

Graphe biparti pour mesurer la similarité entre les vidéos

GAFOUR Yacine¹

Laboratoire SIMPA, Université Mohamed Boudiaf
(USTO) Oran, Algérie, gayacine@yahoo.fr

NOURINE Rachid²

Laboratoire ITIO, Université Oran, Algérie,
rachid.nourine@univ-oran.dz

Résumé

La recherche de vidéos sur le Web fournit des réponses liées essentiellement aux mots utilisés dans la requête. Parmi les réponses, on trouve généralement des vidéos (partiellement ou totalement) similaires, proposées sous différentes appellations. Nous proposons dans cet article une approche qui permet de classer ces vidéos selon leurs similarités. Chaque classe de vidéo représente un contenu plus ou moins similaire. Pour cela nous devons mesurer la dissimilarité entre les vidéos. Ainsi nous établissons le graphe biparti entre les images clés des vidéos deux par deux. Une classification hiérarchique ascendante est ensuite appliquée pour répartir ces vidéos dans un certain nombre de classes. Selon le niveau de la partition on peut alors sélectionner la vidéo la plus représentative de la classe correspondante.

Mots clés : *Graphe biparti, classification, similarité*

I. Introduction

L'évolution rapide de technologies numérique (téléphones portables, Internet à haut débit, la télévision numérique, etc) a mené à une augmentation explosive des données multimédia disponibles dans le domaine public et exploitées par de nombreuses applications journalistiques, éducatives, médicales, etc. Il est donc nécessaire de mettre en place des outils permettant de retrouver rapidement le document recherché dans une base de données gigantesque. Cependant, une recherche effectuée sur le Web est généralement basée sur une requête textuelle ce qui génère de nombreuses réponses plus ou moins semblables même si elles portent des noms différents. L'expérience a montré que sur une vingtaine de requêtes soumises sur un moteur de recherche vidéo comme par exemple YouTube et GoogleVidéo, il y a en moyenne 73 % qui ne sont pas similaires dans les résultats de recherche (table 1).

Il est donc important de pouvoir détecter les vidéos (partiellement ou totalement) similaires et de n'en garder que l'exemplaire complet par exemple ou de les classer selon un ordre de ressemblance.

Cet article présente d'une part une approche pour mesurer la ressemblance entre les vidéos basée sur un couplage maximum (maximum matching) dans un graphe biparti et d'autre part une méthode de classification de ces vidéos basée sur la CHA (Classification Hiérarchique Ascendante), permettant ainsi de sélectionner les vidéos les plus ressemblantes.

II. Etat de l'art

La recherche de vidéos par le contenu a été largement étudiée dans la littérature. Diverses approches ont été proposées. Parmi celles-ci on trouve l'extraction des caractéristiques de vidéos [02] et l'utilisation des algorithmes d'appariement entre les vidéos [03]. Les caractéristiques se divisent en deux classes: globales et locales. Elles permettent d'identifier les vidéos identiques sauf qu'en utilisant les caractéristiques locales le résultat est très cher en temps de réponse par rapport aux caractéristiques globales. Elles sont basées sur le traitement d'intensité des pixels des images. Les caractéristiques globales et locales sont utilisées dans le domaine spatial et temporel où elles sont basées généralement sur les images clés extraites. Les caractéristiques comprennent la signature de la couleur, du mouvement et ordinal [03]. Xiao Wu et al. présentent dans [01] une approche hiérarchique. Un tri initial est développé en se basant sur une primitive extraite de l'histogramme des couleurs. Mais lorsqu'une vidéo ne peut pas être classée de manière précise, ils proposent d'utiliser des caractéristiques locales.

Pour comparer les vidéos il est nécessaire de calculer la similarité entre les signatures. Parmi les méthodes utilisées dans ce cadre on trouve généralement celles qui se basent sur la mesure du pourcentage des images en commun entre deux séquences vidéo [04]. On trouve aussi la programmation dynamique (DTW) [07] utilisée pour mesurer la différence temporelle entre deux séquences.

La distance mesurée dans l'espace vectoriel est généralement utilisée afin de détecter les vidéos similaires pour les caractéristiques globales. Il existe d'autres méthodes basées sur les caractéristiques locales pour indexer un grand ensemble de vidéos avec l'étude de la trajectoire durant les séquences vidéo [06]. J. Law-To et al. présentent une approche pour indexer un grand ensemble de vidéos avec l'étude de trajectoire des caractéristiques locales durant la séquence vidéo [05]. Ils ont démontré l'efficacité de cette approche pour la découverte des copies de vidéos, basée sur le contenu des grandes collections de vidéos (plusieurs centaines d'heures de vidéos).

III. Approche proposée

Dans cette section nous montrons comment classer les vidéos selon leur similarité. Notre approche se fait en deux phases (voir l'algorithme ci-dessous). Dans la première phase, nous construisons une matrice qui représente l'indice de

dissimilarité entre toutes les vidéos deux par deux. A cet effet, nous établissons le graphe biparti complet pour chaque couple de vidéos. Ensuite pour chaque image clé dans une vidéo nous cherchons celle qui correspond le plus dans l'autre vidéo.

requêtes			Proches-Doublans	
ID	Requête	#	#	%
1	The lion saleps tong	792	334	42 %
2	Evolution of dance	483	122	25 %
3	Fol shirt	436	183	42 %
4	Cat massage	344	161	47 %
5	Ok go here it goes again	396	89	22 %
6	Urban ninja	771	45	6 %
7	Real life Simpsons	365	154	42 %
8	Free hugs	539	37	7 %
9	Where the hell is Matt	235	23	10 %
10	U2 and green day	297	52	18 %
11	Little superstar	377	59	16 %
12	Napoleon dynamite dance	881	146	17 %
13	I will survive Jesus	416	387	93 %
14	Ronaldinho ping pong	107	72	67 %
15	White and Nerdy	1771	696	39 %
16	Korean karaoke	205	20	10 %
17	Panic at the disco I write sins not tragedies	647	201	31 %
18	Bus uncle (巴士阿叔)	488	80	16 %
19	Sony Bravia	566	202	36 %
20	Changes Tupac	194	72	37 %
21	Afternoon delight	449	54	12 %
22	Numa Gary	422	32	8 %
23	Shakira hips don't lie	1322	234	18 %
24	India driving	287	26	9 %
Total		12790	3481	27 %

Table 1. 24 requêtes soumises sous YouTube, Google Vidéo et Yahoo! Vidéo (#: nombre de vidéos) [01]

Dans la deuxième phase nous classifions les vidéos en fonction de leur similarité. Pour ce faire, nous créons en plusieurs étapes des partitions en agrégeant deux à deux les

classes de vidéos les plus similaires. Une classe peut être une seule vidéo ou plusieurs (section III.3).

Algorithme

Soit E une base de données de vidéos

Phase 1 : Dissimilarités entre vidéos

1/ Extraire les images clés pour chaque vidéo

2/ Pour chaque couple de vidéos (Vi,Vj)

- Construire un graphe bipartite
- Calculer la ressemblance maximale (algorithme hongrois)

3/ Construire un indice de dissimilarité D entre les vidéos.

Phase 2 : Appliquer une classification hiérarchique ascendante sur D

III.1. Extraction d'images clés

Pour comparer deux vidéos nous extrayons leurs images clés représentatives. A cet effet, nous faisons la segmentation de chaque vidéo en plans. Ces derniers sont définis comme une portion d'une vidéo filmée continûment sans coupure. Dans la Figure 1, nous avons pris une séquence de vidéo qui contient 3 plans. Le 1^{er} commence en image i-10 jusqu'à image i-8 et le 2^{ème} commence en image i-7 jusqu'à l'image i. La segmentation d'une vidéo en plans consiste à déterminer les bornes des plans indiqués (Fig 1) en calculant la différence des valeurs h_i des histogrammes 2D des images successives Eq. (1). Ces valeurs sont calculées uniquement sur les deux composantes **H** et **S** (définies dans l'espace colorimétrique HSV (Hue-Saturation-Value) [08]) quantifiées chacune sur un nombre d'intervalles.

$$Norme(h_i, h_{i+1}) = \sum_{v \in H} \sum_{w \in S} |h_i(v, w) - h_{i+1}(v, w)| \quad (1)$$

$h_i(v, w)$ est le nombre de pixels de la $i^{\text{ème}}$ image sur le $v^{\text{ème}}$ intervalle de la composante **H** et le $w^{\text{ème}}$ intervalle de la composante **S**.

Ainsi, la fin d'un plan vidéo est déterminée par l'image i vérifiant $Norme(h_i, h_{i+1}) > \epsilon$, où ϵ est un seuil que nous avons déterminé de manière expérimentale (Fig 2)(Table 3), tel qu'il est expliqué dans la section 4. Différentes approches sont proposées pour sélectionner l'image clé d'un plan [15]. Pour des raisons d'optimisation du temps de détection des vidéos similaires, nous proposons de prendre la dernière image du plan comme image clé de celui-ci.

III.2 Couplage maximal entre vidéos

Pour chaque couple de vidéos (Vi,Vj), on récupère respectivement deux ensembles d'images clés Eq. (2). On construit alors la matrice de dissimilarités A entre ces deux ensembles d'images clés Eq. (3). Chaque dissimilarité

$A(c_{il}, c_{jk})$ est calculée pour le couple d'images clés (c_{il}, c_{jk}) issues respectivement des vidéos V_i et V_j Eq. (4).



Fig 1- Des images successives capturées de la vidéo BG_34828.mpg de la collection 2007 [11]

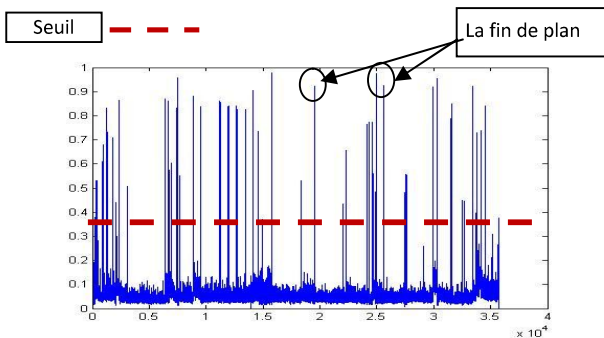


Fig 2 - Détection les bornes des plans de la vidéo par notre méthode de la vidéo BG_34984.mpg [11]

Cette matrice de dissimilarités A peut être représentée sous forme d'un graphe biparti où les sommets sont les images clés des deux vidéos et les arcs représentent les dissimilarités entre les images clés tel que le montre la Fig 3.

$$\begin{aligned} V_i &= \{c_{i1}, \dots, c_{im}\} \\ V_j &= \{c_{j1}, \dots, c_{jn}\} \end{aligned} \quad (2)$$

$$A = \begin{bmatrix} A(c_{i1}, c_{j1}) & \dots & A(c_{i1}, c_{jn}) \\ \vdots & \dots & \vdots \\ A(c_{im}, c_{j1}) & \dots & A(c_{im}, c_{jn}) \end{bmatrix} \quad (3)$$

$$A(c_{il}, c_{jk}) = \sum_{v \in H} \sum_{w \in S} \frac{|h_{il}(v, w) - h_{jk}(v, w)|}{h_{il}(v, w) + h_{jk}(v, w)}, \quad (4)$$

$l = 1..m$ et $k = 1..n$

Avec :

- c_{il} (resp c_{jk}) est la $l^{\text{ème}}$ (resp. $k^{\text{ème}}$) image clé de la $i^{\text{ème}}$ (resp. $j^{\text{ème}}$) vidéo

- $h_{il}(v, w)$, le nombre de pixels de la $l^{\text{ème}}$ image clé de la $i^{\text{ème}}$ vidéo sur le $v^{\text{ème}}$ intervalle de la composante H et le $w^{\text{ème}}$ intervalle de la composante S .

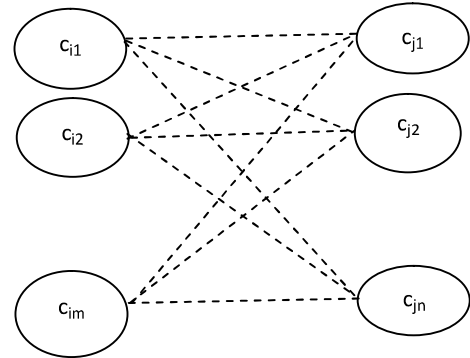


Fig 3- Exemple d'un Graphe biparti complet

Nous cherchons alors à faire correspondre à chaque image clé de la vidéo V_i une seule image clé de la vidéo V_j et vice-versa. Cela revient à chercher un couplage maximal dans le graphe biparti [09]. Pour cela on propose d'utiliser l'algorithme hongrois, appelé aussi algorithme de Kuhn-Munkres [14], que l'on applique à la matrice A . La Figure 4 présente un résultat de l'affectation optimale de chaque image clé de V_i à une image clé correspondante dans V_j .

La dissimilarité que nous avons utilisée entre les deux vidéos est alors représentée par la moyenne des dissimilarités (en couleur rouge dans la Figure 4) entre les images clés mise en correspondances Eq. (5) au lieu de calculer leur moyenne de similarité [16].

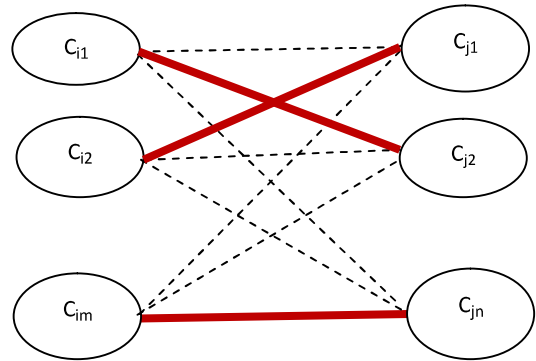


Fig 4- couplage maximum d'un graphe biparti

Ainsi, nous construisons une matrice de dissimilarité D Eq. (6) entre les différentes vidéos récupérées sur le Web.

$$D(v_i, v_j) = \frac{1}{m + n - r} \sum A(c_{il}, c_{jk}) \quad (5)$$

$$D(v_i, v_i) = 0 \text{ pour } i = 0..N.$$

m : nombre des images clés de la 1^{ère} vidéo.

n : nombre des images clés de la 2^{ème} vidéo.

r : nombre d'affectations entre les deux vidéos.

Nous ne garderons que les affectations $A(C_{il}, C_{jk})$ qui ont satisfait le seuil ρ , qui est déterminé de manière expérimentale.

III.3 Classification hiérarchique ascendante

Ayant une mesure de dissimilarité mesurée entre les différentes vidéos Eq. (6), réponses à une requête sur le Web, on y applique une classification hiérarchique ascendante qui fournira différentes partitions.

La CHA permet de construire une hiérarchie des vidéos sous la forme d'un arbre dans un ordre ascendant [10]. On commence en considérant chaque vidéo comme une classe et on essaye de fusionner deux classes appropriées (dissimilarité minimale) pour former une nouvelle classe. On répète cette étape sur les autres classes selon cette dissimilarité jusqu'à ce que tous les vidéos seront réunis dans une seule classe. Cette classification génère un arbre que l'on peut couper à différents niveaux pour obtenir différentes partitions.

$$D = \begin{bmatrix} D(v_1, v_1) & \dots & D(v_1, v_N) \\ \vdots & \dots & \vdots \\ D(v_N, v_1) & \dots & D(v_N, v_N) \end{bmatrix} \quad (6)$$

Algorithme CHA

Soit E ensemble de N vidéos, représentant les N classes initiales.

i. Regrouper les deux classes les plus proches

(qui ont une dissimilarité minimale) \Rightarrow le nombre de classes sera N-1

ii. Calculer les dissimilarités entre la classe formée et les (N-2) autres classes. La distance entre deux classes est la distance minimale entre leurs éléments.

iii. Revenir à i, jusqu'au regroupement des N vidéos dans une seule classe

IV. Expérimentation

IV.1. Détection d'images clés

Afin d'évaluer les performances de notre méthode sur l'extraction des plans nous l'avons testé sur 5 vidéos MPEG-1 avec 25 images par seconde. Nous avons récupéré ces vidéos de la collection de référence TRECVID 2007 [11]. Les plans de ces vidéos sont déjà prédéfinis [12]. En se basant sur cet ensemble de référence, on mesure pour chaque vidéo testée le nombre de limites de plans correctement détectées, le nombre de fausses détections et le nombre des limites oubliées par notre méthode. Nous avons alors choisi comme critère d'évaluation les paramètres Rappel Eq. (7) et Précision Eq. (8)

[13]. Le paramètre Rappel mesure le pourcentage des éléments correctement trouvés par rapport au total souhaité et le paramètre précision définit le pourcentage des éléments correctement trouvés par rapport au total obtenu. Nous avons proposé différents seuils ϵ de comparaison des histogrammes 2D. Sur la base de ces tests, nous avons choisi comme seuil la valeur $\epsilon=0.37$ (Table 3).

$$Rappel = \frac{correctes}{correctes + oubliées} \quad (7)$$

$$Precision = \frac{correctes}{correctes + faussesdetection} \quad (8)$$

- **correcte** : Plans correctement détectés
- **oublié** : Plans de la vidéo non détecté
- **faussesdetection** : Plans faussement détectés

IV.2 Exemple d'application et discussion des résultats

Parmi les nombreux exemples que nous avons traités, on présente ici celui de la requête « l'Algérie vs l'Egypte » qui a été soumise sous YouTube, ce qui a fourni plusieurs vidéos liées à ce match de football entre les deux équipes Algérienne et Egyptienne. La table 2 présente quelques exemples et la Figure 5 donne quelques vidéos utilisées. Nous avons constaté que les réponses sont diverses textuellement. De plus, en visionnant les différentes vidéos, nous avons constaté que les contenus des vidéos sont différents. L'application de notre approche a fourni l'arbre de classification hiérarchique qui indique plusieurs partitions possibles (Figure 6) :

Nous avons constaté que les premiers niveaux ne fournissent pas une classification. Par exemple, dans le niveau 0, chaque vidéo compose une classe ; dans le niveau 1 les deux premières vidéos composent une seule classe et les autres vidéos chacune une classe et ainsi de suite jusqu'à ce qu'on arrive à un niveau contenant des classes se composant uniquement des vidéos similaires. Par exemple dans le 3^{ème} niveau les vidéos 1, 7 et 8 sont des vidéos semblables même si leurs durées et leurs résolutions sont différentes. Cela est dû au fait qu'elles représentent le résumé du match. Les vidéos 10, 11, 12, 15 sont entre le niveau 4 et 7 parlent sur le résultat du match où la vidéo 12 représente une émission télévisée "C'EST DANS L'AIRE" de France 5 et la vidéo 11 montre Marseille après le match (l'Algérie vs l'Egypte). Dans le niveau 8 l'application donne une nouvelle classe qui contient la vidéo (17 et 4). La vidéo 17 représente la 1er mi-temps et la vidéo 4 représente la 2ème mi-temps du match et donc ils sont des vidéos semblables.

On a remarqué que pour le temps de classifications des vidéos est relatif par rapport au nombre des images clés extraites de chaque vidéo. Pour les courtes vidéos (de durée inférieure à 1 minute par exemple) ou les vidéos qui ont la même scène (même plan) influe sur le temps de classification parce que dans ce type de vidéo les images clés extraites sont très petites. Au contraire aux vidéos qui ont un nombre important d'images clés. Dans ce cas le couplage maximum par l'algorithme de hongrois prend un temps important pour faire l'affectation optimale qui influe sur la classification.

V. Conclusion

On aborde dans cet article le problème de détection de vidéos similaires d'une réponse à une requête soumise à un moteur de recherche Web. Nous proposons alors une approche de classification hiérarchique ascendante de l'ensemble de ces vidéos, pouvant permettre à sélectionner les classes de vidéos les plus homogènes. La comparaison des vidéos est basée sur une mesure de dissimilarité calculée sur leurs images clés.

Dans la première phase qui est l'extraction des plans de la vidéo, la méthode proposée est rapide et donne des résultats plus ou moins satisfaisants du au fait qu'on utilise une caractéristique globale qui est l'histogramme. Dans la phase

minimal de plans. Pour y remédier on peut proposer de calculer cette dissimilarité entre les vidéos ayant des durées proches.

Dans la phase de classification hiérarchique on prévoit de faire une analyse des résultats obtenus avec différents stratégies (saut minimum, saut maximum, moyenne des distances).

Cette approche peut être toutefois utilisée pour trouver les vidéos en double dans le Web ainsi on peut l'utiliser comme un bon outil pour contribuer à la protection du droit d'auteur afin de détecter les vidéos piratées. Cette approche pourrait également être utilisée pour détecter une scène.

Vidéo	Nom	Durée
Vidéo1	Algerie Egypte - Matche Complet (Soudan) - Part 111	4 :09
Vidéo2	Algerie VS Egypt Incroyable fin de matche kiki D	3 :16
Vidéo3	ALGERIE vs egypte REPORTAGE DE CANAL PLUS.flv	6 :41
Vidéo4	ALGERIE_vs_EGYPT_2_mi-temps	5 :10
Vidéo5	avant matche Algerie egypt le 14 novembre 2009	6 :52
Vidéo6	france2 et le matche algerieegypte au soudan	3 :40
Vidéo7	Full Matche Algerie vs Egypte 18 11 2009 Part1 Algeria 1 Egypte0 1-تعليق حفيظ دراجي	10 :56
Vidéo8	Full Matche Algerie vs Egypte 18 11 2009 Part3 Algeria 1 Egypte0 2-تعليق حفيظ دراجي	10 :54
Vidéo9	le matche algerieegypte	01 :33
Vidéo10	Le matche Algérie-Egypte s'invite au Forum Arabe de l'Information Dubaï 2010	04 :23
Vidéo11	Marseille apres le matche algerie egypte	4 :50
Vidéo12	Matche Algérie Egypte 350 voitures brûlées guérilla urbaine crise de civilisation	04 :14
Vidéo13	Premiere mi-tempsMatche Algerie Egypt-18 novembre 2009 partie 4	09 :55
Vidéo14	Rappel a la oumma Matche Algerie Egypte HONTE A LA OUMMA	5 :16
Vidéo15	scandale du matche algerie egypte	03 :00
Vidéo16	video fin de matche algeriee gypte hafidderradj	00 :38
Vidéo17	ALGERIE_vs_EGYPT_1_mi-temps	49 :27
Vidéo18	ALGERIE_vs_EGYPT_Matche_-_Aller_Retour_Barage	09 :50

Table 2 - Quelques réponses à la requête « l'Algérie vs l'Egypte »

de comparaison des vidéos la dissimilarité est influencée par la plus petite vidéo ou bien par les vidéos ayant un nombre

Vidéo	Seuil ϵ	Rappel	Précision
g BG_34 835.amp	0.32	0.52	0.63
	0.37	0.47	0.67
	0.41	0.41	0.69
	0.45	0.23	0.65
g BG_35 447.amp	0.32	0.92	0.68
	0.37	0.81	0.68
	0.41	0.77	0.69
g BG_36 511.amp	0.45	0.72	0.74
	0.32	0.70	0.80
	0.37	0.63	0.90
	0.41	0.56	0.91
g BG_34 828.amp	0.45	0.47	0.91
	0.32	0.81	0.69
	0.37	0.78	0.75
	0.41	0.65	0.74
g BG_34 984.amp	0.45	0.57	0.77
	0.32	0.86	0.69
	0.37	0.86	0.80
	0.41	0.81	0.82
	0.45	0.74	0.78

Table3. - Résultat de détection des plans par notre méthode



Vidéo1

Vidéo2



Vidéo9

Vidéo10



Vidéo 11

Vidéo 14



Vidéo 16

Vidéo 17

Fig 5. Exemples de vidéos à la requête "Algérie vs Egypte"

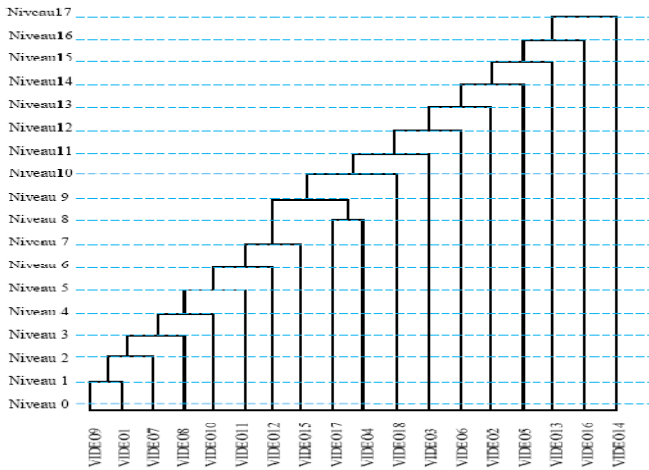


Fig 6 : Classification Hiérarchique Ascendante des vidéos

REFERENCES

- [01] X. Wu, A. G. Hauptmann, and C.-W. Ngo. : Practical elimination of near-duplicates from Web video search. In Proc. ACM Multimedia, 2007.
- [02] S-F. Chang, W. Hsu, L. Kennedy, L. Xie and et al. Columbia University TRECVID-2005 Video Search and High-Level Feature Extraction. TRECVID 2005, Washington DC, 2005.
- [03] A. Hampapur and R. Bolle. Comparison of Sequence Matching Techniques for Video Copy Detection. *Conf. On Storage and Retrieval for Media Databases*, 2002.
- [04] S. Cheung and A. Zakhor. Efficient video similarity measurement with video signature. In IEEE Transactions on Circuits and Systems for Video Technology, 2003.
- [05] J. Law-To, V. Gouet-Brunet, O. Buisson, and N. Boujemaa. Local Behaviours Labelling for Content Based Video Copy Detection. In International Conference Pattern Recognition, 18th Int. Conf. on Pattern Recognition, ICPR'06.
- [06] J. Law-To, V. Gouet-Brunet, O. Buisson, and N. Boujemaa. Local Behaviours Labelling for Content Based Video Copy Detection. In International Conference Pattern Recognition, 18th Int. Conf. on Pattern Recognition, ICPR'06.
- [07] M. R. Naphade, R. Wang, and T. S. Huang. Multimodal pattern matching for audio-visual query and retrieval. In SPIE, pages 188{195, 2001.
- [08] MJ Swain, DH Ballard: Color Indexing, International Journal of Computer Vision, 1991.
- [09] S. Micali and V. V. Vazirani. An $O(\sqrt{|V|} \cdot |E|)$ algorithm for finding maximum matching in general graphs. In Proc. of 21st FOCS, pages 17-27, 1980.
- [10] J.H. WARD. Hierarchical grouping to optimise an objective function ;Journal of the American Statistical Association , (38) : 238-244 ,1963
- [11] <http://vidtest.cloud.itee.uq.edu.au/video/db/trec/>
- [12] http://www-nlpir.nist.gov/projects/tv2007/pastdata/master.s_hot.reference/raw.master.shot_boundaries/
- [13] John S. Boreczky, Lawrence A. Rowe. Comparison of video shot boundary detection techniques. Journal of Electronic Imaging 5(2), 122–128 (April 1996)
- [14] James Munkres, "Algorithms for the Assignment and Transportation Problems", Journal of the Society for Industrial and Applied Mathematics, Vo. 5, N°.1, Mars 1957, pp. 32-38.
- [15] G. Ciocca, R. Schettini, "An innovative algorithm for key frame extraction in video summarization. RealTime IP(1), No. 1, October 2006, pp. 69-88. Springer DOI Link 0001BibRef
- [16] H. Kim, J. Lee, H. Liu, and D. Lee : Video Linkage: Group based copied video detection," in Proc. ACM International Conference on Image and Video Retrieval (CIVR'08), Niagara Falls, Canada, July 7–9, 2008.