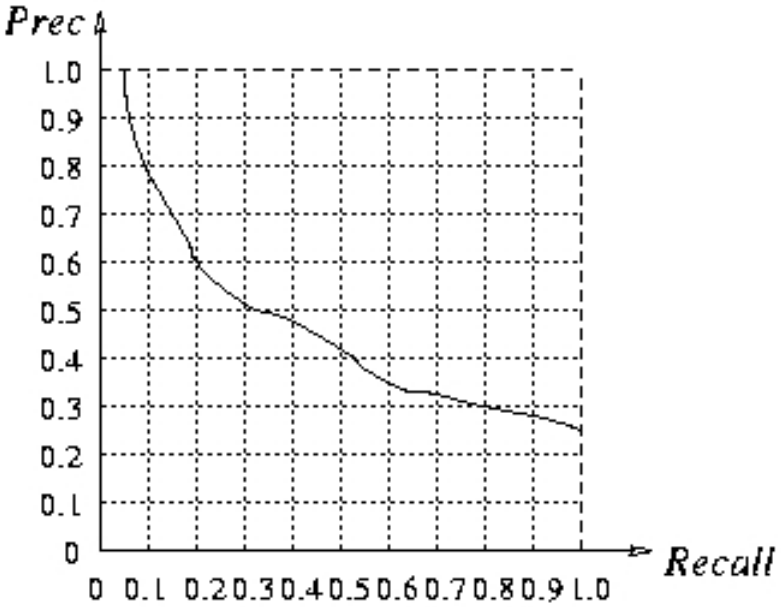


FIG. 3.5 – Exemple de courbe rappel - précision

Classification non supervisée

Dans le cas non-supervisé, on peut évaluer la classification par rapport à certaines de ces caractéristiques. On distingue d'une part, les caractéristiques numériques : le nombre de classes

obtenues, le nombre d'éléments par classe, le nombre moyen d'éléments par classe, l'écart-type des classes obtenues, et d'autre part, les caractéristiques sémantiques. Par exemple, si à un document est associé un ensemble de mots clés, la sémantique associée à une classe pourra se composer des mots les plus fréquents dans la classe.

Pour évaluer l'homogénéité du nombre d'images par classe, on peut utiliser la variance :

$$V = \sigma^2 = \frac{1}{c}\sum_{k=1}^{c}(card(C_k) - moy)^2$$

où $moy = \frac{1}{c}\sum_{k=1}^{c} card(C_k)$ est le nombre moyen d'éléments par classe et c est le nombre de classes obtenues. L'écart-type $\sigma = \sqrt{V}$ permet d'exprimer la dispersion dans la même unité que la moyenne.

3.7 L'exploitation des rushes

L'approche la plus exploitée est basée sur la navigation dans les rushes vidéo. Un premier système est celui proposé par [Allen 2006] : ils commencent par une détection de plans. Beaucoup de plan sont très courts, ils fusionnent donc manuellement certains plans. Puis, il extrait une image clé par plan. Par une méthode simple d'analyse sur ces images clés, les plans noirs, ou encore les mires sont supprimés. Ensuite, manuellement ils annotent les images clés par des mots clés. Les plans sont alors manuellement groupés par histoire. Une interface est finalement développée pour naviguer parmi ces histoires comme le montre la figure 3.6.

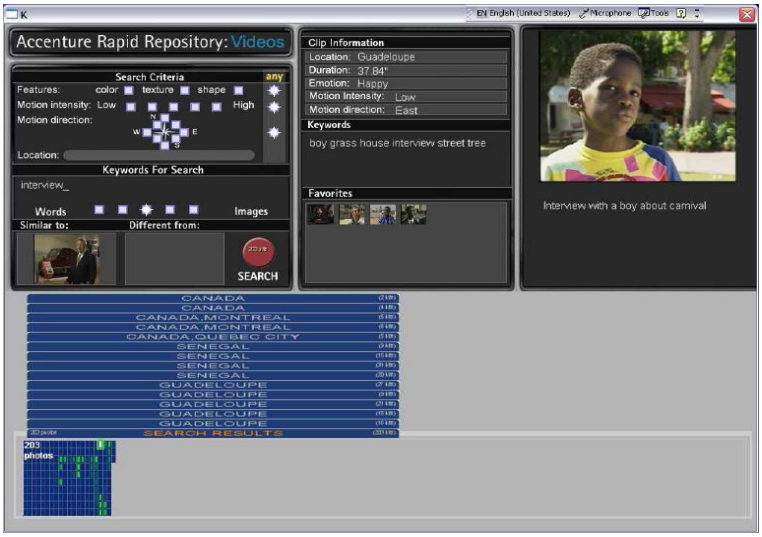


FIG. 3.6 – Schéma du processus proposé par [Allen 2006]

Dans le même esprit, [Liu 2006] détectent un maximum de caractéristiques pour intégrer dans MIRACLE [Gibbon 2006] un logiciel de recherche et de navigation dans les vidéos. Ils commencent par une segmentation en plans, puis extraient les images clés, ensuite ils utilisent des modèles préalablement entrainés pour annoter les images clés par les concepts LSCOM. Ils utilisent aussi un outil pour la détection du texte dans les vidéos. Finalement, ils intègrent le tout dans le logiciel MIRACLE.

Dans [Ewerth 2006], les auteurs ont choisis de travailler sur une unité de temps différente, ils ont choisi de sous-segmenter les plans afin de ne pas garder la redondance temporelle contenue à l'intérieur d'un plan. Ensuite, un grand nombre de caractéristiques sont extraites afin de classifier ces sous-plans, finalement un utilisateur peut naviguer dans les représentants de chaque groupe. Une méthode ressemblante est présentée dans [Bailer 2006] mais ils utilisent l'unité des plans.

[Tang 2006] détectent les séquences d'interview dans la vidéo. Pour cela, les auteurs commencent par une détection de plans, puis ils extraient des images clés. Ensuite, ils effectuent deux processus en parallèle, la détection de la parole dans les plans, et la détection des visages. Puis, grâce à une méthode de fusion, ils détectent le concept interview. La figure 3.7 montre le schéma global de la méthode. Sur l'ensemble des données, ils obtiennent, pour la détection du concept interview un rappel de 0.772 pour une précision de 0.842.

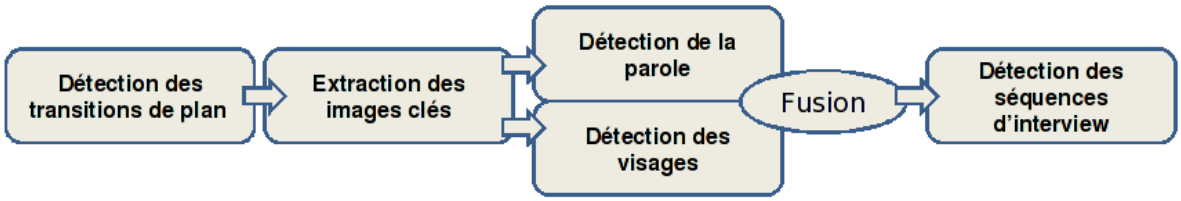


FIG. 3.7 – Schéma du processus proposé par [Tang 2006]

Dans [Cao 2006], une méthode de sélection d'un ensemble de plans non-redondants est présentée : le principe est de segmenter la vidéo en plans, puis d'extraire un certain nombre de caractéristiques permettant une classification. Le médoïde de chaque groupe est alors sélectionné.

Une collaboration de laboratoires [Calic 2006] a proposé une méthode de présentation des rushes vidéos originale. Dans un premier temps, ils effectuent une segmentation basée sur l'audio, puis par des caractéristiques audios et visuels, commencent par détecter les séquences avec de l'excitation, pour ensuite enlever les séquences trop uniformes. Finalement, les images clés sont organisées sur un écran par des images différentes en fonction de leur importance comme le montre la figure 3.8.

[Ulges 2006] utilisent les rushes pour tester une méthode d'extraction de données vidéos basée sur une combinaison de spatiogrammes et la divergence de Jensen-Shannon, [Campbell 2006] propose une classification des plans des rushes basée sur 39 concepts.

Le concept ainsi que le mode d'évaluation actuelle de TRECVID est inspiré de [Truong 2006]. Les auteurs proposent une méthode de résumé vidéo de rushes, puis une méthode d'évaluation basée sur des critères tels que : Le résumé montre-t-il tous les personnages ? Le résumé contient-il beaucoup de redondances ? Le résumé est-il de bonne qualité ?. La méthode de construction des résumés vidéo commence par une détection de plans, une extraction des images clés, ensuite, elle effectue une classification des images clés basée sur la caractéristique SIFT. Pour créer le résumé statique, elle commence par éliminer les singletons, puis sélectionne une image clé par groupe.

FIG. 3.8 – Présentation des images clés proposée par [Calic 2006]

3.8 Les résumés vidéos

La création automatique de résumés multimédia est un outil puissant qui permet de synthétiser le contenu entier d'un document tout en conservant les parties les plus importantes. Dans le domaine de la vidéo, la création d'un résumé vidéo aura pour résultat un nouveau document qui peut consister soit en un arrangement de séquences vidéo, soit en un arrangement d'images. En d'autres termes, un résumé vidéo peut prendre la forme d'un document dynamique ou statique. Cette catégorisation a été effectuée plusieurs fois [Mérialdo 2006, Li 2001, Truong 2007b], le schéma général des méthodes de résumés vidéo est décrit par la figure 3.9. La différence majeure entre ces deux catégories est que les résumés statiques n'ont pas la capacité d'inclure les éléments audio et le mouvement qui améliorent l'expressivité et l'information apportées. De plus, un résumé vidéo dynamique est plus attractif et intéressant à visualiser qu'un ensemble d'images. En contrepartie, les résumés dynamiques ont une capacité limitée dans leur organisation. Les ensembles d'images clés laissent de nombreuses possibilités quant à leur organisation [Calic 2007, Girgensohn 2001, Liu 2007, Uchiashi 1999, Yeung 1997, Krämer 2007, Ionescu 2008].

3.8.1 Les outils

Le contenu vidéo progresse à travers les trois niveaux de la vie d'une vidéo : le tournage, la production et le visionnage. Par conséquent, les résumés vidéo peuvent être effectués pour diverse tâches, et à divers moments de la vie d'une vidéo. Ceci rend la littérature très complète, mais dispersée en fonction du domaine d'utilisation. Le plus souvent, ces techniques sont utilisées dans les domaines du sport, de la musique, des nouvelles télévisées, et des vidéos personnelles. En se basant sur un domaine particulier, le nombre d'ambigüités quand à l'analyse du contenu vidéo est réduit.

Les caractéristiques des images incluant les changements de couleur, de texture, de mouvement d'objets permettent de segmenter une vidéo en plans en identifiant les transitions, qui peuvent être des transitions brusques, ou des transitions progressives. Des objets spécifiques peuvent aussi être identifiés et analysés par de telles caractéristiques dans des vidéos ayant

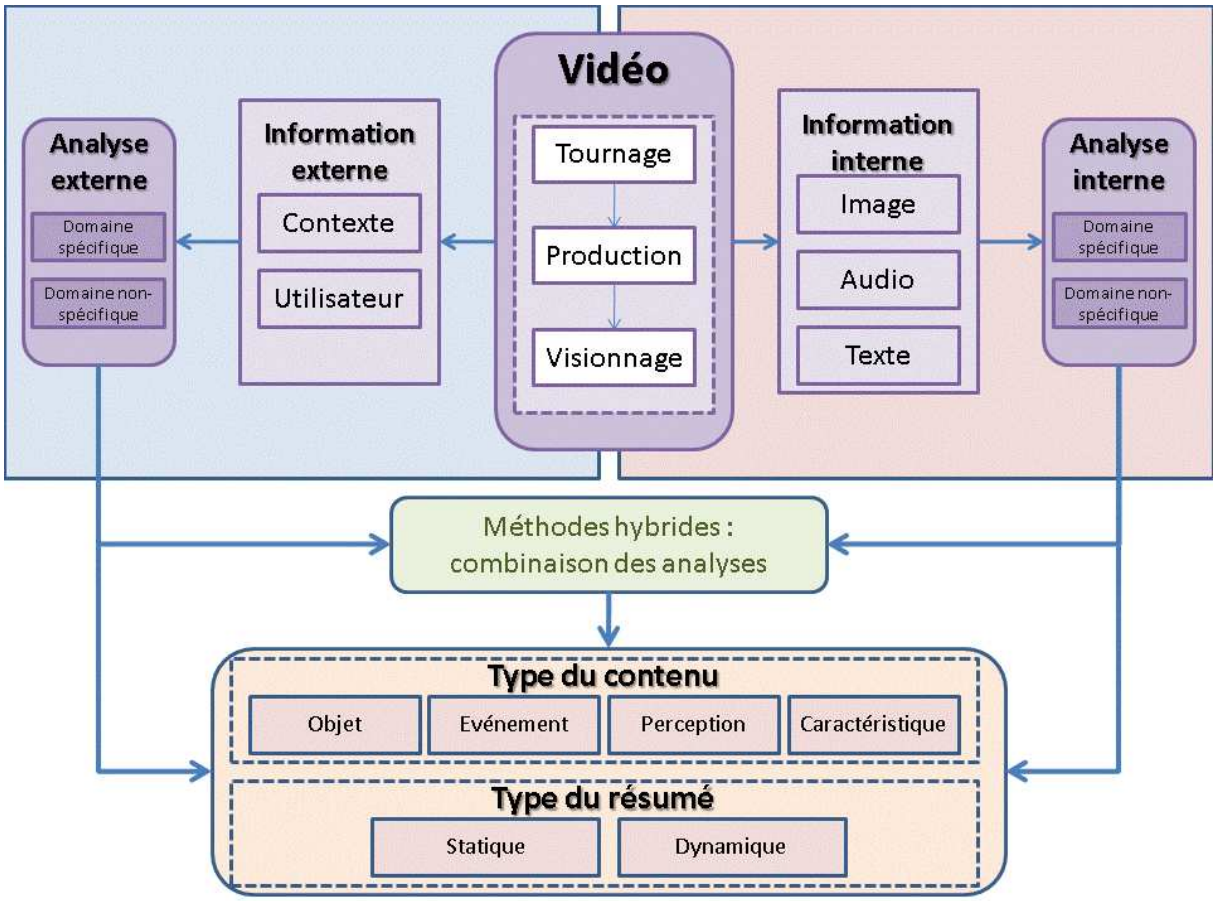


FIG. 3.9 – Schéma général des méthodes de résumés vidéo

une structure connue. Par exemple, dans le domaine du sport, [Ekin 2003] compte sur le fait que la majorité des matchs de football présente des prises de vue longues, moyennes ou des gros-plans. En prenant en compte cette information, ils font une classification des plans sur le pourcentage de pixels de la couleur de l'herbe. La structure des matchs de football est très classique et permet de détecter certains évènements : par exemple la durée des pauses après un but. [Huang 2005] détecte des évènements de vidéo de baseball en utilisant la séquence d'évènements relative au baseball, ainsi que la connaissance des objets spécifiques qui apparait durant un match. La détection d'évènements comme la course, la défense, le lancement est effectuée. La connaissance *a priori* est aussi utilisée dans le domaine des nouvelles télévisées : elles commencent généralement par un résumé des titres durant un certain temps, puis un série de reportages où entre chaque reportage, le présentateur apparait.

Les caractéristiques audio, incluant la parole, le silence, la musique ou encore le son sont utilisées pour détecter des segments candidats pouvant être inclus dans un résumé. En restant dans le domaine du sport, l'audio permet de détecter les coups de sifflet ou l'excitation d'un commentateur. Ces renseignements peuvent aider à déterminer des évènements comme un coup franc, un tir au but, une faute ou un but [Xu 2003].

Le texte qui apparait dans les sous-titres, ou dans les cadres de texte peut contenir des informations sur le contenu ; par exemple, durant un match de boxe, le score défile durant le combat sur une pancarte. [Zhang 2002] utilise les informations textuelles pour détecter des évènements importants durant les matchs de baseball, comme le score et l'annonce du dernier attaquant.

Les méthodes demandant à un utilisateur des informations peuvent être considérées comme coûteuse en terme de temps pour l'utilisateur. Souvent, les systèmes demandant du travail humain ne sont pas considérés comme des solutions réalisables. Par exemple, dans [Tjondronegoro 2004], les auteurs demandent des annotations manuelles de divers évènements et objets dans des vidéos de football : le nom des joueurs, les pénalités, les buts, les plans contenant les buts, les coups de pieds de coin et les fautes. Un autre exemple, dans [Pinzon 2005], les utilisateurs doivent rentrer la description du texte, des annotations manuelles dans [Shipman 2003], le contenu ayant de l'intérêt dans [Lin 2005, Zimmerman 2003]. Cependant, un degré limité d'intervention peut être envisagé. Dans [Aizawa 2001], les auteurs demandent, durant le tournage, d'enregistrer les séquences importantes.

Des informations contextuelles peuvent aussi être utilisées, l'avantage est que cela ne nécessite pas l'intervention d'un utilisateur. Un exemple d'information conceptuelle enregistrée durant le tournage est la localisation GPS attachée à la caméra [Aizawa 2004] ; ou encore des informations provenant de l'environnement, dans [de Silva 2005], la caméra enregistre la pression basée sur des capteurs aux sols permettant d'estimer le mouvement de caméra. Les informations provenant d'internet sont aussi des informations contextuelles. Dans le domaine des matchs de football, les sites spécialisés donnent des informations sur le déroulement du match qui peuvent être associées à la vidéo [Babaguchi 2001, Xu 2006]. Des approches similaires sont utilisées dans le domaine des nouvelles télévisées et des vidéos de musiques [Agnihotri 2004].

3.8.2 Les méthodes

Résumés statiques

Le schéma général des approches statiques est généralement fondé sur les étapes suivantes : identification des plans de la vidéo, sélection des images clés basée sur divers critères, et finalement, une organisation des images est proposée. Un premier aspect important de ce type de résumé est donc de connaitre le nombre d'images clés sélectionnées pour le résumé. Il existe différentes options pour déterminer ce nombre : il peut être fixé *a priori*, ou *a posteriori*.
Dans le premier cas, le nombre d'images est une contrainte, généralement, il dépend de la taille de la vidéo initiale, mais peut aussi être un nombre spécifique. Dans le deuxième cas, le nombre d'images sélectionnées n'est connu qu'à la fin du processus et dépend de la méthode utilisée. Par exemple, avec une méthode basée sur une classification, le nombre d'images sélectionnées dépend des groupes formés, comme dans [Hammoud 2000, Ferman 2003, Yu 2004]. Une autre méthode consiste à déterminer le nombre d'images clés sélectionnées par plans proportionnellement au changement du contenu d'un plan [Liu 2003, Fauvet 2004].
Un autre aspect important des méthodes de résumés statiques est le choix de l'unité temporelle utilisée. L'approche la plus intuitive reste l'utilisation de l'unité du plan vidéo, de plus un avantage de cette méthode est la bonne qualité des méthodes de détection des transitions de

plans. Le schéma 3.10 montre un résumé des différentes étapes.

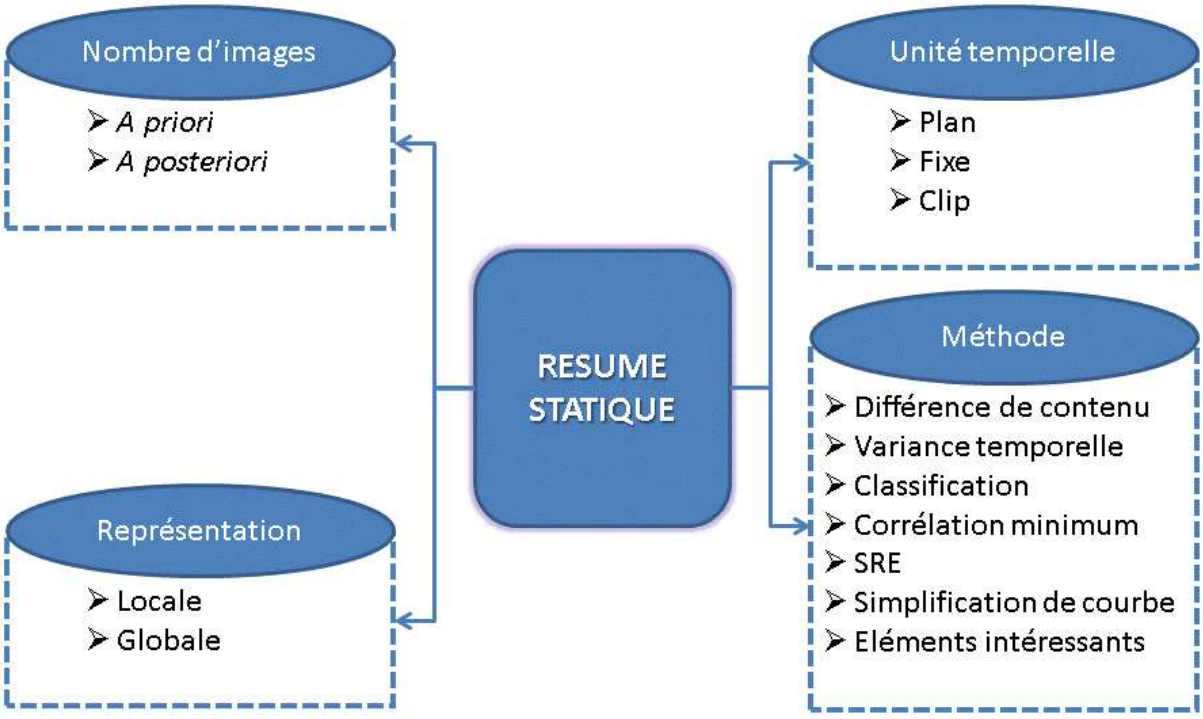


FIG. 3.10 – Attributs des résumés statiques

Selection Nous allons maintenant étudier différentes méthodes de sélections des images représentatives. Une première méthode a pour idée générale de dire qu'une image est une image représentative si son contenu diffère significativement de la précédente image représentative. Il existe plusieurs manières de définir le changement de contenu. La méthode la plus populaire reste la différence d'histogramme [Yeung 1995, Zhang 1997, Kang 1999]. Une autre mesure se définit grâce à une fonction d'accumulation d'énergie sur les mouvements des blocs d'images sur deux images successives [Zhang 2003], ou des changements sur des propriétés géométriques des objets contenus dans les images [Kim 2002].

Une variante de la précédente méthode requiert qu'un nombre d'images clés soit défini *a priori*. Elle permet à l'ensemble des images clés sélectionnées une bonne variance temporelle [Sun 2000, Divakaran 2002, Fauvet 2004]. En général, cette méthode nécessite beaucoup de calculs, mais donne de bons résultats.

Dans la méthode du recouvrement maximal, l'idée est de définir un ensemble d'images qui permet de représenter n'importe quelle image de la vidéo. Cette méthode fut proposée par [Chang 1999], puis améliorée par [Yahiaoui 2001a]. Deux aspects sont importants : déterminer la notion de recouvrement et la procédure d'optimisation. Dans la proposition initiale, le recouvrement consistait à dire que le recouvrement d'une image donnée est l'ensemble des images visuellement similaires, et la procédure d'optimisation consistait itérativement à sélectionner l'image ayant le maximum de recouvrement. Plus récemment, [Rong 2004, Cooper 2005] utilise

l'analogie avec la méthode utilisée pour les textes TF-IDF.

Les méthodes de classification restent les plus populaires : une classification des segments de la vidéo est utilisée. Puis un ensemble d'images clés est extrait des groupes formés. Dans cette méthode, il faut commencer par définir un ensemble de données. [Xiong 1997] extrait un ensemble d'images clés potentiel par la méthode de changement du contenu. Afin de diminuer la taille des données, [David Gibson 2002] utilise une analyse des composantes principales (PCA). Ensuite, il faut déterminer une méthode de classification. [Xiong 1997] utilise une classification séquentielle qui assigne l'image au groupe le plus similaire (en norme L1) sauf si la similarité dépasse un seuil prédéfini, dans ce cas, un nouveau groupe est créé. [Girgensohn 1999] utilise une classification hiérarchique, [David Gibson 2002] des GMMs. Ensuite, un filtrage des groupes trouvés peut-être effectué, [Zhuang 1998] garde seulement les groupes ayant une taille supérieure à la moyenne des clusters. [Girgensohn 1999] retire les groupes ne contenant pas neuf secondes de vidéo consécutive. Finalement, il reste à extraire les représentants des groupes. La méthode la plus répandue est de sélectionner le médoïde de chaque groupe.

La technique de la corrélation minimum sélectionne un ensemble d'images ayant une corrélation minimale entre elles. [Doulamis 1998] considère la notion de corrélation sur toutes les paires d'images. Pour trouver une solution optimale à ce problème, plusieurs solutions furent proposées : une recherche logarithmique ou les algorithmes génétiques [Doulamis 2000a].

L'approche SRE, erreur dans la reconstruction de séquences, est basée sur la capacité d'une image clé à reconstruire une séquence vidéo [Liu 2002, Lee 2003, Liu 2004].

Une autre approche est basée sur la simplification des courbes. Dans cette approche, chaque image est définie par un point multidimensionnel de ces caractéristiques, puis un ensemble de points est recherché tel que si des points sont otés de la courbe, alors, la forme de la courbe change. Ces points correspondent aux images clés [DeMenthon 1998, Calic 2002].

La dernière approche proposée consiste à extraire les évènements intéressants en identifiant les images sémantiquement intéressantes. [Dufaux 2000] propose une approche en deux étapes. Une sélection des plans est effectuée. Elle est basée sur leur longueur, le mouvement, l'activité, les personnes présentes, puis l'image clé ayant la plus faible activité est sélectionnée. Dans [Liu 2003], l'extraction des images clés est basée sur le mouvement des objets. [Kang 2005] apprennent la notion d'image clé par des GMM en utilisant des descripteurs visuels comme la qualité de l'image, le contenu dominant et une mesure d'attention.

Présentation Pour les résumés statiques des images clés sont sélectionnées pour représenter le contenu de la vidéo. Mais ceci requiert de présenter cet ensemble d'une manière compréhensive par l'utilisateur. Les deux méthodes les plus répandues sont les story-boards et les diaporamas. [Komlodi 1998] a montré que les utilisateurs préféraient la méthode des story-boards.

[Yeung 1997] propose une présentation qui consiste en un ensemble de posters vidéos, chacun représentant une scène. [Uchiashi 1999] propose une présentation appelée Video Manga : pour chaque image clé, ils utilisent des tailles différentes en fonction de leur importance, puis leur algorithme de mise en page maintient l'ordre temporel. Cette méthode fut reprise et améliorée

en termes de temps de calcul dans [Girgensohn 2003].

Résumés dynamiques

La recherche, dans le domaine des résumés vidéos dynamiques est plus récente. La longueur du résumé peut-être définie, comme dans le domaine statique, *a priori*, ou *a posteriori*. Cependant, le choix des méthodes de création des vidéos dépend fortement du domaine dans lequel il doit s'appliquer. Dans la littérature, les résumés dynamiques sont crées pour les vidéos de sports, les nouvelles télévisées, les documentaires, les films, les vidéos personnelles, les rushes. Généralement, les systèmes sont constitués de trois étapes : segmentation, sélection, création. Les méthodes les plus classiques restent l'adaptation des méthodes statiques, c'est-à-dire qu'après l'étape de segmentation, une ou plusieurs clés sont extraites, puis utilisées pour la sélection des segments. Dans cette partie, nous ne présenterons pas les méthodes directement dérivées des méthodes statiques.
Le schéma 3.11 montre un résumé des différentes étapes.

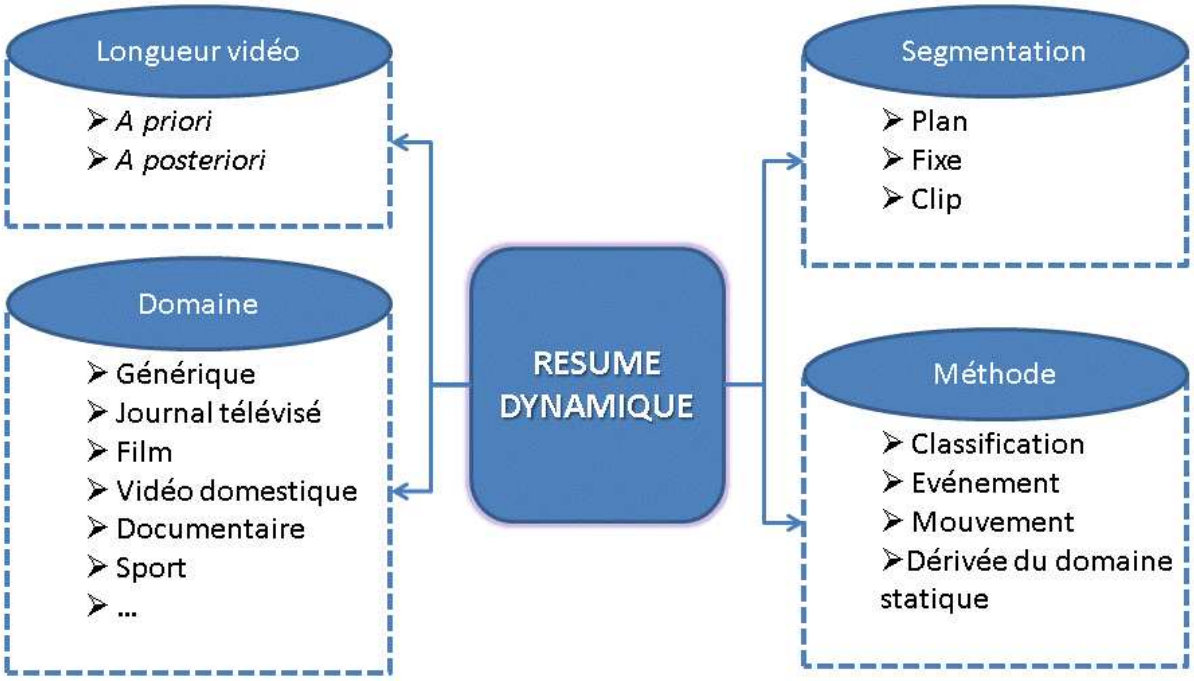


FIG. 3.11 – Attributs des résumés dynamiques

Segmentation L'étape de segmentation consiste à diviser la vidéo en segments vidéo. La segmentation la plus répandue reste la segmentation en plan. [Taskiran 2001] utilise une segmentation basée sur la parole. [Peyrard 2003] effectue une segmentation basée sur la détection de variations du mouvement dominant. Une autre forme de segmentation est déterminée par une factorisation de la matrice de similarité [Cooper 2002].

Sélection Cette étape consiste à sélectionner les segments à inclure dans le résumé. Une méthode classique consiste à classer les plans en se basant sur un critère, puis à sélectionner

le plan le plus long pour chaque groupe [Gong 2003]. Dans les approches basées sur la notion d'événement, les segments les plus pertinents sont sélectionnés [Ariki 2003, Peyrard 2003]. Quelques fois un compromis est effectué entre la pertinence des segments et des facteurs comme la taille du segment. [Babaguchi 2000] sélectionne les segments parmi ceux ayant des évènements pertinents en fonction de leur taille et jusqu'à ce que la longueur totale désirée soit atteinte. Des techniques se basent aussi sur la notion de mouvement : après une classification, le mouvement est calculé pour chaque élément, plan, scène et groupe avec une probabilité d'importance permettant la sélection des segments. [Miura 2003] se base sur la détection de visages pour résumer les vidéos.

3.8.3 L'évaluation automatique

L'évaluation de résumés vidéo est un problème souvent négligé par les chercheurs. Ceci vient du fait qu'il n'y a pas de mesure standard pour évaluer la qualité d'un résumé. De plus, la qualité d'un résumé dépend fortement du but et du domaine d'application, aussi il n'est pas possible de définir une mesure d'évaluation générale. De plus, le processus d'évaluation de résumés est extrêmement subjectif.

L'évaluation la plus commune dans la littérature [Zhuang 1998, Yu 2004] consiste à présenter les résultats de l'approche pour plusieurs vidéos et fournir quelques motivations sur le choix des séquences ou des images-clés proposées. Une évaluation plus complète est quelque fois proposée, elle est subjective mais effectuée par des évaluateurs. Cette évaluation plus réaliste est effectuée selon trois processus différents. Dans un premier cas [Dirfaux 2000, Liu 2003], les évaluateurs résument quelques vidéos pour obtenir une vérité terrain qui peut alors permettre une comparaison automatique. Une deuxième proposition consiste à demander une évaluation de la qualité des résumés par rapport aux vidéos originales. Par exemple, dans [Diklic 1998, Dufaux 2000, Liu 2003], chaque image clé sélectionnée pour être dans une résumé statique est classée en "bonne", "acceptable", "mauvaise" ; cette technique à mis en avant que la sélection de l'image centrale comme image clé donne de bons résultats. Dans le domaine dynamique, des méthodes similaires ont été utilisées comme dans [Lienhart 1997, Sundaram 2001, Agnihotri 2004], les évaluateurs donnent une appréciation sur la qualité et leur satisfaction, telle que "appréciable", "non informatif" ou encore, la précision avec laquelle le résumé a permis l'identification du contenu. Dans la dernière méthode [Ding 1997, Yahiaoui 2001b], les résumés sont présentés aux évaluateurs avec une série critère. La qualité des réponses est alors analysée pour évaluer le résumé et donc la méthodologie de constructions fondamentales.

Néanmoins, il reste possible de définir une mesure permettant l'évaluation de la qualité des résumés. Dans le cas des vidéos statiques, [Liu 2004] compare les performances de sa méthode avec d'autres méthodes en utilisant la SRE. Cette mesure représente la possibilité pour un ensemble d'images clés de reconstruire la vidéo initiale. [Fauvet 2004] propose une mesure simulant la perception humaine en terme de recouvrement et de redondance basée sur l'alignement des images vidéos à un système de coordination commun. [de Silva 2005] a utilisé un ensemble d'images clés provenant d'une vérité terrain sélectionnée par huit personnes. Ils ont montré la qualité de leur système en analysant les correspondances temporelles entre leur ensemble d'images et l'ensemble provenant de la vérité terrain.

Dans le cas des vidéos dynamiques, pour la détection de segments intéressants ou importants, une vérité terrain est effectuée, puis une valeur de rappel - précision est calculée. Dans le cas des vidéos de sports, il est facile de faire une vérité terrain prenant en compte des évènements tels que but, fautes ..., c'est le cas, dans, par exemple, [Chang 2002, Xiong 2003, Ariki 2003, Chen 2004, Shih 2004]. D'autres travaux proposent une création manuelle de résumés vidéos comme vérité terrain. [He 1999] évalue la qualité de son système de résumé de vidéos personnelles directement sur les résumés proposés par les particuliers eux-mêmes. Dans [Miura 2003] les résumés de programme de cuisine sont comparés à ceux de la production.

Deuxième partie

Prétraitement vidéo

Prétraitement vidéo

Travaillant dans le contexte particulier des rushes vidéos, une étape de prétraitement est réalisée afin de nettoyer la vidéo, c'est-à-dire d'enlever de la vidéo, les séquences d'images inutiles. Les rushes d'un film sont constitués des documents originaux (bobines de film, bandes sons, ...) produits au tournage et issus de la caméra et de l'appareil d'enregistrement sonore. Ce sont les documents uniques, bruts, qui seront utilisés au montage et en postproduction. Ils contiennent, donc des séquences vidéos inutiles regroupées en deux catégories : les séquences vidéo poubelles et les séquences vidéos outils. Les séquences vidéos poubelles sont, par exemple, les plans noirs, bleus ou gris. Les séquences vidéos outils sont les mires et les claps : les mires permettent le calibrage de l'affichage d'un écran ou d'un téléviseur ou caractérisent également une absence de signal vidéo ; les claps, quant à eux, permettent d'identifier les plans d'un film et assurer la synchronisation du son et de l'image qui sont enregistrés sur des média séparés.
Ensuite, nous allons enlever la redondance temporelle : les rushes vidéos sont constitués de séquences fixes, par exemple sur un paysage, ou encore de séquences très lentes. Nous allons donc accélérer dynamiquement la vidéo afin de maximiser le contenu visuel par unité de temps.

1 Introduction

Les rushes d'un film sont constitués des documents originaux (bobines de film, bandes sons, ...) produits au tournage et issus de la caméra et de l'appareil d'enregistrement sonore. Ce sont les documents uniques, bruts, qui seront utilisés au montage et en postproduction. Le contexte particulier des rushes vidéo requiert une étape particulière due aux caractéristiques de ces vidéos. Une phase de prétraitement est donc réalisée afin de supprimer les séquences d'images poubelles et outils :

- Les séquences vidéos poubelles, c'est-à-dire, les plans noirs, bleus, ou gris, ou encore les séquences où le caméraman met sa main devant l'objectif de la caméra. Ces séquences vidéos ne contiennent aucune information et peuvent donc être supprimées du flux vidéo sans perdre d'information sur le contenu de la vidéo.

- Les séquences vidéos de mires ; une mire permet de calibrer l'affichage d'un écran ou d'un téléviseur avec des valeurs standardisées. On distingue la mire de type TDF (dégradés de gris, noirs et de couleurs et nom de la station) de la mire de barres (succession de couleurs

verticales). La mire caractérise également une absence de signal vidéo et est accompagnée d'un signal sonore continu de 1000 hz.

- Les séquences vidéos de claps : le clap est un outil utilisé au cinéma permettant d'identifier les plans d'un film lors du tournage et d'assurer la synchronisation du son et de l'image, enregistrés sur des média séparés. Il s'agit d'une ardoise composée de deux parties : une grande où sont inscrits le nom du film, de son réalisateur et du directeur de la photographie, le numéro du plan et de la scène, celui de la prise, l'effet (intérieur/extérieur, jour/soir/nuit) et éventuellement d'autres données techniques ; une plus fine que l'on rabat en produisant un claquement sec au moment où l'on commence à tourner de plus au montage, la synchronisation à l'image près du son et de l'image est rendue possible par la brièveté du claquement.

Après cette phase où les séquences vidéos outils et poubelles ont été enlevées, l'idée est de maximiser le contenu visuel par unité de temps, c'est-à-dire qu'une séquence vidéo montrant une seule information, par exemple une vue sur la mer, n'a pas besoin de durer plusieurs minutes. Seulement toutes les séquences ne sont pas trop longues par rapport à leur contenu informatif. La deuxième étape du prétraitement est donc de maximiser le contenu vidéo par unité de temps en accélérant dynamiquement la vidéo, les séquences actives ne vont pas être accélérées pendant que les séquences molles vont être accélérées, sachant que dans un plan, des séquences molles et actives peuvent se succéder.

Dans ce chapitre, nous commençons par donner un état de l'art des quelques méthodes permettant de détecter les séquences outils et les séquences poubelles. Ensuite, nous détaillerons notre méthode et évaluerons celle-ci. Dans un deuxième temps, nous nous intéresserons à l'accélération dynamique en commençant par parler de quelques méthodes puis en expliquant la notre. Finalement, nous évaluerons l'intégralité de notre système de prétraitement.

2 Etat de l'art

La détection des séquences poubelles et outils, tels que plans noirs, mires, claps ou autres est une tâche très spécifique aux rushes vidéos. Les outils développés sont donc récents. De plus, ils sont utilisés dans une étape de prétraitement, par conséquent les techniques proposées restent assez simples. Le caractère très différents des séquences à détecter mène à trois méthodes de détection indépendantes. Cependant quelques méthodes détectent les images poubelles et outils durant le processus [Beran 2008] : des caractéristiques de couleurs sont extraites, puis une classification est effectuée, finalement les groupes d'images poubelles et outils sont détectés.

La détection des séquences vidéos poubelle utilise l'idée que majoritairement ces images sont des images de couleurs uniformes. L'ensemble des techniques utilisent des caractéristiques visuelles essentiellement dans l'espace de couleur HSV. Par exemple, [Pan 2007] propose de

calculer la proportion de la teinte dominante d'une image pour effectuer une classification des images comme uniformes ou non uniformes grâce à une seuil préfixé. [Beran 2007] a choisi de définir un histogramme de couleur comme modèle d'une image poubelle, puis de définir les images poubelles par leur proximité, en terme de distance Euclidienne, à ce modèle. Une méthode assez répandue est celle de la détection des images poubelles par leur écart type de la distribution des pixels de l'image. [Valdés 2007] utilise cette méthode à l'échelle de l'image pendant que [Wang 2007, Hauptmann 2007] travaille sur l'image clé des plans.

Les méthodes de détection de claps sont plus variables. Cependant, la méthode la plus répandue reste une classification des images en se basant sur la caractéristique SIFT, [Noguchi 2008] utilise les SVM alors que [Wang 2007, Ellouze 2008, Wang 2008] effectuent une correspondance de points clés à partir d'un ensemble d'images de claps ayant des régions annotées manuellement, la figure 2.1 illustre des détections. Les auteurs de [Pan 2007] se basent sur la détection des lignes : si ces lignes forment un rectangle, c'est qu'il y a un clap. La détection des lignes est réalisée par la transformée de Hough, puis ces lignes sont classifiées selon 4 groupes : horizontales (de 0° à 30°), verticales (de 150° à 180°), droites (de 30° à 60°) et gauches (de 120° à 150°). Ce qui leur permet de détecter les rectangles, puis les rectangles doivent satisfaire l'une des deux contraintes suivantes : l'aire du rectangle est assez grande ou le rectangle contient une couleur dominante.

FIG. 2.1 – Détection des claps par correspondance de points clés.

Une autre piste explorée est l'utilisation da la spécificité sonore du clap, la figure 2.2 montre les caractéristiques audio d'un bruit de clap. [Beran 2008] classifie des segments sonore en utilisant l'énergie. [Wang 2008] ont choisi de détecter les claps en combinant les approches visuelle et sonore.

La méthode la plus classique pour la détection des mires reste de décomposer la vidéo en plans, puis de calculer la distance d'une ou plusieurs images clés avec un modèle de mire prédéfini. [Wang 2007, Hauptmann 2007] travaille dans l'espace de couleur RGB, alors que [Truong 2007a] et [?] utilisent un histogramme de six teintes se rapprochant de celle présentent dans les mires.

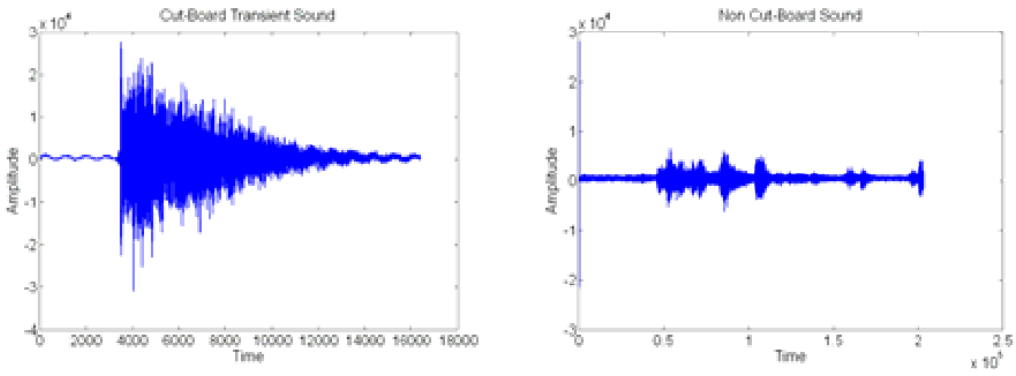


FIG. 2.2 – Caractéristiques d'une séquence audio
A gauche contenant un clap, à droite ne contenant pas de clap.

3 Filtrage des images inutiles

L'identification et la suppression des images poubelles et outils est une étape de prétraitement importante. Notre méthode se décompose en plusieurs étapes : dans un premier temps, nous effectuons une segmentation en plans de la vidéo nous permettant de supprimer les plans courts ne pouvant pas contenir d'information. Puis nous détectons les images poubelles et outils grâce à des méthodes indépendantes.

3.1 Séquences poubelles

3.1.1 Détection des plans poubelles

Pour effectuer la segmentation en plans de la vidéo, nous avons utilisé une méthode développée et présentée lors de la campagne d'évaluation ARGOS [Joly 2007]. Elle s'est appuyée sur la méthode proposée par [Billerbeck 2004]. Cette campagne a évalué la qualité du système pour la détection des transitions brusques et progressives : un taux de rappel et de précision de 0.95. Travaillant dans le contexte particulier des rushes vidéo, les seules transitions présentes sont les transitions brusques, nous avons donc adaptés cette méthode à notre problème.

Pour la détection des transitions brusques, sur les 16 régions d'une image, les 4 régions formant le centre de l'image sont négligées afin que la détection soit moins influencée par des mouvements rapides. Pour chaque image, la similarité entre celle-ci et les autres images de la fenêtre est calculée. Ensuite, les images sont classées par similarité croissante et le nombre

d'images précédant l'image courante étant classée sur la moitié supérieure est enregistré. Une possible transition brusque est détectée si ce nombre chute fortement et est validée si la différence entre l'image précédant la transition et l'image suivant la transition est suffissamment élevée.

Certains plans très courts ne présentent aucune information utile. Par exemple, entre deux prises, un rush vidéo présente quelques images noires, ou quelques images sans contenu visuel. Nous avons donc choisi de supprimer tous les plans courts. Un plan court est tout simplement défini comme un plan de taille inférieure à un seuil.

3.1.2 Détection des images poubelles

Par leur nature, les rushes vidéos contiennent des séquences d'images de couleur uniforme telle que noir, gris, bleu, c'est-à-dire des séquences d'images sans contenus visuels, comme les images de la figure 3.1. Mais d'autres séquences vidéo ne contiennent pas d'information telles que les séquences d'images de la figures 3.2.

FIG. 3.1 – Exemple d'images de couleurs uniformes

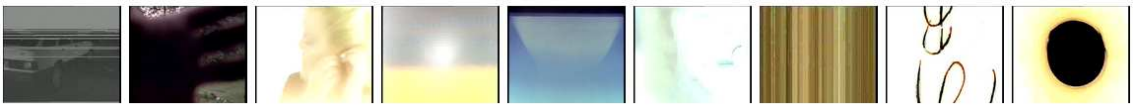


FIG. 3.2 – Exemple d'images diverses non informatives

Toutes ces images n'apportent pas d'information, et ne sont donc pas utiles dans le flux vidéo. Elles sont repérées pour ensuite être supprimées. De plus, la notion de vidéo apporte des indications utiles à la détection de telles séquences : un plan vidéo contient soit des images informatives, soit des images poubelles.

La détection des images poubelles est basée sur le fait qu'une telle image ne contient qu'une quantité d'informations limitée, par exemple un faible nombre de couleurs différentes. Mais, il est aussi important de prendre en compte l'information de la vidéo : toutes les vidéos n'ont pas le même décor, et éclairage, par conséquent une vidéo tournée en pleine journée avec un grand soleil contiendra plus de couleurs qu'une vidéo tournée au milieu de la nuit. Nous quantifions donc l'information contenue dans une image par l'entropie de la distribution des couleurs des pixels contenue dans celle-ci moins l'information moyenne de la vidéo. Puis, nous définissons un seuil permettant de délimiter les images considérées comme informatives à celles considérées comme poubelles.
Nous considérons une image I poubelle si et seulement si :

$$\frac{Ent(I)}{\frac{1}{N}\sum_{i=1}^{N} Ent(i)} < seuil \tag{3.1}$$

où N représente le nombre d'images de la vidéo et $Ent(i)$ l'entropie de Shannon de la distribution des couleurs des pixels de l'image i.

3.2 Séquences outils

3.2.1 Mires

Une mire permet de calibrer l'affichage d'un écran ou d'un téléviseur avec des valeurs standardisées. On distingue la mire de type TDF (dégradés de gris, noirs et de couleurs et nom de la station) de la mire de barres (succession de couleurs verticales). Dans notre cas, les rushes vidéo ne contiennent uniquement des mires de couleurs, comme dans la figure 3.3.

FIG. 3.3 – Exemples de mires présentes dans les rushes vidéo

Notre méthode de détection de mires se base sur la remarque que les mires sont des images très particulières et pour lesquelles nous définissons un vecteur caractéristique. Le vecteur caractéristique est un histogramme de couleur. Pour calculer cet histogramme, nous calculons l'histogramme moyen d'un ensemble d'images de mire. Puis, pour définir si une nouvelle image est une mire, nous définissons une distance entre l'histogramme de cette nouvelle image et le vecteur caractéristique, puis nous fixons un seuil qui définit la limite de la distance d'un vecteur mire au vecteur mire caractéristique.
Nous considérons une image I mire si et seulement si :

$$dist(\mathbf{I_{hist}}, \mathcal{M}_{hist}) < seuil \tag{3.2}$$

où $\mathbf{I_{hist}}$ est le vecteur caractéristique de l'image I, $\mathcal{M}_{hist}$ celui du modèle de mire, et $dist$ la distance utilisée.

3.2.2 Claps

Le clap est un outil utilisé au cinéma permettant d'identifier les plans d'un film lors du tournage et d'assurer la synchronisation du son et de l'image, enregistrés sur des média séparés. Il s'agit d'une ardoise composée de deux parties : une grande où sont inscrits le nom du film, de son réalisateur et du directeur de la photographie, le numéro du plan et de la scène, celui de la prise, l'effet (intérieur/extérieur, jour/soir/nuit) et éventuellement d'autres données techniques ; une plus fine que l'on rabat en produisant un claquement sec au moment où l'on commence à

tourner. Quelques exemples se trouve sur la figure 3.4. Au montage, la synchronisation du son et de l'image est rendue possible par la brièveté du claquement. Les numéros de scène et de plan permettent d'identifier les plans pour ordonner les rushes. Le numéro de prise permet, grâce aux rapports tenus par la scripte de repérer les prises jugées bonnes par le réalisateur au tournage.

FIG. 3.4 – Exemples de claps présents dans les rushes vidéo

Pour détecter les images de claps dans une vidéo, une méthode de classification supervisée est utilisée. Le modèle est entrainé sur un ensemble d'images représentant des claps et d'images diverses (mais ne contenant pas de clap) où chaque image est représentée par un histogramme de couleur de la région centrale.
Ensuite, nous appliquons le modèle à chaque image de la vidéo ce qui nous donne une valeur de confiance sur le fait que l'image contienne un clap. Mais il est à noter que dans une vidéo, une image de clap n'est pas isolée : une image sera catégorisée clap si la moyenne des confiances des images précédentes, des images suivantes et l'image courante est supérieure à un seuil.

4 Etude expérimentale

4.1 Base de données

Pour évaluer la qualité de la détection des images poubelles et outils, nous avons annotés un ensemble de 8 vidéos proposées par la campagne d'évaluation TRECVID 2008 ayant les descriptions du tableau 4.1. La première colonne est le nom des vidéos, la deuxième est le nombre d'images contenu dans la vidéo, les troisième, quatrième et cinquième sont le nombre d'image des différentes catégories, respectivement, poubelle, mire et clap. La dernière colonne représente le pourcentage d'images inutiles de la vidéo.

4.2 Evaluation

Pour évaluer la qualité des différentes détections, nous utilisons une courbe de rappel-précision. Le rappel représente la proportion d'images outils et poubelles retrouvées, et la précision représente la proportion d'images poubelles et outils correctement classées. Idéalement,

Vidéo	Nb images	poubelles	mires	claps	% parasites
MRS035126	46425	991	1554	2352	0.105
MRS048773	41265	2892	5471	0	0.203
MRS151585	24856	3272	2913	2296	0.341
MRS157479	54522	429	3746	1150	0.098
MRS044499	18585	1022	533	612	0.117
MRS145229	20325	689	1305	906	0.143
MRS157450	54585	5323	3767	1359	0.191
MS206370	23310	3313	1554	1495	0.273
Total *soit*	283 873 *3 h 9 min 15 sec*	17931 *11 min 57 sec*	20843 *13 min 54*	10170 *6 min 47*	0.172

TAB. 4.1 – Nombre d'images inutiles par rapport au nombre d'image total dans les vidéos de tests.

nous voudrions qu'un système donne de bons taux de précision et de rappel en même temps. Un système qui aurait une précision de 1 et un rappel de 1 signifie qu'il classe correctement toutes les images. La courbe rappel - précision est un ensemble de points (rappel, précision) obtenus en faisant varier le seuil.

Une deuxième évaluation est effectuée pour laquelle nous utilisions la notion d'éléments d'histoire proposé par la campagne TRECVID : ces éléments représentent les séquences de la vidéo où il se passe quelque chose d'intéressant et devant être conservés. Ils sont choisis et identifiés par un humain. La deuxième mesure d'évaluation calcule la précision, par la proportion d'image n'appartenant pas à un élément d'histoire correctement enlevées. Elle permet d'évaluer l'influence des images enlevées à tort : certaines images sont enlevées mais cela ne touche pas aux éléments d'histoires, et par conséquent même si le détecteur fait une erreur, cela n'influence pas la qualité de la vidéo nettoyée.

4.3 Plans poubelles

Les plans très courts sont supprimés, cependant, il reste à définir la longueur d'un plan court. Nous avons donc fait varier le seuil délimitant un plan très court et évalué les résultats. Le but est de déterminer le meilleur seuil permettant de supprimer les images poubelles sans perdre d'images d'éléments d'histoire. Le rappel et la précision sont donc calculés à partir de l'ensemble des images poubelles. Cette étape ne permet pas d'obtenir un rappel élevé, le but étant juste de supprimer quelques séquences poubelles grâce à la détection de plans. La courbe 4.1 montre les résultats obtenus, la courbe en pointillés bleus est le rappel - précision des images poubelles ; en rouge, la courbe rappel - précision des images d'éléments d'histoire.

Nous pouvons voir grâce à cette courbe qu'il est possible de trouver un bon compromis entre la suppression d'un maximum d'images poubelles sans pour autant perdre les éléments d'histoire d'une vidéo. Le meilleur compromis se situe au point de rappel égal à 0.153, pour une précision de 0.663 / 0.993. Ce qui correspond à une longueur de 100 images, soit 2 secondes.

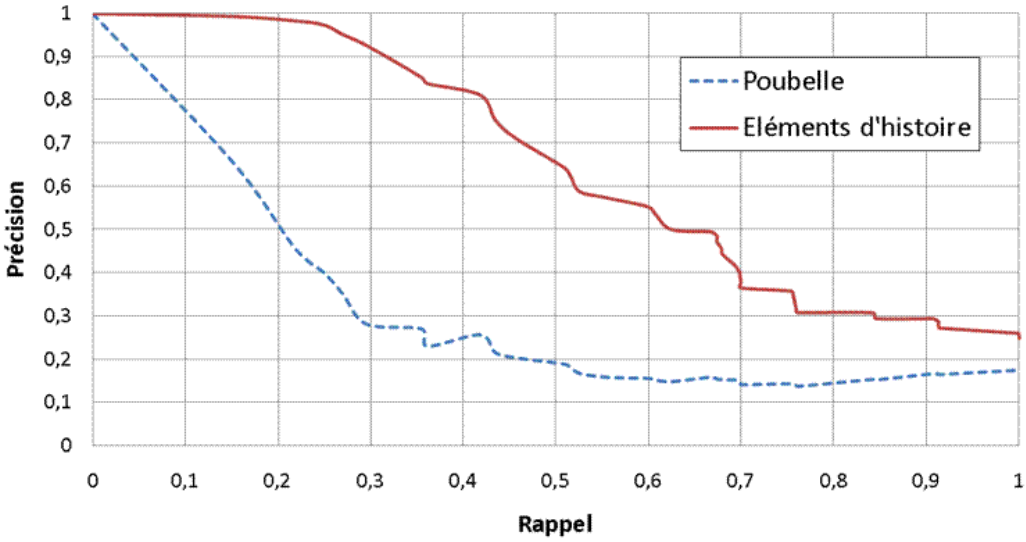


FIG. 4.1 – Détection des plans poubelles

4.4 Images poubelles

Dans la méthode que nous proposons, il reste à définir le meilleur espace de couleur, ainsi que le seuil. Nous avons donc testé notre méthode sur 3 espaces de couleurs différents, HSV, RGB, YUV, ainsi que sur les niveaux de gris. Nous avons utilisé une division des espaces couleurs en 64 tranches. Les résultats sont présentés par la courbe de la figure 4.2 qui montre le rappel - précision des images poubelles.

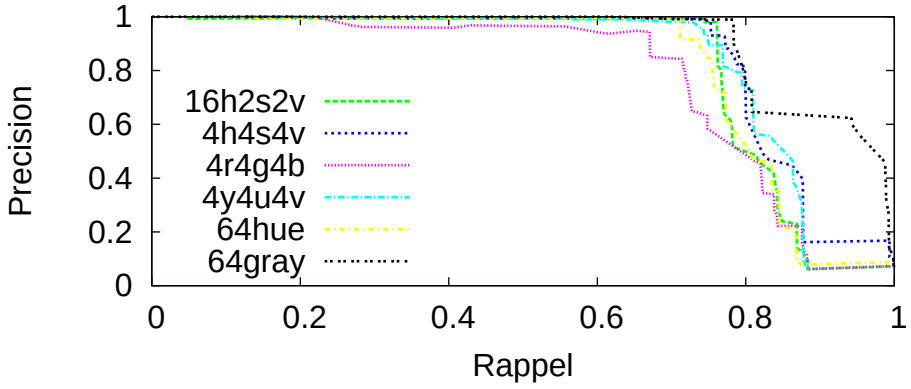


FIG. 4.2 – Détection des images poubelles pour différents espace de couleurs

L'espace le plus performant pour ce problème est les niveaux de gris. Nous avons donc affiné les expériences en niveau de gris en utilisant différentes valeurs pour le nombre de tranches : 4, 8, 16, 32, 64, 128, 256, 512. Les résultats sont présentés dans la figure 4.3 qui montre le rappel - précision pour différents nombre de niveau de gris.

La figure 4.4 montre les résultats des deux méthodes d'évaluation pour un espace de couleur en 256 niveaux de gris grâce au rappel en fonction de la précision liée aux images poubelles et la

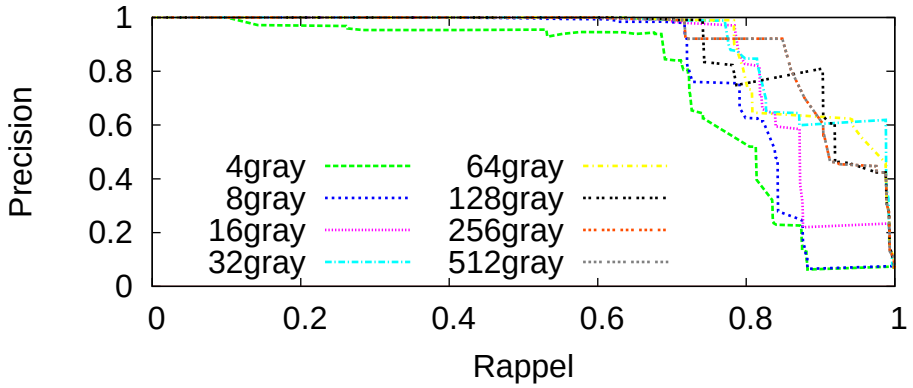


FIG. 4.3 – Détection des images poubelles pour différents niveaux de gris

précision liée aux images d'éléments d'histoire. Ce second taux de précision permet de montrer que les images perdues ne font en majorité pas partie des séquences dites intéressantes. Il est donc possible d'obtenir un bon compromis au point de rappel 0.849, pour une précision de 0.921 / 1, ce qui correspond à un seuil de -1.42.

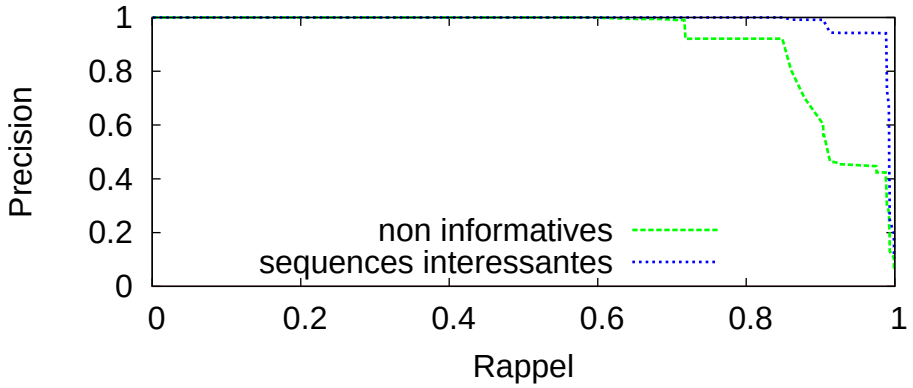


FIG. 4.4 – Détection des images poubelles

4.5 Images de mires

Pour évaluer la qualité de la détection, nous utilisons un taux de rappel-précision. Le rappel représente la proportion d'images de mire retrouvées et la précision représente la proportion d'images de mire correctement classées. Le but est d'enlever toutes les séquences de mires de la vidéo, les résultats présentés ont donc un rappel de 1.
Pour créer le modèle du vecteur mire, nous avons utilisés un ensemble de 28905 images de mire. Nous avons utilisé 3 espaces de couleurs différents, HSV, RGB, YUV, et les niveaux de gris, utilisé des histogrammes de couleurs de 64 tranches et testé avec les distances Euclidienne, Manhattan, Bhattacharyya. Les résultats sont présentés par le graphique 4.5 par la précision pour différents espace de couleur, et différentes distances pour un rappel égal à 1. L'espace de couleur le plus performant pour ce problème est HSV, mais en utilisant que la composante Hue,

qui signifie teinte et la meilleure distance est Bhattacharyya.

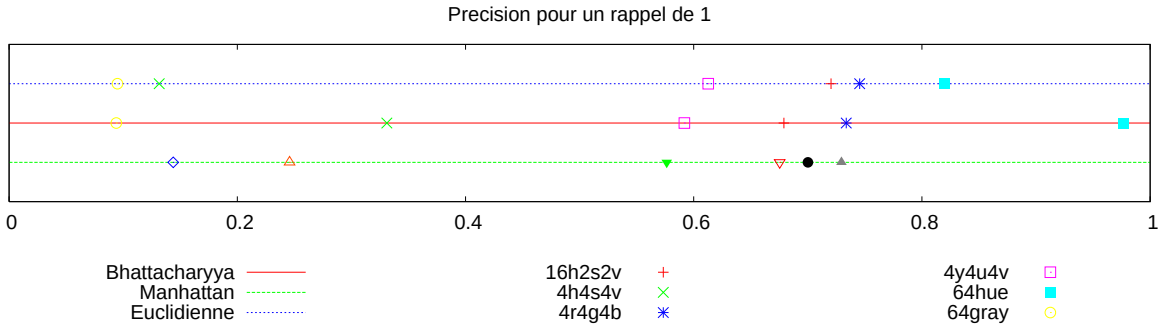


FIG. 4.5 – Détection des images mire.

Nous avons approfondi les expériences en testant plusieurs nombres de tranches : 4, 8, 16, 32, 64, 128, 256, 512, mais cela ne donne pas d'amélioration. Le meilleur détecteur est donc de travailler sur la teinte d'une image en 64 tranches en utilisant la distance de Bhattacharyya. Nous obtenons un rappel de 1, pour une précision de 0.98 / 1, le seuil est fixé à 0.7.

4.6 Images de claps

Pour entrainer le modèle de détection des claps, nous avons utilisés 9972 images de claps et 15467 images diverses. De plus, nous utilisons l'information des 12 images précédentes et des 12 images suivantes pour prédire la présence d'un clap sur une image. Nous avons utilisé 3 espaces de couleurs différents, HSV, RGB, YUV, et les niveaux de gris, utilisé des histogrammes de couleurs de 64 tranches et testé avec différents noyaux : linéaire, gaussien, polynomial, sinusoïdal. Les meilleurs résultats sont obtenus avec un noyau polynomial, et l'espace de couleur HSV divisé en 12 pour H et 2 pour S et V. La courbe rappel - précision du graphique 4.6 montre les résultats obtenus. Les résultats ne sont pas de grandes qualités, nous choisissons donc de garder des claps plutôt que perdre des images intéressantes. Le meilleur compromis se trouve au point de rappel 0.170, pour une précision de 0.486 / 1, ce qui correspond à un seuil de -2.75.

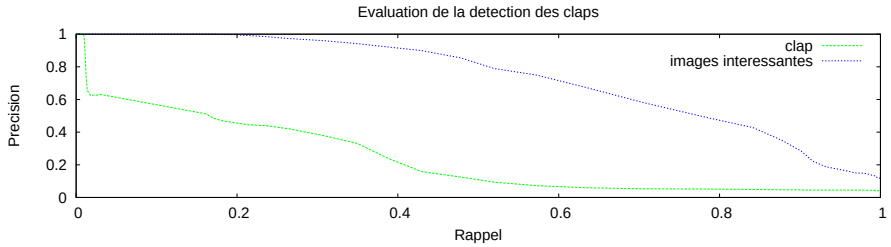


FIG. 4.6 – Détection des images de claps

5 Accélération dynamique

Les rushes vidéos présentent une grande redondance temporelle : il existe un fossé entre une scène entière d'un rush qui peut durer plusieurs minutes et la séquence retenue lors du montage qui va durer seulement quelques secondes. Les rushes vidéos contiennent des séquences où il n'y a pas d'action, par exemple, un paysage est filmé, les acteurs sont en place et écoutent les instructions du réalisateur. Les méthodes d'exploration de rushes et de résumé de rushes présentées dans les parties suivantes ne prennent pas en compte la bande sonore, nous proposons donc d'effectuer une accélération de la vidéo. Les remarques précédentes montrent bien que certaines séquences vidéos doivent être accélérées plus rapidement que d'autre, pour cette raison, l'accélération réalisée n'est pas linéaire, mais dynamique afin de maximiser le contenu visuel par unité de temps.

Dans [Doulamis 2000b], les auteurs présentent un algorithme permettant l'extraction d'un ensemble d'images représentatives pour un plan donné basé sur les variations temporelles. Ils calculent la magnitude de la seconde dérivée des vecteurs caractéristiques basés sur la couleur par rapport au temps, puis sélectionnent uniquement les images correspondant à un minimum ou un maximum local.
Dans la même idée, c'est-à-dire, sous-échantillonner les images d'un plan afin d'accélérer la vidéo, les auteurs de [Peker 2001] utilisent la moyenne de la magnitude des vecteurs mouvement d'une image pour mesurer l'activité de celle-ci. Les images sont ensuite sous-échantillonnées en deux catégories : celle ayant une forte activité, et celle ayant une faible activité, la limite étant déterminée par un seuil. Seules les images du premier ensemble sont sélectionnées.

Les séquences vidéos contiennent une très grande redondance statistique, aussi bien dans le domaine temporel que dans le domaine spatial. Beaucoup de travaux ont été réalisés dans le domaine de la compression vidéo, mais la littérature actuelle ne permet pas une évaluation générique des systèmes d'accélération dynamique.

Dans un premier temps, nous devons définir l'accélération voulue pour toute la vidéo, c'est-à-dire l'accélération moyenne désirée $ACCmean$. De même, une accélération maximale doit être définie $ACCmax$ afin que les séquences durant une minute dans la vidéo initiale ne soit pas réduite à une seule image et par conséquent que la séquence disparaisse.
L'idée de l'accélération dynamique est d'accélérer les séquences vidéos proportionnellement à leur activité, c'est-à dire que les séquences calmes doivent être plus accélérée que les séquences d'action. Une notion d'activité entre deux images est donc utilisée et calculée pour chaque image par rapport à l'image précédente.
Pour une accélération linéaire de n, une image est sélectionnée toutes les n images. Dans notre cas, n est variable. Une image t est sélectionnée, puis la prochaine image t' sélectionnée est celle dont la somme des activités des images t à t' est supérieur à $jump(v)$ ou que $|t'-t| >= ACCmax$,

avec :

$$jump(v) = \sum_{f \in v} act(f)/F * ACCmean \tag{5.1}$$

6 Evaluation expérimentale

Nous avons effectué la fusion des détecteurs, c'est-à-dire que pour chaque image de la vidéo, si celle-ci a été détectée comme étant une image poubelle ou outil par un de nos détecteur, alors elle est supprimée. Ensuite, nous avons effectué l'accélération dynamique avec une accélération moyenne $ACCmean$ de 3, et une accélération maximale $ACCmax$ de 5.
Nous avons effectué une évaluation globale de notre système de prétraitement : nous avons appliqué le système sur nos 8 vidéos annotées, puis évalué la qualité finale. Le tableau 6.1 propose une vision globale des résultats : pour chaque vidéo, le rappel, la précision des images inutiles et la précision des images d'éléments d'histoire du système sont montrés, ainsi que le pourcentage de la vidéo finale par rapport à la vidéo initiale.

	Rappel	Précision (inutile)	Précision (élts histoire)	Pourcentage gardé
MRS035126	0.839323	0.789662	0.990779	28.39%
MRS151585	0.776994	0.887603	0.981896	23.90%
MRS157479	0.544304	0.986042	1	32.32%
MRS044499	0.935440	0.999581	1	24.66%
MRS145229	0.824629	0.987378	0.998643	26.55%
MRS157450	0.892716	1	1	29.55%
MS206370	0.844217	0.938779	0.974338	24.89%
MRS048773	0.793140	0.948718	1	32.04%
Moyenne	0.806345	0.942220	0.993207	27.79%

TAB. 6.1 – Evaluation globale du système de prétraitement pour les vidéos tests.

Les résultats du tableau sont très satisfaisants, nous pouvons voir que rares sont les images supprimées à tort, la précision moyenne est de 0.94 / 0.99, c'est-à-dire que parmi les images supprimées à tort, un très faible nombre représente des éléments d'histoire. Le rappel moyen est de 0.81, c'est-à-dire que 80% des images inutiles sont correctement supprimées. La figure 6.1 montre des exemples de mauvaises classifications des détecteurs.

Nous avons testé l'intégralité de notre méthode de prétraitement sur un ensemble de 7 vidéos complètement indépendantes des vidéos précédentes, et pour lesquelles seules les éléments d'histoire sont annotés. Le tableau 6.2 montre les résultats obtenus : pour chaque vidéo, la précision du système est montré, ainsi que le pourcentage de la durée de la vidéo finale par rapport à la vidéo initiale.

(a) Détecteur de mires : exemple de faux positifs

(b) Détecteur d'images non informatives : exemple de faux négatifs

(c) Détecteur d'images non informatives : exemple de faux positifs

(d) Détecteur de claps : exemple de faux négatifs

(d) Détecteur de claps : exemple de faux positifs

FIG. 6.1 – Exemples de mauvaises classifications

Vidéo	Précision (élts histoire)	Pourcentage gardé
MRS035132	1	28.96%
MRS043400	0.999441	30.43%
MRS150148	0.999952	29.29 %
MRS157443	1	30.37%
MRS157475	1	32.47%
MS212920	1	27.08%
MS237650	1	24.22%

TAB. 6.2 – Evaluation globale du système de prétraitement

7 Conclusion

Dans ce chapitre, nous avons présenté une méthode de détection des séquences vidéos inutiles qui se répartissent en deux grandes catégories : les images poubelles, c'est-à-dire, les images toutes noires, grises, bleues ainsi que les séquences parasites ; les images outil composées de mires qui permettent le calibrage de l'affichage d'un écran ou d'un téléviseur ou caractérisent également une absence de signal vidéo et de claps qui permettent d'identifier les plans d'un film et d'assurer la synchronisation du son et de l'image qui sont enregistrés sur des média séparés. Nous avons aussi proposé une méthode d'accélération dynamique permettant d'enlever la redondance temporelle en maximisant le contenu visuel par unité de temps.

Une expérimentation complète a permis de mettre en avant les meilleurs critères de sélection de caractéristiques, outils ou encore les seuils. L'étude sur l'échantillon test n'est pas complète, mais l'essentiel est montré : la qualité de la détection des images poubelles et outils n'altère pas le contenu visuel des vidéos originales malgré l'accélération effectuée. Les résultats montrent une bonne qualité de détection, en particulier pour les images non informatives et surtout pour les mires. La détection des claps restent le point faible de cette phase de prétraitement.
Il est important de prendre en compte la notion de prétraitement, c'est-à-dire que cette étape n'est pas l'étape primordiale d'un algorithme, c'est pour cette raison que nous avons choisi de privilégier une complexité temporelle et spatiale faible. Nous aurions pu extraire un maximum de caractéristiques puis les fusionner pour améliorer les résultats, mais cela aurait grandement augmenter la complexité en temps et en espace, pour cette raison, nous avons privilégié l'utilisation d'une seule caractéristique.

Le but de cette phase est de nettoyer une vidéo de rush. Ce prétraitement va être utilisé comme point de départ et principal primitif pour les algorithmes développé pour les rushes, la qualité d'un tel algorithme est dépendant de la qualité de ce prétraitement. Il serait donc intéressant de s'attarder sur l'amélioration de la détection des claps, en choisissant des caractéristiques plus adaptées, ou encore en utilisant plus efficacement l'information apportée par la vidéo. Une excellente détection des claps permettrait pour la suite d'extraire des informations très importantes : les claps donnent des informations sur le rush vidéo, c'est-à-dire le nom du film, de son réalisateur et du directeur de la photographie, le numéro du plan et de la scène, celui de la prise, l'effet (intérieur/extérieur, jour/soir/nuit) et éventuellement d'autres données techniques. Toutes ces informations permettraient certainement de donner d'excellentes indications *a priori* pour les algorithmes subséquents à cette phase de prétraitement.

Troisième partie

Dictionnaire visuel

Dictionnaire visuel

La première exploitation de rushes vidéo proposée fut un système de recherche de plans vidéo par requête de mots clés visuels. Ce système est basé sur le paradigme inspiré de la recherche documentaire et la recherche de plans vidéo. En utilisant l'ensemble des plans vidéo des rushes, nous construisons un dictionnaire visuel global, c'est-à-dire un dictionnaire de l'ensemble des mots visuels des plans vidéo. A partir de cet ensemble de mots visuels, nous sélectionnons les plus représentatifs afin de construire un dictionnaire visuel utilisé pour effectuer des requêtes. Un utilisateur peut alors composer des requêtes de mots visuels, puis le système sélectionne les plans vidéo les plus discriminants et les renvoie à l'utilisateur. Une méthode d'évaluation automatique est proposée afin d'évaluer la qualité du système.

1 Description générale

1.1 Introduction

"Un bon croquis vaut mieux qu'un long discours". Cet adage, attribué à Napoléon Bonaparte, fait allusion à l'idée que les histoires complexes peuvent être décrites avec une image, ou qu'une image peut se substituer à une quantité substantielle de mots. C'est pour cela que les images sont très souvent utilisées pour illustrer du texte. Or, grâce au développement des outils numériques tels que les appareils photos ou encore les caméras vidéos, elles sont de plus en plus nombreuses et il devient très difficile de faire une recherche efficace par mots clés dans une grande base d'images non classées. Par exemple, Google [5] propose un outil de recherche d'images qui se base uniquement sur le texte associé à l'image, il est donc impératif que le texte entourant l'image corresponde effectivement au contenu visuel de l'image. Ce système est assez efficace sous ces conditions, mais les limites sont vite atteintes, même avec des mots clés assez précis ; par exemple, la requête "hamster" nous renvoie le résultat suivant (voir figure 1.1) :

La recherche actuelle se focalise donc sur les systèmes de recherche d'images par le contenu visuel appelé CBIR [6] qui ont fait de grand progrès ces dernières années en ce qui concerne la recherche d'images visuellement similaires. La recherche de plans vidéo est directement liée

[5] http ://images.google.fr/
[6] Content-Based Image Retrieval

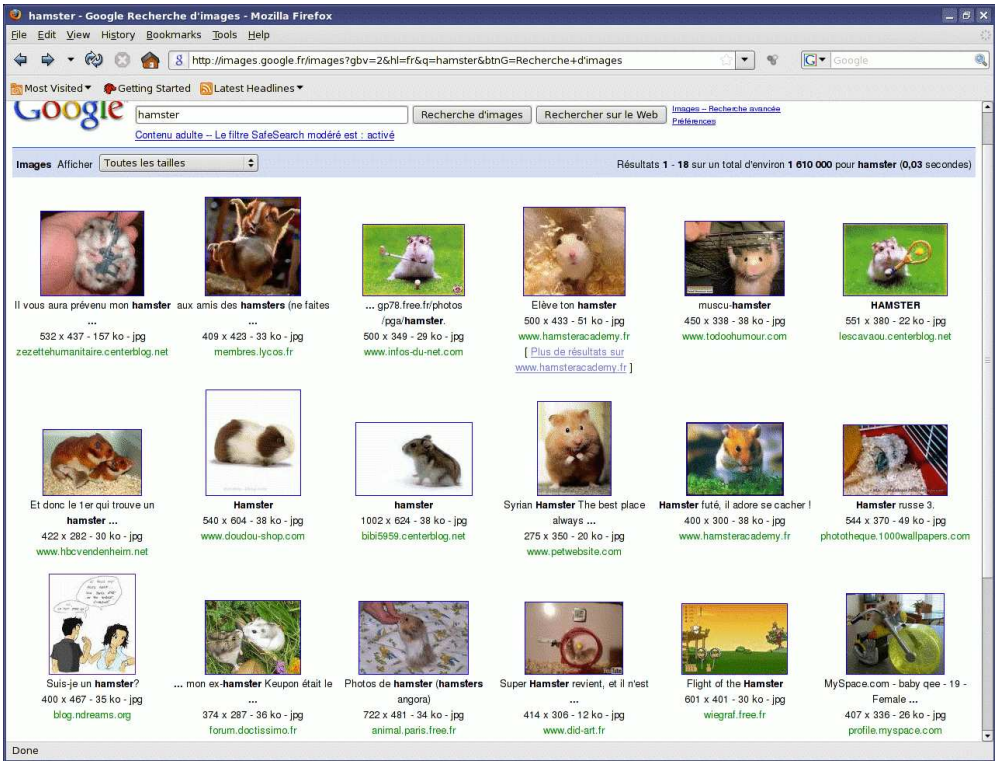


FIG. 1.1 – Résultat de la requête “hamster” sous google image

à l'évolution de la recherche d'images. Cependant, les systèmes de recherche par le contenu sémantique restent moins performants. Une des raisons de cette différence provient de la description des plans vidéo : ils sont décrits par leur contenu numérique, c'est-à-dire par la couleur de chacun des pixels des images les composant. La distribution de la couleur est une caractéristique très utilisée pour la représentation de plans vidéos. Il s'agit vraisemblablement du descripteur le plus répandu en indexation. Il est en théorie invariant aux translations et rotations, et change seulement légèrement en cas de changements de la prise de vue ou de l'échelle, alors qu'un plan vidéo pourrait être décrit par son contenu sémantique, c'est-à-dire par sa signification. La corrélation entre le contenu numérique et le contenu sémantique d'un plan vidéo reste difficile à interpréter, c'est ce qui est appelé “le fossé sémantique”.

Ce chapitre est organisé comme suit : dans un premier temps, nous allons expliquer le contexte de cette recherche, ensuite nous rappellerons les méthodes les plus pertinentes d'utilisation d'un dictionnaire visuel. Puis, nous expliquerons notre méthode pour construire et optimiser le dictionnaire visuel. Alors, nous proposerons une méthode d'évaluation pour notre dictionnaire visuel. Et enfin, nous évaluerons la qualité de notre dictionnaire visuel sur les données de TRECVID.

1.2 Motivation

La campagne d'évaluation TRECVID 2006 [7] a proposé une nouvelle tâche : l'exploitation de rushes vidéos. Les groupes participant à cette tâche devaient développer un outil fondamental pour l'exploration de rushes vidéo tout en démontrant la qualité de cet outil. L'outil proposé ne doit pas forcément être entièrement interactif et complet.
Les objectifs minimaux exigés de cet outil sont la capacité à :

- enlever la redondance des rushes vidéos autant que possible
- présenter / organiser le matériel non redondant selon plusieurs caractéristiques.
- exécuter sa propre évaluation et présenter les résultats

Le but de cette tâche pour l'année 2006 est de fournir des systèmes permettant d'ouvrir le débat sur la planification d'une tâche plus stricte pour les années futures.

Nous avons donc choisi de proposer un système de recherche de plans vidéos adapté aux rushes utilisant le succès des méthodes de recherche de documents textuels par mots clés afin d'effectuer un parallèle avec la recherche de plans vidéos par mots clés visuels. Cependant, les séquences d'images différent significativement des documents textuels aussi bien syntaxiquement que sémantiquement dans leur mode de représentation et leur façon d'exprimer l'information. Effectuer le paradigme entre la recherche textuelle et visuelle n'est donc pas trivial.

Dans le domaine des documents textuels, la représentation ordinaire de ceux-ci est un vecteur où chaque composante est relative à l'importance d'un certain mot dans un document. Nous allons choisir de représenter un plan vidéo par une seule image du plan appelée image clé. Une image clé est décrite par un vecteur où chaque composante représente l'importance d'un certain mot visuel pour cette image.

Pour la recherche de documents textuels, un utilisateur tape des mots pour composer une requête et le système retourne une liste de documents où les documents sont classés par pertinence. Pour la recherche de plans vidéos, nous proposons à l'utilisateur d'effectuer une requête composée de mot visuels choisis parmi un dictionnaire visuel, puis le système retourne une liste de plans vidéos classés par pertinence.

Pour créer notre dictionnaire visuel, nous effectuons deux étapes.

- Premièrement, nous créons un Dictionnaire Visuel Global (DVG) contenant tous les mots visuels provenant de l'ensemble d'entrainement.
- Deuxièmement, un nouvel ensemble appelé Dictionnaire Visuel de Requête (DVR) est construit en limitant le nombre de mots visuels afin que l'utilisateur puisse choisir facilement les mots pour construire et constituer des éléments les plus pertinents.

Par l'utilisation du DVR, une image peut être codée comme un vecteur du nombre d'éléments visuels le constituant. Basées sur cette représentation de plan vidéo, les techniques du domaine textuel peuvent être généralisées au domaine visuel.

[7] http ://www-nlpir.nist.gov/projects/trecvid/

1.3 Etat de l'art

La recherche d'images et de vidéos est un domaine très actif de la recherche grâce à la croissance continuelle de vidéos, collections de photos, médias ... Le succès phénoménal de la recherche de site internet augmente l'intérêt de rechercher de nouvelles solutions pour la recherche de documents visuels. Actuellement, les approches proposées pour la recherche d'images ou de vidéos prennent en compte les médias associés, par exemple, les transcriptions de discours, les sous-titres, ... Cependant, de telles informations textuelles ne peuvent pas être automatiquement fournies avec une image ou une vidéo. Il est donc important de développer des techniques basées sur d'autres aspects tels que le contenu visuel ou audio.

La recherche d'images par le contenu, en anglais : Content Based Image Retrieval (CBIR) [Carson 1999, Gong 1996, Smeulders 2000], est une technique visant à effectuer des recherches d'images à l'aide de requêtes portant sur les caractéristiques visuelles d'une image : texture, couleur, forme... La figure 1.2 montre l'architecture d'un système de recherche par la contenu.

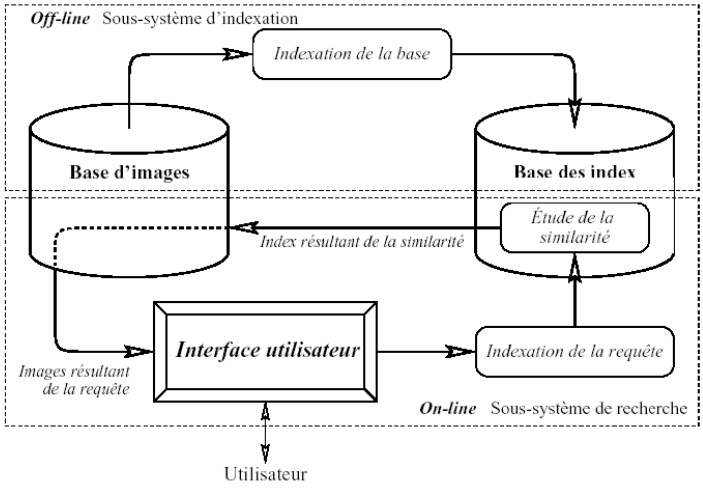


FIG. 1.2 – Architecture d'un système de recherche par le contenu.

Le cas typique d'utilisation de ces systèmes est lorsque l'on dispose d'une image pour laquelle on souhaiterait obtenir des images visuellement similaires. Il s'oppose à la recherche d'images par mots clés, qui est typiquement ce qui est proposé actuellement par les moteurs de recherche tels que Google ou Yahoo !, où les images sont retrouvées en utilisant le texte qui les entoure plutôt que le contenu de l'image elle-même. Du fait des caractéristiques calculées, qui sont de bas-niveau, ces techniques obtiennent des résultats satisfaisant pour certains types de requêtes et certains types de base d'images. Par exemple rechercher des images de paysages enneigés, parmi une base d'image de paysages. Toutefois ces systèmes rendent souvent des réponses extravagantes, et souvent éloignées de l'idée qu'avait l'utilisateur lorsqu'il a soumis sa requête. Ce genre de système permet aussi de rechercher des images sans forcément avoir une image requête, par exemple rechercher des images plutôt bleues, ou alors dessiner une forme et demander de chercher toutes les images qui possèdent un objet de forme similaire. La figure 1.3 montre différents exemples de requête. Mais ces méthodes ne peuvent pas capturer l'information sémantique dans les images.

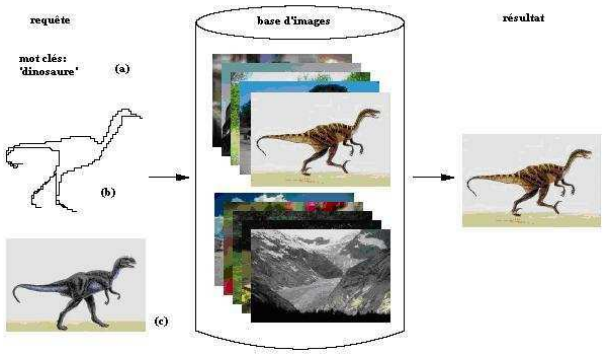


FIG. 1.3 – Exemple de requête.

Picard fut le premier à développer le concept général d'un dictionnaire visuel en transformant l'idée principale de dictionnaire textuel à un dictionnaire visuel [Picard 1995]. Un an plus tard, elle a proposé des exemples de dictionnaires visuels fondés sur la texture, en particulier le système de FourEyes [Picard 1996]. Mais aucune expérience n'a été réalisée pour montrer la qualité de ces systèmes.

Une première méthode consiste dans la construction d'un dictionnaire visuel des vecteurs de caractéristiques provenant de la segmentation d'images en régions. Dans [Zhang 2004], les auteurs utilisent un SMO [8] pour choisir des éléments visuels ; dans [Lim 1999] des SVMs [9] sont entraînés sur des régions d'image d'un petit ensemble d'images appartenant à sept catégories sémantiques et dans [Fauqueur 2003], les régions sont regroupées par la similarité de leurs caractéristiques visuelles avec une classification hiérarchique. Les images sont alors représentées comme des vecteurs basés sur ce dictionnaire. Le contenu sémantique de ces éléments visuels dépend principalement de la qualité de la segmentation.

Des régions affines elliptiques sont représentées par la caractéristique SIFT [10]. Les régions détectées dans chaque image de la vidéo sont suivies, et l'estimation du descripteur pour une région est calculée en faisant la moyenne des descripteurs à travers le suivi. Et enfin, pour créer le vocabulaire visuel, ils utilisent l'algorithme K-Moyenne pour regrouper les éléments dans [Sivic 2003]. Cette méthode ne peut pas être utilisée sur une série d'images et exige une méthode de suivi.

Pour créer un dictionnaire visuel, les auteurs de [Zhu 2000] segmentent les images en blocs et utilisent une combinaison entre l'algorithme Generalized Lloyd et Pairwise Neighbor sur un

[8] Sequential Minimal Optimization
[9] Support Vector Machine
[10] Scale-Invariant Feature Transform

ensemble d'entrainement. Les résultats présentés utilisent des blocs de très petites tailles (moins de 4x4 pixels), ces blocs ne peuvent pas être utilisés comme des requêtes visuelles.

2 Dictionnaire visuel

Afin de pouvoir appliquer le paradigme de la recherche de documents textuels à la recherche de documents visuels, nous devons créer un dictionnaire visuel permettant à un utilisateur de composer des requêtes. Le dictionnaire est composé de mots visuels qui doivent être différents et en nombre limité afin de faciliter la tâche de l'utilisateur. Notre approche pour créer le dictionnaire visuel est donc décomposée en deux étapes :

- La création d'un dictionnaire visuel global **DVG** : dans la recherche de documents textuels, les documents sont composés de plusieurs milliers de mots. Un utilisateur connait l'ensemble des mots et les utilise pour composer une requête sans avoir accès à l'ensemble des mots du dictionnaire. Dans le cas de la recherche de plans vidéo, il n'existe pas de dictionnaire universel, il faut donc créer un dictionnaire global.
- La création d'un dictionnaire visuel de requête **DVR** : un utilisateur ne peut pas connaitre l'ensemble des éléments d'un dictionnaire visuel global, et les "taper" comme requête, il est donc important de proposer un dictionnaire de taille réduite à un utilisateur. Ce dictionnaire doit comporter les mots les plus discriminants.

2.1 Plans vidéo

Les documents vidéos sont hiérarchiquement structurés en scènes, plans et images. Les plans sont définis comme des séquences continues d'images prises sans arrêter la caméra. Une scène est définie comme une suite de plans contigus qui sont sémantiquement reliés. Le but de le segmentation en plans est de trouver une méthode de détection automatique des plans dans la vidéo. Pour cela, il faut identifier les effets de transitions. La segmentation en plan et l'identification des effets de transition fournissent assez peu d'éléments directement exploitables. Il s'agit cependant d'un traitement préliminaire d'un bon nombre d'outils d'analyse.

2.1.1 Détection des transitions de plans

Travaillant dans le contexte particulier des rushes vidéos, les seules transitions présentes sont les transitions brusques ; nous avons adapté la méthode du système de fenêtrage proposée par [Billerbeck 2004] à notre problème. Pour la détection des transitions brusques, sur les 16 régions d'une image, les 4 régions formant le centre de l'image sont négligées afin que la détection soit moins influencée par des mouvements rapides. Pour chaque image, la similarité entre celle-ci et les autres images de la fenêtre est calculée. Ensuite, les images sont classées par similarité croissante. Le nombre d'images précédants l'image courrante classées sur la moitié supérieure, est enregistrée. Une possible transition brusque est détectée si ce nombre chute fortement et est

validée si la différence entre l'image précédant la transition et l'image suivant la transition est suffisament élevée.

2.1.2 Sélection d'images clés

Une séquence vidéo est une succession d'images redondantes, nous utilisons donc un processus de simplification d'une séquence vidéo qui consiste à ne sélectionner qu'une ou plusieurs images représentatives pour une séquence vidéo.
Idéalement, les images clés devraient être les images ayant le contenu sémantique le plus fort, mais les techniques actuelles ne le permettent pas ; les algorithmes actuels utilisent les caractéristiques brutes des images, tels que textures, couleurs, mouvement. Beaucoup de techniques ont été proposées, des techniques très simples telle que celle proposée par [Tonomura 1993] qui consiste à ne sélectionner que la première image de chaque plan, ou encore [Ueda 1991] qui représente un plan par la première et la dernière image ; des techniques un peu plus complexes telle [Ferman 1997] qui sélectionne l'image la plus proche du centre du groupe formé par les images du plan. Plus récemment, des méthodes plus complexes ont été développées. Mais finalement les méthodes empiriques sélectionnant la première image, l'image du milieu et la dernière image du plan restent les plus utilisées.

2.2 Mot visuel

Notre approche se fonde sur l'idée d'utiliser un nombre fixe et limité d'éléments visuels. Ces éléments visuels doivent être calculés de manière automatique, et une séquence d'images doit pouvoir être décrite grâce à ces éléments visuels. Ils doivent être interprétables par l'utilisateur : lorsque l'utilisateur effectue une requête de mots visuels, il perçoit la relation entre la requête et les plans vidéos recherchés afin d'effectuer une requête efficace. Un mot visuel est un morceau d'image rectangulaire qui peut-être interprété comme un concept par un utilisateur. La figure 2.1 montre des exemples de mots visuels : à gauche, des mots visuels basés sur la caractéristique couleur, et à droite sur la caractéristique texture.

FIG. 2.1 – Exemple de mots visuels.

Les mots visuels sont crées dans le but de représenter des concepts sémantiques pour effectuer des requêtes. Par exemple, si un utilisateur cherche un plan vidéo contenant de l'eau, la forêt et le ciel, alors il choisira trois éléments : un par concept. Chacun sera identifié par un élément visuel donné. L'illustration de ce processus est présenté par la figure 2.2. L'exemple montre bien l'importance du choix de la taille du bloc.

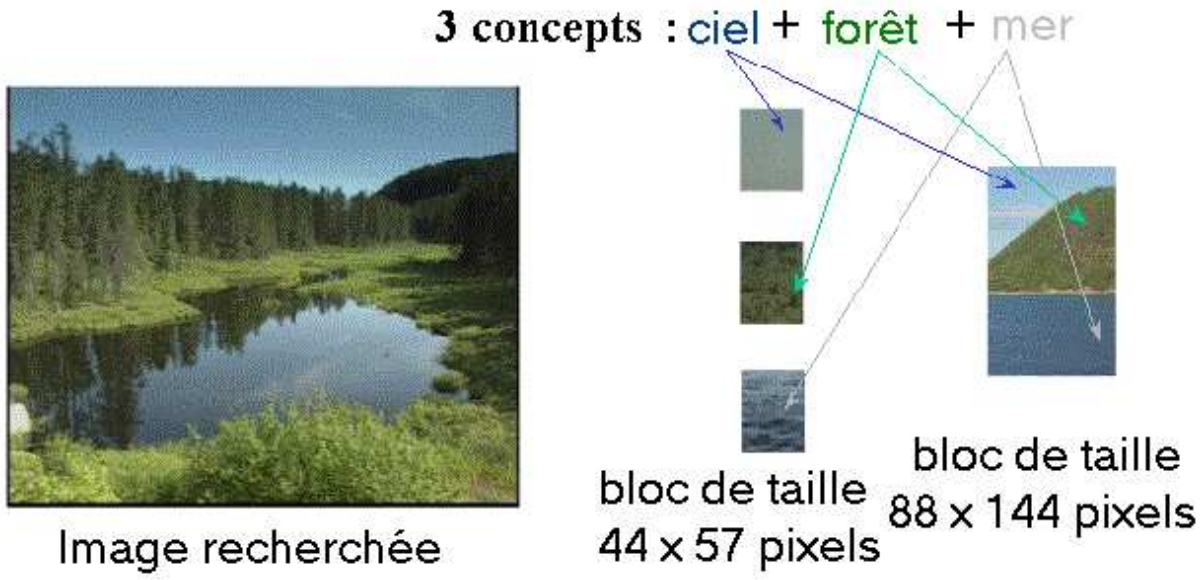


FIG. 2.2 – Exemple de requête pour la recherche d'un plan vidéo contenant le ciel, la forêt et le ciel.

2.3 Représentation d'un plan vidéo

Un plan vidéo est représenté par une image clé qui est décrite par un vecteur de fréquence d'apparition des mots visuels du dictionnaire visuel de requête, le DVR. L'image clé est donc divisée en blocs rectangulaires de taille identique aux mots visuels. Pour chaque bloc de l'image, nous affectons le mot visuel le plus similaire. La figure 2.3 montre un exemple de transcription d'une image par des mots visuels basée sur la caractéristique de couleur ; à gauche, l'image originale à droite l'image encodée. Un plan vidéo est représenté par le nombre d'occurrence des mots visuels du dictionnaire visuel de requête dans l'image clé encodée.

FIG. 2.3 – Image décrite en mots visuels.

2.4 Dictionnaire visuel global

La création du dictionnaire visuel global **DVG** est la première étape du système, il doit être composé de tous les mots visuels existants. Nous utilisons donc tous nos plans vidéo pour le

construire. Tous les mots visuels sont extraits des images clés et utilisés. Nous décrivons chaque mot visuel par un ou plusieurs vecteurs caractéristiques. Le vecteur caractéristique peut-être basé sur la couleur, une caractéristique textuelle, ou autre ; la seule contrainte est de pouvoir représenter ce mot visuel pour le proposer à un utilisateur. Tous ces éléments sont ensuite regroupés en plusieurs groupes grâce à l'algorithme des K-Moyenne afin de supprimer les mots visuels quasi-identiques.

2.5 Dictionnaire visuel de requête

Le dictionnaire visuel de requête **DVR** est construit par la sélection d'un sous-ensemble de mots du DVG. Nous nous basons sur l'idée de sélectionner les mots visuels les plus discriminants. Le pouvoir discriminant d'un mot peut s'interpréter de différentes manières, nous avons choisi de nous baser sur l'idée que plus un mot visuel est présent dans les plans vidéos, moins il est discriminant. Nous définissons donc le pouvoir discriminant d'un mot v est défini par :

$$dis(v) = \log(\frac{1}{1 + tf(v)}) \tag{2.1}$$

où $tf(v)$ est le nombre total d'occurrences de cet élément visuel dans l'ensemble des plans vidéos.
Le DVR est composé d'une ou plusieurs caractéristiques visuelles qui peuvent être prises en compte ensemble ou séparément durant le processus. La figure 2.4 est une illustration du processus en utilisant deux caractéristiques séparément.

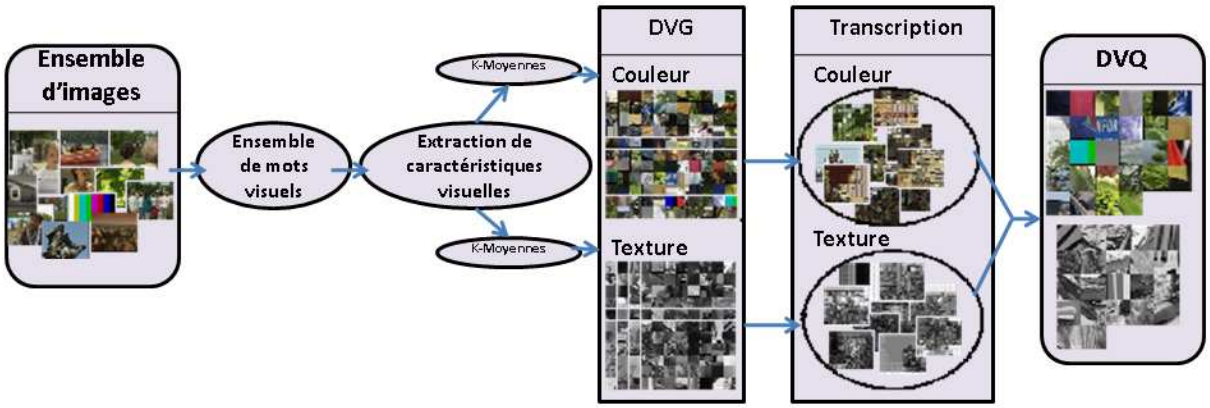


FIG. 2.4 – Illustration du processus de construction d'un dictionnaire visuel avec deux caractéristiques.

3 Méthode de recherche automatique

Afin de répondre correctement à la tâche de TRECVID 2006, nous devions proposer une méthode d'évaluation automatique du système. Pour satisfaire à cette demande, nous proposons une méthode simulée de recherche interactive. C'est-à-dire que nous demandons à un utilisateur automatique de simuler des requêtes, puis nous évaluons la pertinence des plans vidéo renvoyés.

3.1 Recherche interactive

L'utilisation du dictionnaire visuel pour la recherche de plans vidéo pourrait se faire de différentes manières. La méthode utilisée pour l'évaluation est la suivante : dans un premier temps, tous les plans vidéo sont disponibles sous forme d'une liste de micro-icônes. L'utilisateur peut alors choisir des éléments visuels afin d'affiner sa recherche, le système identifie alors, les plans vidéo contenant ces éléments, puis reclasse les documents en fonction de leur pertinence. Certains mots visuels ne sont pas présents dans les plans vidéo restants ; dans ce cas, ils sont retirés du DVR. Les mots visuels sélectionnés sont aussi retirés du DVR, et l'utilisateur peut choisir un nouvel élément. La figure 3.1 montre un exemple d'un tel système. Le DVR est présenté à l'utilisateur sur la gauche, celui-ci est composé de mots visuels de textures, et de mots visuels de couleurs. Les mots visuels de requêtes sélectionnés sont présentés au milieu. La liste des plans vidéo répondant à la requête est affichée sur la droite.

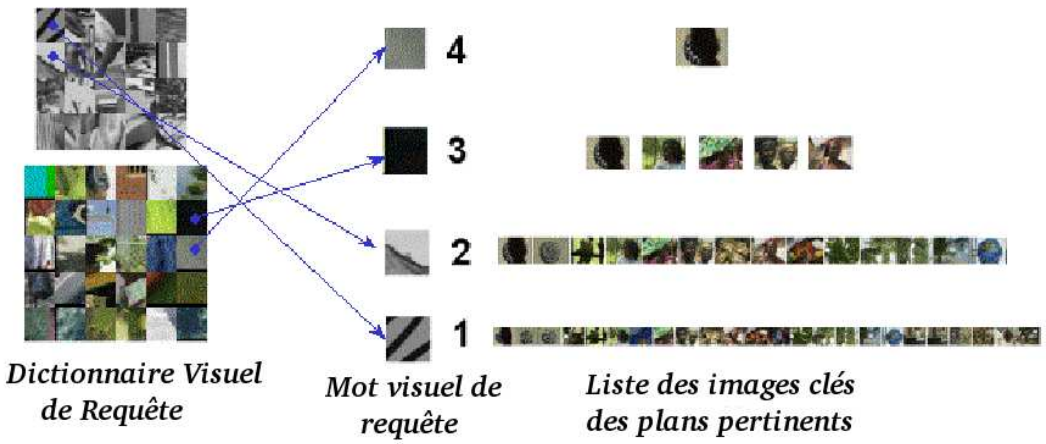


FIG. 3.1 – Interface simulée pour la recherche de plans vidéo.

3.2 Sélection d'un ensemble de test

Afin de construire un ensemble d'entrainement, nous utilisons toutes les vidéos disponibles. Nous effectuons une détection de plans, puis sélectionnons pour chaque plan vidéo une image clé. Afin de limiter la redondance, nous effectuons une classification hiérarchique, comme le montre la figure 3.2. Les images redondantes sont représentées sur le bas, et l'ensemble des images non redondantes sélectionnées sont sur le haut. Pour effectuer cette classification, nous représentons les images clés par un histogramme de couleur, la distance entre deux images est calculée comme la distance Euclidienne, et la distance entre deux groupes est la distance moyenne à travers toutes les paires possibles d'images de chaque groupe. Nous gardons le médoïde de chaque groupe.

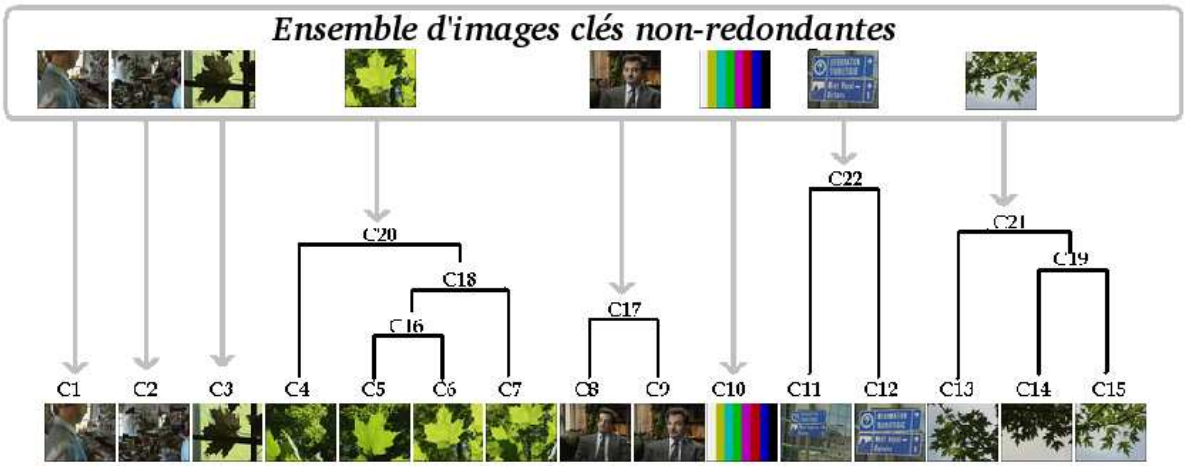


FIG. 3.2 – Illustration de l'algortihme de classification hiérarchique.

3.2.1 Evaluation automatique

Nous proposons d'évaluer la qualité du DVR par une procédure de Recherche Artificielle (RA). L'idée des expériences est la suivante : supposons vouloir identifier un plan vidéo des données d'entraînement, c'est-à-dire que nous savons quel plan nous souhaitons rechercher. Grâce à l'image clé de ce plan, nous pouvons identifier les mots visuels les plus discriminants pour ce plan. Et finalement, nous évaluons la qualité du système grâce à la position du plan voulu dans la liste des plans considérés comme pertinents par le système. Nous effectuons la moyenne des positions pour plusieurs requêtes, ce qui donne une mesure globale de la qualité du système. La figure 3.3 montre une illustration du processus de recherche artificielle. A gauche, l'image clé du plan vidéo recherché, au milieu les mots visuels sélectionnés à partir du DVR, et à droite le rang du plan recherché dans la liste retournée par le système.

Ce processus peut être facilement simulé, mais il nécessite un mécanisme raisonnable pour composer les requêtes. Il est automatisé par le mécanisme suivant : pour chaque mot visuel de DVR, nous calculons la qualité $q_i(v)$ de cet élément pour appartenir à une requête potentielle pour le plan vidéo considéré par :

$$q_i(v) = \begin{cases} \log(\frac{1+N}{1+tf(v)}) & \text{si } v \in i \\ 0 & \text{sinon} \end{cases} \tag{3.1}$$

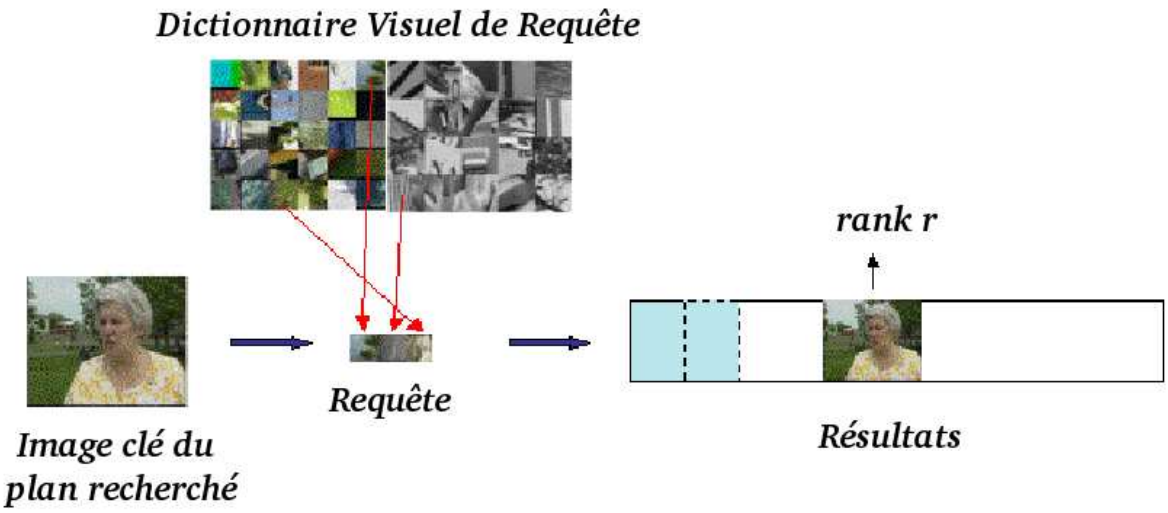


FIG. 3.3 – Recherche Artificielle

où i est le plan vidéo, v le mot visuel, N le nombre total de blocs dans toutes les images et $tf(v)$ le nombre d'occurrences du mot visuel v dans l'image clé du plan vidéo i.

Pour chaque image i, nous pouvons définir $rank(i)$ comme le rang du plan original i dans la liste de résultats. Alors, la qualité du DVR est le rang moyen pour tous les plans vidéos :

$$AverageRank = \frac{1}{N}\sum_{i=1}^{N} rank(i) \tag{3.2}$$

où N est le nombre de plans vidéo.
Cette évaluation peut être complètement automatique.

4 Résultats expérimentaux

4.1 Base de données

Les expériences ont été effectuées sur les données vidéo de TRECVID 2006. Elles représentent un total de plus de 40 heures de vidéo. Les rushes vidéo contiennent les données brutes lors de l'enregistrement de programmes vidéos. Par la méthode décrite auparavant, nous traitons tous ces rushes vidéo, nous exécutons la détection de transition de plan et nous extrayons une série d'images clés non-redondantes. Pour ces rushes vidéo, nous avons trouvé une série de 1759 images non-redondantes.

4.2 Protocole expérimental

Nous avons considéré plusieurs paramètres dans nos expérimentations :
- la taille des mots visuels : les images sont fractionnées par une grille régulière de taille $12 * 10$, $10 * 8$, $8 * 5$, $5 * 4$ ou $4 * 2$;
- le nombre de mots dans le dictionnaire global : K-Moyenne a été lancé avec un nombre de groupe variant de 25 à 1500.
- le nombre de mots dans le dictionnaire de requête : 25, 50, 75, 100 et 200.

A partir de l'image originale, nous construisons la requête en choisissant *Nclics* mots visuels importants qui apparaissent dans l'image.

La figure 4.1 montre un exemple de recherche. L'image recherchée i permet de choisir les mots de la requête dans le dictionnaire visuel. Le système nous renvoie une liste d'images classées suivant leur pertinence par rapport à la requête. Dans notre cas, 7 images sont considérées comme plus pertinente que l'image recherchée et 1 image a la même pertinence que l'image recherchée. Le rang minimal de l'image i est donc 8 et le rang maximal est 9. Le rang de i est alors de 8.5.

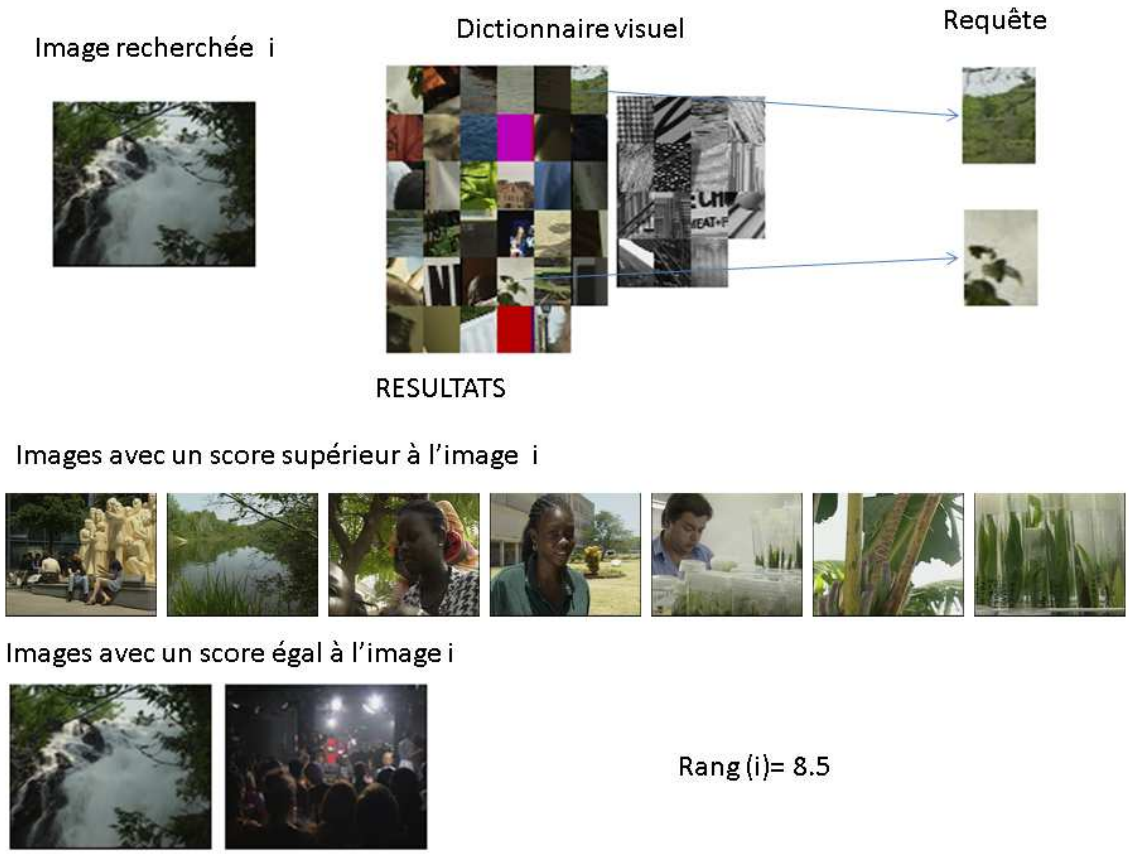


FIG. 4.1 – Exemple de recherche

4.3 Résultats expérimentaux

4.3.1 Dictionniare Visuel Global

Le dictionnaire visuel global, DVG, est obtenu par une méthode de K-Moyenne de tous les mots. Nous avons choisi d'utiliser deux caractéristiques : la couleur HSV, et la texture calculée par 12 filtres de Gabor. Ces deux caractéristiques sont utilisées en parallèle ; le DVG comportant N mots est composé de deux ensembles de mots visuels, un de mots visuels basés sur la couleur et l'autre sur la texture et ayant des tailles identiques $N_c = N_t$, avec $N = N_c + N_t$.
Ensuite, le dictionnaire visuel de requête DVR est composé des mots les plus discriminants puis une recherche artificielle est réalisée avec $Nclics = 2$. Dans cette expérience, les images sont fractionnées selon une grille $8 * 5$ régulière. La figure 4.2 montre le rang moyen de l'image originale par rapport à la taille du DVG, et chaque courbe se réfère à une taille donnée pour le DVR.

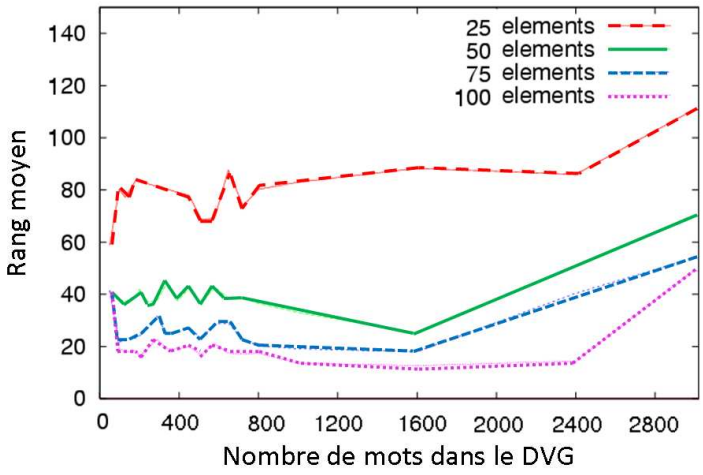


FIG. 4.2 – Rang moyen par rapport à la taille du DVG.

Cette figure montre que la taille du dictionnaire visuel global n'est pas un facteur critique pour la qualité du système. En effet, le comportement de la courbe est assez stable, sauf quand la taille du DVG devient très grande, parce qu'alors les éléments visuels deviennent bien trop particuliers.

4.3.2 Dictionnaire Visuel de Requête

Le nombre de mots du DVR est le nombre d'éléments avec lequel une image peut être encodée ainsi que le nombre d'éléments visuels proposé à un utilisateur pour composer des requêtes. La courbe 4.3 montre le rang moyen de l'image originale pour les diverses tailles de DVR dans le cas où $Nclics = 2$; les images sont fractionnées par une grille $8 * 5$. L'axe des abscisses indique la taille du DVR, et les différentes courbes indiquent le nombre de groupes utilisés dans K-Moyenne.

La courbe d'évaluation pour le DVR montre que le rang moyen diminue rapidement en fonction du nombre d'éléments visuels dans le DVR. Ceci est prévisible puisque ces éléments

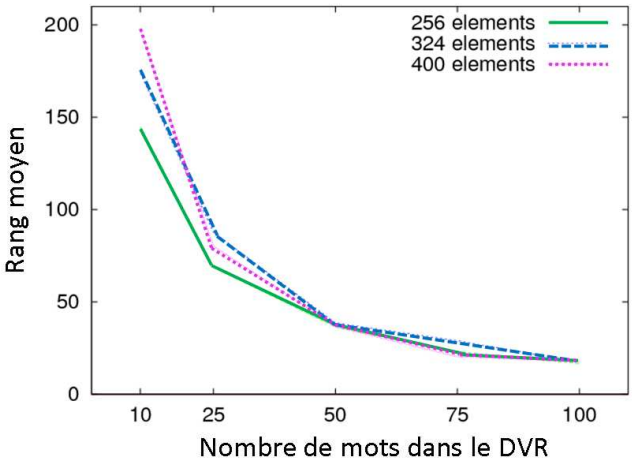


FIG. 4.3 – Rang moyen par rapport à la taille du DVR.

visuels sont utilisés pour coder les images et composer les requêtes. Plus le nombre d'éléments visuels est grand, moins le nombre d'images comportant cet élément est grand, et donc plus le rang moyen des images originales est bas. Mais, les éléments visuels sont proposés à un utilisateur pour composer une requête, donc ce nombre d'éléments visuels doit être raisonnable. Si trop de mots sont proposés à un utilisateur, alors il aura du mal à formuler des requêtes : il lui faudra chercher son mot parmi un grand ensemble, et il ne faudra pas qu'il se trompe entre deux mots trop similaires pour l'oeil humain.

4.3.3 Mots visuels

La taille des mots visuels est évidemment un facteur très important dans le processus. La courbe 4.4 montre le rang moyen de l'image originale pour les diverses tailles de mot et les diverses valeurs de $Nclics$. La taille du DVG est 648 et la taille du DVR est 50.

Nous pouvons voir que pour chaque taille de mots visuels, il y a un optimum en terme de nombre de mots composant la requête $Nclics$. Quand les mots sont grands, cette valeur optimale est petite, parce que ces blocs sont très discriminants. Au contraire, quand les mots sont petits, un plus grand nombre est exigé pour identifier l'image désirée. Il est important de préciser que la requête est considérée comme un ET logique. Donc dès que le nombre de mots caractéristiques d'une image est dépassé, le rang moyen augmente. Le système retourne toutes les images avec un rang égal, ce qui augmente le rang moyen. Le tableau 4.1 montre le nombre d'images ayant un nombre de mots visuels inférieur à la requête.

Une image est en moyenne composée de 12.97 mots visuels. En moyenne, le nombre de mots visuels d'une image est de 8.87 ; ce nombre varie de 3.67 à 23, 35. De plus, un mot visuel est présent dans environ 456, 18 images sur 1759 images (soit 0.26). Le mot visuel le moins fréquent est contenu dans 18 images, alors que le plus fréquent est contenu dans 1589 images.

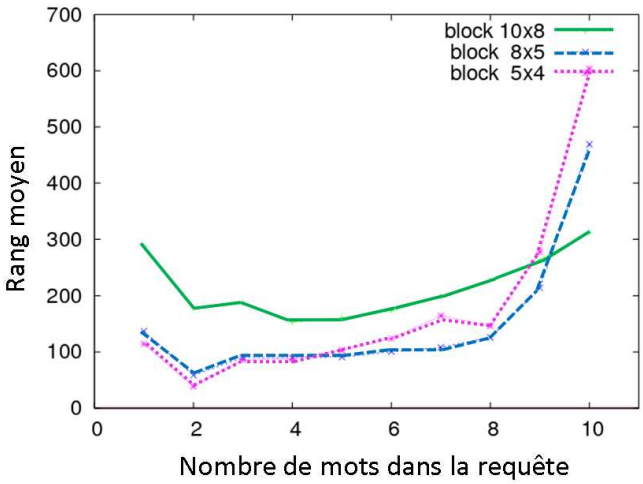


FIG. 4.4 – Evaluation du DVR pour différentes taille de mots

Nombre de mots visuels	Nombre d'images en ayant un nombre inférieur
1	0
2	0
3	114
4	132
5	151
6	171
7	204
8	238
9	287
10	382

TAB. 4.1 – Nombre d'images ayant un nombre de mots visuels inférieur à la requête.

La figure 4.5 propose des exemples de mots avec des tailles différentes. Par exemple, le premier mot à gauche est obtenu en divisant l'image en 10x8 blocs, soit une taille de 35x36 pixels.

FIG. 4.5 – Exemple de mots de tailles différentes.

4.3.4 Caractéristiques visuelles

Jusqu'à maintenant, nous avons considéré un DVR composé des éléments les plus discriminants choisis parmi la couleur et les vecteurs de caractéristiques de texture. Nous comparons

donc cette combinaison avec l'utilisation individuelle de ces caractéristiques. Nous utilisons des mots visuels de taille $8*5$, un DVG de taille 648 et un DVR de taille 50. La figure 4.6 compare la combinaison des caractéristiques avec l'utilisation d'une seule caractéristique. Cette courbe montre nettement une amélioration substantielle pour la combinaison des caractéristiques. Notons que le dictionnaire combiné contient 35% de vecteurs de texture et 65% de vecteurs de couleur.

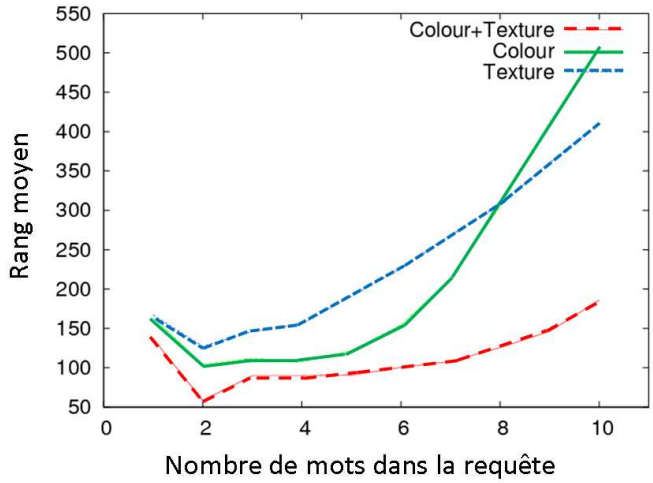


FIG. 4.6 – Evaluation du DVR pour différentes caractéristiques.

4.4 Perceptive

Nous avons continué notre étude de l'utilisation du dictionnaire visuel en nous consacrant à l'amélioration de la qualité des mots visuels d'un point de vue utilisateur. Nous avons donc choisi de ne plus utiliser des blocs d'images rectangles n'ayant pas une interprétation toujours évidente, mais d'utiliser une segmentation en région de l'image afin de séparer les concepts sémantique des images. Pour effectuer la segmentation des images, nous avons utilisé la méthode [Felzenszwalb 2004]. Le figure 4.7 montre une exemple de segmentation en région d'une image extraite d'une vidéo.

Puis nous avons utilisé la même méthode qu'expliquée précédemment en utilisant la caractéristique de couleur :

- K-Moyennes sur la caractéristique couleur des régions des images afin de créer un DVG.
- Une sélection des mots visuels les plus discriminants afin de créer le DVR.

La figure 4.8 montre un exemple de DVR crée par cette méthode.

La courbe 4.9 montre la qualité de notre dictionnaire à travers les résultats préliminaires. La première remarque est que les résultats sont de moins bonne qualité que pour la méthode des blocs. Cependant l'adaptation de la méthode a été faite de manière immédiate, par conséquent l'étude de cette nouvelle piste reste à étendre. Mais les nouvelles orientation de TRECVID nous

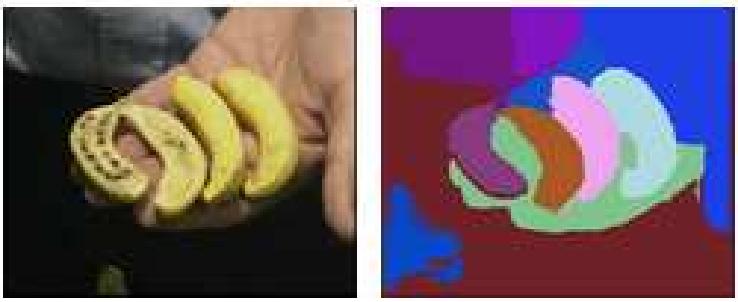


FIG. 4.7 – Exemple de segmentation en région d'une image.

FIG. 4.8 – Exemple de DVR utilisant une segmentation en régions des images.

ne l'ont pas permis.

Le but de cette nouvelle approche est de faciliter la compréhension et le choix d'un mot visuel lors de la composition d'une requête. Dans un deuxième temps, une perceptive très intéressante est de lier les mots visuels à des mots textuels [Tollari 2005].

5 Conclusion

Dans ce chapitre, nous avons expliqué la méthode proposée lors de la campagne d'évaluation TRECVID 2006 pour la tâche pilote de l'exploitation des rushes. Cette méthode est basée sur le paradigme entre la recherche de documents textuels et la méthode de recherche de documents visuels. Le principe général que nous avons développé est la construction d'un dictionnaire de mots visuels, puis la sélection d'un sous-ensemble de mots discriminants.

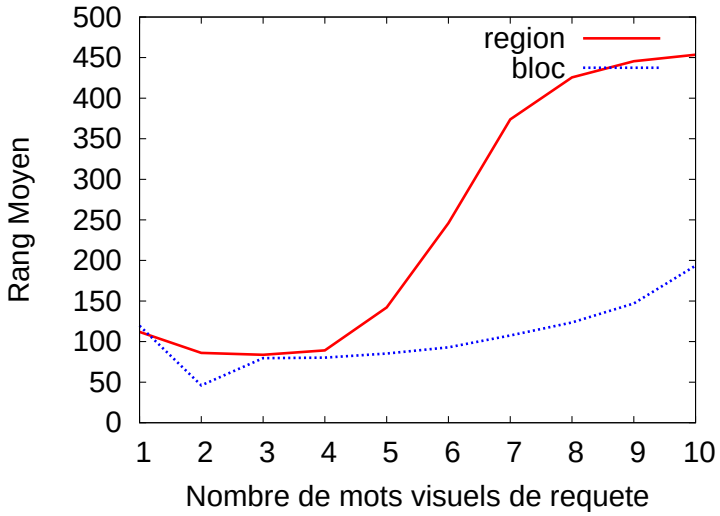


FIG. 4.9 – Evaluation du DVQ en utilisant les régions des images.

Conformément à la proposition pilote de la campagne TRECVID, la tâche consacrée à l'exploitation des rushes vidéo se focalise sur les résumés vidéos. Pour cette raison, ce travail a été mis de côté. Beaucoup d'améliorations sont envisageables, en particulier améliorer les mots visuels : il serait intéressant que les mots aient plus de sens. Et finalement, faire le lien entre les mots visuels et les mots textuels permettrait un système de recherche de plans vidéo très intéressants.

Quatrième partie

Construction de résumés vidéo de rushes

Construction de résumés vidéo de rushes

Un résumé vidéo doit permettre de donner un aperçu rapide du contenu de la vidéo initiale. Afin de répondre à ce besoin, nous avons développé deux méthodes. La première méthode est basée sur une mesure du contenu sémantique d'une vidéo ; après avoir effectué le prétraitement sur la vidéo, nous segmentons la vidéo en segments d'une seconde, puis utilisons une classification hiérarchique pour grouper ces segments. Finalement, les séquences recouvrant le plus de contenu sémantique sont sélectionnées pour créer le résumé vidéo.
Dans la deuxième méthode, nous utilisons un algorithme de structuration de la vidéo grâce à une méthode d'alignement de séquences vidéo. Après avoir groupé les segments d'une seconde, nous alignons les suites de segments similaires. Ces alignements de séquences nous permettent de retrouver la structure de la vidéo et donc, d'enlever la redondance.
Ces deux méthodes ont été utilisées pour les soumissions à TRECVID 2007 et TRECVID 2008, nous parlerons donc de la qualité de ces méthodes à travers la campagne d'évaluation TRECVID.

1 Introduction

Avec les avancés rapides en matière de technologie des documents vidéo numériques et malgré le fait que des technologies puissantes existent dorénavant pour créer, lire, stocker et transmettre ces documents, l'analyse du contenu vidéo est toujours un défi de recherche ouvert et actif. La création automatique de résumés vidéo est un outil puissant qui permet de synthétiser le contenu d'une vidéo tout en conservant l'essence importante. Dans cet esprit, le contenu de la séquence vidéo doit être analysé, et sa structure doit être identifiée pour que les segments vidéos les plus pertinents puissent être choisis.

Dans ce travail, développé pour répondre à la tâche de la campagne TRECVID de résumé vidéo, nous nous limiterons aux vidéos de rushes. Le premier aspect que nous avons abordé est celui du choix des séquences de la vidéo à sélectionner afin de représenter au mieux l'ensemble d'une vidéo. L'idée générale s'appuie sur le fait que certaines séquences sont redondantes par rapport à d'autres et que certains critères permettent un choix judicieux d'une séquence parmi un ensemble redondant.

Le deuxième aspect est celui de la détection des séquences similaires permettant de retrouver la structure du rush. Ce qui va nous permettre de sélectionner une prise par scène et donc d'éliminer la redondance.

Dans ce chapitre, nous expliquerons les outils que nous avons développés : la mesure de similarité du contenu sémantique d'une séquence, puis celui de la détection des séquences répétitives. Pour chacune d'entre elle, nous proposerons une évaluation de la méthode permettant le calibrage de celle-ci. Finalement, nous parlerons de notre participation à TRECVID à travers les soumissions basées sur ces méthodes.

2 Classification vidéo

Une étape préliminaire commune à nos deux systèmes consiste à grouper les segments vidéos par similarité visuelle. Pour se faire :

1 - Le prétraitement est appliqué.

2 - La vidéo est segmentée.

3 - Une méthode de classification non-supervisée permet de grouper les segments.

2.1 Unité temporelle

Notre première étape consiste à définir l'unité de temps sur laquelle nous allons travailler. Cette unité de temps doit au mieux répondre au compromis d'être visuellement perceptible, c'est-à-dire qu'elle doit être assez longue pour qu'un cerveau humain puisse l'interpréter, mais en même temps, cette unité ne doit pas être trop grande afin de ne pas être redondante, c'est-à-dire que le cerveau doit juste avoir le temps de l'interpréter. Une étude a mis en avant le fait qu'une seconde suffisait pour voir un concept dans un résumé [Hauptmann 2007]. Nous avons donc choisis d'utiliser cette unité temporelle d'une seconde.

Dans une premier temps, le prétraitement, présenté dans la première partie, est appliqué à la vidéo. Les séquences "poubelles" et "outils" sont éliminées et la vidéo est dynamiquement accélérée. De plus ce prétraitement nous donne une segmentation en plans de la vidéo. La vidéo est alors segmentée en segments d'une seconde. Les segments ne peuvent pas être à cheval sur deux plans, par conséquent si le dernier segment d'un plan dure moins d'une seconde, il est tout simplement inutilisé.
Une vidéo V est alors définie par une liste de segments s_i. Soit n le nombre de segments, alors :

$$V = (s_1, s_2, ..., s_n)$$

2.2 Classification

Afin de grouper les segments vidéos d'une seconde, il est important de définir un vecteur caractéristique relatif à un segment. Plusieurs approches sont possibles, soit utiliser le principe des images clés ; c'est-à-dire extraire un nombre prédéfini d'images provenant du segment, soit utiliser l'information de toutes les images, par exemple en calculant une moyenne sur les 25 images. Le choix de la ou des caractéristique(s) dépend du problème. Pour notre problème, nous souhaitons grouper les segments vidéos par similarité visuelle, nous avons donc choisis d'utiliser la caractéristique couleur représentée sous forme d'un histogramme. Deux approches ont été proposées, soit calculer l'histogramme sur l'image centrale, soit de faire la moyenne sur les 25 images. La vidéo V est alors représentée par :

$$V = (\vec{s_1}, \vec{s_2}, ..., \vec{s_n})$$

Maintenant que les segments sont représentés sous forme de vecteur, il reste à définir la méthode de classification non-supervisée. Nous avons choisi de faire une classification hiérarchique ascendante. Au début, chaque groupe est composé d'un segment, puis à chaque étape, deux groupes sont fusionnés jusqu'à ce que tous les segments soient dans le même groupe. La distance moyenne est utilisée pour calculer les distances entre groupes. Nous avons choisis cette méthode de classification, puisque à la fin de l'algorithme, nous obtenons un arbre. Cet arbre permet de pouvoir utiliser plusieurs niveaux de classification et jouer sur ces différents niveaux : c'est-à-dire que nous n'avons pas besoin de préfixer le nombre de groupes, ou de fixer un seuil sur la similarité pour arrêter la classification. Soit c_i^l le groupe du segment i au niveau de classification l, alors, le vidéo est définie par :

$$V = (c_1^l, c_2^l, ..., c_n^l)$$

3 Mesure du contenu visuel

L'idée de cette première méthode est de sélectionner des séquences de segments recouvrant au maximum le contenu visuel de la vidéo tout en utilisant la notion d'intérêt visuel d'un segment. Nous définissons donc une mesure d'intérêt visuel relatif à un groupe de segments. Finalement, nous choisissons de manière itérative les séquences de segments les plus informatives en terme de contenu visuel, mais aussi recouvrant au mieux les informations de la vidéo.

3.1 Intérêt visuel

Nous affectons un coefficient d'intérêt visuel à un groupe de segments ; cet intérêt est basé sur des notions intuitives. Une séquence vidéo est généralement importante lorsque des personnes

sont présentes. De même plus il y a de l'action dans la vidéo, plus le segment doit contenir un évènement informatif. La notion d'encombrement dans une vidéo est très importante aussi, c'est-à-dire qu'une séquence vide en terme de contenu ne doit pas être très informative. L'intérêt visuel d'une séquence est alors basé sur :

- $face(s)$ représente la probabilité que le segment s contienne un visage. Pour chaque image d'un segment, nous détectons les visages présents par la méthode [Rowley 1996], ce qui nous permet de calculer la probabilité que le segment contienne un visage.

- $act(s)$ représente le degré moyen d'activité dans le segment s. Pour chaque image, l'activité visuelle est calculée, puis normalisée à l'échelle de la vidéo. L'activité moyenne du segment est calculée.

- $ent(s)$ représente le degré d'encombrement de la vidéo. Pour chaque image, l'entropie de la distribution des couleurs est calculée, puis normalisée à l'échelle de la vidéo. L'encombrement moyen du segment est calculée.

L'importance de chacun de ces éléments n'est pas forcément équivalente, des poids sont donc affectés, de manière manuelle Wf, $Wact$ et $Went$, respectivement à l'intérêt de la présence de visage, au degré d'activité, et au degré d'encombrement. Finalement, l'attribut visuel d'un groupe c est défini par :

$$Vatt(c) = \frac{\sum_{s \in c}(face(s) * Wf + act(s) * Wact + ent(s) * Went)}{\sum_{s \in c} 1} \tag{3.1}$$

Cependant, il est important de noter qu'une séquence d'images très calme ne comportant pas de visage et une très faible diversité de couleur, par exemple une vue sur la mer, ne doit pas être négligée pour autant. L'intérêt visuel d'un groupe est donc un compromis entre l'attribut visuel et l'existence du groupe :

$$Vint(c) = \alpha * \frac{Vatt(c)}{(Wf + Wact + Went)} + (1 - \alpha) \tag{3.2}$$

avec $0 \leq \alpha \leq 1$, et où α est le poids de l'importance de l'attribut visuel, alors que $(1 - \alpha)$ est celui de l'existence du groupe.

3.2 Sélection des segments

Cette étape consiste à sélectionner les segments utilisés pour la création du résumé final et à fixer la durée maximale du résumé $\mathcal{D}_{max}$. Cette durée doit se rapprocher au mieux de la durée de l'information non redondante dans la vidéo.

Nous définissons, la notion de séquence par un nombre prédéfini de segments. Pour chaque séquence p et niveau de classification l, nous calculons l'importance de cette séquence $I^l(p)$ par la somme des intérêts visuels des groupes qu'elle recouvre $\mathcal{R}_p^l$.

$$I^l(p) = \sum_{c \in \mathcal{R}_p^l} Vint(c) \tag{3.3}$$

Pour un niveau de classification l donné, nous sélectionnons itérativement les séquences les plus importantes $\mathcal{S}^l$. A chaque itération, les groupes recouverts par la séquence n'ont plus d'intérêt visuel à être sélectionnés puisque le contenu du groupe a déjà un représentant, par conséquent leur intérêt devient nul. Ce processus s'arrête lorsque tous les groupes sont couverts par une séquence sélectionnée. Ensuite, nous calculons la durée associée à cet ensemble $\mathcal{D}^l$ par la somme des durées des segments des séquences de $\mathcal{S}^l$.

La classification hiérarchique a produit un arbre, ce qui implique de choisir un niveau l dans la classification. Nous allons donc choisir le niveau qui représente au moins la redondance visuelle. C'est-à-dire que tous les segments vidéos similaires doivent être dans le même groupe et tous les segments non similaires doivent être dans des groupes différents. Afin de déterminer ce niveau, nous commençons par déterminer pour chaque niveau $\mathcal{D}^l$ et finalement choisir l tel que :

$$l = \text{argmax}\{\mathcal{D}^l | \mathcal{D}^l < \mathcal{D}_{max}\} \tag{3.4}$$

3.3 Evaluation expérimentale

Nous proposons d'évaluer l'impact des différents paramètres du système proposé : la taille des séquences, la taille du résumé final, et les différents poids permettant de déterminer l'intérêt visuel d'une séquence.

3.3.1 Protocole

Nous avons effectué les tests sur les vidéos proposées en 2007 pour la tâche des résumés vidéo de TRECVID. Le problème de l'évaluation d'un tel système reste ouvert. Nous avons adapté l'évaluation manuelle proposée par TRECVID à notre problème. TRECVID propose de prendre le critère du pourcentage d'éléments d'histoire visibles dans le résumé. Et dans [Hauptmann 2007], une proposition de calcul automatique de ce critère a été proposée. Cette méthode a une forte corrélation avec la méthode manuelle. Le principe est très simple, si un élément d'histoire apparait durant au moins une seconde dans le résumé, alors celle-ci est visible. Une telle méthode nécessite une annotation manuelle des éléments d'histoire ; 7 vidéos ont été annotées pour faire les expériences.
Nous avons utilisé la valeur du pourcentage d'éléments d'histoire retrouvés dans les résumés comme une valeur de rappel. La précision est définie comme le nombre d'éléments d'histoire retrouvés sur le nombre de séquences sélectionnées.

3.3.2 Résultats

La taille d'une séquence vidéo est définie par un nombre fixé de segments d'une seconde. Intuitivement, cette valeur correspond à la longueur vidéo d'un élément d'histoire. La figure 3.1 montre l'impact de ce critère sur une vidéo. Les différentes courbes font référence aux différentes tailles des séquences vidéo donnant les meilleurs résultats (3, 4 ou 5 secondes), la courbe du haut correspond à la valeur de précision en fonction de la taille du résumé (en pourcentage

par rapport à la vidéo initiale), alors que celle du bas correspond à la valeur de rappel en fonction de la taille du résumé. Cette courbe nous montre qu'évidement, au plus la taille du résumé est grande au plus le nombre d'éléments d'histoire visibles dans le résumé est grand. Mais inversement, au plus la taille du résumé est allongée, au plus des séquences inutiles sont sélectionnées.

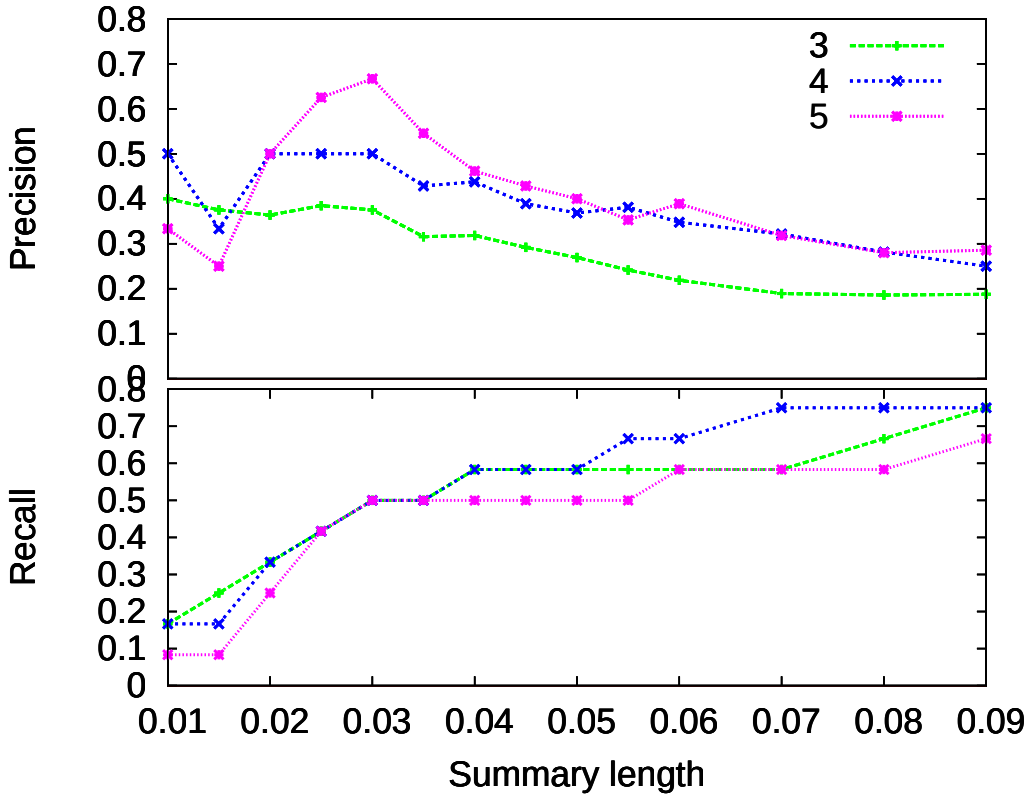


FIG. 3.1 – Impact de la longueur du résumé

Notre système a besoin de connaitre la taille du résumé final : l'impact de cette valeur est présenté sur la figure 3.2. Les différentes courbes nous montrent les résultats pour différentes tailles de résumés. Inversement, nous voyons clairement que la précision augmente avec la taille des séquences. En effet, une séquence trop courte ne contiendra pas l'élément d'histoire ou du moins pas de manière suffisante pour le visualiser. Mais si on part dans l'excès, le nombre de séquences sélectionnées sera très faible et par conséquent la moindre séquence sélectionnée à tort fera chuter la valeur de la précision.

Ces deux courbes nous permettent de montrer que les critères de la longueur d'une séquence en terme de nombre de segments d'une seconde et la longueur du résumé final sont des valeurs très importantes à fixer et ayant un gros impact sur la qualité du résumé final. Au regard des résultats, nous avons décidé de fixer pour une qualité optimal la taille des résumés finaux à 3%,

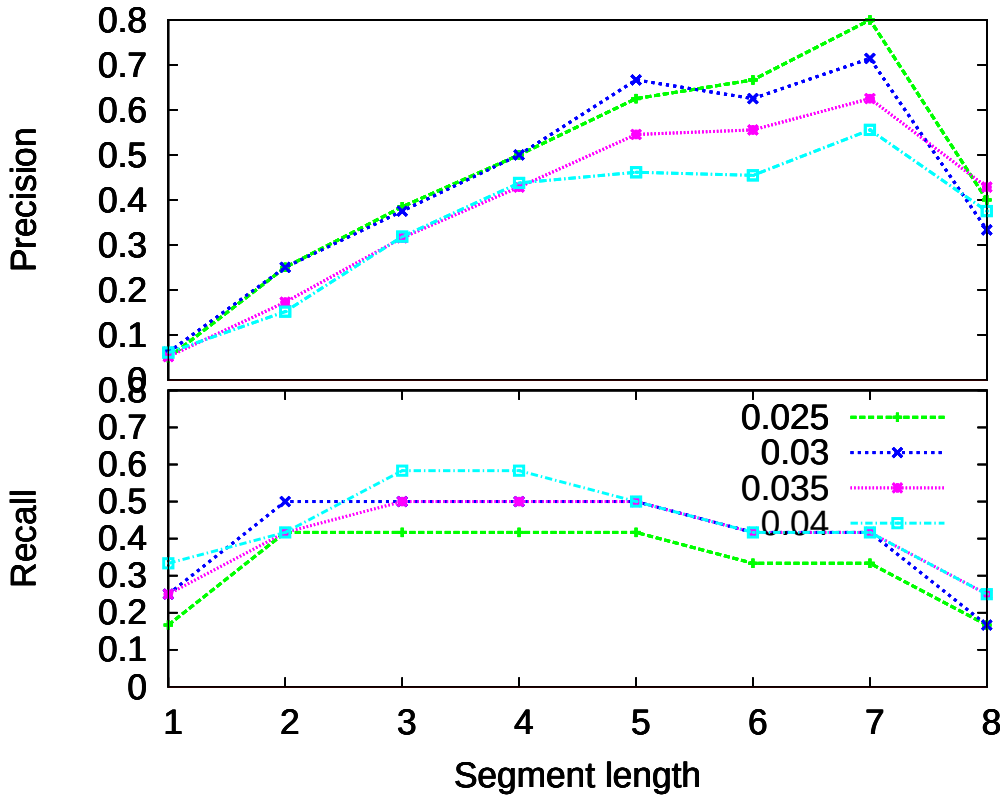


FIG. 3.2 – Impact de la longueur d'une séquence

et la taille des séquences à 4 secondes.

4 Alignement des séquences vidéo

Pour cette deuxième méthode, l'idée est de retrouver la structure de la vidéo, c'est-à-dire arriver à délimiter les transitions de scène, et à l'intérieur de chaque scène réussir à retrouver une prise complète. Ce qui permet d'obtenir un bout-à-bout sans redondance. Pour arriver à retrouver la structure d'une vidéo, nous avons développé un algorithme d'alignement de séquences vidéo dérivé d'un algorithme du domaine de la bioinformatique permettant l'alignement local de séquences de protéines. Ensuite, grâce à ces alignements, nous pouvons déterminer les différentes scènes ; et finalement choisir une prise par scène.

4.1 Alignement

En 1966, Levenshtein introduit la notion de distance d'édition (edit distance) par la question : "Quel est le nombre minimal d'opérations pour transformer une séquence en un autre ?". La

distance de Levenshtein entre deux séquences est donnée par le nombre minimum d'opérations nécessaires pour transformer une séquence en une autre, où une opération peut-être une insertion, une suppression, ou une substitution. Elle permet de mesurer la quantité de différence entre deux séquences en associant à chacune de ces opérations un coût. Cette distance est d'autant plus grande que le nombre de différences entre les deux chaînes est grand. La distance de Levenshtein peut être considérée comme une généralisation de la distance de Hamming.

En 1970, l'algorithme de Needleman-Wunsch [Needleman 1970] est proposé ; c'est un exemple de programmation dynamique [Bellman 1957], tout comme l'algorithme de Levenshtein auquel il est apparenté. Il garantit de trouver l'alignement global de score maximal entre deux séquences. Ce fut la première application de la programmation dynamique pour la comparaison de séquences biologiques. Les scores pour les caractères alignés sont spécifiés par une matrice de similarité.

Afin de trouver le meilleur alignement global, il faut calculer la matrice de score C. Soit deux séquences $A = (a_1, a_2, ..., a_n)$ et $B = (b_1, b_2, ..., b_m)$, la matrice de score est alors initialisée en plaçant chaque caractère a_i de la séquence A au dessus de chaque colonne de la matrice en commençant à la deuxième case, et chaque caractère b_i de la séquence B à gauche de chaque ligne en commençant à la deuxième. La case $C[i][j]$ correspondra alors au score de l'alignement entre $(a_1, a_2, ..., a_i)$ et $(b_1, b_2, ..., b_j)$. La dernière étape de l'initialisation consiste à :

$$C[i][0] = 0, \forall i \in 0, ..., n \text{ and } C[0][i] = 0, \forall i \in 0, ..., m$$

Pour remplir la matrice de score, il faut définir :
- $d(a_i, b_j)$: le score de l'alignement du caractère a_i avec la caractère b_j.
- $d(a_i, e\ )$: le score de l'alignement du caractère a_i avec un trou.
- $d(e\ , b_j)$: le score de l'alignement du caractère b_j avec un trou.

Et finalement, la matrice de score peut-être calculée grâce à la formule récursive :

$$C[i][j] = \max(C[i-1][j-1] + d(a_i, b_j), C[i-1][j] + d(a_i, e), C[i][j-1] + d(e, b_j))$$

La figure 4.1[11] montre un exemple de matrice de score avec A =GAATTCAGTTA et B=GGATCGA, et $d(a_i, b_j) = 5$ si $a_i = b_j$; sinon $d(a_i, b_j) = -3$. $d(a_i, e) = d(e, b_j) = -4$.

Le score maximal de l'alignement entre A et B est donné par la case $C[n][m]$. Il ne reste plus qu'à reconstruire le chemin donnant ce score maximal noté $path(n, m)$. Ceci est réalisé en regardant la direction dans laquelle chaque case a été atteinte. Dans notre exemple, deux alignements globaux sont de score maximal : GAATTCAGTTA avec GGA-TC-G–A ou GAATTCAGTTA avec GGAT-C-G–A. La figure 4.2 montre le chemin parcouru pour ces deux alignements.

En 1981, Smith-Waterman [Smith 1981] a proposé une variation de cet algorithme permettant un alignement local des séquences : au lieu de regarder la séquence dans sa globalité, l'algorithme de Smith-Waterman compare des segments de toutes longueurs possibles et optimise la mesure de similarité. Ceci est fait en créant une matrice de score dans laquelle les cases indiquent le coût associé au changement d'une sous-séquence en une autre sous-séquence. La différence principale par rapport à l'algorithme de Needleman-Wunsch est le fait que les

[11] source : http ://www.acm.org/crossroads/wikifiles/13-1-CE/13-1-11-CE.html

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	5	1	-3	-3	-3	-3	-3	5	1	-3	-3
G	0	5	2	-2	-6	-6	-6	-6	2	2	-2	-6
A	0	1	10	7	3	-1	-5	-1	-2	-1	-1	3
T	0	-3	6	7	12	8	4	0	-4	3	4	0
C	0	-3	2	3	8	9	13	9	5	1	0	1
G	0	5	1	0	4	5	9	10	14	10	6	2
A	0	1	10	6	2	1	5	14	10	11	7	11

FIG. 4.1 – Exemple de matrice de score pour l'alignement global de séquences ADN

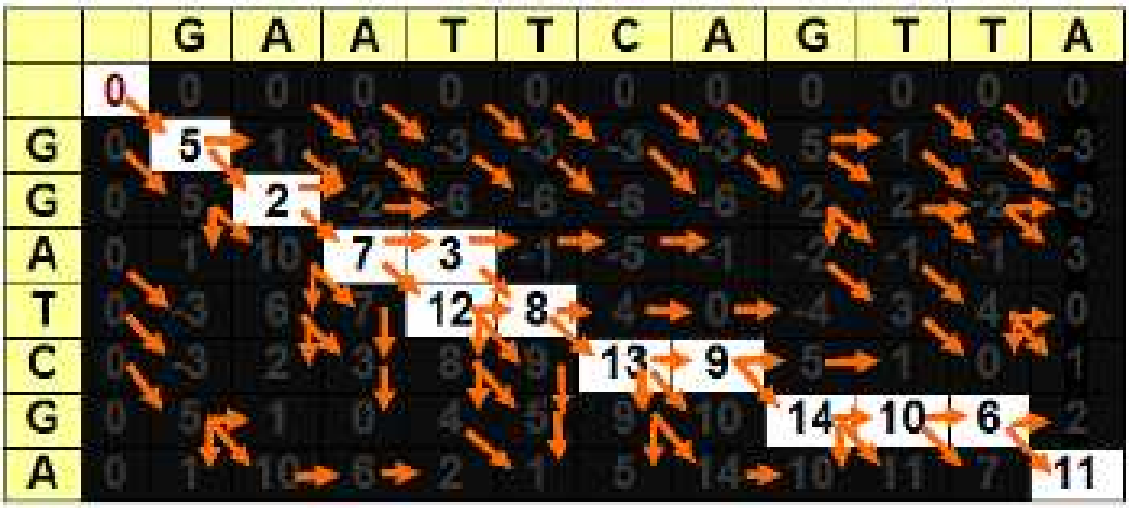


FIG. 4.2 – Exemple de reconstruction du chemin de l'alignement global de séquences ADN

scores d'alignement sont seuillés à zéro ce qui rend les alignements locaux visibles. Pour notre exemple, les alignements locaux maximaux sont : GAATTCAG avec GGA-TC-G ou GCAT-C-G et GAATTC-A avec GGA-TCGA ou GGAT-CGA. La figure 4.3 montre la matrice de score.

Plusieurs groupes de travail ont adaptés de tels algorithmes à l'alignement de séquences de rushes vidéo. Par exemple, [Chasanis 2008] décomposent les plans en images clés, puis effectue des alignements globaux entre tous les couples de plans, la qualité de l'alignement permet de construire une matrice de similarité des plans. Sur une idée similaire, [Liu 2008] effectuent un alignement local des images clés des plans successifs, une qualité d'alignement est attribuée à chaque résultat. Ces coefficients de qualités sont utilisés comme distance pour effectuer un classification des plans. Dans [Bailer 2009], les auteurs détectent les différentes prises d'une même scène grâce à l'algorithme LCSS.

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	5	1	0	0	0	0	0	5	1	0	0
G	0	5	2	0	0	0	0	0	5	2	0	0
A	0	1	10	7	3	0	0	5	1	2	0	5
T	0	0	6	7	12	8	4	1	2	6	8	4
C	0	0	2	3	8	9	13	9	5	2	4	5
G	0	5	1	0	4	5	9	10	14	10	6	2
A	0	1	10	6	2	1	5	14	10	11	7	11

FIG. 4.3 – Exemple de matrice de score pour l'alignement local de séquences ADN

4.2 Adaptation aux séquences vidéo

L'idée est d'appliquer l'algorithme de l'alignement local des séquences d'ADN à l'alignement de séquences vidéo. Le principe est de remplacer les protéines par le numéro du groupe affecté à un segment vidéo d'une seconde durant la procédure de classification et de chercher les alignements locaux à l'intérieur d'une même vidéo. Cependant pour adapter correctement ce mécanisme, nous posons les assertions suivantes :

- Une vidéo est définie comme une liste de segment d'une seconde $V = (s_1, s_2, ..., s_n)$.
- Deux segments s_i et s_j sont dits alignés s'ils appartiennent à deux séquences alignées entre elles.
- Deux séquences alignées doivent avoir une taille minimale.
- Deux séquences alignées ne peuvent pas contenir le même segment.
- Deux segments ne peuvent être alignés qu'une seule fois entre eux.

Une vidéo est représentée comme une liste $V = (c_1^l, c_2^l, ..., c_n^l)$ où c_i^l représente le groupe du ième segment de V au niveau de classification l. En faisant varier l, nous obtenons une définition plus ou moins grossière de la similarité visuelle. Nous proposons de faire varier l tout au long de la détection des alignements ; c'est-à-dire qu'une fois tous les alignements fins trouvés, nous augmentons la valeur de l pour continuer les alignements, mais à un niveau légèrement plus imprécis.

Nous construisons donc une matrice de score dépendant du niveau de classification l comme suit :

$$M^l[i][0] = 0, \;\; M^l[0][i] = 0 \;\text{ et }\; M^l[i][i] = 0 \;\; \forall i \in 0, ..., n \tag{4.1}$$

$$M^l[i][j] = max \left(\begin{array}{l} 0 \\ \left\{ \begin{array}{ll} M^l[i-1][j-1] + match\_cost(i,j) & \text{si } c_i = c_j \\ M^l[i-1][j-1] + no\_match\_cost(i,j) & \text{si } c_i \neq c_j \end{array} \right. \\ M^l[i][j-1] + gap\_cost(i,j) \\ M^l[i-1][j] + gap\_cost(i,j) \end{array} \right)$$

match_cost est le coût pour aligner deux segments qui appartiennent au même groupe. *no_match_cost* est le coût pour aligner deux segments qui appartiennent à des groupes différents. *gap_cost* est le coût pour ajouter un trou dans l'alignement.

Nous proposons aussi d'utiliser une version normalisée de cette matrice afin de privilégier la qualité de l'alignement plutôt que la longueur de l'alignement. La normalisation est effectuée par la longueur du chemin. C'est à dire que soit $length(i,j)$ la longueur de $path(i,j)$:

$$\bar{M}[i][j] = \frac{M[i][j]}{length(i,j)} \tag{4.2}$$

Finalement, l'algorithme que nous proposons est défini par la figure 4.4.

Entrée : Une vidéo V définie par une liste de n segments d'une seconde : $V = s_1 s_2 ... s_n$
- Classification hiérarchique : $V = (c_1^l, c_2^l, ..., c_n^l)$ où c_i^l représente le groupe du segment s_i au niveau l de la classification.
- $l = 0$.
- Calcul de la $(n+1) * (n+1)$ matrice de score normalisée $\bar{M}^l$.
- Itérativement : trouver le meilleur alignement, c'est-à-dire la valeur maximale dans $\bar{M}^l$, M_{max}.
  - Si $M_{max} > threshold$, cet alignement est enregistré et la matrice de score est mise à jour.
  - Sinon $l = l + 1$ et la matrice de score est mise à jour.

Sortie : Une liste ordonnée de séquences alignées.

FIG. 4.4 – VSA Algorithme d'alignement de séquence vidéo

4.3 Structuration de la vidéo

Chaque scène est généralement enregistrée plusieurs fois, ou enregistrée sous une version différente. Nous proposons de retrouver la structure de la vidéo, c'est-à-dire les limites entre les scènes, puis de sélectionner une prise par scène grâce à l'information apportée par les alignements. De plus, cette détection des transitions de scènes va nous permettre de supprimer les faux alignements. Il est à noter que maintenant, nous travaillons avec l'unité de l'image.

4.3.1 Matrice d'alignement

Une séquence vidéo est définie comme une liste d'images ayant un ordre temporel $V = f_1 ... f_m$. Nous construisons la $m * m$ matrice d'alignement A où $A[f_i][f_j]$ représente l'indice de l'alignement entre l'image f_i et l'image f_j ou le nombre d'alignements plus un si les images f_i et f_j ne sont pas alignées. La figure 4.5 montre un exemple d'une matrice d'alignement, l'intensité du niveau de gris dépend de l'indice de l'alignement.

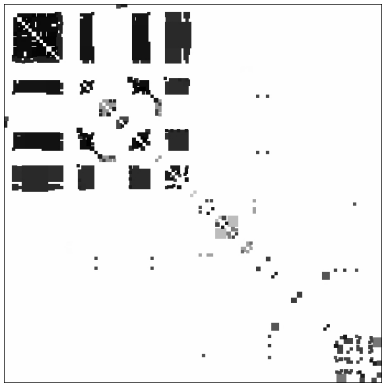


FIG. 4.5 – Exemple d'une matrice d'alignement

4.3.2 Détection des transitions de scènes

La matrice d'alignement nous donne des informations sur la structure de la vidéo. Pour déterminer les transitions de scène, nous partons de l'idée qu'à l'intérieur des scènes, les prises sont visuellement similaires et par conséquent représentées par des rectangles noirs dans la matrice d'alignement. Plus précisement, nous calculons pour chaque image fi :

$$rect(f_i) = \frac{\sum\limits_{\forall f1 \in [first, f_i]} \sum\limits_{\forall f2 \in [last, F]} A[f1][f2]}{\sum\limits_{\forall f1 \in [first, f_i]} \sum\limits_{\forall f2 \in [f_i, last]} 1} \tag{4.3}$$

Les transitions de scènes sont détectées récursivement : l'algorithme commence à $first = 0$ et $last = m$ où m est le nombre total d'images dans la vidéo, tous les rectangles sont calculés, la plus grande valeur $rect[f]$ est gardée.

- $rect[f] \leq threshold$: une transition de scène est détectée à l'image f et l'algorithme recommence avec d'une part $first = 0$ et $last = f$, et d'autre part $first = f$ et $last = m$.
- $rect[f] < threshold$: il n'y a pas d'appel récursif.

La figure 4.6 donne un aperçu du mécanisme.

4.3.3 Sélection des prises

La détection des scènes effectuée, le choix de la sélection de la prise se base sur les commentaires suivants :

- Les différentes prises d'une même scène ont un contenu très similaire, par conséquent, les différentes prises doivent apparaître de manière alignée dans la matrice d'alignement. Une prise doit être une séquence d'images qui ne contient pas d'images alignées.
- Certaines prises sont plus courtes que d'autres à cause par exemple d'erreurs des acteurs, ou encore du fait que certains bouts de scènes sont pris de différentes manières. L'idée est donc de prendre la prise la plus complète, c'est-à-dire la plus longue.

En se basant sur ces remarques, nous ne cherchons pas une décomposition précise des scènes en prises, mais nous cherchons juste à extraire la prise la plus complète, c'est-à-dire, la suite d'images la plus longue sans contenir d'images alignées entre elles.

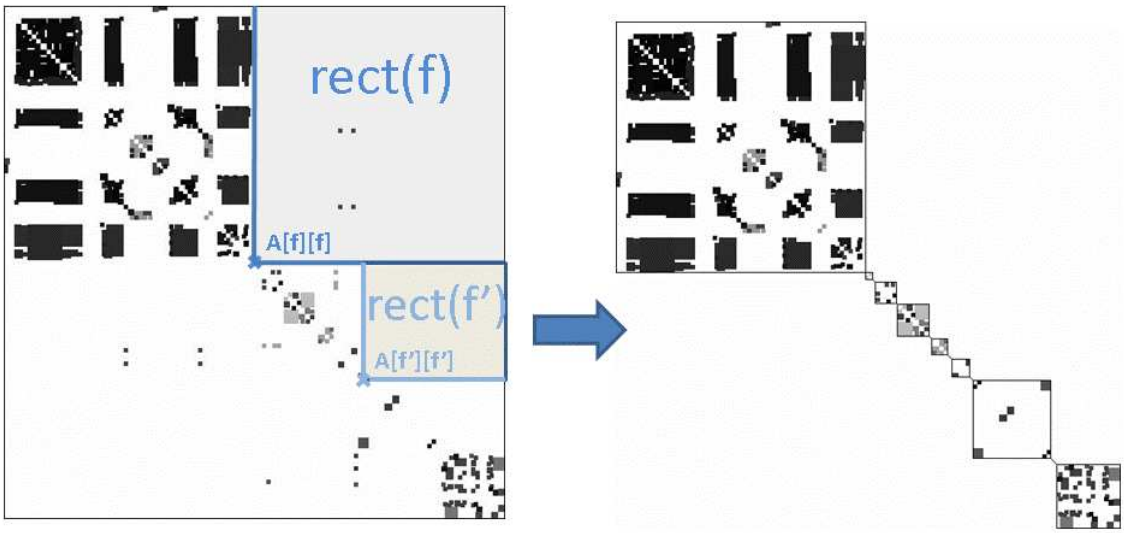


FIG. 4.6 – Illustration du système de détection des transitions de scène.

4.4 Evaluation expérimentale

Notre système est organisé en deux grandes étapes, la première étant l'alignement des séquences et la deuxième la détection des transitions de scène. Nous proposons une évaluation expérimentale de ces deux étapes.

4.4.1 Protocole

Pour effectuer notre évaluation, nous utilisons les vidéos provenant de la tâche de résumé vidéo proposée par TRECVID 2008. Nous disposons de 6 vidéos annotées pour le développement et 8 vidéos annotées pour les tests.

Pour créer la vérité terrain, nous avons manuellement décomposé les vidéos en scènes et prises. Les prises sont aussi décomposées en sous-prises. Nous utilisons cette décomposition et nous considérons que toutes les sous-prises d'une même scène contiennent des images qui doivent être alignées entre elles. Ce qui nous permet d'obtenir une matrice d'alignement comme l'illustre la figure 4.7.

4.4.2 Evaluation de l'alignement

L'évaluation de l'alignement de séquences reste un problème difficile : lorsque deux prises de la même scène sont alignées, il n'est pas possible de définir un alignement image par image à cause de la grande similarité des images successives. Nous avons choisis de définir un alignement entre séquences d'images, c'est-à-dire que lorsque deux séquences sont alignées, toutes les images d'une séquence sont alignées avec toutes les images de l'autre séquence. Afin d'évaluer la qualité de l'algorithme VSA, nous comparons la matrice d'alignements trouvés par VSA à la vérité terrain. Nous utilisons une mesure de rappel - précision. Le taux de rappel est le rapport entre le nombre de paires d'images correctement alignée par VSA et le nombre de paires alignées dans la vérité terrain. La précision est le rapport entre le nombre de paires d'images correctement alignées par VSA et le nombre de paires d'images alignées par VSA.

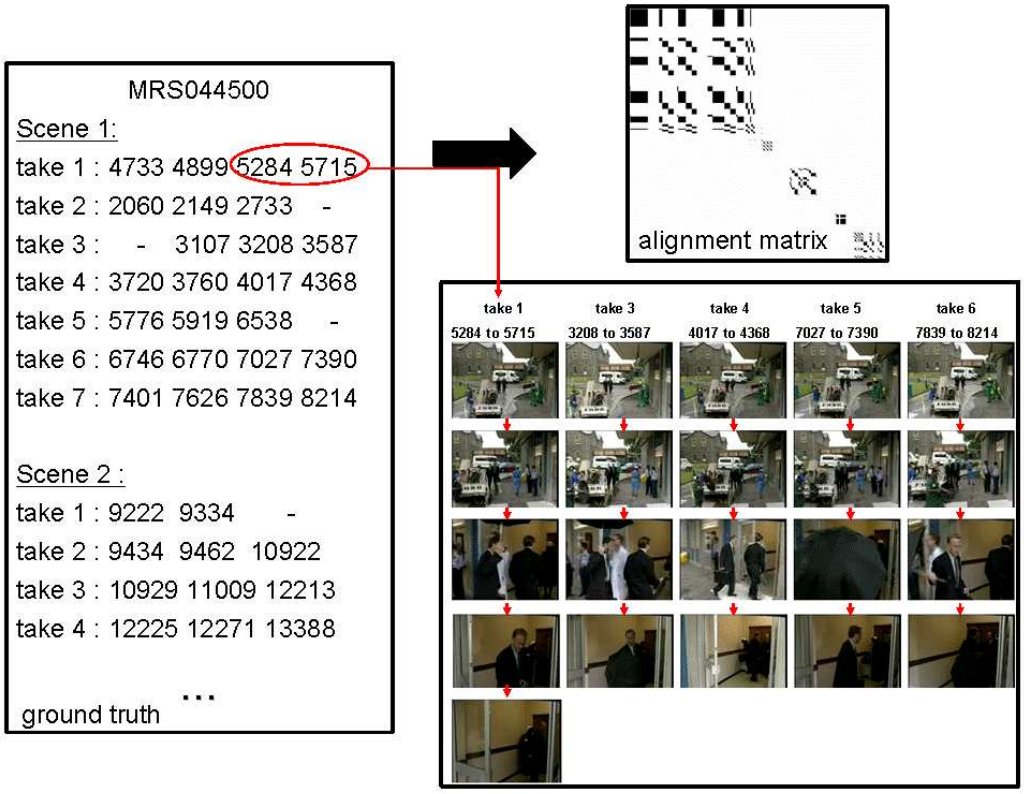


FIG. 4.7 – Illustration du système de construction de la vérité terrain pour la matrice d'alignement.

Grâce à différentes expériences sur l'ensemble de test, dont les résultats sont montrés par la courbe 4.8, nous avons montré que l'utilisation d'un niveau de classification dynamique augmentait considérablement les résultats, et que l'utilisation d'une matrice de scores normalisée donnait de meilleurs résultats et supprime les faux alignements grâce à la détection des scènes. De plus, l'ensemble d'entrainement nous a permis de fixer les seuils au mieux, ainsi que le choix des distances.

4.4.3 Evaluation de la décomposition en scène

Pour l'évaluation de la détection des scènes, nous comparons la surface des scènes de la vérité terrain à celle trouvée : si deux images appartiennent à la même scène aussi bien dans la vérité terrain que par notre algorithme, alors cette allocation est correcte. Nous utilisons aussi un taux de rappel - précision. Le rappel est le rapport entre le nombre de paires d'images correctement allouées à la même scène par notre algorithme et le nombre de paires d'images allouées à la même scène dans la vérité terrain. Le taux de précision est le rapport entre le nombre de paires d'images correctement allouées à la même scène par notre algorithme et le nombre total de paires d'images allouées à la même scène.
Les vidéos de développement nous ont permis de fixer au mieux le seuil du processus de détection de scènes. La figure 4.9 montre la comparaison entre la détection des scènes dans la

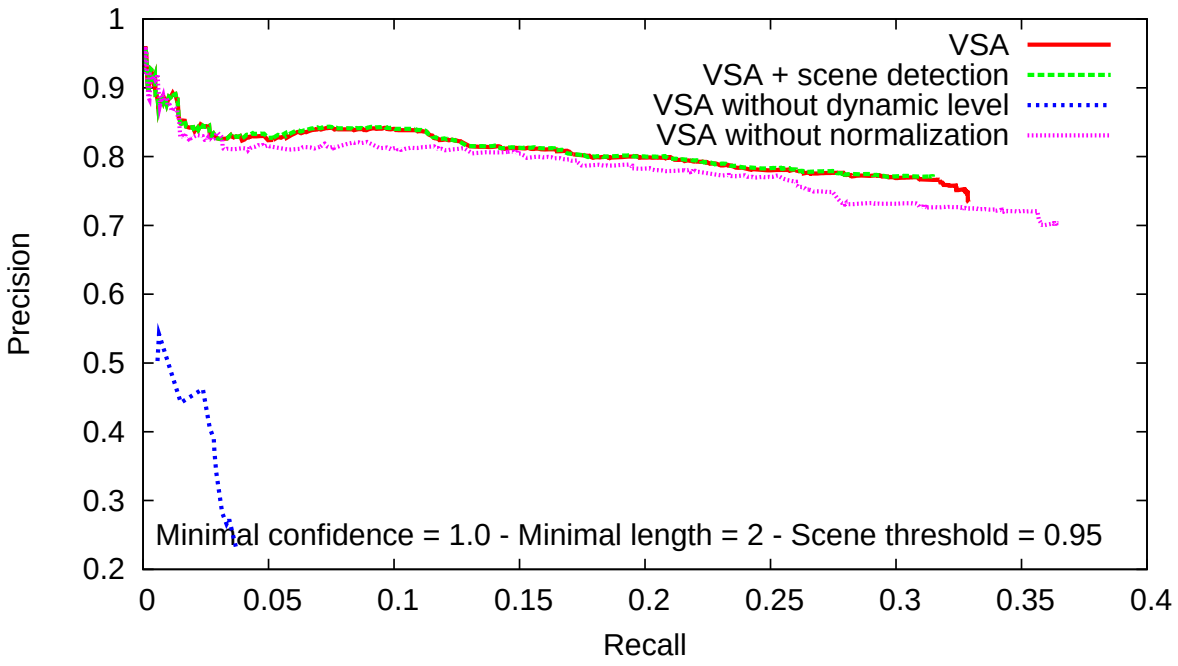


FIG. 4.8 – Résultats obtenus sur l'ensemble de développement

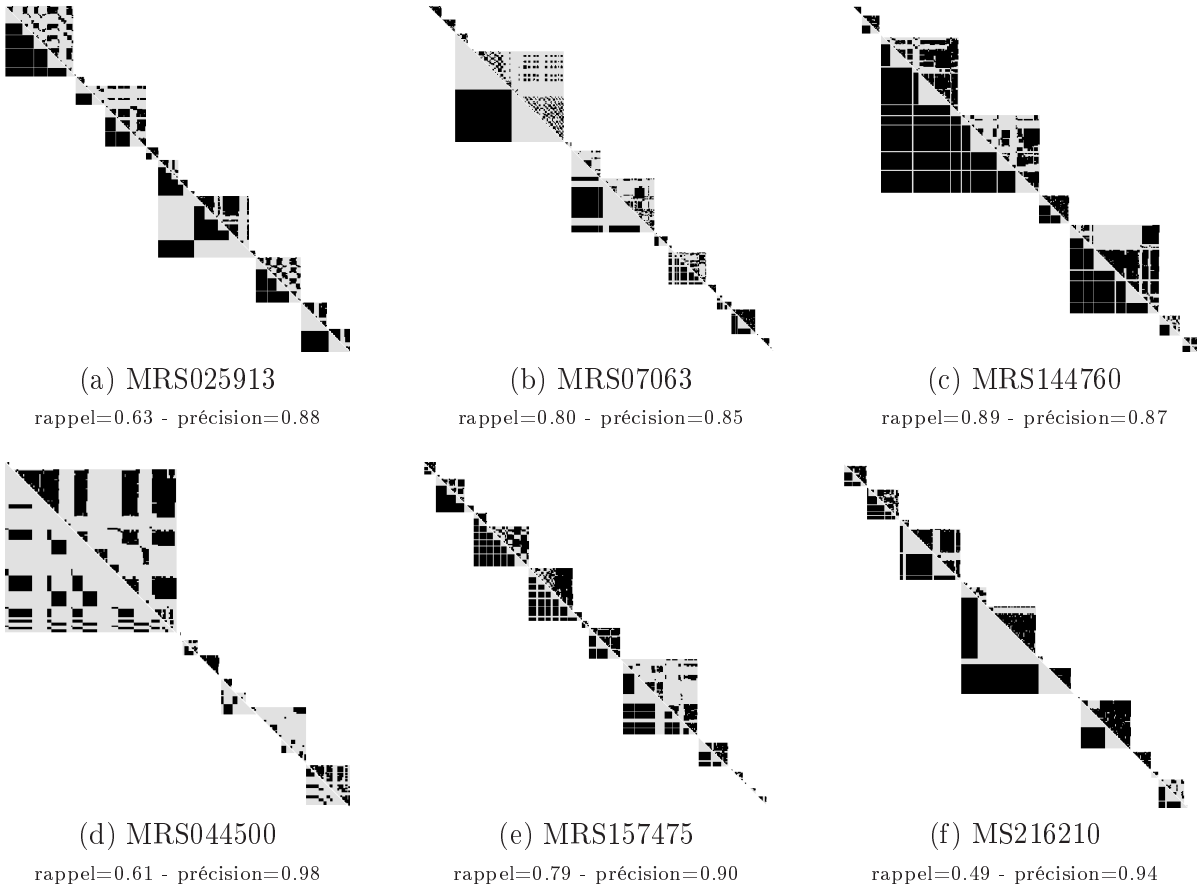
vérité terrain et celles trouvées par notre algorithme ; la différence des scènes est définie grâce au gris dans la figure.

4.4.4 Résultats

L'ensemble d'entrainement nous a permis de fixer les seuils correctement. Pour ces données tests, nous avons pu comparer nos résultats avec ceux de JRS [Bailer 2009]. Le tableau 4.1 donne un récapitulatif de nos résultats et la comparaison avec la méthode de JRS.

		MRS025913	MRS07063	MRS157475	MRS144760	MS216210
Evaluation des alignements de JRS	Rappel	0.241	0.027	0.063	0.369	0.343
	Précision	0.228	0.019	0.059	0.310	0.493
Evaluation des alignements de VSA	Rappel	0.293	0.211	0.154	0.273	0.246
	Précision	0.513	0.487	0.360	0.389	0.660
Evaluation de la détection des scènes de JRS	Rappel	0.30	0.87	0.47	0.46	0.42
	Précision	0.74	0.40	0.61	0.94	0.91
Evaluation de la détection des scènes de VSA	Rappel	0.63	0.80	0.75	0.89	0.49
	Précision	0.88	0.85	0.90	0.87	0.94

TAB. 4.1 – Evaluation de VSA et de la structuration



(a) MRS025913
rappel=0.63 - précision=0.88

(b) MRS07063
rappel=0.80 - précision=0.85

(c) MRS144760
rappel=0.89 - précision=0.87

(d) MRS044500
rappel=0.61 - précision=0.98

(e) MRS157475
rappel=0.79 - précision=0.90

(f) MS216210
rappel=0.49 - précision=0.94

FIG. 4.9 – Comparatif des alignements de la vérité terrain (en dessous de la diagonale) par rapport aux résultats de VSA (en dessus de la diagonale) et de la surface des scènes en gris.

Nous avons appliqué l'algorithme VSA, puis la détection des scènes sur 8 vidéos tests. Le tableau 4.2 présente les résultats de l'évaluation sur ces données tests. VSA aligne correctement les séquences, mais ne les aligne pas toutes : VSA aligne chaque image avec, en moyenne, de 1.54 à 3.87 autres images, alors que pour la vérité terrain ce nombre varie de 2.74 à 6.77 ; le rappel de l'alignement varie de 0.154 à 0.444 quant à la précision, elle varie de 0.422 à 0.762. La structuration de la vidéo est de bonne qualité puisque le rappel varie de 0.554 à 0.932 et la précision de 0.771 à 0.978.

		MRS035126	MRS048773	MRS151585	MRS157479	MRS044499	MRS145229	MRS157450	MS206370
Nombre d'alignements moyen par image	Vérité terrain	5.77	6.77	3.81	3.92	4.92	2.74	4.75	3.97
	VSA	2.47	3.87	2.16	2.35	1.63	1.54	2.85	1.84
Evaluation des alignements	Rappel	0.233	0.277	0.301	0.281	0.444	0.154	0.244	0.163
	Précision	0.747	0.762	0.606	0.595	0.693	0.422	0.581	0.463
Evaluation de la détection des scènes	Rappel	0.716	0.932	0.566	0.749	0.554	0.677	0.765	0.805
	Précision	0.920	0.807	0.883	0.942	0.921	0.766	0.978	0.771

TAB. 4.2 – Evaluation de VSA et de la structuration

5 Participation à TRECVID

5.1 Système basé sur la mesure du contenu visuel

5.1.1 Architecture

La première soumission fut basée sur la méthode du contenu visuel et soumise à TRECVID 2007. L'architecture globale du système est représentée sur la figure 5.1. Une première étape de prétraitement est faite, cette méthode est une version préliminaire de notre méthode de prétraitement expliquée dans le premier chapitre. Nous effectuons une détection des plans de mires, des plans noirs, et des plans courts. Puis, nous appliquons la méthode de sélection des segments basée sur le contenu visuel. Une séquence est définie comme un plan vidéo. Finalement, le résumé est présenté suivant l'idée originale de montrer les quatre plans les plus visuellement dissimilaires et temporellement similaires en même temps, et pour rendre cela possible, nous accélérons dynamiquement les plans d'un groupe de quatre à la durée la plus courte.

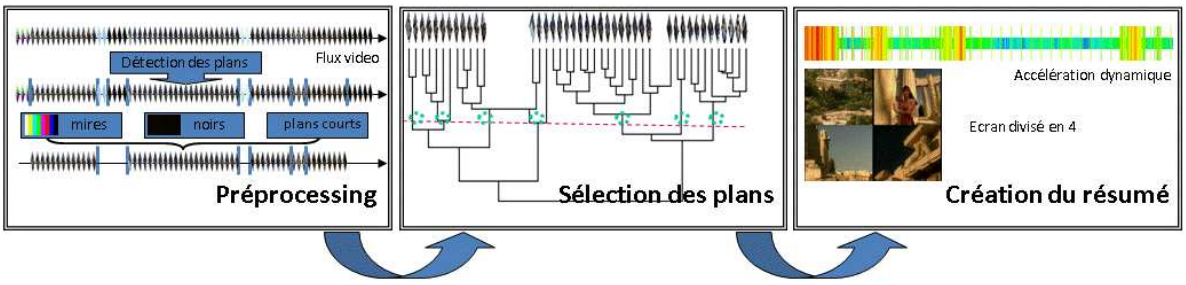


FIG. 5.1 – Architecture du système soumis à TRECVID 2007

5.1.2 Evaluation

La campagne de TRECVID 2007 utilisait les critères d'évaluation suivants :

- DU : durée du résumé (en seconde)
- XD : durée entre la taille du résumé et la taille maximale (en seconde)
- TT : temps total pour juger un résumé (en seconde)
- VT : temps total de lecture utilisé pour juger une vidéo (en seconde)
- IN : pourcentage d'éléments d'histoire retrouvés dans le résumé (0-1)
- EA : le résumé est-il simple à comprendre (mauvais 1-5 bon)
- RE : le résumé contient-il beaucoup de redondances (1-5)

5.1.3 Résultats

Le tableau 5.1 montre les résultats moyens obtenus pour ces différents critères. La première remarque que nous pouvons faire est que nos résumés sont plus courts que la taille maximale proposée, ceci est dû à la mise en pratique de l'idée de montrer quatre plans en même temps. De plus cette idée a augmenté le temps d'évaluation des résumés. Au niveau de la qualité du résumé, le choix de grouper les plans par 4 plans de même longueur a obligé à trop accélérer certains plans et donc de rendre la compréhension de ceux-ci difficile. La méthode de prétraitement n'était quand à elle pas optimisée ce qui a laissé beaucoup de redondance absolue. Le pourcentage de contenu retrouvé IN présente de bons résultats. L'utilisation de 4 plans en même temps a permis une optimisation du contenu visuel présenté et la méthode de mesure du contenu visuel a quant à elle permis une bonne sélection des séquences.

	DU	XD	TT	VT	IN	EA	RE
Eurecom	42	18	119	43	0.53	1.97	3.02
Moyenne	50.5	9.3	93	52	0.48	3.18	3.65
Maximal	64	33.8	119.3	66.6	0.68	3.6	3.98
Minimal	26	-4.34	61.7	28.4	0.25	1.97	3.02

TAB. 5.1 – Résultats moyens obtenus à la soumission TRECVID 2007.

5.2 Système basé sur l'alignement des séquences

5.2.1 Architecture

La deuxième soumission fut basée sur la méthode d'alignement des séquences et soumise à TRECVID 2008. L'architecture globale du système est présentée sur la figure 5.2. Une première étape de prétraitement est faite, cette méthode est la version de notre méthode de prétraitement expliquée dans le premier chapitre. Nous effectuons une détection des plans de mires, des plans uniformes, des plans courts, et des claps. Puis, nous appliquons la méthode d'alignement de séquence basée sur l'alignement des séquences. Cependant la taille des résumés obtenus dépasse la limite des 2% imposé par TRECVID, aussi afin d'arriver à cette limite, nous avons décomposé les prises sélectionnées en segments de deux secondes, puis classifié ces segments et sélectionné le médoide de chaque groupe formé durant la classification.

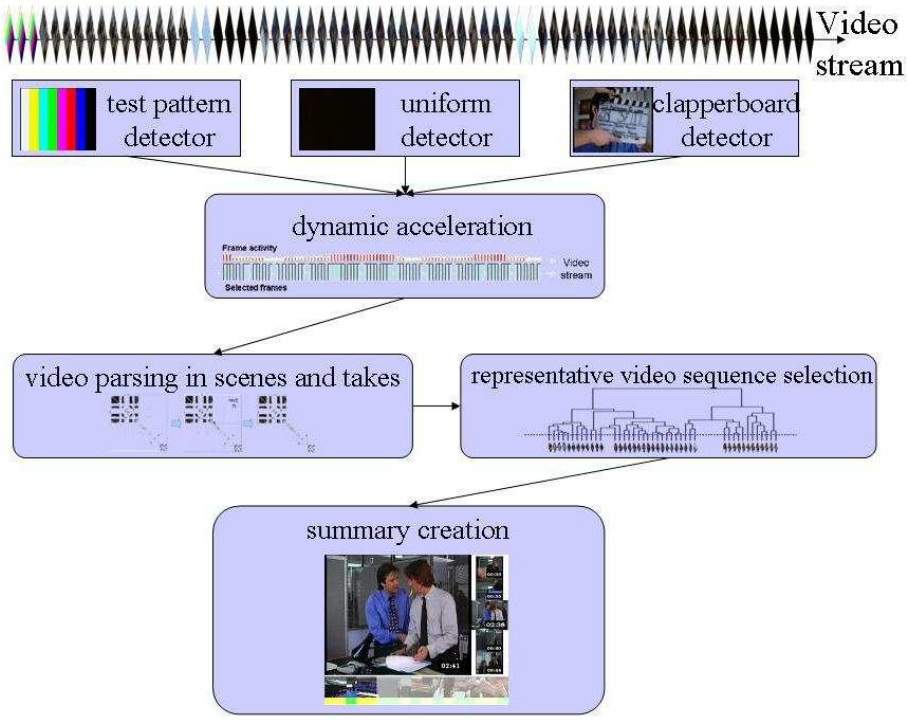


FIG. 5.2 – Architecture du système soumis à TRECVID 2008

Finalement, nous avons présenté les résumés en ajoutant quelques renseignements : un indice temporel, les images clés, l'évolution temporelle, ainsi que la séparation entre les segments. La figure 5.3a, nous propose un exemple. Puis, sur toute la durée restante, c'est-à-dire quelques images, nous avons mis le résumé des images clés ; la figure 5.3b montre un exemple de cette présentation.

(a) Présentation du résumé (b) Images clés

FIG. 5.3 – Exemple de présentation du résumé proposé à TRECVID2008

5.2.2 Evaluation

La campagne de TRECVID 2008 utilisait les critères d'évaluation suivants :

- DU : durée du résumé (en seconde)
- XD : durée entre la taille du résumé et la taille maximale (en seconde)
- TT : temps total pour juger un résumé (en seconde)
- VT : temps total de lecture utilisé pour juger une vidéo (en seconde)
- IN : Pourcentage de séquences présentes dans le résumé (1-0)
- JU : Présence de séquences dites parasites (mauvais 1-5 bon)
- RE : Présence de redondances (mauvais 1-5 bon)
- TE : Qualité visuelle du résumé (mauvais 1-5 bon)

5.2.3 Résultats

Le tableau 5.2 montre les résultats moyens obtenus pour ces différents critères. En montrant durant le temps imparti, les images clés des segments sélectionnés, nous avons utilisé entièrement le temps disponible pour chaque résumé. Notre méthode de prétraitement n'est pas optimale en particulier au niveau de la détection des claps. Le système de présentation du résumé permet une bonne qualité visuelle de celui-ci et la concaténation des segments sélectionnés permet d'avoir un rythme / tempo correct. Le pourcentage de contenu retrouvé est un peu faible : ceci vient de l'idée de découper les prises en segments de deux secondes, puis de sélectionner les médoïdes.

	DU	XD	TT	VT	IN	JU	RE	EA
Eurecom	31.72	0	23.33	34.19	0.39	2.62	3.5	2.75
Moyenne	27.76	4.71	42.13	28.64	0.46	3.23	3.35	2.79
Maximal	47,81	18,15	75,09	53,25	0,83	3,64	3,99	3,38
Minimal	13,56	-16,1	22,56	0,4	0,07	2,52	2,02	1,44

TAB. 5.2 – Résultats moyens obtenus à la soumission TRECVID 2008.

6 Conclusion

Dans ce chapitre, nous avons proposé deux outils permettant la construction de résumés vidéo. Le premier aspect est la mesure du contenu visuel d'une séquence vidéo. Nous avons proposé une méthode de sélection de séquences vidéos se focalisant sur l'idée de sélectionner les séquences les plus informatives. Grâce à une évaluation du système, nous avons pu mettre en avant la taille optimale d'un résumé vidéo, mais aussi une taille de séquence d'images suffisante pour visualiser un élément d'histoire.
Le deuxième outil proposé se base sur la nature des rushes vidéos : ils sont composés de scène

contenant plusieurs sous-prises. Nous proposons donc un outils de décomposition des rushes en scènes et prises permettant d'enlever la redondance dans ces vidéos. Cette méthode permet de bien retrouver les transitions de scènes dans les vidéos.

Nous avons aussi présenté la mise en pratique de ces outils lors de soumission à la campagne d'évaluation TRECVID. Cependant, leur mise en pratique durant les campagnes d'évaluations TRECVID 2007 et 2008 reste délicate. Nous avons appliqués ces deux méthodes séparément, il serait donc intéressant de fusionner ces deux méthodes afin d'optimiser la qualité de la sélection des séquences vidéos.

Cinquième partie

Une approche collaborative

Une approche collaborative

Un résumé vidéo est un outil utile qui permet à un utilisateur de saisir rapidement l'essence du contenu d'une vidéo. Dans le développement de ce sujet de recherche, nous proposons une nouvelle méthode fondée sur la segmentation temporelle et les outils de sélection de contenus sémantiques. L'innovation principale de ce travail est de fusionner les résultats de différentes approches afin de profiter de leurs qualités respectives. Ce système a été développé par différents membres du réseau d'excellence Européen K-Space [12].
Notre système est organisé en deux phases : la première phase est la segmentation temporelle de la vidéo, la deuxième est l'identification des segments pertinents et redondants. Puis, la liste finale des segments pertinents est utilisée pour concaténer les segments vidéo et construire le résumé final. Pour évaluer l'efficacité de cette organisation, nous avons effectué une soumission à la campagne d'évaluation TRECVID 2008.

1 Introduction

Les systèmes de construction automatique de résumés vidéo montrent une grande variabilité dans les résultats produits. Un premier aspect de cette variabilité est lié aux caractéristiques utilisées durant le processus de sélection des séquences importantes. Comme nous l'avons détaillé auparavant, la majorité des systèmes se basent sur les caractéristiques des images (couleur, texture, mouvement, ...), d'autres sur les caractéristiques audio (parole, silence, musique, ...) et certains systèmes utilisent des informations contextuelles ou provenant de la saisie d'un utilisateur. Certains besoins sont difficiles à satisfaire pour une majorité de systèmes, certains systèmes ont des comportements spécifiques leur permettant de mieux répondre à certains objectifs.

Dans le domaine de la recherche d'information, [Fox 1994] ont montré que la combinaison de résultats provenant de plusieurs systèmes améliore les performances par rapport aux résultats individuelles. Les membres du réseau d'excellence Européen K-Space se sont basés sur cette idée et ont décidé de travailler ensemble au développement d'un système commun permettant de fusionner les différentes approches des différents membres.

[12] Knowledge Space of Semantic Inference for Automatic Annotation and Retrieval of Multimedia Content

Dans ce chapitre, nous allons détailler la méthode proposée par K-Space à la campagne d'évaluation TRECVID 2008. Dans un premier temps, nous allons présenter la méthode dans sa globalité, puis chacune des étapes en détail. Nous finirons par montrer les résultats obtenus ainsi qu'une discussion sur ces derniers.

2 Une approche collaborative

2.1 Etat de l'art

Le principe général des approches collaboratives est d'utiliser les compétences de différents centres de recherche en les fusionnant de manière à utiliser au mieux les différentes sources d'information. L'architecture générale d'un système collaboratif est divisée en plusieurs étapes. Par exemple, dans le cas particulier de la construction de rushes vidéos, les étapes sont généralement : la segmentation de la vidéo, l'extraction d'images clés, la détection des images outils et poubelles, la détection de la redondance, puis la sélection des segments vidéo. Ces étapes peuvent être réalisées selon deux grands schéma différents : en parallèle ou en séquence.

La première catégorie est formée des méthodes séquentielles : le système est découpé en différentes phases et chaque phase est exécutée par un seul laboratoire, comme le montre la figure 2.1. Ces architectures sont les plus répandues [Toharia 2008, Laganière 2008, Ren 2008, Sano 2008, Gorisse 2008, Putpuek 2008, Naci 2008]. L'avantage d'un tel système est sa simplicité à mettre en place : après avoir défini les différentes étapes, chacune d'entre elle est attribuée au laboratoire le plus adaptés à répondre à la tâche demandée. Une telle méthode est efficace : chaque laboratoire ne traitant que la tâche dont il a des outils adaptés et efficaces.

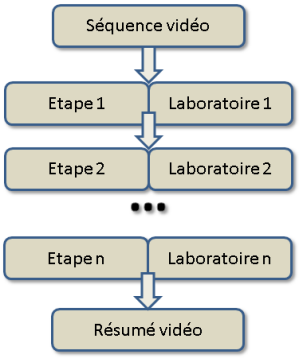


FIG. 2.1 – Schéma global d'un système collaboratif séquentiel

La deuxième catégorie est celle des systèmes basés sur un processus parallèle : chaque laboratoire répond à la tâche demandée puis les résultats sont fusionnés. Le système peut-être découpé en plusieurs phase ou non, comme le montre la figure 2.2. Cette catégorie d'architecture est plus difficile à mettre en place et demande plus d'interaction entre les différents acteurs. De plus, des méthodes de fusion doivent être mise en place pour combiner les différents résultats. La qualité des ces architectures dépend beaucoup de ces méthodes de fusion. Pour ces raisons ces architectures dont plus rarement utilisées. Pour la construction de résumés vidéo, nous avons proposé une telle architecture.

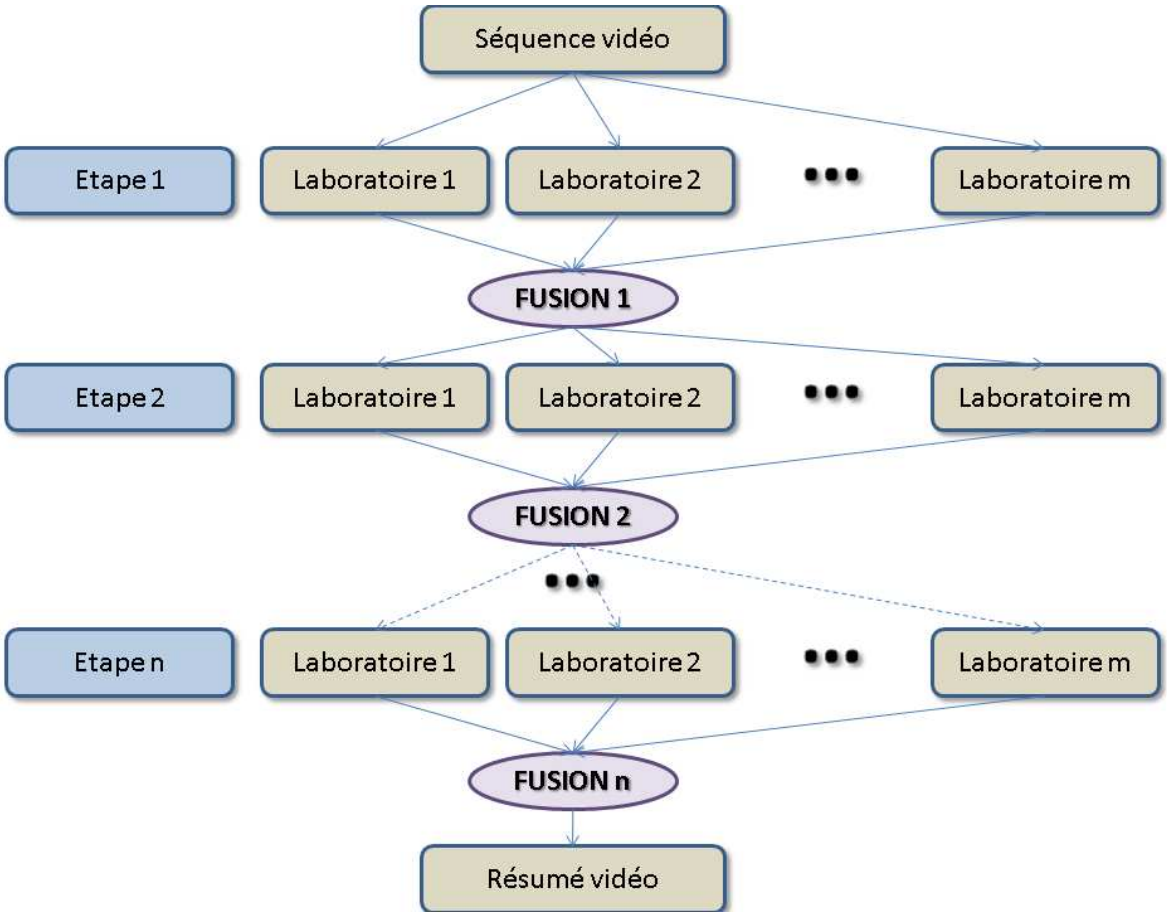


FIG. 2.2 – Schéma global d'un système collaboratif parallèle

Le consortium IRIM, une collaboration entre plusieurs laboratoires français, utilise l'idée générale de combiner leurs ressources [Quénot 2008] dans une architecture mixe. Des caractéristiques de bas niveau sont extraites par les différents acteurs, dont des indicateurs audio et l'activité basée sur le mouvement, afin de détecter les transitions de plans. Pour supprimer les plans poubelles et outils, ils utilisent des caractéristiques de moyen niveau, tels que la détection des visages ou encore les mouvements de caméra. Ensuite, les segments redondants sont détectés par une méthode de classification. Finalement, ils utilisent une méthode de fusion pour choisir les segments à inclure dans le résumé. Quatre critères sont extraits parallèlement : l'activité visuelle, l'activité sonore, les mouvements de caméra, et la détection de visages. Indépendamment, ces critères permettent de sélectionner les plans à inclure dans le résumé. Ces

résultats sont alors fusionnés par le mécanisme présenté par la figure 2.3.

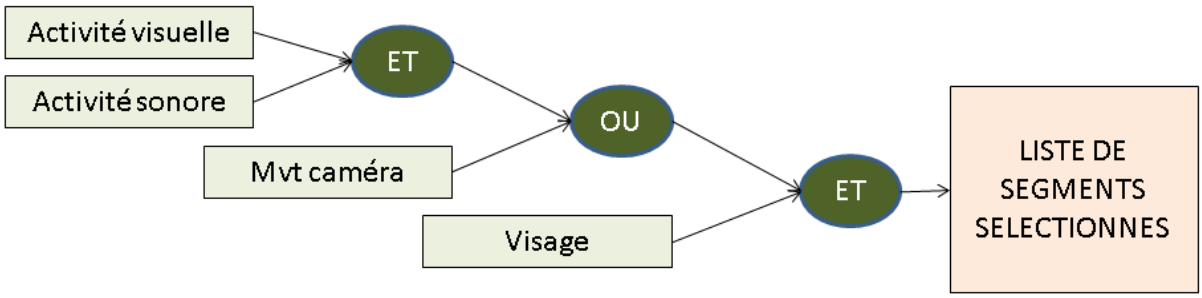


FIG. 2.3 – Méthode de fusion pour la sélection des plans.

2.2 La collaboration K-Space

Ce travail intervient dans le cadre du réseau européen d'excellence K-Space. L'objectif général du réseau est de réduire l'écart entre les descripteurs de bas niveau et ceux de haut niveau reflétant l'interprétation humaine des médias audiovisuels. Le but de cette collaboration est de développer un système de construction de résumés vidéo automatiques de rushes. L'idée principale est de fusionner les résultats des diverses approches provenant de différents laboratoires de recherche. Nous prévoyons qu'une approche combinée de systèmes sera plus précise et plus robuste que des approches individuelles. Pour appliquer cette stratégie dans le cadre de la campagne d'évaluation TRECVID, nous avons conçu une architecture en trois phases :

- Premièrement, nous construisons une segmentation temporelle commune de la vidéo. Ceci est réalisé en fusionnant des segmentations temporelles fondées sur différents indicateurs.
- Deuxièmement, les segments provenant de la segmentation temporelle commune sont évalués séparément en terme de redondance et pertinence. Chaque partenaire contribue en suggérant des listes de segments pertinents et redondants. Ces listes sont unifiées pour composer une liste classée finale des segments choisis.
- Troisièmement, le résumé est assemblé en mettant bout-à-bout les segments sélectionnés puis accéléré afin d'obtenir le temps désiré.

La figure 2.4 donne un aperçu général de cette organisation. La construction des résumés vidéo est donc réalisée par les étapes suivantes :

- Chaque partenaire propose une ou plusieurs segmentations temporelles de la vidéo originale, basée sur divers indicateurs, ainsi que les valeurs de confiance pour chaque transition suggérée.
- Ces segmentations sont fusionnées pour produire une segmentation temporelle commune de la vidéo originale.
- Chaque partenaire analyse les segments communs pour détecter des redondances afin de produire une liste de segments redondants qui ne seront pas inclus dans le résumé parce qu'ils ne présentent pas de contenu intéressant, ou parce qu'ils sont similaires aux autres segments.
- Chaque partenaire évalue la pertinence de chaque segment et produit une liste classée des segments choisis avec un indicateur sur la pertinence de chaque segment commun relatif à l'information contenue dans la vidéo originale.

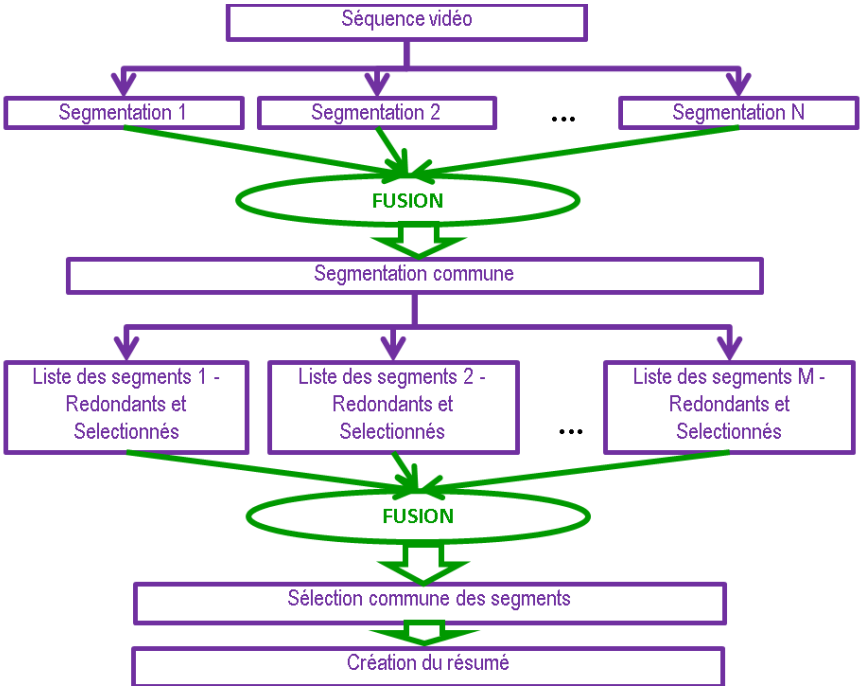


FIG. 2.4 – Schéma global du système collaboratif

- Ces listes sont prises en compte pour produire une liste classée de segments communs choisis.
- Finalement, un résumé vidéo est construit en concaténant et accélérant les clips des segments choisis.

3 Segmentation temporelle

Dans un premier temps, plusieurs segmentations temporelles de la vidéo originale sont produites ; elles sont fondées sur divers indicateurs et diverses caractéristiques. Elles prennent en compte le fait qu'un rush vidéo ne comporte que des transitions brutes. Et les différentes segmentations temporelles sont fusionnées afin de produire une segmentation temporelle commune. La segmentation temporelle, dans ce cas, n'est pas une détection de plans. L'idée est d'identifier les segments représentant un élément d'histoire : un objet, un événement ou un mouvement de caméra. L'avantage de travailler sur l'unité temporelle des plans est qu'un plan représente une séquence complète d'un évènement. Cependant, dans le cas particulier des rushes, l'unité de longueur des plans est beaucoup trop longue, et un résumé ne pourrait contenir qu'un ou deux plans.

3.1 Approches individuelles

Une première approche est proposée par le laboratoire JOANNEUM Research. La détection des transitions se base sur un SVM entrainé sur la différence de couleurs des trois images précédentes. Ce SVM est entrainé grâce à l'implémentation de LIBSVM [Chang 2001]. L'ensemble d'entrainement provient de la base de données proposées par TRECVID2006 pour la tâche de détection de plans.

Une deuxième méthode est proposée par Technische Universität Berlin. La caractérisation des mouvements de caméra est utilisée pour identifier des segments avec un mouvement de caméra cohérent. Les mouvements de caméra globaux sont estimés et décrits par les caractéristiques appropriées. Ces caractéristiques sont alors utilisées pour une classification multi-classe par SVM , les différentes classes sont panoramiques à droite, à gauche, en haut, en bas, nul et zoom avant, arrière, nul [Haller 2007]. L'identification des segments se base sur les résultats obtenus et l'idée qu'un segment comporte un même mouvement de caméra. La notion de confiance de la transition proposée est déterminée par l'approche de Platt [Platt 2000] durant la classification par SVM.

L'approche que nous proposons pour la segmentation temporelle est la même que celle décrite dans les chapitres précédents. Pour la détection des transitions brusques, sur les 16 régions d'une image, les 4 régions formant le centre de l'image sont négligées afin que la détection soit moins influencée par des mouvements rapides. Pour chaque image, la similarité entre celle-ci et les autres images de la fenêtre est calculée. Ensuite, les images sont classées par similarité croissante et le nombre d'images précédant l'image courante étant classée sur la moitié supérieure est enregistré. Une possible transition brusque est détectée si ce nombre chute fortement et est validée si la différence entre l'image précédant la transition et l'image suivant la transition est suffisament élevée. Cette différence représente la valeur de confiance que nous associons à chaque transition. Nous avons proposé une deuxième segmentation plus fine qui utilise la même méthode mais les seuils ont été baissés, afin que la détection soit plus sensible.

3.2 Fusion

Les différentes segmentations temporelles sont fusionnées en se basant sur les confiances associées à chaque transition proposée afin de produire une segmentation temporelle commune. Dans un premier temps, le GET normalise les valeurs de confiance qui correspondent à chaque système par une méthode de normalisation gaussienne, c'est-à-dire, en les centrant et en réglant leur variance à 1. Ensuite une classification hiérarchique agglomérative permet de regrouper les transitions les plus proches [Reas 2007]. Initialement, chaque segment forme un groupe. A chaque étape, les deux groupes les plus proches temporellement sont fusionnés, le système s'arrête lorsque les groupes les plus proches ont une différence temporelle supérieure à 5 secondes.

La dernière étape consiste à choisir le meilleur représentant de chaque groupe : les singletons sont gardés si la valeur de confiance est supérieure à un seuil (fixé à -1), et pour les groupes de non-singleton, la transition ayant la plus grande confiance est gardée. La figure 3.1 montre un aperçu général du système de fusion.

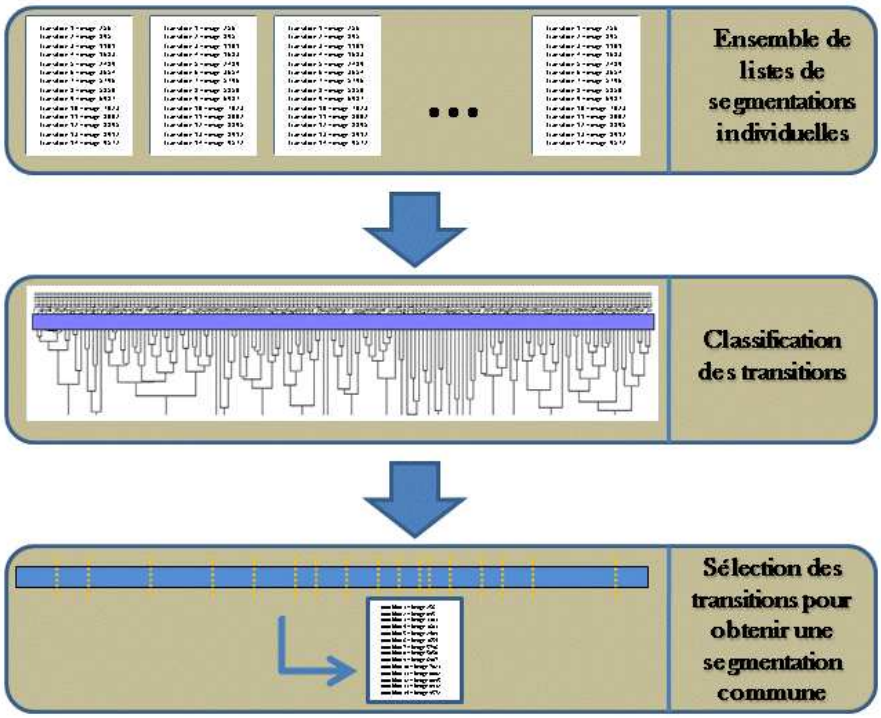


FIG. 3.1 – Schéma global du système de fusion des segmentations temporelles

4 Sélection des segments

Nous utilisons deux stratégies pour déterminer les segments qui sont inclus dans le résumé. L'une est la sélection explicite des segments qui sont classés comme pertinents. Pour chacun de ces segments, une valeur de pertinence est déterminée. L'autre est de déterminer les segments redondants qui ne seront pas inclus. La redondance d'un segment peut être absolue, c'est-à-dire que le contenu n'est pas informatif, par exemple une mire ou relatif à une série de segments, c'est-à-dire que ces segments contiennent le même contenu qu'un autre et n'a pas d'intérêt à être sélectionné. Nous commençons par décrire les méthodes individuelles, puis la fusion de ces méthodes.

4.1 Approches individuelles des segments pertinents

Une première méthode est proposée par JOANNEUM Research. Une liste de segments pertinents basée sur les caractéristiques visuelles est créée à partir de l'activité visuelle et de

la détection de visage. Le détecteur de visage est basé sur l'algorithme développé par Viola et Jones [Viola 2004]. Le résultat de la détection de visage forme une fonction discrète. Pour éliminer les fausses détection (c'est-à-dire l'apparition ou la disparition soudaine de visages), les opérateurs mathématiques de fermeture et ouverture morphologique ont été appliqués à la séquence de résultats de la détection de visage. La pertinence d'un segment est proportionnelle à l'activité visuelle, mais cette valeur est réduite si aucun visage n'est présent dans le segment.

L'université Queen Mary de Londres a proposé une deuxième approche. Une étape fondamentale de celle-ci est de créer la matrice de similarité et d'organiser les images de la vidéo par une méthode d'arbre de classification. Ils utilisent une combinaison de descripteurs MPEG-7 [MPEG-7 2002], par exemple la répartition des couleurs et les histogrammes de contour, pour représenter une image vidéo. En commençant par la racine, les images sont petit à petit réparties dans l'arbre, la classification est basée sur les valeurs de similarité, des informations temporelles et des informations sur les images voisines. Pour trouver le nombre optimal de branches et pour augmenter la qualité de la classification, le seuil de similarité est optimisé. Les résultats de l'algorithme de classification sont des groupes qui sont utilisés dans le processus de décision pour la classification de segments pertinents/redondants : les segments vidéo ayant des images directement reliées à la racine sont les plus pertinents, la valeur de la pertinence est directement liée à la redondance relative de chaque segment, c'est-à-dire que la pertinence d'un segment est définie grâce au nombre d'images des segments redondants dans le groupe correspondant.

La méthode que nous proposons est basée sur la notion de la maximisation du recouvrement de contenu que nous avons détaillée dans le chapitre précédent. Nous divisons la vidéo originale en segments vidéo d'une seconde, et nous regroupons ces segments par une classification agglomérative hiérarchique. Pour ce faire, nous utilisons des histogrammes HSV et la distance entre deux segments est calculée comme la distance Euclidienne d'histogrammes, et la distance entre deux groupes est la distance moyenne à travers toutes paires possibles d'un deuxièmes segments de chaque groupe. Nous choisissons itérativement les segments communs qui recouvrent au maximum le contenu. L'importance d'un segment est définie comme le nombre de classes qu'il recouvre.

4.2 Approches individuelles de la détection de la redondance

Les approches de détection des segments redondants se décomposent en deux phases : l'identification de la redondance relative et l'identification de la redondance absolue. La redondance relative est dépendante de la vidéo, cette redondance est liée à la notion de segment répétitif. Des groupes de redondances sont donc définis par les différentes méthodes. La redondance absolue regroupe les segments ne contenant pas d'information, par exemple, les mires, ou encore les séquences vidéos toutes noires.

Redondance relative

JOANNEUM Research propose une méthode d'identification et de groupement des séquences vidéos identiques comme la méthode de [Damnjanovic 2007]. Le problème de détecter et rassembler les séquences répétitives est formulé comme un problème de séquences identiques en terme

de caractéristiques visuelles de segments vidéos. L'algorithme utilise l'algorithme LCSS [13] pour déterminer la similarité entre deux segments. Une classification hiérarchique est appliqué à la matrice résultante des distances et produit une série de groupes qui correspond aux segments répétitifs.

Les segments redondants déterminés par l'université Queen Mary de Londres découlent directement de l'algorithme proposé pour déterminer les segments pertinents ; de même pour notre approche. Les segments étant regroupés par similarité, les redondances visuelles en résultent.

Redondance absolue

Une méthode classique est utilisée par JOANNEUM research pour déterminer les segments de mires, et les segments à couleur uniforme : si l'écart type des colonnes d'un nombre suffisant d'images de la séquence est inférieur à un seuil, alors ce segment est redondant.

L'université Queen Mary de Londres propose de calculer l'écart type sur l'intégralité des images du segment, puis également grâce à un seuil, détermine les segments redondants.

Nous avons appliqué la méthode développée dans le premier chapitre de cette thèse "Prétraitement vidéo". Trois détecteurs indépendants sont utilisés : un détecteur de mire se basant sur les couleurs HSV des images, un détecteur d'images non informatives se basant sur l'entropie de la distribution des couleurs dans l'image, et enfin, un classifieur de clap appris par SVM.

4.3 Fusion

Partant des différentes listes proposées, JOANNEUM Research execute une fusion afin de produire une liste de sélection commune des segments qui seront inclus dans le résumé final. Les étapes de fusion sont représentées par le schéma 4.1, et se décomposent comme suit :

- Les listes de redondances relatives sont transformées en listes de redondances absolues.
- Les listes de segments pertinents et redondants sont combinées.
- Un seuil sur la pertinence, approprié et adapté à la sélection de segments aux contraintes de longueur du résumé, est fixé.

L'information relative de redondance ne peut pas être directement utilisée, c'est-à-dire que pour un segment visuel donné, plusieurs segments sont visuellement redondant à celui-ci, mais parmi ce groupe de segments redondants, un seul doit être gardé et considéré comme pertinent. Cette liste prend en entrée tous les segments communs de la vidéo à l'exception des segments considérés comme trop courts. Les segments trop longs sont quant à eux segmentés. L'idée étant que les segments trop courts ne peuvent pas être correctement perçus par un utilisateur, il est donc plus intéressant de sélectionner un segment visuellement identique, mais perceptible. De plus, un résumé vidéo ne doit pas proposer de redondance visuelle : un segment qui dure plusieurs minutes (comme cela arrive très régulièrement dans les rushes vidéo) est décomposé en plusieurs segments qui vont être considérés comme redondants les uns par rapport aux autres. Par exemple, une vue sur une maison durant 3 minutes sera détectée comme un seul et unique segment, cependant montrer 3 secondes dans le résumé est suffisant. A chacun de ces segments

[13] Longuest Common Subsequence

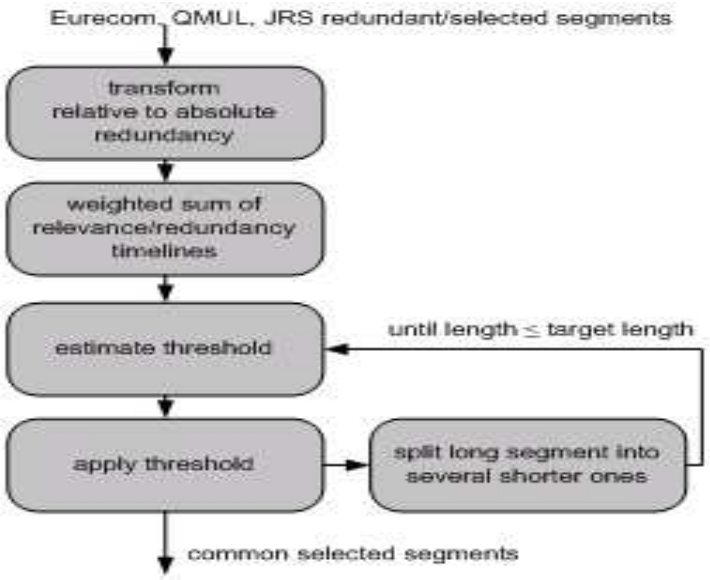


FIG. 4.1 – Schéma global du système de la fusion des listes de segments redondants et pertinents.

est associé la somme des pertinences proposées par chaque méthode individuelle, les segments considérés comme de la redondance absolue ont une valeur de pertinence nulle.

Parmi un groupe de segments redondants, il faut sélectionner le segment le plus représentatif, et ôter les autres. Deux aspects sont proposés :

- Soit utiliser les segments les plus longs parmi un groupe de redondances en partant de l'idée qu'il sera plus complet. Certaines prises vidéos sont des ratés et sont donc arrêtées avant la fin, pour cette raison un segment peut avoir seulement une grande partie de la prise, mais pas l'intégralité.

- Soit utiliser le segment le plus représentatif d'un groupe de segments redondants entre eux en terme de confiance déterminée par les méthodes individuelles.

La liste des segments redondants et pertinents est interprétée comme des fonctions de $rel(t)$ et $red(t)$ où t représente un indice temporel. La fonction de sélection est définie comme suit :

$$\text{select}(t) = \begin{cases} 1 & \text{si} \frac{w_{rel}}{n_{rel}} \sum_{k=1}^{n_{rel}} rel(t) + \frac{w_{rel}}{n_{rel}} \sum_{k=1}^{n_{red}} rel(t) \geq \theta \\ 0 & \text{sinon} \end{cases} \tag{4.1}$$

où n_{rel} est le nombre total d'entrées pertinentes, n_{red} celui des entrées redondantes, w_{rel} et w_{red} sont les poids relatifs associés à la pertinence et à la redondance des segments et θ est un seuil.

5 Présentation visuelle du résumé

A ce stade, la liste des segments vidéo sélectionnés pour le résumé final est définie. Deux stratégies qui sont dépendantes de la méthode de fusion ont été adoptées par Dublin City University : soit sélectionner un petit nombre de segments longs, soit sélectionner un plus grand nombre de segments mais plus courts. Dans la vidéo finale, les segments sont uniformément accélérés. Dans la première configuration, les segments sont assez longs, ils ont une durée minimale de 10 secondes, l'accélération peut être assez rapide, une accélération de 4 fois la vitesse originale a été choisie. Dans la deuxième configuration, les segments ne dépassent pas 2 secondes, par conséquent une accélération beaucoup plus faible à été réalisée : la vitesse a été multipliée par 1.5.

Chaque résumé inclut en plus une chronologie en bas de l'écran. La chronologie est extrêmement transparente pour empêcher l'occlusion de l'image mais est suffisamment visible pour fournir une indication utile correspondant à l'emplacement du segment en cours de lecture par rapport à la longueur initiale de la vidéo. La figure 5.1 montre une image extraite d'un résumé vidéo.

FIG. 5.1 – Exemple du format de présentation du résumé vidéo final

6 Evaluation

6.1 Protocole

L'approche collaborative K-Space présentée à été testée grâce à la collaboration de six laboratoires de recherche différents : JOANNEUM Research (JRS), Technische Universität Berlin (TUB), TELECOM ParisTech (GET), l'université Queen Mary (QMUL), Dublin City University (DCU) et Eurécom. La répartition des tâches a été effectuée comme le montre le schéma 6.1. La mise en place d'un tel système ne permet pas une évaluation rapide du système. Les résumés vidéos évalués sont donc basés sur la campagne d'évaluation TRECVID 2008 [Over 2007] dans laquelle 39 résumés vidéos ont été évalués pour chacune de nos deux méthodes.

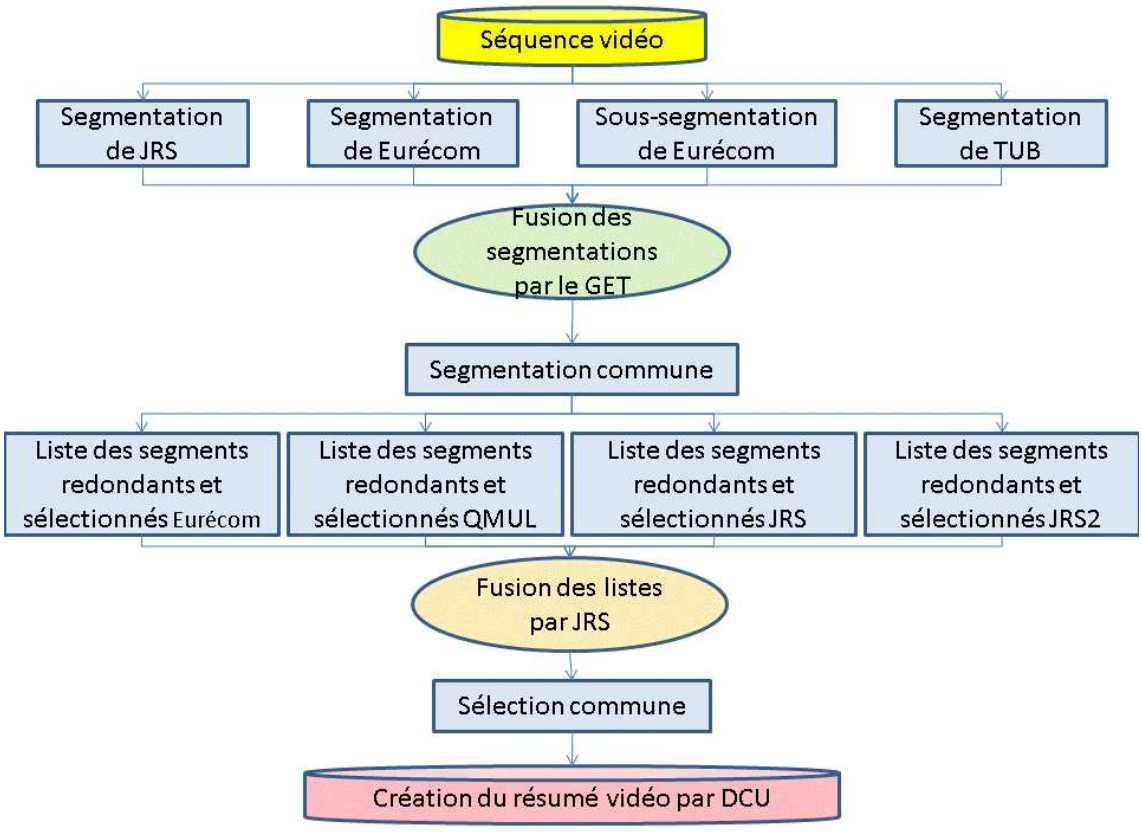


FIG. 6.1 – Schéma global du système collaboratif K-Space.

6.2 Résultats

6.2.1 Segmentation temporelle

La première étape du système consiste à fusionner différentes segmentations, le tableau 6.1 donne un comparatif des résultats des méthodes individuelles et de la fusion.

Méthode	Nombre moyen de segments par vidéo	Taille moyenne des segments (nb d'images)
Segmentation de JRS	56.125	709.500
Segmentation de Eurecom	104.075	382.613
Sous-segmentation de Eurecom	185.725	214.405
Segmentation de TUB	561.700	70.890
Segmentation commune	70.450	565.23

TAB. 6.1 – Comparatif des résultats des méthodes individuelles de segmentation temporelle avec la méthode de la fusion.

Ce tableau met en évidence la notion de méthode individuelle, chaque laboratoire de recherche a proposé une méthode de segmentation basée sur ses points forts et sur sa vision du problème. Le plus grand contraste est entre la méthode proposée par JRS et celle proposée par TUB. JRS propose des segments d'une durée moyenne de 28 secondes, alors que TUB propose des segments d'un peu plus de 3 secondes. Nous avons proposé deux méthodes de segmentation intermédiaires, l'une proposant des segments de 15 secondes en moyenne, et l'autre d'environ 9 secondes. Le seuil fixé dans la méthode de fusion se rapportant au regroupement de toutes les transitions détectées dans un voisinage de 5 secondes a limité l'impact de la segmentation de TUB dans la segmentation finale. Les segments communs ont une durée moyenne de 22 secondes, une vidéo est segmentée, en moyenne, en 70 segments.

6.2.2 Sélection des segments pertinents

La deuxième étape du système consiste à fusionner différentes listes de segments pertinents et redondants, le tableau 6.2 donne un comparatifs des résultats des méthodes individuelles et de la fusion. JRS sélectionne un "maximum" de segments : leur méthode a tendance à sélectionner les segments les plus courts. Inversement notre système se focalise sur les segments ayant le plus de contenu, il sélectionne plus facilement les longs segments. La deuxième segmentation commune se rapproche de la moyenne des méthodes individuelles. La première méthode a, quant à elle, découpé certains segments : elle a donc sélectionné plus de segments, mais d'une durée plus courte.

6.2.3 Evaluation de TRECVID

Les résultats proposés sont donc basés sur la campagne d'évaluation TRECVID 2008 [Over 2007] dans laquelle 39 résumés vidéos ont été évalués pour chacune de nos deux méthodes. Plusieurs critères sont utilisés pour l'évaluation de résumé :

- DU - durée du résumé (en secondes)
- XD - différence entre la durée maximale autorisée et la durée du résumé (en seconde)

Méthode	Nombre moyen de segments sélectionnés par vidéo	Nombre moyen de segments redondants par vidéos
Sélection de JRS	13.38	63.98
Sélection de Eurecom	3.52	70.45
Sélection de QMUL	5.30	65.50
Sélection commune 1	19.67	n/a
Sélection commune 2	6.68	n/a

TAB. 6.2 – Comparatif des résultats des méthodes individuelles de sélection des segments avec la méthode de la fusion.

- TT - durée de l'évaluation de IN (en seconde)
- VT - durée de lecture utilisée pour l'évaluation de IN (en seconde)
- IN - fraction d'inclusions trouvées dans le résumé (0 - 1)
- JU - le résumé contient beaucoup d'images parasites : 1 (oui) - 5 (non)
- RE - redondances visuelles présente dans le résumé : 1 (oui) - 5 (non)
- TE - le résumé a un tempo/rythme agréable : 1 (oui) - 5 (non)

Le système de référence, la baseline, est la vidéo initiale accélérée 50 fois. Les résultats suggèrent que notre système est raisonnable, surtout étant donné le fait que cette évaluation n'est pas facilement reproductible. Pour une première soumission, il ne nous a pas été possible de vérifier la qualité des systèmes individuels, même si cela reste évaluable par d'autres moyens, mais une évaluation spécifique à une tâche reste la meilleure évaluation. Surtout, il a été très délicat de paramétrer correctement les systèmes de fusion.

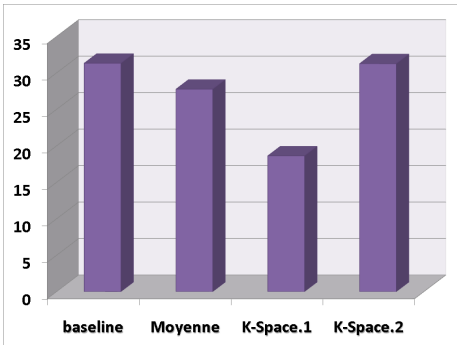
Critères temporels

La moyenne des résultats sur les 39 vidéos pour les différents critères temporels est présentée par le graphique 6.2.

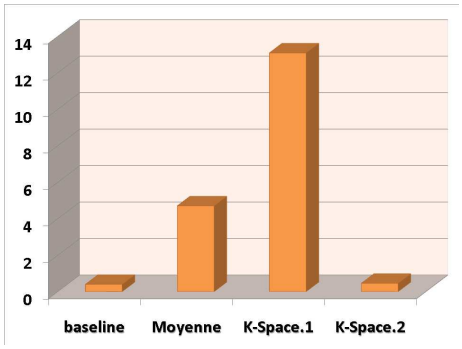
Pour la durée des résumés, la baseline donne une estimation sur la durée maximale autorisée. La méthode utilisant des segments courts permet clairement d'inclure un maximum d'information durant le temps imparti contrairement à l'utilisation de segments longs. En moyenne, les participants ont des résumés plus court de 5 secondes par rapport à la baseline et K-Space2, alors que K-Space1 est plus court de 15 secondes.
Concernant le durée consacrée à l'évaluation, nous pouvons nous apercevoir que pour évaluer les résumés de K-Space1 et K-Space2, les évaluateurs ont pris, en moyenne et respectivement, 7 et 8 secondes de pause. La moyenne de l'ensemble des participants est de 13 secondes et la difficulté visuelle de la baseline a porté cette valeur à 28 secondes. Ceci nous donne une indication sur la facilité à visualiser et interpréter le contenu de nos résumés : les évaluateurs n'ont pas besoin de faire de pause pour réfléchir au contenu, contrairement à la baseline si rapide et si difficile à visualiser, que des pauses s'imposent.

Qualité visuelle du résumé

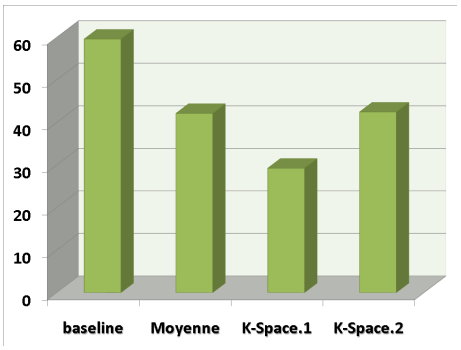
Le graphique 6.3 donne un comparatif entre les résultats obtenus par K-Space, la baseline et la moyenne des participants pour les critères TE (le résumé a un tempo/rythme agréable), JU (le



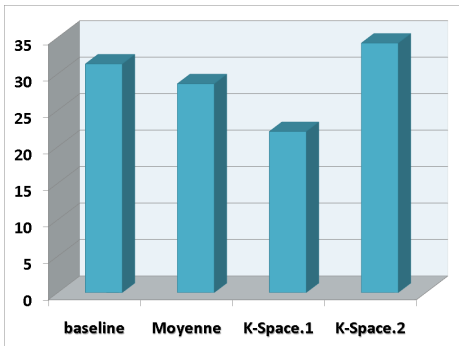
(a) DU : Durée du résumé en secondes.



(b) XD : Durée autorisée non utilisée



(c) TT : Durée de l'évaluation de IN



(d) VT : Durée de lecture utilisée pour l'évaluation

FIG. 6.2 – Comparaison des résultats obtenus par K-Space avec la baseline et la moyenne des participants pour les critères temporels.

résumé contient beaucoup d'images parasites), et RE (le résumé vidéo contient des redondances visuelles).

La première remarque que nous pouvons faire est que les résultats obtenus pour ces trois critères sont bons, et tous au dessus de la moyenne des participants. Nous avons correctement enlevé la redondance aussi bien absolue que relative. Ce qui signifie que les méthodes individuelles sont de bonnes qualités et que la méthode de fusion proposée garde ces bonnes qualités, mais surtout qu'elle est robuste face aux différentes approches proposées. Certaines méthodes individuelles sont basées sur le contenu visuel alors que d'autres se basent sur les mouvements de caméra. La fusion est de bonne qualité et réalise un bon compromis entre ces différentes approches. Mais, il nous est, actuellement, pas possible de comparer les qualités individuelles des systèmes à celle du système fusionné. Malgré tout, nous savons que la méthode fusionnée détermine correctement les séquences parasites ainsi que la redondance. Le critère concernant TE est égal à la moyenne des participants.

Critère sur le contenu visuel

Le graphique 6.4 donne un comparatif entre les résultats obtenus par K-Space, la baseline et la moyenne des participants pour le IN.

Pour ce critère la baseline n'est pas idéale : l'ensemble de la vidéo est sélectionné pour être dans le résumé final mais avec une très grande accélération : par conséquent, seules les actions

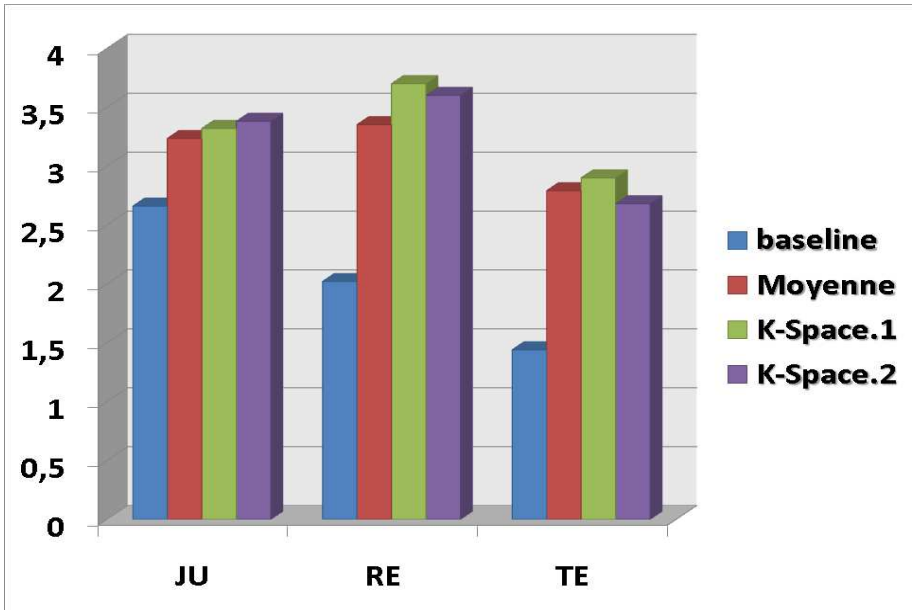


FIG. 6.3 – Comparatif entre les résultats obtenus par K-Space, la baseline et la moyenne des participants pour les critères TE (le résumé a un tempo/rythme agréable), JU (le résumé contient beaucoup d'images parasites), et RE (le résumé vidéo contient des redondances visuelles).

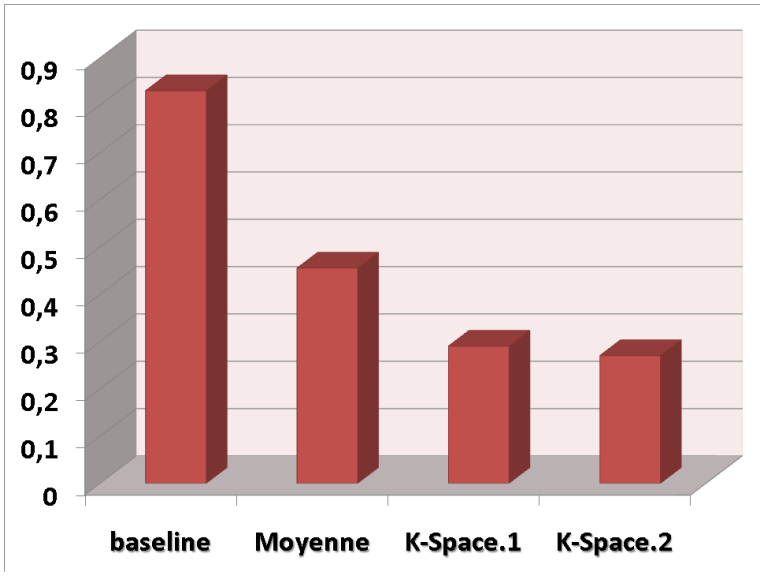


FIG. 6.4 – IN - fraction d'inclusions trouvées dans le résumé

très rapides et certains mouvements de caméra sont perdus. La fraction d'inclusions IN reste faible par rapport à la moyenne alors que nous avons vu précédemment que le contenu proposé dans les résumés n'est pas redondant. Ceci provient du fait que nous avons des segments trop longs par rapport à l'information contenue dans ceux-ci. Un découpage plus fin des segments ainsi qu'une accélération plus rapide permettrait d'inclure plus de contenu et par conséquent

d'améliorer ce critère.

7 Conclusion

Dans ce chapitre, nous avons présenté une méthode collaborative de construction automatique de résumés vidéo de rushes. Notre système est organisé en deux phases : la première phase est la segmentation temporelle de la vidéo, la deuxième est l'identification des segments pertinents et redondants. Ces deux étapes majeures nous donnent une liste de segments pertinents qui est utilisée pour concaténer les segments vidéo et construire le résumé final. Afin de maximiser le contenu visuel du résumé, nous avons accéléré linéairement les résumés. L'efficacité de cette organisation a été montrée grâce aux expérimentations : notre système est efficace en terme de reconnaissance des segments redondants. Cependant, il reste légèrement en dessous de la moyenne en ce qui concerne la sélection des segments.
Cette collaboration a conduit à de bon résultats, il reste cependant des améliorations à apporter. En particulier, définir plus précisément le notion de segment. Les résultats ont montré qu'il est important d'utiliser des unités de temps courtes, alors que notre segmentation commune propose des segments trop longs.

Sixième partie

Evaluation automatique

Evaluation automatique

Dans le chapitre précédent, nous avons proposé des méthodes de construction de résumés vidéo de rushes, domaine de recherche très récent dont le développement induira des outils utilisés régulièrement dans la phase de postproduction du montage vidéo ou dans le domaine de l'archivage vidéo. Cependant, l'une des grandes problématiques de ce domaine reste la notion d'évaluation de la qualité de ces résumés vidéo de rushes. La campagne d'évaluation TRECVID tente de donner une réponse à cette problématique en proposant une méthode d'évaluation manuelle permettant la comparaison annuelle des différents systèmes développés. Mais, il reste malgré tout, le problème de l'évaluation durant le processus de développement d'un système, étape incontournable à son optimisation. Dans ce chapitre, nous proposons une méthode originale d'automatisation de l'évaluation manuelle proposée par la campagne d'évaluation TRECVID.

1 Introduction

La création de résumés automatiques est un challenge actuel, mais le problème de l'évaluation limite l'évolution de cette recherche. Le problème du développement de méthodes d'évaluation reste délicat dans la définition même de l'évaluation qui implique de définir les qualités requises d'un système pour créer de bons résumés : une méthode d'évaluation doit prendre des décisions sur le contenu sémantique des séquences vidéo et leur importance. Ce facteur complique le développement de méthodes d'évaluation, et en particulier les méthodes d'évaluation entièrement automatiques.

La campagne d'évaluation TRECVID a choisi ses critères de qualité des résumés en prenant en compte les conclusions des travaux tels que [Jing 1998], [Taskiran 2006], [Christel 2006], [Ferman 2003], [Ekin 2003] ou encore [Truong 2007b] pour définir et identifier quel est le contenu le plus important dans une vidéo. La qualité des critères et le choix de ceux-ci peuvent être discutés, mais ce n'est pas le sujet de ce travail qui se focalise uniquement sur l'automatisation de la méthode d'évaluation manuelle proposée par TRECVID, et en particulier sur le critère concernant le pourcentage d'éléments d'histoire retrouvés dans un résumé.

Les travaux sur ce sujet sont donc rares, seuls deux laboratoires ont proposés des solutions. Dans [Hauptmann 2007], l'idée est simple, mais présente une bonne corrélation avec les résultats

manuels, une séquence intéressante est visible dans le résumé si une partie de celle-ci est dans le résumé et dure au moins une seconde. Nous avons en parallèle proposé une méthode qui utilisait un critère quasi identique, puisque nous proposions de dire que 25 images successives dans le résumé permettaient de voir un élément d'histoire pour nous l'idée était de prendre en compte la notion d'accélération. Cette méthode nous a conduit à des résultats préliminaires [Dumont 2007].

Le but de ce travail est donc de proposer une méthode d'automatisation du critère d'évaluation *IN* selon la campagne TRECVID. Nous commencerons par décrire les améliorations apportées à la vérité terrain, puis décrirons notre évaluateur automatique. Une étude complète de la qualité de notre évaluateur sera finalement proposée.

2 Automatisation de l'évaluation

2.1 Nouvelle vérité terrain

La vérité terrain fournie par TRECVID est une liste chronologique simple d'éléments d'histoire ("topic" en anglais) définis par des objets, évènements ou mouvement de caméra. Ceci n'est pas suffisant pour une évaluation automatique, nous avons donc ajouté des informations à cette liste provenant d'une annotation manuelle, la figure 2.1 montre un exemple de la nouvelle vérité terrain :

- Pour chaque élément d'histoire de la liste, nous avons délimité les séquences sur la vidéo initiale en terme de numéro d'image.
- Pour chaque élément d'histoire de la liste, nous avons ajouté la nature de celui-ci : comporte-t-il un événement ? comporte-t-il un mouvement de caméra ?

2.2 Evaluateur manuel

Chaque résumé soumis est jugé par trois juges humains différents, les évaluateurs. Pour effectuer l'évaluation, l'évaluateur a un résumé ainsi que la liste de 12 éléments d'histoire choisis aléatoirement parmi l'intégralité des éléments d'histoire d'une vidéo. Il visionne le résumé dans une fenêtre de taille 125mm x 102mm grâce au logiciel mplayer à une fréquence de 25 images par seconde en utilisant seulement les fonctions de lecture et pause. Pour chaque élément d'histoire qu'il voit, l'évaluateur coche cette séquence sur la liste. Finalement, en faisant la moyenne des éléments d'histoire visualisés, chaque évaluateur obtient une valeur de *IN*. La valeur attribuée à un résumé est la moyenne de ces 3 résultats.

2.3 Evaluateur automatique

Afin d'automatiser le processus, nous proposons de créer un évaluateur automatique qui pour chaque élément d'histoire prend une décision sur sa présence. La valeur de *IN* attribuée

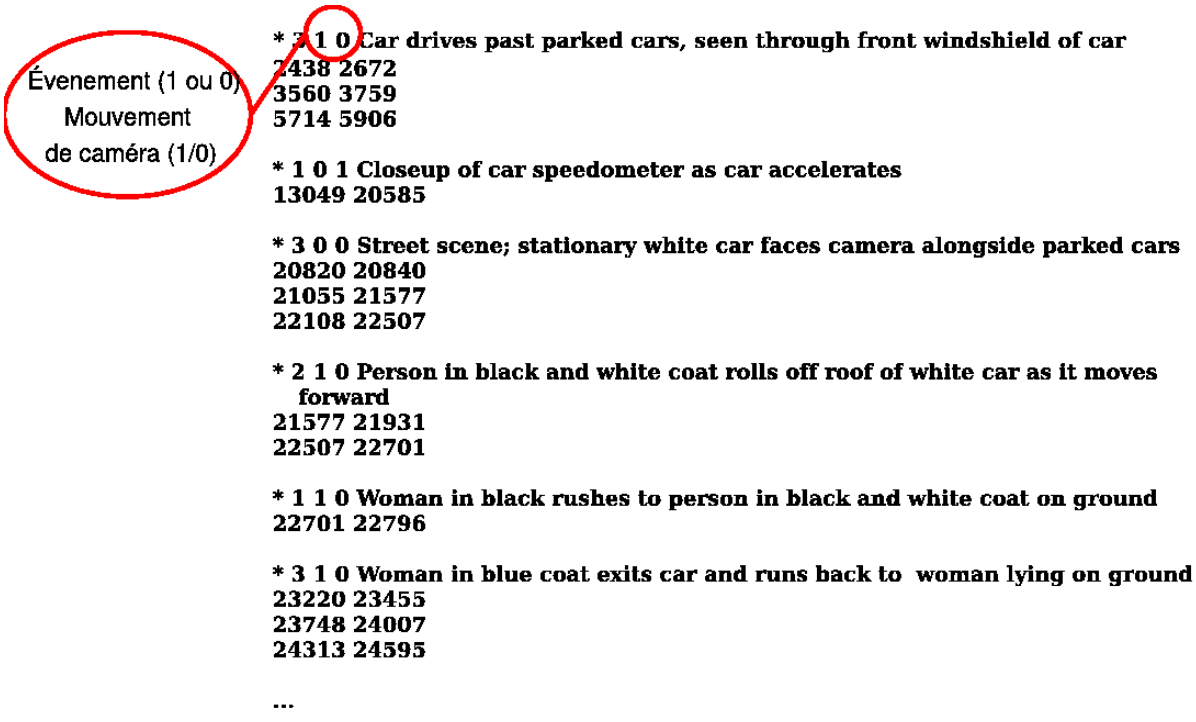


FIG. 2.1 – Nouvelle vérité terrain pour une vidéo

à un résumé correspond au pourcentage d'éléments d'histoire présents défini par notre évaluateur.

2.3.1 Modélisation du problème

Pour modéliser notre problème, nous définissons un élément d'histoire i comme un couple $(\mathbf{x}_i, y_i)$ où :

- $\mathbf{x}_i \in \mathcal{X}$ est un vecteur contenant la description d'un élément d'histoire i,
- $y_i \in \{\text{presence}, \text{absence}\}$ est le résultat de la décision d'un évaluateur dépendant du vecteur $\mathbf{x}_i$.

La décision de détecter un élément d'histoire est directement dépendante de la localisation de ces séquences dans le résumé, de leur durée, mais aussi de certains critères liés à la vidéo initiale. La décision de la présence d'un élément d'histoire durant 5 secondes dans le résumé peut être différente si dans la vidéo initiale celui-ci dure 10 minutes ou 10 secondes. De même, il est intéressant de prendre en compte les caractéristiques d'un élément d'histoire : y-a-t-il beaucoup d'activité, la séquence est-elle encombrée ? La décision peut aussi être influencée par la nature d'un élément d'histoire, y-a-t-il un mouvement de caméra ou un évènement ? Dans le modèle que nous proposons, nous avons donc choisi de décrire un élément d'histoire par :

x_0 : Y-a-t-il un mouvement de caméra ? : oui ou non
x_1 : Y-a-t-il un évènement ? : oui ou non
x_2 : Nombre de répétition d'un élément d'histoire dans la vidéo initiale
x_3 : Taille de la plus petite séquence parmi les répétitions dans la vidéo initiale
x_4 : Taille de la plus longue séquence parmi les répétitions dans la vidéo initiale
x_5 : Taille moyenne d'un élément d'histoire dans la vidéo initiale
x_6 : Activité moyenne d'un élément d'histoire dans la vidéo initiale

x_7 : Entropie moyenne d'un élément d'histoire dans la vidéo initiale
x_8 : Nombre de répétition d'un élément d'histoire dans le résumé
x_9 : Taille de la plus petite séquence parmi les répétitions dans le résumé
x_{10} : Taille de la plus longue séquence parmi les répétitions dans le résumé
x_{11} : Taille moyenne d'un élément d'histoire dans le résumé

L'ensemble de ces descriptifs peuvent être automatiquement calculés à partir de la vérité terrain et du résumé dont on désire évaluer la qualité. Cette modélisation permet à un évaluateur automatique de décider de l'absence ou de la présence y_i d'un élément d'histoire i dans un résumé à partir des données $\mathbf{x}_i$.

2.3.2 Entrainement d'un évaluateur automatique

Un évaluateur automatique doit définir une fonction *prediction* prédisant la présence 1 ou l'absence 0 d'un élément d'histoire. Une fois cette fonction définie, nous pouvons calculer automatiquement IN pour un résumé vidéo v par :

$$IN(v) = \frac{1}{N} \sum_{i=1}^{N} prediction(i) \tag{2.1}$$

où N est le nombre d'éléments d'histoire à retrouver.

Grâce à un ensemble d'entrainement constitué d'un ensemble de couples $(\mathbf{x}_i, y_i)$, nous pouvons utiliser une méthode d'apprentissage supervisée pour créer le modèle de prédiction. Puis appliquer ce modèle à un ensemble de test.

3 Expériences

3.1 Qualité de la prédiction

Le but de ces expériences est de prédire automatiquement la valeur IN d'un résumé grâce à la prédiction de la présence des éléments d'histoire dans le but de pouvoir comparer des systèmes. Idéalement, nous souhaitons avoir une forte corrélation entre l'évaluation automatique et l'évaluation manuelle tant au niveau du classement, que du calcul de IN et qu'au niveau de la prédiction de la présence d'un élément d'histoire.

Pour évaluer la corrélation, nous utilisons le coefficient de Bravais-Pearson r. Cet indice statistique exprime l'intensité et le sens (positif ou négatif) de la relation linéaire entre deux variables quantitatives. Il assume des valeurs se situant dans l'intervalle de -1 à $+1$. Une valeur égale à -1 ou à $+1$ indique l'existence d'une relation linéaire parfaite (fonctionnelle) entre les deux variables. En revanche, ce coefficient est nul ($r = 0$) lorsqu'il n'y a pas de relation linéaire entre les variables. L'intensité de la relation linéaire sera donc d'autant plus forte que la valeur

du coefficient est proche de +1 ou de - 1, et d'autant plus faible qu'elle est proche de 0. Par ailleurs, le coefficient est de signe positif si la relation est positive (directe, croissante) et de signe négatif si la relation est négative (inverse, décroissante).
Le coefficient r de Bravais-Pearson entre deux variables X et Y se calcule en appliquant la formule suivante :

$$r = \frac{\text{cov}(X,Y)}{\sqrt{(V(X)V(Y))}} \tag{3.1}$$

où $\text{cov}(X,Y)$ est la covariance entre X et Y, et $V(X)$, $V(Y)$ respectivement la variance de X, et la variance de Y.

La qualité de l'évaluation peut se mesurer à 3 niveaux différents :
- élément d'histoire : corrélation entre les prédictions de la présence d'un élément d'histoire manuelles et automatiques.
- IN : corrélation entre les prédictions du pourcentage d'éléments d'histoire présents manuels et automatiques.
- rang : corrélation entre les classements manuels et automatiques.

3.2 Base de données

Les expériences sont effectuées sur 8 vidéos de la base de données TRECVID 2008, et sur 10 systèmes de résumés vidéos différents. Les méthodes d'apprentissage supervisée sont listées dans le tableau 3.2 et sont utilisées grâce au software WEKA [URL 1]. Afin d'utiliser au maximum les données et que les ensembles de tests et d'entrainement soient indépendants, nous utilisons un système de validation croisée : le calcul de la valeur de IN d'un résumé v produit par un système s est basé sur la prédiction de la présence des 12 séquences intéressantes, le modèle pour un résumé est entrainé sur toutes les vidéos sauf v et tous les systèmes sauf s, soit 9 systèmes x 7 vidéos x 12 séquences intéressantes x 3 évaluateurs = 2268 couples $(\mathbf{x}_i, y_i)$.

3.3 Classifieur

Nous commençons les expériences par le choix du meilleur classifieur pour ce problème. Le graphique 3.1 montre une vision globale des résultats obtenus par les différents classifieurs pour les 3 niveaux d'évaluations : rang, IN, élément d'histoire.

Nous pouvons constater que le choix du classifieur influence beaucoup les résultats finaux pour ce problème. Nous allons donc nous consacrer sur le plus performant pour ce problème : l'utilisation des stumps est le meilleur système pour le classement des systèmes, pour la prédiction de la valeur IN, ainsi que au niveau d'un élément d'histoire.

Au niveau de la prédiction d'un élément d'histoire, la corrélation entre l'évaluation manuelle et l'évaluation automatique est de 0.535, elle est donc modérée. Ce qui correspond à 80% de prédiction en accord entre les deux méthodes. La figure 3.2 montre les résultats de l'évaluation automatique en fonction de l'évaluation manuelle au niveau de IN :la corrélation est forte (0.876). Le dernier niveau étudié est celui du classement dont la corrélation est forte (0.913), les résultats sont montrés par la figure 3.3.

AdaBoost	AdaBoost
Alternating Decision Tree [Freund 1999]	ADTree
Bayes Network learning	BayesNet
simple meta-Classifier Via clustering	CVClustering
single Conjunctive Rule	ConjunctiveRule
Decision Stump	DecisionStump
simple Decision Table majority [Kohavi 1995]	DecisionTable
HyperPipe classifier	HyperPipes
K-nearest neighbours [Aha 1991]	IBk
C4.5 Decision Tree [Quinlan 1993]	J48
repeated incremental pruning [Cohen 1995]	JRip
instance-based classifier [Cleary 1995]	KStar
Lazy Bayesian Rules [Zheng 2000]	LBR
Logistic Model Trees	LMT
multinomial Logistic regression [le Cessie 1992]	Logistic
Multilayer Perceptron	MultilayerPerceptron
Naive Bayes [John 1995]	NaiveBayes
NT-AdaBoost [Dumont 2005]	NT-AdaBoost
Simple Naive Bayes [Duda 1973]	NaiveBayesSimple
decision Tree with Naive Bayes classifiers at the leaves [Kohavi 1996]	NBTree
Nearest-neighbor-like algorithm using non-nested generalized exemplars [Roy 2002]	NNge
uses the minimum-error attribute for prediction [Holte 1993]	OneR
PART decision list [Frank 1998]	PART
Forest of Random trees [Breiman 2001]	RandomForest
a tree that considers K randomly chosen attributes at each node	RandomTree
Normalized Gaussian Radial Basis function Network	RBFNetwork
Fast Decision Tree	REPTree
RIpple-DOwn Rule learner. [Gaines 1995]	Ridor
Logistic Model Trees	SimpleLogistic
Sequential Minimal Optimization	SMO
Voting Feature Intervals	VFI
Voted Perceptron	VotedPerceptron
0-R classifier	ZeroR

TAB. 3.1 – Méthodes de classification supervisée

3.3.1 Les stumps

Les stumps sont des arbres de décision à un niveau, c'est-à-dire qu'un seul test permet de prédire la présence d'un élément d'histoire. Pour chaque couple (vidéo, système), un modèle de prédiction de la présence d'un élément d'histoire a été créé. La figure 3.4 montre les différents stumps crées, la valeur au dessus du stump représente le proportion de fois où le stump a été créé. L'ensemble des modèles crées sont basés sur une caractéristique extraite du résumé. Dans la majorité des modèles construits, l'évaluateur considère qu'un élément d'histoire est présent dans le résumé si la plus longue séquence du résumé le représentant dure au moins 2 images.

3.4 Evaluation automatique contre manuelle

Nous avons vu que la corrélation, au niveau du classement et du calcul de IN, entre notre évaluateur automatique et l'évaluation manuelle était forte, mais que l'accord de notre évaluateur automatique était modéré par rapport à l'évaluation manuelle au niveau de la prédiction de la présence d'un élément d'histoire. Les évaluations ont été effectuées par

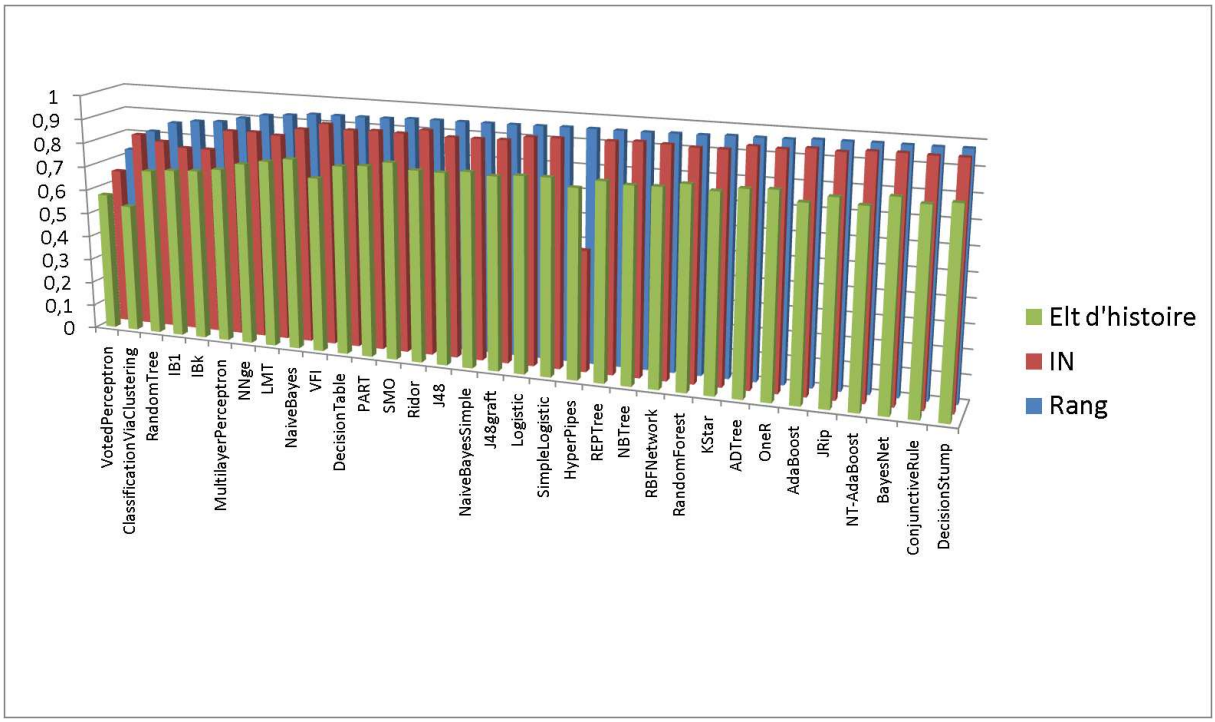


FIG. 3.1 – Vision global des résultats obtenus par les différents classifieurs

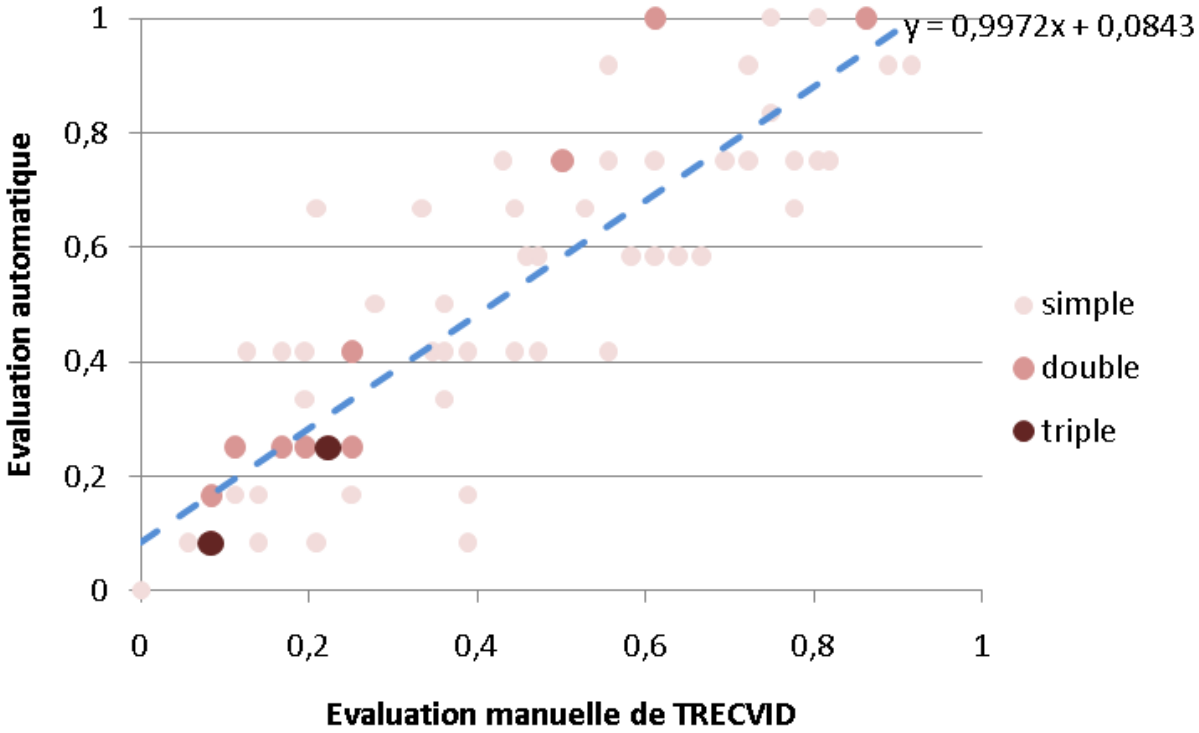


FIG. 3.2 – Evaluation automatique en fonction de l'évaluation manuelle

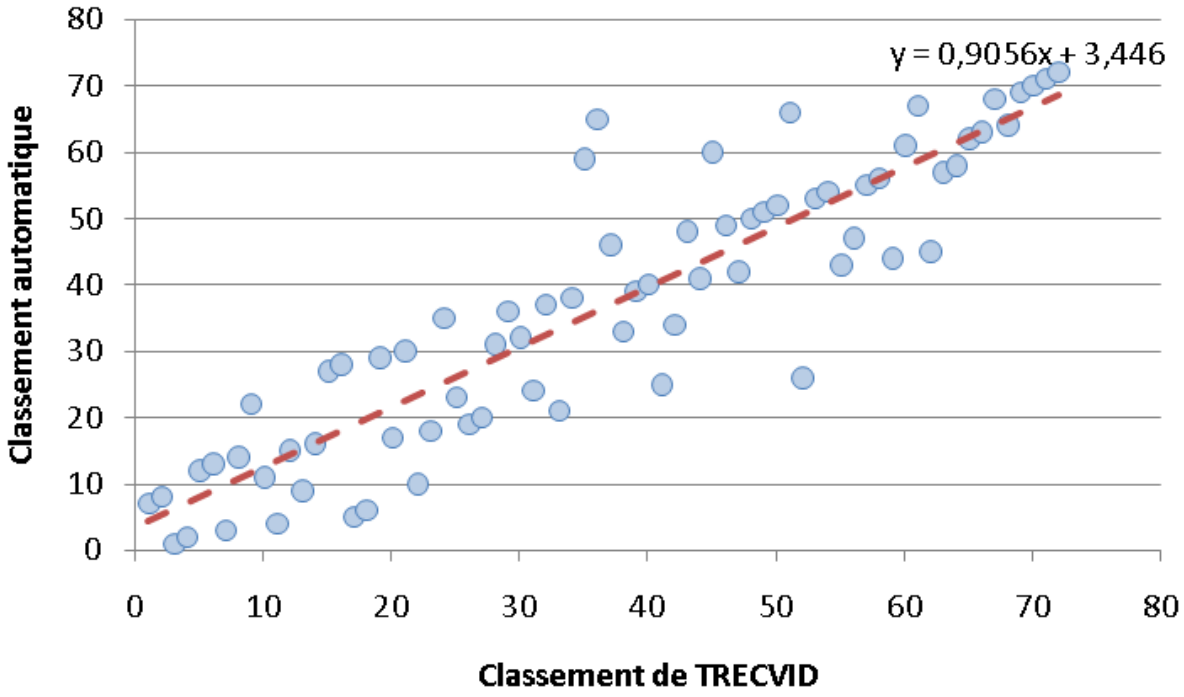


FIG. 3.3 – Evaluation automatique en fonction de l'évaluation manuelle

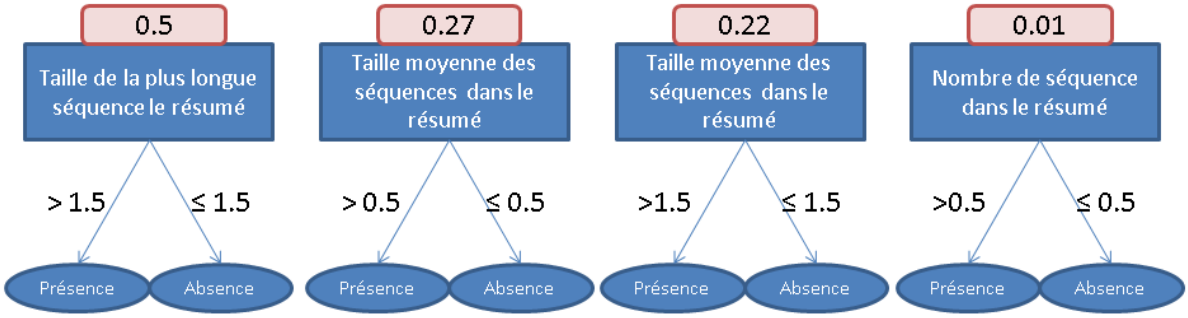


FIG. 3.4 – Stump pour la prédiction de la présence d'une séquence intéressante

dix évaluateurs manuels dont 3 par résumé. Nous allons donc étudier la corrélation entre les évaluateurs manuels. Le tableau 3.2 montre les résultats obtenus. Pour évaluer la qualité des évaluateurs humains, nous avons, pour chaque évaluateur, comparé ses prédictions avec toutes les prédictions effectuées par d'autres évaluateurs pour un même élément d'histoire et ceci sur toutes les évaluations de TRECVID 2008, soit 1710 résumés évalués par 3 évaluateurs.

Nous pouvons donc voir que les évaluateurs automatiques sont, au niveau du classement et en terme de corrélation, comparables aux évaluateurs manuels : la corrélation varie de 0.81 à 0.92 pour les évaluateurs alors que l'évaluateur manuel que nous proposons a une corrélation de 0.91. De même au niveau du calcul de IN : la corrélation varie de 0.78 à 0.90 pour les évaluateurs manuels alors que l'évaluateur que nous proposons a une corrélation de 0.88. La corrélation au niveau de la prédiction d'un élément d'histoire, modérée pour notre évaluateur

Evaluateur	Élément d'histoire	IN	Rang
Evaluateur 1	0.755961	0.878687	0.914713
Evaluateur 2	0.789278	0.875702	0.917511
Evaluateur 3	0.770808	0.870860	0.911205
Evaluateur 4	0.775011	0.860053	0.895711
Evaluateur 5	0.790169	0.865818	0.897900
Evaluateur 6	0.750509	0.805306	0.833046
Evaluateur 7	0.715957	0.781069	0.824572
Evaluateur 8	0.702580	0.804130	0.811149
Evaluateur 9	0.726755	0.855810	0.892882
Evaluateur 10	0.790546	0.901866	0.926379
DecisionStump	0.535261	0.875906	0.913306

TAB. 3.2 – Evaluation des évaluations manuelles

manuel, n'est pas aussi forte pour les évaluateurs manuelles que la prédiction au niveau d'un élément d'histoire, ou au niveau du classement.

3.5 Conclusion

Dans ce chapitre, nous avons présenté une méthode pour automatiser une évaluation. Notre approche s'est focalisée sur un des critères défini par TRECVID pour l'évaluation des systèmes de résumés automatiques de rushes. Nous avons commencé par présenter une méthode de modélisation d'un tel problème, puis nous avons, par des expériences sur différentes méthodes de classification supervisée, choisis deux méthodes ayant de bonnes performances. Nous avons montré que notre méthode automatique est fiable dans un but de classement ou de calcul de IN.
La méthode d'évaluation peut être améliorée par un ensemble d'entrainement plus grand, plus de systèmes et plus de vidéo annotées. Il est possible aussi d'accroître les résultats grâce à une augmentation des descripteurs dans la modélisation du problème, ou encore une méthode de classification plus adaptée à ce problème. Il serait intéressant aussi de pouvoir tester le modèle à plus grande échelle. Une validation correcte de cette méthode d'évaluation permettrait d'accélérer le développement de systèmes performants.

Conclusion générale

Tout au long de ce manuscrit, nous avons présenté le travail développé durant cette thèse. Le fil directeur est l'exploitation des rushes vidéo proposée par la campagne d'évaluation internationale TRECVID.

Un premier aspect étudié fut celui des spécificités des rushes vidéo. Les rushes d'un film étant constitués des documents originaux produits au tournage d'un film, ils sont donc constitués de beaucoup de séquences outils telles que les mires ou les claps, ou encore des séquences dites poubelles telles que des plans de couleurs uniformes. Nous avons proposé une méthode rapide et efficace quant à la détection des séquences poubelles. La détection des séquences outils est composée de deux méthodes. L'une proposant de détecter les séquences de mires ; cette méthode est rapide et très efficace. Contrairement à la détection des séquences de claps qui reste le point faible. Il serait intéressant d'étudier plus précisément la détection des claps ainsi que l'extraction des informations qu'ils contiennent, puisque ils recèlent d'informations très riches aussi bien au niveau de la structure de la vidéo que sur la nature de celle-ci.
De plus, la majorité des séquences provenant de rushes sont temporellement très redondantes, c'est-à-dire, que certains plans peuvent durer plusieurs minutes où visuellement les changements restent très faibles. Afin de limiter de telles redondances, nous proposons une accélération dynamique. Cette méthode est visuellement satisfaisante, mais sa qualité reste difficile à définir de manière absolue. L'évaluation de la qualité d'une accélération reste un problème ouvert qui serait intéressant de résoudre afin de faciliter l'amélioration des méthodes proposées. Notre méthode propose une accélération moyenne qui reste assujettie à une valeur préfixée, alors qu'il serait plus naturel que l'accélération moyenne dépende de la nature de la vidéo.

Afin de faciliter l'utilisation d'une multitude de vidéos de rushes, nous avons proposé une méthode de recherche de plans vidéo via un dictionnaire visuel. L'idée principale qui a guidé le développement de notre système fut la bonne qualité des méthodes développées dans le domaine textuel. Nous avons donc proposé une méthode se basant sur le paradigme des recherches de documents textuels : un utilisateur compose des requêtes de mots visuels qui permettent au système de proposer à celui-ci la liste ordonnée par pertinence des plans vidéo répondant à cette requête. La première amélioration est de lier les mots visuels à des mots textuels afin de permettre à un utilisateur de composer une requête textuelle et au système de retourner une liste de plans répondant aux requêtes sémantiques de l'utilisateur.

L'exploitation des rushes vidéo s'est rapidement orientée vers la notion de résumé vidéo. Nous avons proposé deux outils pour l'aide à la création de résumés vidéo. Le premier est une mesure du contenu visuel d'une séquence vidéo. Le deuxième est une structuration de la vidéo permettant de supprimer la redondance dans une vidéo en se basant sur l'alignement de séquences vidéo. Ces méthodes ont été testées grâce à la campagne d'évaluation TRECVID.

Les résultats furent satisfaisants mais dépendent des artifices ajoutés lors de la mise en place du système complet. Grâce à des évaluations intermédiaires, nous avons pu montrer la force de ces outils. L'un des aspects intéressant à développer mais qui reste peu étudié est la sélection sémantique des prises d'un groupe répétitif.

Afin d'utiliser les compétences de différents laboratoires, nous avons proposés, en collaboration, une architecture permettant la création de résumés vidéo en fusionnant des méthodes issus de divers laboratoires. Cette architecture a été testée dans le cadre de TRECVID et a montré ses bonnes qualités. Les résultats obtenus nous encouragent à améliorer le système, et à le tester à une plus grande échelle.

Le développement des méthodes de construction de résumé vidéo reste limité à cause du manque de méthode d'évaluation. Nous avons décrit la méthode d'évaluation proposée par TRECVID qui tend à devenir la méthode d'évaluation des résumés vidéo. Cependant, son grand handicap reste sa dimension humaine : cette méthode est manuelle. Nous avons proposé une méthode permettant d'automatiser cette évaluation, les résultats montrent que nos évaluations automatiques sont très fortement corrélées à celles de TRECVID effectuées manuellement. Il serait donc intéressant de pouvoir créer le modèle sur l'ensemble des données de TRECVID, puis de l'étendre à l'ensemble des critères d'évaluations subjectifs.

Bibliographie

[Agnihotri 2004] L. Agnihotri, N. Dimitrova and J. Kender. *Design and evaluation of a music video summarization system.* In International Conference on Multimedia & Expo (ICME'04), pages 1943–1946 Vol.3, Taipei, Taiwan, June 27-30 2004.

[Aha 1991] D. Aha and D. Kibler. *Instance-based learning algorithms.* Machine Learning, vol. 6, pages 37–66, 1991.

[Aizawa 2001] K Aizawa, K-I Ishijima and M Shiina. *Summarizing Wearable Video.* In IEEE International Conference on Image Processing (ICIP'01), Thessaloniki, Greece, oct 7-10 2001.

[Aizawa 2004] Kiyoharu Aizawa, Datchakorn Tancharoen, Shinya Kawasaki and Toshihiko Yamasaki. *Efficient retrieval of life log based on context and content.* In 1st ACM workshop on Continuous archival and retrieval of personal experiences (CARPE'04), pages 22–31, New York, New York, October 15th 2004.

[Allen 2006] B. P. Allen, V. A. Petrushin, G. Wei and D. Roqueiro. *Semantic Web Techniques for Searching and Navigating Video Shots in BBC Rushes.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Ariki 2003] Y. Ariki, M. Kumano and K. Tsukada. *Highlight Scene Extraction in Real Time from Baseball Live Video.* In 5th ACM SIGMM international workshop on Multimedia information retrieval (MIR'03), pages 209–214, Berkeley, California, USA, November 7 2003.

[Babaguchi 2000] Noboru Babaguchi. *Towards Abstracting Sports Video by Highlights.* In IEEE International Conference on Multimedia and Expo (ICME'00), pages 1519–1522, New York, August 2000.

[Babaguchi 2001] N. Babaguchi, Y. Kawai and T. Kitahashi. *Generation of personalized abstract of sports video.* Multimedia and Expo, 2001. ICME 2001. IEEE International Conference on, pages 619–622, Aug. 2001.

[Bailer 2006] W. Bailer, C. Schober and G. Thallinger. *Video Content Browsing Based on Iterative Feature Clustering for Rushes Exploitation.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Bailer 2009] Werner Bailer, Felix Lee and Georg Thallinger. *A Distance Measure for Repeated Takes of One Scene.* The Visual Computer, vol. 25, pages 53–68, 2009.

[Barron 1992] J.L. Barron, D.J. Fleet, S.S. Beauchemin and T.A. Burkitt. *Performance of optical flow techniques.* Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society Conference on, pages 236–242, Jun 1992.

[Bellman 1957] R. E. Bellman. *Dynamic programming.* In Princeton University Press, Princeton, NJ., 1957.

[Beran 2007] Vítezslav Beran, Michal Hradis, Adam Herout, Stanislav Sumec, Igor Potúcek, Pavel Zemcdík, Josef Mlích, Ales Láník and Petr Chmelar. *Video Summarization at Brno University of Technology.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Beran 2008] Vítezslav Beran, Michal Hradis, Pavel Zemcík, Adam Herout and Ivo Reznícek. *Video summarization at Brno university of technology.* In ACM International Conference on Multimedia, pages 31–34, Vancouver, BC, Canada, october 2008.

[Bhaskar 2001] Raghav Bhaskar, Damien Paulin, Dinesh Kumar and Georges M. Quenot. *Recovering Camera Motion and Mobile Objects in Video Documents.* International Workshop on Content-Based Multimedia Indexing, 19-21 September 2001, University of Brescia, Italy, 2001.

[Billerbeck 2004] B. Billerbeck, A. Cannane, A. Chatteraj, N. Lester, W. Webber, H. E. Williams, J. Yiannis and J. Zobel. *RMIT University at TREC 2004.* In TREC Video Retrieval Evaluation Forum (TRECVID'04), Gaithersburg, Maryland, November 2004.

[Breiman 2001] Leo Breiman. *Random Forests.* Machine Learning, vol. 45, no. 1, pages 5–32, 2001.

[Calic 2002] Janko Calic and Ebroul Izquierdo. *Efficient Key-Frame Extraction and Video Analysis.* In Information Technology : Coding and Computing (ITCC'02), pages 28–33, Las Vegas, USA, 8-10 April 2002.

[Calic 2006] J. Calic, P. Kramer, U. Naci, S. Vrochidis, S. Aksoy, Q. Zhangk, J. Benois-Pineau adn A. Saracoglu, C. Doulaverakis, R. Jarina, N. Campbell, V. Mezaris, I. Kompatsiaris, E. Spyrou, G. Koumoulos, Y. Avrithis, A. Dalkilic, A. Alatan, A. Hanjalic and E. Izquierdo. *COST292 experimental framework for TRECVID 2006.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Calic 2007] Janko Calic, David P. Gibson and Neill W. Campbell. *Efficient Layout of Comic-Like Video Summaries*, Jul. 2007.

[Campbell 2006] M. Campbell, S. Ebadollahi, D. Joshi, M. Naphade, A. Natsev, J. Seidl, J. R. Smith, K. Scheinberg, J. Tesic, L. Xie and A. Haubold. *IBM T.J. Watson Research Center.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Canny 1986] J. Canny. *A computational approach to edge detection.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 8, no. 6, pages 679–698, November 1986.

[Cao 2006] J. Cao, Y. Lan, J. Li, Q. Li, X. Li, F. Lin, X. Liu, L. Luo, W. Peng, D. Wang, H. Wang, Z.Wang, Z. Xiang, J. Yuan, W. Zheng, B. Zhang, J. Zhang, L. Zhang and X. Zhang. *Intelligent Multimedia Group of Tsinghua University at TRECVID 2006.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Carson 1999] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein and Jitendra Malik. *Blobworld : a system for region-based image indexing and retrieval.* In Third International Conference on Visual Information Systems (VISUAL'99), pages 509–516, Shangai, China, June 1999.

[Chang 1999] Hyun Sung Chang, Sanghoon Sull and Sang Uk Lee. *Efficient video indexing scheme for content-based retrieval.* IEEE Transactions on Circuits and Systems for Video Technology, vol. 9, no. 8, pages 1269–1279, December 1999.

[Chang 2001] Chih-chung Chang and Chih-jen Lin. *LIBSVM : a library for support vector machines*, 2001.

[Chang 2002] Peng Chang, Mei Han and Yihong Gong. *Extract Highlights From Baseball Game Video With Hidden Markov Models.* In IEEE International Conference on Image Processing (ICIP'02), Rochester, NY, sep 22-25 2002.

[Chasanis 2008] Vasileios Chasanis, Aristidis Likas and Nikolas P. Galatsanos. *Video rushes summarization using spectral clustering and sequence alignment.* In ACM International Conference on Multimedia, pages 75–79, Vancouver, BC, Canada, october 2008.

[Chen 2004] Shu-Ching Chen, Mei-Ling Shyu, Min Chen and Chengcui Zhang. *A Decision Tree-based Multimodal Data Mining Framework for Soccer Goal Detection.* In International Conference on Multimedia & Expo (ICME'04), Taipei, Taiwan, June 27-30 2004.

[Christel 2006] M.G. Christel. *Evaluation and user studies with respect to video summarization and browsing.* In Multimedia Content Analysis, Management and Retrieval, Proceedings of SPIE, vol. 6073, pp. 196-210., 2006.

[Cleary 1995] John G. Cleary and Leonard E. Trigg. *K* : An Instance-based Learner Using an Entropic Distance Measure.* In The Twelfth International Conference on Machine Learning (ICML'95), pages 108–114, Tahoe City, California, USA, June 1995.

[Cohen 1995] William W. Cohen. *Fast Effective Rule Induction.* In The Twelfth International Conference on Machine Learning (ICML'95), pages 115–123, Tahoe City, California, USA, June 1995.

[Cooper 2002] Matthew Cooper and Jonathan Foote. *Summarizing Video using Non-Negative Similarity Matrix Factorization.* In IEEE Workshop on Multimedia Signal Processing (MMSP'02), pages 25–28, International Workshop on Multimedia Signal Processing (MMSP'02), St. Thomas, Virgin Islands, USA 2002.

[Cooper 2005] M. Cooper and J. Foote. *Discriminative Techniques for Keyframe Selection.* In International Conference on Multimedia & Expo (ICME'05), pages 502–505, Amsterdam, The Netherlands, July 2005.

[Damnjanovic 2007] U. Damnjanovic, T. Piatrik, D. Djordjevic and E. Izquierdo. *Video Summarisation for Surveillance and News Domain.* In The 2nd international conference on Semantics And digital Media Technologies (SAMT'07), Genova, Italy, December 2007.

[David Gibson 2002] Neill Campbell David Gibson and Barry Thomas. *Visual Abstraction of Wildlife Footage using Gaussian Mixture Models.* In 15th International Conference on Vision Interface (VI'02), Calgary, Canada, May 27-29 2002.

[de Silva 2005] Gamhewage C. de Silva, Toshihiko Yamasaki and Kiyoharu Aizawa. *Evaluation of video summarization for a large number of cameras in ubiquitous home.* In ACM International Conference on Multimedia (ACMMM'05), Singapore, November 2005.

[Del Bimbo 1995] A. Del Bimbo, P. Nesi and J.L.C. Sanz. *Analysis of optical flow constraints.* Image Processing, IEEE Transactions on, vol. 4, no. 4, pages 460–469, Apr 1995.

[DeMenthon 1998] Daniel DeMenthon, Vikrant Kobla and David Doerman. *Video Summarization by Curve Simplification.* In ACM International Conference on Multimedia (ACMMM'98), pages 211–218, Bristol, UK, September 1998.

[Diklic 1998] D. Diklic, D. Petkovic and R. Danielson. *Automatic extraction of representative keyframes based on scene content.* Signals, Systems & Computers, 1998. Conference Record of the Thirty-Second Asilomar Conference on, vol. 1, pages 877–881 vol.1, Nov 1998.

[Ding 1997] W.and G. Marchionini Ding and T. Tse. *Previewing video data : Browsing key frames at high rates using a video slide show interface.* In International Symposium on Research, Development and Practice in Digital Libraries (ISDL'97), Tsukuba, Japan, November 1997.

[Dirfaux 2000] F. Dirfaux. *Key frame selection to represent a video.* Image Processing, 2000. Proceedings. 2000 International Conference on, vol. 2, pages 275–278 vol.2, 2000.

[Divakaran 2002] A. Divakaran, R. Radhakrishnan and K. A. Peker. *Motion activity-based extraction of key-frames from video shots.* In IEEE International Conference on Image Processing (ICIP'02), pages 932–935, Rochester, NY, sep 22-25 2002.

[Doulamis 1998] Nikolas D. Doulamis, Anastasios D. Doulamis, Yannis S. Avrithis and Stefanos D. Kollias. *Video Content Representation using Optimal Extraction of Frames and Scenes.* In IEEE International Conference on Image Processing (ICIP'98), pages 875–879, Chicago, Illinois, oct 1998.

[Doulamis 2000a] A. Doulamis, N. Doulamis, Y. Avrithis and S. Kollias. *A Fuzzy Video Content Representation for Video Summarization and Content-Based Retrieval.* Signal Processing, vol. 80, pages 1049–1067, June 2000.

[Doulamis 2000b] Anastasios D. Doulamis, Nikolaos Doulamis, Georgios Akrivas and Stefanos Kollias. *Non-sequential video content representation using temporal variation of feature vectors.* IEEE Transactions on Consumer Electronics, vol. 46, page 2000, 2000.

[Duda 1973] Richard Duda and Peter Hart. Pattern classification and scene analysis. Wiley, New York, 1973.

[Dufaux 2000] F. Dufaux. *Key frame selection to represent a video.* In IEEE International Conference on Image Processing (ICIP'00), volume 2, pages 275 – 278, Vancouver, BC, Canada, septembre 2000.

[Dumont 2005] Emilie Dumont. *Noise Tolerant AdaBoost.* Technical report, Université de Provence, Marseille, France, 2005.

[Dumont 2007] Emilie Dumont and Bernard Mérialdo. *Split-screen dynamically accelerated video summaries.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Durik 2001] M. Durik and J. Benois-Pineau. *Robust motion characterisation for video indexing based on MPEG-2 optical flow.* Proceedings of the International Workshop on Content Based Multi-media Indexing (CBMI '01), pp. 57-64, Brescia, Italy, September, 2001.

[Ekin 2003] A. Ekin and R. Tekalp A.M.and Mehrotra. *Automatic soccer video analysis and summarization.* Image Processing, IEEE Transactions on, vol. 12, no. 7, pages 796–807, 2003.

[Ellouze 2008] Mehdi Ellouze, Hichem Karray and Adel M. Alimi. *Regim, research group on intelligent machines, tunisia, at TRECVID 2008, BBC rushes summarization.* In ACM International Conference on Multimedia, pages 105–108, Vancouver, BC, Canada, october 2008.

[Ewerth 2006] R. Ewerth, M. Mühling, T. Stadelmann, B. Freisleben, University of Siegen E. Qeli, B. Agel, D. Seiler and B. Freisleben. *University of Marburg at TRECVID 2006 : Shot Boundary Detection and Rushes Task Results.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Fauqueur 2003] Julien Fauqueur and Nozha Boujemaa. *New image retrieval paradigm : logical composition of region categories.* In IEEE International Conference on Image Processing (ICIP'03), pages 601–604, Barcelona, Spain, sep 14-17 2003.

[Fauvet 2004] B. Fauvet, P. Bouthemy, P. Gros and F. Spindler. *A geometrical key-frame selection method exploiting dominant motion estimation in video.* In International Conference on Image and Video Retrieval (CIVR'04), Dublin, Ireland, July 21-23 2004.

[Felzenszwalb 2004] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. *Efficient Graph-Based Image Segmentation.* In International Journal of Computer Vision, 59(2), September 2004.

[Ferman 1997] A.M. Ferman and A.M. Tekalp. *Multiscale Content Extraction and Representation for Video Indexing.* In SPIE Multimedia storage and archiving systems II, Dallas, TX, USA, Nov 1997.

[Ferman 2003] A.M. Ferman A.M.; Tekalp. *Two-stage hierarchical video summary extraction to match low-level user browsing preferences.* Multimedia, IEEE Transactions on, vol. 5, no. 2, pages 244–256, June 2003.

[Fox 1994] E.A. Fox and J.A. Shaw. *Combination of Multiple Searches.* 1994.

[Frank 1998] Eibe Frank and Ian H. Witten. *Generating Accurate Rule Sets Without Global Optimization.* In The Fifteenth International Conference on Machine Learning (ICML'98), pages 144–151, Madison, Wisconsin USA, July 1998.

[Freund 1999] Y. Freund and L. Mason. *The alternating decision tree learning algorithm.* In The Sixteenth International Conference on Machine Learning (ICML'99), pages 124–133, Bled, Slovenia, June 1999.

[Gaines 1995] Brian R. Gaines and Paul Compton. *Induction of Ripple-Down Rules Applied to Modeling Large Databases.* J. Intell. Inf. Syst., vol. 5, no. 3, pages 211–228, 1995.

[Gibbon 2006] D.C. Gibbon, Zhu Liu and B. Shahraray. *The MIRACLE video search engine.* In IEEE Consumer Communications and Networking Conference (CCNC'06), volume 1, pages 277–281, Las Vegas, Nevada, USA, January 2006.

[Girgensohn 1999] Andreas Girgensohn and John Boreczky. *Time-Constrained Keyframe Selection Technique.* In IEEE International Conference on Multimedia Computing and Systems, volume 1, pages 756–761, Florence, Italy, 7-11 June 1999.

[Girgensohn 2001] Andreas Girgensohn, John Boreczky and Lynn Wilcox. *Keyframe-Based User Interfaces for Digital Video.* Computer, vol. 34, no. 9, pages 61–67, 2001.

[Girgensohn 2003] Andreas Girgensohn. *A Fast Layout Algorithm for Visual Video Summaries.* In IEEE International Conference on Multimedia and Expo (ICME'03), volume 2, pages 77–80, Baltimore, NY, July 6-9 2003.

[Gong 1996] Yihong Gong, H.C. Chua and X.Y. Guo. *Image Indexing and Retrieval Based on Color Histograms.* International Journal of Multimedia Tools and Applications, vol. 2, 1996.

[Gong 2003] Yihong Gong and Xin Liu. *Video summarization and retrieval using singular value decomposition.* Multimedia Systems, vol. 9, no. 2, pages 157–168, 2003.

[Gorisse 2008] David Gorisse, Frédéric Precioso, Sylvie Philipp-Foliguet and Matthieu Cord. *Summarization scheme based on near-duplicate analysis.* In ACM International Conference on Multimedia, pages 50–54, Vancouver, BC, Canada, october 2008.

[Haller 2007] Martin Haller, Andreas Krutz and Thomas Sikora. *A Generic Approach for Motion-based Video Parsing.* In 15th European Signal Processing Conference (EUSIPCO'07), Poznan, Poland, September 2007.

[Hammoud 2000] Riad Hammoud and Roger Mohr. *A probabilistic framework of selecting effective key frames for video browsing and indexing.* In International workshop on Real-Time Image Sequence Analysis, Oulu, Finland, August 31 - September 1 2000.

[Harris 1988] C. Harris and M Stephens. *A combined corner and edge detector.* Alvey Vision Conference pp. 147-151, 1988.

[Hauptmann 2007] Alexander G. Hauptmann, Michael G. Christel, Wei-Hao Lin, Bryan Maher, Jun Yang, Robert V. Baron and Guang Xiang. *Clever clustering vs. simple speed-up for summarizing rushes.* In ACM International Conference on Multimedia, pages 20–24, Augsburg, Germany, September 2007.

[He 1999] Liwei He, Elizabeth Sanocki, Anoop Gupta and Jonathan Grudin. *Auto-summarization of audio-video presentations.* In ACM International Conference on Multimedia (ACMMM'99), pages 489–498, Orlando, Florida, Nov 1999.

[Heng 1999] W. J. Heng and K. N. Ngan. *The implementation of object-based shot boundary detection using edge tracing and tracking.* In IEEE International Symposium on Circuits and Systems (ISCAS'99), pages 439–442, Orlando, Florida, June 1999.

[Heng 2003] K.N Heng W.J.and Ngan. *High accuracy flashlight scene determination for shot boundary detection*, 2003.

[Holte 1993] R.C. Holte. *Very simple classification rules perform well on most commonly used datasets.* Machine Learning, vol. 11, pages 63–91, 1993.

[Horn 1980] Berthold K.P. Horn and Brian G. Schunck. *Determining Optical Flow.* Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1980.

[Huang 2005] Huang-Chia Shih ; Chung-Lin Huang. *MSN : statistical understanding of broadcasted baseball video using multi-level semantic network.* IEEE Transactions on Image Processing, vol. 51, pages 449–459, 2005.

[Ionescu 2008] Bogdan Ionescu, Didier Coquin, Patrick Lambert and Vasile Buzuloiu. *Fuzzy color-based approach for understanding animated movies content in the indexing task.* J. Image Video Process., vol. 8, no. 2, pages 1–17, 2008.

[Jing 1998] H. Jing, R. Barzilay, K. McKeown and M. Elhadad. *Summarization evaluation methods experiments and analysis*, 1998.

[John 1995] George H. John and Pat Langley. *Estimating Continuous Distributions in Bayesian Classifiers.* In 11th Conference on Uncertainty in Artificielle Intelligent (UAI'95), pages 338–345, Quebec, Canada, August 1995.

[Joly 2007] P. Joly, J. Benois-Pineau, E. Kijak and G. Quenot. *The Argos Campaign : Evaluation of Video Analysis Tools.* Content-Based Multimedia Indexing, 2007. CBMI '07. International Workshop on, pages 130–137, 2007.

[Kang 1999] Eung Kwan Kang, Sung Joo Kim and Joon Soo Choi. *Video Retrieval Based on Scene Change Detection in Compressed Domain.* IEEE Transactions on Consumer Electronics, vol. 45, no. 3, pages 932–936, 1999.

[Kang 2005] Hong-Wen Kang and Xian-Sheng Hua. *To Learn Representativeness of Video Frames.* In ACM International Conference on Multimedia (ACMMM'05), Singapore, November 2005.

[Kim 2002] Changick Kim and Jenq-Neng Hwang. *Object-Based Video Abstraction for Video Surveillance Systems.* IEEE CSVT, vol. 12, no. 12, pages 1128–1138, 2002.

[Kohavi 1995] Ron Kohavi. *The Power of Decision Tables.* In 8th European Conference on Machine Learning, Heraclion (Ecml'95), pages 174–189, Heraclion, Crete, Greece, April 1995.

[Kohavi 1996] Ron Kohavi. *Scaling Up the Accuracy of Naive-Bayes Classifiers : A Decision-Tree Hybrid.* In The Second International Conference on Knowledge Discovery and Data Mining (KDD'96), pages 202–207, Portland, Oregon, USA, August 1996.

[Komlodi 1998] Anita Komlodi and Gary Marchionini. *Key frame preview techniques for video browsing.* In 3rd ACM International Conference on Digital Libraries (DL'98), pages 118–125, Pittsburgh, PA, USA, June 23-26 1998.

[Krämer 2007] P. Krämer, O. Hadar, J. Benois-Pineau and J. P. Domenger. *Super-resolution mosaicing from MPEG compressed video.* Image Commun., vol. 22, no. 10, pages 845–865, 2007.

[Laganière 2008] Robert Laganière, Raphael Bacco, Arnaud Hocevar, Patrick Lambert, Grégory Païs and Bogdan Ionescu. *Video summarization from spatio-temporal features.* In ACM International Conference on Multimedia, pages 144–148, Vancouver, BC, Canada, october 2008.

[le Cessie 1992] S. le Cessie and J.C. van Houwelingen. *Ridge Estimators in Logistic Regression.* Applied Statistics, vol. 41, no. 1, pages 191–201, 1992.

[Le Gall 1991] Didier Le Gall. *MPEG : a video compression standard for multimedia applications.* Commun. ACM, vol. 34, no. 4, pages 46–58, April 1991.

[Lee 2003] Hun-Cheol Lee and Seong-Dae Kim. *Iterative Key Frame Selection in the Rate-Constraint Environment.* Signal Processing : Image Communication, no. 18, pages 1–15, 2003.

[Li 2001] Ying Li, Tong Zhang and Daniel Tretter. *An overview of video abstraction techniques.* Technical report, Report HPL-2001-191, HP Laboratories, 2001.

[Lienhart 1997] Rainer Lienhart. *Dynamic Video Summarization of Home Video.* In SPIE Multimedia storage and archiving systems II, pages pp. 378–389, Dallas, TX, USA, Nov 1997.

[Lienhart 2001] Rainer Lienhart. *Reliable Dissolve Detection*, 2001.

[Lim 1999] Joo-Hwee Lim. *Categorizing Visual Contents by Matching Visual "Keywords".* In Third International Conference on Visual Information Systems (VISUAL'99), pages 367–374, Shangai, China, June 1999.

[Lin 2005] Ching-Yung Lin and Belle L. Tseng. *Optimizing user expectations for video semantic filtering and abstraction.* In IEEE International Symposium on Circuits and Systems (ISCAS'05), pages 1250–1253, Madison, Wisconsin USA, June 2005.

[Liu 2002] Tiecheng Liu and John R. Kender. *An Efficient Error-Minimizing Algorithm for Variable-Rate Temporal Video Sampling.* In IEEE International Conference on Multimedia and Expo (ICME'02), Lausanne, Switzerland, August 26-29 2002.

[Liu 2003] Tianming Liu, Hong-Jiang Zhang and Feihu Qi. *A Novel Video Key-Frame Extraction Algorithm Based on Perceived Motion Energy Model.* IEEE Transaction on Circuits and Systems for Video Technology, vol. 13, no. 10, pages 1006–1013, October 2003.

[Liu 2004] T. Liu, X. Zhang, J. Feng and K.T. Lo. *Shot Reconstruction Degree : A Novel Criterion for Keyframe Selection.* accepted for publication, Pattern Recognition Letter, vol. 25, no. 12, pages 1451–1457, September 2004.

[Liu 2006] Z. Liu, D. Gibbon, E. Zavesky, B. Shahraray and P. Haffne. *AT&T RESEARCH AT TRECVID 2006.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Liu 2007] Xueliang Liu, Tao Mei, Xian-Sheng Hua, Bo Yang and He-Qin Zhou. *Video collage.* In ACM International Conference on Multimedia (ACMMM'07), pages 461–462, Augsburg, Germany, September 24 - 29 2007.

[Liu 2008] Yang Liu, Yan Liu, Tongwei Ren and Keith C. C. Chan. *Rushes video summarization using audio-visual information and sequence alignment.* In ACM International Conference on Multimedia, pages 114–118, Vancouver, BC, Canada, october 2008.

[Makhoul 1999] John Makhoul, Francis Kubala, Richard Schwartz and Ralph Weischedel. *Performance measures for information extraction.* DARPA Broadcast News Workshop, Herndon, VA, February 1999.

[Marr 1980] D. Marr and E. Hildreth. *Theory of Edge Detection.* Journal Proceedings of the Royal Society of London. Series B, Biological Sciences (1934-1990)- Issue Volume 207, Number 1167 / February 29, 1980.

[Mérialdo 2006] Bernard Mérialdo and Benoit Huet. Automatic video summarization. Chapter in "Interactive Video, Algorithms and Technologies" by Hammoud, Riad (Ed.), 2006, XVI, 250 p, ISBN : 3-540-33214-6, Jun 2006.

[Miura 2003] Koichi Miura, Reiko Hamada, Ichiro Ide, Shuichi Sakai and Hidehiko Tanaka. *Motion Based Automatic Abstraction of Cooking Videos.* In IPSJ Transactions on Computer Vision and Image Media, volume 44, 2003.

[Moravec 1980] H.P. Moravec. *Obstacle avoidance and navigation in the real world by a seeing robot rover.* Tech. Rept, CMU-RI-TR-3, The Robotic Institute, Carnegie-Mellon University, 1980.

[MPEG-7 2002] MPEG-7. *Multimedia Content Description Interface, Part 3.* Information Technology, ISO/IEC 15938-3, 2002.

[Naci 2008] Suphi Umut Naci, Uros Damnjanovic, Boris Mansencal, Jenny Benois-Pineau, Christian Kaes, Marzia Corvaglia, Eliana Rossi and Naiara Aginako. *The COST292 experimental framework for rushes summarization task in TRECVID 2008.* In ACM International Conference on Multimedia, pages 40–44, Vancouver, BC, Canada, october 2008.

[Needleman 1970] Saul Needleman and Christian Wunsch. *A general method applicable to the search for similarities in the amino acid sequence of two proteins.* Journal of Molecular Biology, 1970.

[Noguchi 2008] Akitsugu Noguchi and Keiji Yanai. *Rushes summarization based on color, motion and face.* In ACM International Conference on Multimedia, pages 139–143, Vancouver, BC, Canada, october 2008.

[Oh 2002] JungHwan Oh and Praveen Sankuratri. *Computation of Motion Activity Descriptors in Video Sequences.* Advances in Multimedia, Video and Signal Processing Systems, 2002.

[Over 2007] Paul Over, Alan F. Smeaton and Philip Kelly. *The trecvid 2007 BBC rushes summarization evaluation pilot.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Pan 2007] Chen-Ming Pan, Yung-Yu Chuang and Winston H. Hsu. *NTU TRECVID-2007 Fast Rushes Summarization System.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Peker 2001] K.A. Peker, A. Divakaran and Huifang Sun. *Constant pace skimming and temporal sub-sampling of video using motion activity.* In IEEE International Conference on Image Processing (ICIP'01), volume 3, pages 414–417, Thessaloniki, Greece, oct 7-10 2001.

[Peker 2003] Kadir A. Peker, Kadir A. Peker, Ajay Divakaran and Ajay Divakaran. *Framework for measurement of the intensity of motion activity of video segments.* Journal of Visual Communications and Image Representation, vol. 14, 2003.

[Peyrard 2003] Natalie Peyrard and Patrick Bouthemy. *Motion Based Selection of Relevant Video Segments for Video Summarization.* In IEEE International Conference on Multimedia and Expo (ICME'03), Baltimore, NY, July 6-9 2003.

[Pfeioeer 1996] S. Pfeioeer, R. Lienhart, S. Fischer, W. Eoeelsberg, Praktische Informatik Iv, D-Mannheim, Silvia Pfeioeer, Rainer Lienhart, Stephan Fischer, Wolfgang Eoeelsberg and Praktische Informatik. *Abstracting Digital Movies Automatically*, 1996.

[Picard 1995] Rosalind W. Picard. *Toward a visual thesaurus.* In Proceeding of the Final Workshop on Multimedia Information Retrieval (MIRO'95), pages 35–8, Glasgow, Scotland, September 1995.

[Picard 1996] Rosalind W. Picard. *A Society of Models for Video and Image Libraries*, 1996.

[Pinzon 2005] Juan Camilo Pinzon and Rahul Singh. *Designing an experiential annotation system for personal multimedia information management*, 2005.

[Platt 2000] J. C. Platt. Probabilities for sv machines. 2000.

[Prewitt 1970] J M S Prewitt. *Object enhancement and extraction*, 1970.

[Putpuek 2008] Narongsak Putpuek, Duy-Dinh Le, Nagul Cooharojananone, Shin'ichi Satoh and Chidchanok Lursinsap. *Rushes summarization using different redundancy elimination approaches.* In ACM International Conference on Multimedia, pages 100–104, Vancouver, BC, Canada, october 2008.

[Quinlan 1993] Ross Quinlan. C4.5 : Programs for machine learning. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[Quénot 1996] Georges M. Quénot. *Computation of Optical Flow Using Dynamic Programming.* IAPR Workshop on Machine Vision Applications pages 249-52, Tokyo, Japan, 12-14 nov, 1996.

[Quénot 2008] G. Quénot, J. Benois-Peneau, B. Mansencal, F. Precioso, D. Gorisse, P. Lambert, B. Augereau, L Granjon, D. Pellerin, M. Rombaut and S. Ayache. *Rushes Summarization by IRIM Consortium : Redundancy Removal and Multi-feature Fusion.* In ACM International Conference on Multimedia, Vancouver, BC, Canada, october 2008.

[Reas 2007] C. Reas, B. Fry and J. Maeda. Processing : A programming handbook for visual designers and artists. 2007.

[Ren 2008] Jinchang Ren, Jianmin Jiang and Christian Eckes. *Hierarchical modeling and adaptive clustering for real-time summarization of rush videos in trecvid'08.* In ACM International Conference on Multimedia, pages 26–30, Vancouver, BC, Canada, october 2008.

[Rong 2004] Jiawei Rong, Wanjun Jin and Lide Wu. *Key Frame Extraction using Inter-Shot Information.* In International Conference on Multimedia & Expo (ICME'04), Taipei, Taiwan, June 27-30 2004.

[Rowley 1996] Henry Rowley, Shumeet Baluja and Takeo Kanade. *Neural Network-Based Face Detection.* In Computer Vision and Pattern Recognition '96, June 1996.

[Roy 2002] Sylvain Roy. Nearest neighbor with generalization. University of Canterbury, 2002.

[Sano 2008] Masanori Sano, Yoshihiko Kawai, Nobuyuki Yagi and Shin'ichi Satoh. *Video rushes summarization utilizing retake characteristics.* In ACM International Conference on Multimedia, pages 95–99, Vancouver, BC, Canada, october 2008.

[Shih 2004] Huang-Chia Shih and Chung-Lin Huang. *Detection of the highlights in baseball video program.* In International Conference on Multimedia & Expo (ICME'04), Taipei, Taiwan, June 27-30 2004.

[Shipman 2003] F. Shipman, A. Girgensohn and L. Wilcox. *Creating navigable multi-level video summaries.* In IEEE International Conference on Multimedia and Expo (ICME'03), pages 753–756, Baltimore, NY, July 6-9 2003.

[Sivic 2003] Josef Sivic and Andrew Zisserman. *Video Google : A Text Retrieval Approach to Object Matching in Videos.* In Ninth IEEE International Conference on Computer Vision (ICCV'03), Washington, DC, USA, April 2003.

[Smeulders 2000] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta and Ramesh Jain. *Content-Based Image Retrieval at the End of the Early Years.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 12, pages 1349–1380, December 2000.

[Smith 1981] T. F. Smith and M. S. Waterman. *Identification of common molecular subsequences.* Journal of Molecular Biology, 1981.

[Sobel 1968] I. Sobel and G. Feldman. *A 3x3 Isotropic Gradient Operator for Image Processing*, 1968.

[Sun 2000] Xinding Sun and Mohan. S. Kankanhalli. *Video Summarization Using R-Sequences.* J.l of Real Time Imaging, vol. 6, pages 449–459, 2000.

[Sundaram 2001] H. Sundaram and Shih-Fu Chang. *Condensing computable scenes using visual complexity and film syntax analysis.* In IEEE International Conference on Multimedia and Expo (ICME'01), pages 273–276, Tokyo, Japan, Aug 22-25 2001.

[Swain 1991] Michael J. Swain and Dana H. Ballard. *Color indexing.* Int. J. Comput. Vision, vol. 7, no. 1, pages 11–32, 1991.

[Tan 1995] Yap-Peng Tan, Sanjeev R. Kulkarni and Peter J. Ramadge. *A new method for camera motion parameter estimation.* In IEEE International Conference on Image Processing (ICIP'95), pages 406–409, Washington D.C., oct 1995.

[Tan 2000] Y. Tan, D. Saur, S. Kulkami and P. Ramadge. *Rapid Estimation of Camera Motion from Compressed Video with Application to Video Annotation*, 2000.

[Tang 2006] S. Tang, Y.-D. Zhang, J.-T. Li, X. Pan, T. Xia, M. Li, A. Liu, L. Bao, Q. Yan, L. Tan and S. Liu. *TRECVID 2006 Rushes Exploitation by CAS MCG.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Taskiran 2001] C. M. Taskiran, A. Amir, D. B. Ponceleon and E. J. Delp. *Automated video summarization using speech transcripts.* In Storage and Retrieval for Media Databases, pages 371–382, San Jose, USA, dec 2001.

[Taskiran 2006] Cuneyt M. Taskiran. *Evaluation of automatic video summarization systems.* In Edward Y. Chang, Alan Hanjalic and Nicu Sebe, editeurs, SPIE, 2006.

[Tjondronegoro 2004] Dian Tjondronegoro, Yi-Ping Phoebe Chen and Binh Pham. *Integrating Highlights for More Complete Sports Video Summarization.* IEEE MultiMedia, vol. 11, no. 4, pages 22–37, 2004.

[Toharia 2008] Pablo Toharia, Oscar David Robles, Luis Pastor and Angel Rodríguez. *Combining activity and temporal coherence with low-level information for summarization of video rushes.* In ACM International Conference on Multimedia, pages 70–74, Vancouver, BC, Canada, october 2008.

[Tollari 2005] Sabrina Tollari, Hervé Glotin and Jacques Le Maitre. *Maitre. Enhancement of textual images classification using segmented visual contents for image search engine.* Multimedia Tools and Applications, vol. 25, pages 405–417, 2005.

[Tonomura 1993] Yoshinobu Tonomura, Akihito Akutsu, Kiyotaka Otsuji and Toru Sadakata. *VideoMAP and VideoSpaceIcon : tools for anatomizing video content.* In Human Factors in Computing Systems (CHI'93), pages 131–136, Amsterdam, April 1993.

[Truong 2000] Ba Tu Truong, Chitra Dorai and Svetha Venkatesh. *New enhancements to cut, fade, and dissolve detection processes in video segmentation.* In ACM International Conference on Multimedia (ACMMM'00), pages 219–227, Los Angeles, Nov 2000.

[Truong 2006] B.T. Truong and S. Venkatesh. *Curtin at TRECVID 2006 Rushes Summarization.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Truong 2007a] Ba Tu Truong and Svetha Venkatesh. *Generating Comprehensible Summaries of Rushes Sequences based on Robust Feature Matching.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Truong 2007b] Ba Tu Truong and Svetha Venkatesh. *Video abstraction : A systematic review and classification.* TOMCCAP - ACM Trans. Multimedia Comput. Commun. Appl., vol. 3, no. 1, 2007.

[Uchiashi 1999] Shingo Uchiashi, Jonathan Foote, Andreas Girgensohn and John Boreczky. *Video Manga : Generating Semantically Meaningful Video Summaries.* In ACM International Conference on Multimedia (ACMMM'99), pages 383–392, Orlando, Florida, Nov 1999.

[Ueda 1991] Hirotada Ueda, Takafumi Miyatake and Satoshi Yoshizawa. *IMPACT : an interactive natural-motion-picture dedicated multimedia authoring system.* In Human Factors in Computing Systems (CHI'91), pages 343–350, New Orleans, Louisiana, United States, April 1991.

[Ulges 2006] A. Ulges, C. Lampert and D. Keysers. *Spatiogram-Based Shot Distances for Video Retrieval.* In TrecVid 2006, 10th International Workshop on Video Retrieval Evaluation (TRECVID'06), Gaithersburg, USA, Nov 2006.

[Valdés 2007] Víctor Valdés and José M. Martínez. *On-line Video Skimming Based on Histogram Similarity.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Vasconcelos 1997] Nuno Vasconcelos. *Towards semantically meaningful feature spaces for the characterization of video content.* In International Conference on Image Processing (ICIP '97), pages 26–29, Washington, DC, USA, October 1997.

[Viola 2004] Paul Viola and Michael J. Jones. *Robust Real-Time Face Detection.* International Journal of Computer Vision, vol. 57, no. 2, pages 137–154, May 2004.

[Wang 1999] Roy Wang and Thomas S. Huang. *Fast Camera Motion Analysis in Mpeg Domain.* In IEEE International Conference on Image Processing (ICIP'99), pages 691–694, Kobe, Japan, oct 1999.

[Wang 2007] Feng Wang and Chong-Wah Ngo. *Rushes video summarization by object and event understanding.* In ACM International Conference on Multimedia, Augsburg, Germany, September 2007.

[Wang 2008] Tao Wang, Shangping Feng, Patricia P. Wang, Wei Hu, Shuang Zhang, Wei Zhang, Yangzhou Du, Jianguo Li, Jianmin Li and Yimin Zhang. *THU-intel at rushes summarization of TRECVID 2008.* In ACM International Conference on Multimedia, pages 124–128, Vancouver, BC, Canada, october 2008.

[Wolf 1996] W. Wolf. *Key frame selection by motion analysis.* In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'96), pages 1228–1231 vol. 2, Atlanta, Georgia, 7-10 May 1996.

[Wu 1998] Min Wu, Wayne Wolf and Bede Liu. *An algorithm for wipe detection.* In IEEE International Conference on Image Processing (ICIP'98), pages 893–897, Chicago, Illinois, oct 1998.

[Xiong 1997] Wei Xiong, John Chung-Mong Lee and Rui-Hua Ma. *Automatic video data structuring through shot partitioning and key frame computing.* Machine Vision and Applications, vol. 10, no. 2, pages 51–65, 1997.

[Xiong 2003] Ziyou Xiong, R. Radhakrishnan and A. Divakaran. *Generation of sports highlights using motion activity in combination with a common audio feature extraction framework.* In IEEE International Conference on Image Processing (ICIP'03), volume 1, pages 5–8, Barcelona, Spain, sep 14-17 2003.

[Xu 2003] Min Xu, N. C. Maddage, Changsheng Xu, M. Kankanhalli and Qi Tian. *Creating audio keywords for event detection in soccer video.* In IEEE International Conference on Multimedia and Expo (ICME'03), pages 281–284, Baltimore, NY, July 6-9 2003.

[Xu 2006] Changsheng Xu, Jinjun Wang, Kongwah Wan, Yiqun Li and Lingyu Duan. *Live sports event detection based on broadcast video and web-casting text.* In ACM International Conference on Multimedia (ACMMM'06), pages 221–230, Santa Barbara, CA, USA, October 2006.

[Yahiaoui 2001a] Itheri Yahiaoui, Bernard Merialdo and Benoit Huet. *Automatic Video Summarization.* In Multimedia Content-based Indexing and Retrieval (MMCBIR'01), Rocquencourt, France, September 24-25 2001.

[Yahiaoui 2001b] Itheri Yahiaoui, Bernard Mérialdo and Benoit Huet. *Generating summaries of multi-episodes video.* In IEEE International Conference on Multimedia and Expo (ICME'01), Tokyo, Japan, Aug 22-25 2001.

[Yeung 1995] M.M. Yeung and Bede Liu. *Efficient Matching and Clustering of Shots.* In IEEE International Conference on Image Processing (ICIP'95), volume 2, pages 338–341, Washington D.C., oct 1995.

[Yeung 1997] M. M. Yeung and B.-L. Leo. *Video Visualization for compact representation and fast browsing of pictorial content.* In IEEE Transactions on Circuits and Systems for Video Technology (TCSV'97), pages 7(5) :771–785, Oct. 1997.

[Yu 2004] Xiao-Dong Yu, Lei Wang, Qui Tian and Ping Xue. *Multi-Level Video Representation with Application to Keyframe extraction.* In The 10th International Multi-Media Modelling Conference (MMM'04), pages 117–121, Brisbane, Australia, January 2004.

[Zhang 1997] H. Zhang, J. Wu, D. Zhong and S.W. Smoliar. *An Integrated System for Content-Based Video Retrieval and Browsing.* Pattern Recognition, vol. 30, no. 4, pages 643–658, 1997.

[Zhang 2002] Dongqing Zhang and Shih-Fu Chang. *Event detection in baseball video using superimposed caption recognition.* In ACM International Conference on Multimedia (ACMMM'02), pages 315–318, Juan Les Pin, France, Dec 1-6 2002.

[Zhang 2003] Xu-Dong Zhang, Tie-Yan Liu, Kwok-Tung Lo and Jian Feng. *Dynamic selection and effective compression of key frames for video abstraction.* Pattern Recognition Letters, vol. 24, no. 9-10, pages 1523–1532, 2003.

[Zhang 2004] Ruofei Zhang and Zhongfei (Mark) Zhang. *Hidden Semantic Concept Discovery in Region Based Image Retrieval.* cvpr, vol. 02, pages 996–1001, 2004.

[Zheng 2000] Zijian Zheng and G. Webb. *Lazy Learning of Bayesian Rules.* Machine Learning, vol. 4, no. 1, pages 53–84, 2000.

[Zhu 2000] Lei Zhu, Aidong Zhang, Aibing Rao and Rohini K. Srihari. *Keyblock : an approach for content-based image retrieval.* eighth ACM international conference on Multimedia table of contents, pages 157–166, 2000.

[Zhuang 1998] Y. Zhuang, Y. Rui, T.S. Huang and S. Mehrotra. *Adaptive Key Frame Extraction Using Unsupervised Clustering.* In IEEE International Conference on Image Processing (ICIP'98), pages 866–870, Chicago, Illinois, oct 1998.

[Zimmerman 2003] John Zimmerman, Nevenka Dimitrova, Lalitha Agnihotri, Angel Janevski and Lira Nikolovska. *Interface design for MyInfo : A personal news demonstrator combining Web and TV content.* In 10th International Conference on Human-Computer Interaction (INTERACT '03), pages 41–48, Heraklion, Crete, Greece, June 22-27 2003.

[URL 1] weka. *http ://www.cs.waikato.ac.nz/ ml/weka/.*

Table des figures

Liste des tableaux

Résumé

Cette thèse se situe dans le contexte de l'analyse de vidéos ; en particulier des vidéos appelées rushes. Les rushes d'un film sont constitués des documents originaux (bobines de film, bandes sons, ...) produits au tournage et issus de la caméra et de l'appareil d'enregistrement sonore. Ce sont des documents uniques, bruts, qui seront utilisés au montage et en postproduction.
Nous proposons différents outils pour l'exploitation des rushes tels que des méthodes pour supprimer les séquences outils et poubelles du flux vidéo ; une méthode de recherche de plans vidéos grâce à l'utilisation d'un plan vidéo ; une mesure du contenu visuel d'une séquence vidéo ainsi qu'une structuration de la vidéo permettant de supprimer la redondance dans une vidéo en se basant sur l'alignement de séquences vidéos. Ensuite ces outils ont été incorporés dans des systèmes pour la création de résumés vidéo de rushes. Le premier système se base uniquement sur la mesure du contenu vidéo, le deuxième utilise l'alignement des séquences ; en parallèle, nous avons développé une architecture permettant une collaboration entre laboratoires.
Nous avons soumis ces différents systèmes à la campagne d'évaluation internationale TRECVID. Les résultats obtenus furent satisfaisants. Cependant cette méthode d'évaluation est manuelle, nous avons donc étudié de l'automatisation de cette évaluation.

Mots-clés: résumé vidéo, TRECVID

Abstract

The purpose of this document is video analysis and in particular analysis of video rushes. In filmmaking, rushes is the term used to describe the raw, unedited, footage shots which are created during the making of a motion picture.
We propose several tools to explore rushes. The first one is a tool to remove redundancy : the redundancy can be absolute (i.e. the content is not needed) or relative (i.e. the content is repetitive). An other method is a shot video search using a visual dictionary based on the paradigm of textual document search.
In order to create video summarization, we propose a method to represent the quantity of the relevant visual content of a video sequence. A second technique is to align repetitive video sequences in order to parse the video and remove repetitive takes. At the same time, we present a collaborative architecture allowing to fuse different partner analysis in order to exploit their different competences. We use these methods to create systems submitted to TRECVID.
These systems were evaluated by TRECVID. Results encouraged us to continue on this direction. The main problem is that the TRECVID evaluations are currently performed by human judges. This creates fundamental difficulties because evaluation experiments are expensive to reproduce, and subject to the variability of human judgment. Therefore, we propose an approach to automate this evaluation procedure using the same quality criteria. Through experiments, we show a good correlation with the manual evaluation.

Keywords: video summarization, TRECVID