

Rushes Redundancy Detection

Reede Ren
University of Glasgow
17 Lilybank Gardens
Glasgow, UK
reede@dcs.gla.ac.uk

P. Punitha
University of Glasgow
17 Lilybank Gardens
Glasgow, UK
punitha@dcs.gla.ac.uk

Joemon Jose
University of Glasgow
17 Lilybank Gardens
Glasgow, UK
jj@dcs.gla.ac.uk

ABSTRACT

The rushes is a collection of raw material videos. Various video redundancies exist, such as rainbow screen, clipboard shot, white/back view, and unnecessary re-takes. This paper develops a set of solutions to identify and remove these video redundancies as well as create a summary video. We consider the manual editing effects, such as clipboard shots, as a differentiator in the visual language. A rushes video is therefore divided into a group of subsequences, each of which stands for a re-take instance. A rough graphic matching algorithm is developed to estimate the similarity between re-take instances. The experiments on the Rushes 2008 collection show that a video can be shortened to 4%-16% of the original size by these redundancy detection solutions. This significantly reduces the complexity in content selection and leads to an effective and efficient video summarisation system.

Categories and Subject Descriptors

H.4 [Information Systems Applications]: Video Summarisation

General Terms

video summarisation

Keywords

manual edit effect detection, re-take detection, attention-based content selection, rushes collection

1. INTRODUCTION

As a raw material collection, the rushes is made up by many re-takes of similar scenes. This indicates that only a very small ratio of a rushes video will appear in a final edited version [1]. Rushes summarisation is therefore a removal of redundant or unnecessary contents in many aspects. In the experiments, we find that it can cut a video to 4%-16% of the original size by removing redundant contents in Rushes 2008 collection. This significantly decreases the complexity

of video content selection and thus eases video abridgement.

The redundancy in the rushes could be roughly categorised into two classes, manual editing effects and unnecessary re-takes. Manual editing effects are short and usually meaningless video segments, such as clap shots, rainbow and black/white screens. These effects are artificially created during the production for later processing purposes, e.g. marking the boundary of a re-take. They can be regarded as *commas* in the visual language. We therefore employ these manual editing effects to separate different re-takes. re-takes are multiple records of a given scene to: (1) provide different camera viewpoints; (2) correct actor's mistakes; or (3) remove some technical failures. This indicates that re-takes of a scene are similar but not identical. In addition, the shot sequence of re-takes varies because of occasional omissions and extra insertions. Moreover, one and only one instance of a re-take could be accepted in the final summary, according to the requirement of video summarisation. This is also a problem, how to identify the best presentation of video contents among multiple re-take instances. In short, the challenges of redundancy removal in the rushes collection are to: (1) detect manual editing effects; (2) identify instances of a re-take; and (3) decide the *best* re-take instance for video content representation.

The remains of this paper are organised as follows. Section 2 describes the framework of our video summarisation system. Section 3 states a cluster algorithm to remove direct repetitions. Algorithms for manual editing effect detection are presented in Section 4, involving three cases: rainbow, clipboard and meaningless views. Section 5 presents the technique of re-take discrimination. A visual similarity measurement is defined to compare visual key frames (Section 5.1). A rough graph matching algorithm is developed to match two similar but not identical re-take instances and therefore decides an optimised content representation (Section 5.2). Based on the work [7], an attention-based content topic selection is introduced by Section 6. Discussion and future work are found in Section 7.

2. SYSTEM FRAMEWORK

The framework of video summarisation system is shown in Figure 1. Based on the prior system of attention-based video abstraction [7], this system improves two new function modules, the adaptive adjustment of replaying speed and the identification of content redundancy. Replay speed adjustment is a part of summary composition, which in-

creases replaying frame rate to reduce the overall size of a summary video. However, this module is only activated in the case of MRS148797.mpg, due to the effectiveness of redundancy identification. As we have mentioned, the removal of redundant contents can reduce most rushes videos to 4-16% of the original size. It is easy to reach the pre-defined 2% duration limitation without changing replaying frame rate, when a proper topic selection algorithm is applied, i.e. choosing video clips with a high attention intensity. The module of content redundancy identification removes manual editing effects, repetitive shots and unnecessary re-takes. Three sub-components are involved, direct repetitive removal(Section 3), manual editing detection (Section 4), and re-take redundancy identification (Section 5).

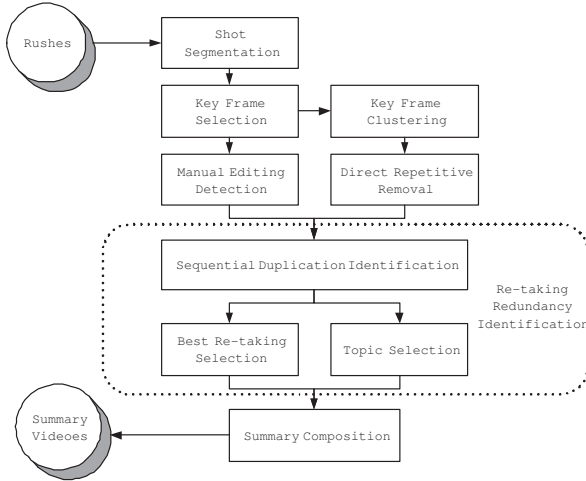


Figure 1: System Framework

The entire video summarisation process can be described as follows. A two-threshold shot segmentation algorithm is employed to allocate shot boundaries [5]. We favour relatively low thresholds in order to extract as many key frames as possible. These key frame are clustered according to visual similarity. Adjacent shots with similar key frames will be removed as direct repetitions. This step will decrease the computational complexity in the later sequential similarity measurement. Meanwhile, a group of salient features, such as average local motion, shot boundary frequency and colour moment, are computed to estimate video attention intensity. Then, manual editing effects are identified to divide a long video into a group of sub-sequences. We suppose each sub-sequence denotes a possible re-take instance. The module of retaking identification computes the sequential similarity between these sub-sequences and therefore removes unnecessary re-take instances. The later topic selection module chooses video clips with a high attention intensity and composes a summary. If the overall size of selected video clips exceeds the given 2% duration limitation, the module of replaying speed adjustment will increase frame rate to shorten summary length.

3. SHOT CLUSTERING

As evident from [4], principal component analysis captures the most relevant features to use in classifying a group of objects to be recognised. Principal component analysis is a

linear method for data feature extraction. It is a mathematical technique used to analyse correlated random variables to reduce the dimensionality of a data set. This reduction is achieved by selecting the first few principal components. The mathematical background of PCA lies in eigen analysis. Thus to achieve good recognition rate, for all N , key frames representing the N shots of a rushes video, $N \times N$ eigen distance matrix, is computed, where each cell entry corresponds to the pairwise eigen distance of the respective key frames. The eigen distance matrix is then used for clustering shots. The clustering algorithm always tries to find the best fit for a fixed number of clusters. Fixing up a static value for the number of clusters, irrespective of the video and the number of shots detected in the video, is not meaningful. This is because, either the number of clusters might be wrong, or the clusters might not correspond to that of the data. There are two main approaches to determine the appropriate number of clusters, compatible cluster merging and validity measures. We use the validity measure to find the appropriate number of clusters for each video. Depending on the number of shots detected for each rushes video, minimum and maximum number of expected clusters is calculated. This minimum and maximum number of clusters reflects to the percentage of coverage of a rushes video. With this initial set up and a combination of the scalar validity measures, separation index, Alternative Dunn’s index [2] and Xie and Beni’s index [9] the most appropriate and optimal number of clusters for each video is found. K-means clustering is then used to categorise the shots into the most optimal number of clusters.

4. MANUAL EDITING DISCRIMINATION

In rushes collection, three types of manual editing effects are observed, namely rainbow, clipboard shot and meaningless low quality view, i.e. black/white screen. These effects should be removed because they contribute few contents. The following sections will discuss related algorithms in details.

4.1 Rainbow

Rainbow is the colour bar screen mostly appearing at the beginning of a video, although some instances are also randomly found in the middle of video documents. Figure 2 shows some examples of rainbow screen.

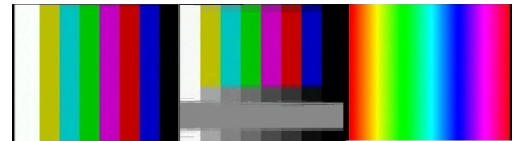


Figure 2: Rainbow Samples

A significant character of rainbow is the high similarity in colour component distribution. This means colour histograms of R,G,B colour channels are almost the same (Figure 3). We therefore compute bin difference between R-G and R-B histograms to create a feature vector for rainbow discrimination. 41 rainbow screens are collected from the Rushes 2007 collection to train a two-class Gaussian mixed model(GMM).

The recall of rainbow detection in the Rushes 2007 collection is 100% with the precision above 87%. Additionally, most false detections are acceptable, which are various meaningless screens, i.e. white/black screen.

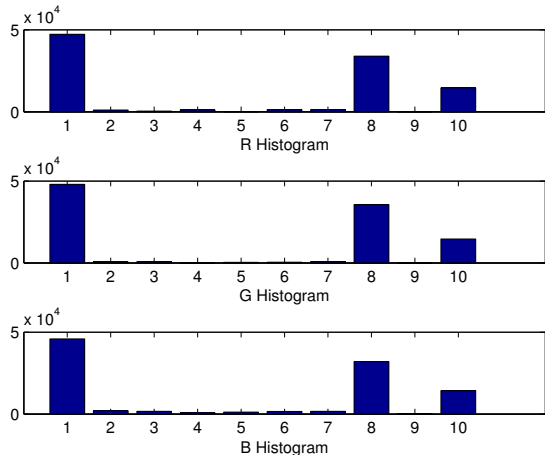


Figure 3: Colour Histograms of A Rainbow

4.2 Clipboard Detection

We categorise clipboard shots into two groups, small board and large board, according to clipboard appearance in visual frames. In the case of small board, a clipboard is displayed entirely and occurs less than one half area in a visual frame (Figure 4). More than 300 video objects of small clipboards are manually cut from visual frames and normalised to the size of 20×15 pixels. A support vector machine detector (SVM) is trained by these samples and more than 1000 randomly cut 20×15 image regions. A four-layer pyramid is created for each visual frame and the SVM detector scans the pyramid from coarse to high resolutions. If a clipboard is found on a given layer, such a scan process will stop and the visual frame is labelled as with clipboard.



Figure 4: Small Clipboard Examples

In the case of large board, only a part of a clipboard is shown and takes most visual area (Fig 5). Three region-based low-level features are computed, black ratio, white ratio, and block variance. We divide a visual frame into a 3×3 region map and qualify the image into 12 gray scales [0, 11]. A pixel with gray intensity zero will be regarded as black and that with 11 as white. Black and white ratio is the rate of black/white pixels in the given region. Block variance is the standard deviation of gray intensity. The feature

vector therefore consists of 27 dimensions. A sample collection is built to train a SVM for large clipboard detection, including 216 large clipboard examples and more than 600 non-clipboard visual frames.



Figure 5: Large Clipboard Examples

A dynamic programming [8] is employed as post-processing to smooth the clipboard/non-clipboard label sequence. Since most clipboard shots includes more than 25 frames (longer than 1 second), we set the length of dynamic steps to 6 and the transmission probability between different states to 0.2. Additionally, a rushes video is sampled at 1/5 for clipboard detection. This means that we extract one from every five visual frames to detect small and large clipboards. Figure 6 show the whole process of clipboard detection.

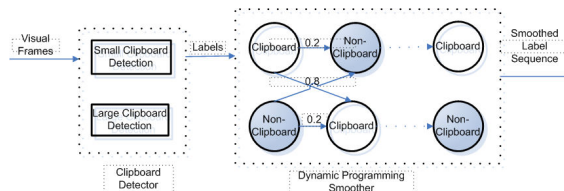


Figure 6: Clipboard Shot Detection

4.3 Meaningless View Discrimination

Meaningless views are low quality visual frames due to technical failures, e.g. over-exposure. Although most meaningless views are black and white screens, many variations are observed in the rushes collection. Some examples are shown in Figure 7. The image quality measurement called local harmonic activity map [3] are therefore employed to detect meaningless views. We will label a visual frame with a low local harmonic activity as a meaningless view.

5. RE-TAKE IDENTIFICATION

After manually effect detection, a long video stream is divided into a group of video segments, each of which usually involves several shots. We assume each of video segments denotes a re-take instance.

Generally, re-take identification clusters video segments with similar content structures. Such a content similarity consists of two aspects: (1) the visual similarity between related key frames; and (2) the sequential similarity between video segments. However, Some shot/key frames are missed between different re-take instances as well as extra shot/key frames are inserted. The sequential similarity could hardly be estimated by a string matching.

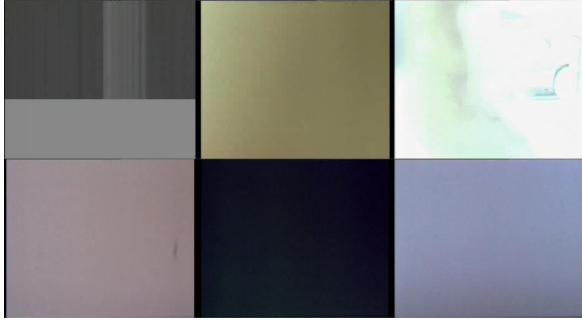


Figure 7: Black/White Samples

5.1 Key Frame Similarity Estimation

We extract one key frame from every shots in a video segment. These key frames are divided into 3×3 region maps as shown in Figure 8. For each region, a colour histogram is computed and a weight is assigned (Figure 8). The similarity between key frames is therefore estimated as a weighted sum of intersection distances between region based colour histograms. In addition, the intersection distance will be one if two histograms are the same and zero for totally different histograms. We normalise such a measurement into $[0, 1]$ throughout a video, which is one when visual frames are the same and zero for entirely different ones.

2	1	2
1	2	1
2	1	2

Figure 8: Weights of Region Map

A two-class Gaussian mixed model is learnt from key frame similarity measurement throughout a video. The threshold for similar vs non-similar decision is therefore computed as the average of Gaussian model means.

An optional step is carried out to reduce the number of video segments. For short video segments with less than five shots, we test whether all key frames of a short video segment could be found in other long video segments. If such a key frame coverage is high, these short video segments will be labelled as redundancy. This step can reduce the computational cost in the following sequential similarity measurement and avoid the problem of multiple overlaying.

5.2 Sequential Matching

Sequential matching decides whether a video segment could be replaced by another video segment. Such a matching process is similar to the comparison between two strings,

because a video segment is represented by a ordered appearance sequence of key frames. However, there are unpredictable omissions and insertions of video shots during production. This indicates: (1) some character may be lost in one of appearance strings; (2) a random number of characters may be inserted between two adjacent characters. The search of a possible match is equivalent to finding a bi-directional pathway in a $N \times M$ directional graph, where N and M are the length of key frame appearance string. The computational complexity will be $O(n^{NM})$. This indicates string matching algorithm is not so effective in the appearance sequence matching. Additionally, there is absent a common video structure in the rushes collection. It is hard to train an effective markov model to adopt these omissions and insertions.

We develop a bi-directional searching algorithm to find possible matches between two video segments. This searching algorithm should decide (1) whether all key frames of a video segment could be found similar frames in another video segment; and (2) whether the appearance order of key frames between two video segments are similar. Such an algorithm involves three steps. Firstly, we computes the frame similarity of all key frames between video segments. A candidate collection is therefore created for each key frame, which lists the appearance order of the top N most similar key frames in the other video segment. Secondly, we search these candidate collections in the order of key frame appearance to assume a possible matching sequence. The appearance will be taken as a possible match, which is the smallest candidate but larger than any other orders already in the matching sequence. If such an appearance order could not be found, we set the related key frame as unique. A similar matching sequence is computed for the other video segment as well. Thirdly, we compare the number of unique key frames and the length of possible matching sequence. If the unique key frame number is larger than one third of matching sequence length, these video segments will be labeled as non-relevance, otherwise relevant. The relevant video segment with more unique key frames will be held as a better re-take instance. The algorithm is presented in Algorithm 1 for the computation of matching sequence.

6. TOPIC SELECTION

The module of topic selection is to abridge selected video segments. We estimate the attention intensity of shots [7] and offer a long presentation period for shots with a high attention score. In this system, we increase the weight of two aspects in attention estimation: (1) the duration of a re-take instance; and (2) audio-visual variations, e.g. strong motion. On one hand, a long re-take instance indicates that ad-hoc directors are satisfied with this sequence when production. On the other hand, a complex visual or audio situation usually provides rich information about video contents.

7. DISCUSSION AND FUTURE WORK

We present our system for the Rushes 2008 collection. The major contribution is to develop an approach for various video redundancy detection, including rainbow, clipboard, low quality view and unnecessary re-takes. These instances covers most redundancy in the rushes collection. In addition, we regard manual editing effects as *comas*. This leads to an efficient rough matching algorithm between key frame

Data: A pair of video segments $\{S_1, S_2\}$, each of which consists of a key frame group
 $K_n = \{k_{in} | k_{in} \in S_n, i = 1 \dots N_n, n \in \{1, 2\}\}$, Q the candidate collection size

Result: The number of unique key frames U_1 and the matching sequence M_{12} ;

```

foreach  $k_i1 \in K_1$  do
  create a candidate collection  $QK_{i1}$ ;
  calculate visual similarity from  $k_i1$  to elements in  $K_2$  ;
  insert the appearance order of the top  $Q^{th}$  element in
   $K_2$  into  $QK_{i1}$  ;
end
 $U_1=0$ ;
for  $i = 1$  to  $N_1$  do
  if  $M_{12}$  is empty then
    insert  $M_{12}$  the minimum element of  $QK_{i1}$ ;
    continue;
  end
  while  $QK_{i1}$  is not empty do
    find the minimum element  $q$  of  $QK_{i1}$ ;
    if  $q$  larger than all elements in  $M_{12}$  then
      add  $q$  to  $M_{12}$ ;
      break;
    end
    remove  $q$  from  $QK_{i1}$ ;
  end
  if  $QK_{i1}$  is empty then
     $U_1++$ ;
  end
end

```

Algorithm 1: Searching Possible Match Sequence

sequences. A re-take detection is therefore successfully created. The evaluation report [6] shows that our solution finds a good balance between the summary duration and content topic coverage. Moreover, a short judgement period indicates these summaries are easy to be understood. This shows the effectiveness of our attention-based topic selection strategy.

re-take detection is a complex task. A few challenges can seriously decrease the algorithm performance of matching sequence search: (1) too many short video segments; (2) incomplete video segments; (3) key frame extraction. We compute the coverage of similar key frame collection to remove short video segments. However, a short video segment is mostly caused by some technique failure and thus consists of some unique frames. This results in a possible low set coverage, given the short video segment duration. Moreover, an incomplete video segment stands for such a case that a re-take instance consists of several video segments. A better content representation may be found if all video segments are merged to create a unified re-take instance. A cross-segment search is therefore required to process incomplete video segments. We leave this for future work. Nevertheless, the selection of key frames plays a prominent role in the video summarisation system. A good collection of key frames can significantly improve the performance of re-take detection, vice versa. This may lead to a fragile system, since it is hard to define what kind of key frame is ideal for video summarisation.

We ignore audio information in this work, because audio clips can hardly be incorporated into a summary video. Additionally, an audio stream in the rushes collection is full of background noise and meaningless speeches during production, such as "ready...camera". It may incur extra complexity to translate audio to text for content matching. However, manual editing effect detection can partially remove these helpless audio clips. This will lead to a possible pathway for the employment of audio information and thus indicate an efficient approach for re-take detection and content topic selection.

8. ACKNOWLEDGMENTS

The research leading to this paper was supported by European Commission under contracts FP6-045032 (Semedia).

9. REFERENCES

- [1] W. Bailer, F. Lee, and G. Thallinger. Detecting and clustering multiple takes of one scene. In *MMM*, pages 80–89, 2008.
- [2] A. Bensaid, L. Hall, J. Bezdek, L. Clarke, M. Silbiger, J. Arrington, and R. Murtagh. Validity-guided (re)clustering with applications to image segmentation. *IEEE Transactions on Fuzzy Systems*, 4(3):112–123, 1996.
- [3] I. P. Gunawan and M. Ghanbari. Reduced-reference picture quality estimation by using local harmonic amplitude information. In *in Proc. London Communications Symposium*, pages 137–140, University College London, UK, September 2003.
- [4] K. Han and A. H. Tewfik. Eigen image based video segmentation and indexing. In *Intl conference on Image Processing*, Santa Barbara, CA, USA, 1997.
- [5] R. Lienhart. Comparisons of automatic shot boundary detection algorithms. In *Proc of SPIE Storage and Retrieval for Image and Video Database*, volume 3656, pages 290–301, 1999.
- [6] P. Over, A. F. Smeaton, and G. Awad. The TRECVID 2008 BBC rushes summarization evaluation, booktitle = TVS '08: Proceedings of the International Workshop on TRECVID Video Summarization, year = 2008, pages = 1–??, location = Vancouver, British Columbia, Canada, publisher = ACM, address = New York, NY, USA,.
- [7] R. Ren, P. Punitha, J. M. Jose, and J. Urban. Attention-based video summarisation in rushes collection. In *TVS '07: Proceedings of the international workshop on TRECVID video summarization*, pages 89–93, New York, NY, USA, 2007. ACM.
- [8] L. Xie, L. Kennedy, S.-F. Chang, A. Divakaran, H. Sun, and C.-Y. Lin. Discovering meaningful multimedia patterns with audio-visual concepts and associated text. In *IEEE International Conference on Image Processing (ICIP)*, Singapore, Oct 2004.
- [9] X. L. Xie and G. A. Beni. Validity measure for fuzzy clustering. *IEEE Trans. PAMI*, 3(8):841–846, 1991.