MULTI-DOCUMENT VIDEO SUMMARIZATION

Feng Wang and *Bernard Merialdo*
Multimedia Communications Deptartment
Institute Eurecom, Sophia-Antipolis, France
{Feng.Wang, Bernard.Merialdo} @ eurecom.fr

ABSTRACT

Most previous works on video summarization target on a single video document. With the popularity of video corpus (*e.g.* news video archives) and web videos, video article that consists of a set of relevant videos are frequently confronted by users. By the traditional single-document summarization, these videos are treated independently and the results are usually redundant due to the lack of inter-video analysis. To efficiently manage video articles, in this paper, we propose an approach for multi-document video summarization by exploring the redundancy between different videos. The importance of keyframes is first measured by the content inclusion based on intra- and inter-video similarities. We then propose a Minimum Description Length (MDL) for automatically determining the appropriate length of the summary. Finally a video summary is generated for users to browse the content of the whole video article. We show that multi-document video summarization presents more elegant and informative summaries compared with single-document approach.

Index Terms— Multi-document, Video Summarization, Minimum Description Length.

1. INTRODUCTION

Nowadays video summarization has become the key tool for efficient browsing, access and manipulation of large video collections. Since 1990s, video summarization has attracted numerous researchers' attention. Most existing works focus on the summarization of a single video based on various features such as motion [1], audio [4] or multi-modality [2]. A systematic review of these works can be found in [5].

In the management of video corpus, the similar or related videos are usually grouped into one article according to the content since the users of these videos may also be interested in others related to them. For instance, in news video archives, all clips about the same story from different channels and news sessions are put together to present the evolution of the story. Another scenario is the search of web videos where given a query a list of relevant videos are presented to users. Here a video article is defined as a set of videos that share some common properties or contents. They might be similar in visual, audio, text, or event. Potential users would like to quickly browse the content of the whole video article in a short summary. By the traditional single-document video summarization, one sub-summary can be produced for each video and then concatenated to represent the content of the article. However, the relevancy between different videos and the property of the whole video article are not employed. For instance, some clips may be repeated in different videos and included in multiple sub-summaries. This makes the summary unnecessarily longer and more redundant than expected.

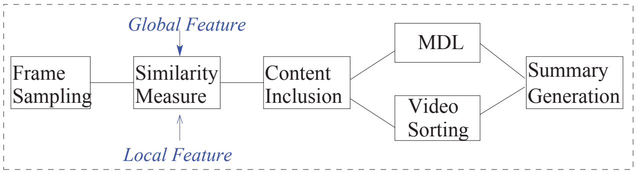


Fig. 1. Our framework for multi-document video summarization.

During past few years, numerous efforts have been devoted to multi-document text summarization [3, 6]. However, few works have been done in the summarization of multiple video documents. A solution for rapid browsing of news topics is proposed in [8]. The videos of the same topic are first clustered. The topic structure is then presented by exploring the textual-visual novelty and redundancy of the videos. In [7], from the search results of web videos, the highly similar videos are detected and eliminated. A more explicit set of videos are then presented to the users. However, summarization is not carried out. In [9], the common and unique materials in different episodes of TV series are identified for independent and dependent selection of keyframes.

In this paper, we present our approach for summarization of multiple videos in an article by exploring the redundancy between different videos. The framework is illustrated in Figure 1. Different from single-video summarization, to summarize a video article, we investigate the redundancy of video content not only inside a video, but also among different videos. Global and local visual features are employed to measure intra- and inter-video similarities (Sec. 2). For keyframe selection, we then propose a content inclusion measure to weight the keyframe importance. Both the amount and the distribution of the covered information in different videos are considered (Sec. 3). Finally, for summary generation, on one hand, we employ minimum description length (MDL) to automatically determine the summary length by balancing the content inclusion and the size (Sec. 4.1). On the other hand, we sort the presence of the videos in the final summary according to their importance in the article (Sec. 4.2).

978-1-4244-4291-1/09/$25.00 ©2009 IEEE

2. MULTI-VIDEO SIMILARITY MEASURE

Given a video article with a set of relevant videos, we first evenly extract 1 frame every second from each video. The sampled frames are used to represent the content of the corresponding 1-second clips. In this paper, we focus on employing the visual information for video summarization. Global and local features are extracted for frame similarity measure.

2.1. Global Features

Global features are extracted over the whole frame. We employ color and texture information to measure the visual similarity between frames. For color feature, in HSV space, we construct a histogram of $16(H) \times 4(S) \times 4(V)$ bins and form a $256 - d$ feature vector. For texture information, a given keyframe is splited into 3×3 grids and each grid is represented by the variances in 9 Haar wavelet sub-band to form a $81{-}d$ feature vector. Given two frames f_1, f_2, Cosine similarities are then calculated as $Sim_{cr}(f_1, f_2)$ and $Sim_{wv}(f_1, f_2)$ based on the color and texture features respectively. Their average is taken as the global similarity

$$Sim_g(k_1, k_2) = \frac{Sim_{cr}(k_1, k_2) + Sim_{wv}(k_1, k_2)}{2}$$

2.2. Local Feature

The global features are used to measure similarities between frames with similar color or texture distribution. However, in some cases, although the content of two frames might be similar, the color and texture are different. This is frequently encountered especially in multi-video summarization since the videos may come from different sources where they are captured and edited by different people. Figure 2 gives an example where neither color nor texture is reliable to identify the redundancy of the two frames although the main contents are exactly the same. To cope with this problem, we employ keyframe matching based on SIFT (Scale Invariant Feature Transformation) features [11]. In each frame, the local interest points (LIPs) are detected by DoG (Difference of Gaussian) and described by SIFT. Point-to-point matching is then used to detect the similar content (LIPs) between two frames.

In Figure 2, the matched keypoints between two keyframes are illustrated. By keyframe matching based on local features, we can find the contents of the two keyframes are actually quite similar since there are many matching points distributed in both frames. The number of matching lines $\mathcal{M}$ between two keyframes can be used to measure the content similarity between them. A problem with local features is that the number of keypoints detected in each image is quite different which can range from tens to thousands due to different image natures. Furthermore, the keypoint matching may be affected by some certain patterns, such as texts in images. In these cases, the global features are needed for more robust similarity measure. Thus, in next section, we combine local and global features to measure the frame similarity for calculating the content inclusion of a summary.

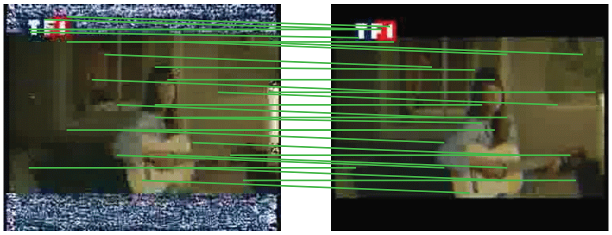


Fig. 2. Image matching based on local feature. The global similarity is 0.55 and not enough to detect the content duplication.

3. KEYFRAME SELECTION BY CONTENT INCLUSION MEASURE

Let $A = \{v_1, v_2, \cdots, v_n\}$ be a video article. In each video $v_i \in A$, a set of frames $\mathcal{F}_i = \{f_{i1}, f_{i2}, \cdots, f_{il_i}\}$ are sampled at 1 frame per second to represent video content. In this section, we define content inclusion to measure the importance of keyframes. The most important keyframes are then selected from different videos for summary generation.

3.1. Content Inclusion Measure

By traditional single-document video summarization, a sub-summary can be generated for each video and then concatenated as the summary of the video article. In this work, we explore the inter-video relations to produce a summary for the video article. A direct approach is to first combine all frames from different videos into one set $\mathcal{F} = \cup_{i=1}^{n} \mathcal{F}_i$. Keyframes are then selected according to their content inclusion for $\mathcal{F}$ instead of $\mathcal{F}_i$ from a single video.

Given a set of selected keyframes $S = \{k_1, k_2, \cdots, k_L\}$ where $S \subseteq \mathcal{F}$, we can then calculate the content inclusion of S for $\mathcal{F}$ by estimating how much visual information in $\mathcal{F}$ is included in S based on global and local information respectively as

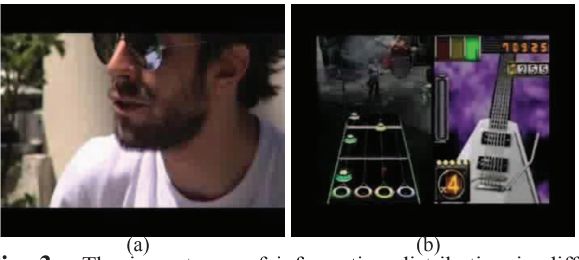
$$Inc_g(S, \mathcal{F}) = \frac{\sum_{f \in \mathcal{F}} \max_{k_j \in S} Sim_g(f, k_j)}{|\mathcal{F}|} \quad (1)$$

$$Inc_l(S, \mathcal{F}) = \frac{\sum_{f \in \mathcal{F}} \max_{k_j \in S} \mathcal{M}(f, k_j)}{T} \quad (2)$$

where $\mathcal{M}(\cdot)$ is the number of matching points between two frames, and $T = \sum_{f_a \in \mathcal{F}} \max_{f_b \in \mathcal{F}} \mathcal{M}(f_a, f_b)$ is used to estimate the amount of local information captured by LIPs in the video article. By combining global and local features, content inclusion of S for $\mathcal{F}$ is defined as

$$Inc(S, \mathcal{F}) = \sqrt{Inc_g(S, \mathcal{F}) \cdot Inc_l(S, \mathcal{F})} \quad (3)$$

This approach employs the inter-video relations by connecting all videos into one. However, the information distribution in different videos is not considered. For instance, in Figure 3, Frame (a) has 37 duplicates in 1 video, while frame (b) has 32 duplicates distributed in 5 videos. By the approach above, frame (a) has the larger content inclusion since it covers information of more frames in $\mathcal{F}$. However, since frame (b) carries information from more videos, it is thought as more important than (a).



(a) (b)

Fig. 3. The importance of information distribution in different videos. (a) 37 duplicates in 1 video; (b) 32 duplicates in 5 videos.

To solve this problem, we take into account the distribution of the content included by S in different videos. Instead of treating the video article as one video, we calculate the content inclusions of S for each single video v_i as $Inc_g(S, \mathcal{F}_i)$, $Inc_l(S, \mathcal{F}_i)$, and $Inc(S, \mathcal{F}_i)$ by substituting $\mathcal{F}$ in Equations (1-3) with $\mathcal{F}_i$ respectively. Thus, the content inclusion of S for the video article A is defined as

$$Inc(S, A) = \left(\prod_{i=1}^{n} Inc(S, \mathcal{F}_i)\right)^{1/n} \tag{4}$$

Compared with arithmetic average, the geometry average can be maximized when the variation of $Inc(S, \mathcal{F}_i)$ is minimized or the selected keyframes in S cover the content from more videos. By Equation 4, although frame (a) in Figure 3 has more duplicates in the video article, frame (b) will be selected with higher priority than (a) since it can increase the content inclusion of S for more videos.

3.2. Keyframe Selection

Based on the content inclusion defined in Equation 4, we select and put the keyframes in an ordered list according to their importance. The algorithm is described as below. In each loop, we expand the keyframe set S by appending the keyframe which improves the content inclusion the most.

1. *Initialize* $S = \phi$.
2. *Select the keyframe* k *from* $\mathcal{F} - S$ *so that* $k = \arg\max(Inc(S \cup \{k\}, A))$.
3. $S = S \cup \{k\}$. *Goto 2.*

With the sorted list S, we can generate a summary S_L with any length of L seconds by selecting the first L keyframes in S and concatenating the video clips into a short video. In next section, we then propose our approach to automatically determine the summary length and sort the selected keyframes to compose a video summary.

4. SUMMARY GENERATION

4.1. Determining Summary Length by Minimal Description Length

A good summary should contain as much information in the original videos as possible. However, this inevitably requires more keyframes in the summary. There is a tradeoff between the content inclusion and the summary length. To balance these two aspects, we propose a Minimum Description Length (MDL) for the summary to automatically determine the optimal summary length. In [10], MDL has been used to determine the number of scenes in rushes videos where each scene is usually taken many times.

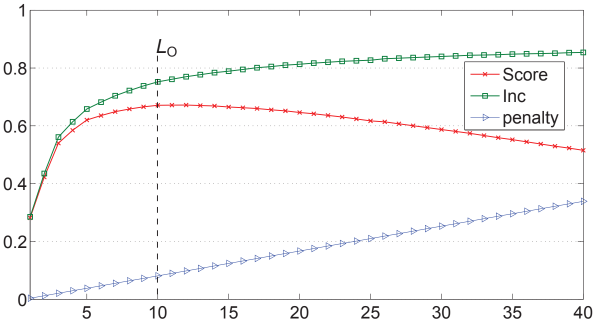


Fig. 4. MDL to determine summary length.

Given a summary S_L of length L generated in Sec. 3.2, the content inclusion of S_L is used to measure the description ability of the video article. The longer the summary, the larger the content inclusion. However, the larger content inclusion will be penalized by the extra size. We define the penalty function of the summary S as

$$Penalty(S_L) = \frac{L}{|\mathcal{F}|} \cdot (1 - MIN_{Inc}) \tag{5}$$

where $MIN_{Inc} = Inc(S_1, A)$ is used for normalization. The optimal length of the summary according to MDL is then determined by

$$Lo = \arg\max\left(Inc(S_L, A) - Penalty(S_L)\right) \tag{6}$$

Figure 4 illustrates an example of this procedure. As can be seen in Figure 4, when more keyframes are included in the summary, the content inclusion gets larger. Since the most important keyframes are selected first, the increment of the content inclusion by adding one keyframe actually becomes less and less. At the same time, when the summary length increases, the content inclusion is penalized more. As a result, the score $(Inc(S) - Penalty(S))$ increases at the beginning, and then decreases when the increment of content inclusion is not significant enough to compensate for the penalty due to the additional summary length. Thus, after the optimal length L_o, it is not worth adding more keyframes to the summary.

4.2. Sorting Summary by Video Importance

To generate a single video summary using the selected keyframes from the video article, we need to arrange the keyframes from different videos in an appropriate order. In our video data, there is no other cues such as timeline for video sorting. Thus, we sort the selected keyframes according to the importance of the original videos they belong to. The importance of a video v_i is weighted by $Inc(\Psi_i, A)$ calculated by Equation 4 where $\Psi_i \subseteq \mathcal{F}_i$ is the set of keyframes selected from v_i. We then sort the videos in A in decreasing order of their weights as $\{\hat{v}_1, \hat{v}_2, \hat{v}_3, ..., \hat{v}_n\}$.

For summary composition, the keyframes from the most important video are placed at the beginning of the summary $Sum = \{k_{11}, k_{12}, \cdots, k_{1s_1}, \cdots, k_{n1}, k_{n2}, \cdots, k_{ns_n}\}$ where k_{ij} is selected from $\hat{v}_i$. By this arrangement, the more important videos which cover larger portions of the article content, preserve the most structural information of the storytelling in the video article. Thus, most information of the video article is presented in the same order as in the original video while other complementary content is appended to it.

5. EXPERIMENTS

Our experiments are carried on the videos downloaded from the wikio website [12], where videos are grouped into different articles. Each of them consists of a number of relevant video segments. We downloaded 66 videos that belong to 12 articles for experiments. Each article has 3 to 8 videos.

We compare two different approaches for summarizing video articles based on single- and multi-document summarization respectively. The former first generates a sub-summary for each video. Content inclusion measure in Equation 3 and MDL in Section 4.1 are employed. The sub-summaries are then concatenated as the summary for the article. The latter produces a summary by exploring the relationship among different videos as described in Sections 2-4. By this experiment, we show that multi-document video summarization provides more efficient representation of the given video articles.

Table 1. Comparison between multi- and single-document video summarization.

Article	Video	Single-doc summary		Multi-doc summary	
(Video #)	Length	Length	Inclusion	Length	Inclusion
A (4)	617	8	0.878	4	0.872
B (5)	469	68	0.880	46	0.864
C (8)	2304	226	0.871	137	0.856
D (8)	269	64	0.813	19	0.810
E (4)	3645	37	0.825	18	0.803
F (5)	699	47	0.905	31	0.895
G(6)	5179	202	0.918	163	0.903
H(6)	568	22	0.830	10	0.828
I (5)	408	69	0.886	41	0.865
J (6)	737	104	0.897	67	0.884
K(8)	1717	111	0.856	49	0.847
L (3)	1990	139	0.849	114	0.844
Mean	1550	**91**	**0.867**	**58**	**0.856**

Table 1 presents our experimental results. For each video article, we compare the lengths and the content inclusions (as defined in Section 3.1) of the summaries generated by two approaches. Overall, the average length of the 12 summaries by multi-document summarization is reduced by 36% compared with single-document approach. Meanwhile, the content inclusion is just slightly reduced by 1.3%. This shows that it is necessary to explore the relevance among different videos when summarizing a video article in order to make a more elegant and informative summary. On the other hand, as can be seen in Table 1, for some video articles (G and L), the difference between two approaches is less significant. This is because the videos in these articles are visually different. They are put in the same article because they belong to the same video category (music videos). Since only visual cues are employed in our current approach, the relevancy between these videos cannot be fully explored. To cope with this problem, features other than visual cues are needed.

6. CONCLUSION

We have presented our approach for multi-video summarization. A content inclusion measure is proposed for keyframe selection by exploring the content relevancy and information distribution among different videos. A single summary is then produced, which proves to be more elegant and informative in representing the content of a video article. In our current approach, only visual cue is employed for similarity measure. In the future, other cues such as text, audio, motion, semantic features and category information can be used to deal with different kinds of video articles. Furthermore, we will also extend our work by modelling the relationship among a large number of videos for summarization.

7. REFERENCES

[1] T. Liu, H. J. Zhang, and F. Qi, “A Novel Video Keyframe Extraction Algorithm based on Perceived Motion Energy Model”, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 10, 2003.

[2] Y. F. Ma, L. Lu, H. J. Zhang, and M. Li, “A User Attention Model for Video Summarization”, *ACM Multimedia*, 2002.

[3] Ani Nenkova, “A Compositional Context Sensitive Multi-document Summarizer: Exploring the Factors that Influence Summarization”, *SIGIR*, 2006.

[4] C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, and E. Delp, “Automated Video Program Summarization using Speech Transcripts”, *IEEE Trans. on Multimedia*, vol. 8, no. 4, 2006.

[5] B. Truong and S. Venkatesh, “Video Abstraction: A Systematic Review and Classification”, *ACM Trans. on Multimedia Computing, Communication and Applications*, Feb. 2007.

[6] D. Wang, “Multi-document Summarization via Sentence-based Semantic Analysis and Symmetric Matrix Factorization”, *SIGIR*, 2008.

[7] X. Wu, A. G. Hauptmann, and C. W. Ngo, “Practical Elimination of Near-Duplicates from Web Video Search”, *ACM Multimedia Conf.*, 2007.

[8] X. Wu, C. W. Ngo, and Q. Li, “Threading and Autodocumenting in News Videos”, *IEEE Signal Processing Magazine*, vol. 23, no. 2, 2006.

[9] I. Yahiaoui, B. Merialdo, and B. Huet, “Generating Summaries of Multi-Episode Video”, *IEEE Int. Conf. on Multimedia & Expo.*,2001.

[10] K. Yamasaki, K. Shinoda, and S. Furui, “Automatically Estimating Number of Scenes for Rushes Summarization”, *TRECVID Workshop on Rushes Summarization*, 2008.

[11] W. L. Zhao, C. W. Ngo, H. K. Tan and X. Wu, “Near-Duplicate Keyframe Identification with Interest Point Matching and Pattern Learning”, *IEEE Trans. on Multimedia*, vol. 9, no. 5, pp. 1037-1048, 2007.

[12] Wikio videos. “http://www.wikio.fr/videos”.