Automatic Summarization of Rushes Video using Bipartite Graphs[1]

Liang Bai[1,2], Songyang Lao[1], Alan F.Smeaton[2] , Noel E. O'Connor[2]

[1] Sch. of Information System & Management, National Univ. of Defense Technology, ChangSha, 410073, R.P China
lbai@computing.dcu.ie, laosongyang@vip.sina.com
[2]Centre for Digital Video Processing & CLARITY: Centre for Sensor Web Technologies, Dublin City University, Glasnevin, Dublin 9, Ireland
asmeaton@computing.dcu.ie, oconnorn@eeng.dcu.ie

Abstract. In this paper we present a new approach for automatic summarization of rushes video. Our approach is composed of three main steps. First, based on a temporal segmentation, we filter sub-shots with low information content not likely to be useful in a summary. Second, a method using maximal matching in a bipartite graph is adapted to measure similarity between the remaining shots and to minimize inter-shot redundancy by removing repetitive retake shots common in rushes content. Finally, the presence of faces and the motion intensity are characterised in each sub-shot. A measure of how representative the sub-shot is in the context of the overall video is then proposed. Video summaries composed of keyframe slideshows are then generated. In order to evaluate the effectiveness of this approach we re-run the evaluation carried out by the TREC, using the same dataset and evaluation metrics used in the TRECVID video summarization task in 2007 but with our own assessors. Results show that our approach leads to a significant improvement in terms of the fraction of the TRECVID summary ground truth included and is competitive with other approaches in TRECVID 2007.

Keywords: Video summarization, Evaluation

1 Introduction

Decreasing capture and storage costs have led to significant growth in the amount and availability of video content in recent years. One consequence is that video summarization has recently emerged as an active research field. Video summaries provide a condensed version of a full-length video and should include the most important content from within the original video. Summaries can be used in different applications such as browsing and search, TV program editing, and so on. A variety

[1] This work is © Springer Verlag and appears in Semantic Multimedia: Proceedings of the third international conference on semantic and digital media technologies, SAMT, 2008, Koblenz, Germany, December 3-5, 2008, LNCS 5392, pp3-14.

of approaches have been proposed based on redundancy detection [1], frame clustering [2], speech transcripts [3], and multiple information streams [4]. Interest in this area has grown to such an extent that recently the TRECVID global benchmarking initiative initiated a work item on summarization, targeting rushes content i.e. extra video, B-rolls footage, etc. Rushes are captured by professional cameramen during the video production lifecycle. As an unedited version of the final video, they include many useless and redundant shots. Although the structure of the video and threading of the story are not directly available, the rushes are organized based on the traditional shot structure.

In 2007, the National Institute of Standards and Technology (NIST) coordinated an evaluation of automatic video summarization for rushes. This took place as part of the larger video benchmarking activity known as TRECVID. The overall video summarization task, data used, evaluation metrics, etc., are described in [5]. Importantly, in the TRECVID guidelines for rushes summarization, several criteria are used for evaluating the generated summaries, including the fraction of objects and events included by the summary (IN), the ease of understanding the summary (EA), the time needed for subjective judgment (TT, VT), and the compactness of the summary (DU, XD).

For our participation in this task, we proposed a relatively straightforward keyframe-based approach [1]. However, this approach did not perform as well as expected, especially in the IN and EA criteria. The inclusion results placed our approach (mean: 0.38; median: 0.38) among the 5 lowest scoring participants. Our low EA scores (mean: 2.53; median: 2.67) placed us second worst out of 25 participants. This poor performance encouraged us to undertake detailed failure analysis and motivated us to re-analyze the characteristics of rushes videos. This paper reports on the new algorithm that we developed adapted from [1] on the basis of the results of this analysis.

There are two types of redundant information in rushes video. The first is content such as clapperboards, color bars, monochromatic shots and very short shots. This content is not related to the main content of the video and is not of value in a summary. The second type of redundant content is repetition of some shots with near-identical material appearing in the second and subsequent shots. During program production, the same shot is often taken many times. For summarization purposes, retake shots should be detected and only one kept, removing others from the final summary.

Our enhanced approach described in this paper focuses on representative frames selection, useless content removal, retake detection and content filtering and ranking in the selected shots. In order to select representative frames, which represent video content with as much precision as possible, we calculate the difference between consecutive frames based on color features at the pixel level in each shot and use a geometrical method to select representative frames. Although we don't explicitly segment sub-shots, our method for key frame selection guarantees that representative frames in each sub-shot are selected as both the sum of differences and length of the shot are considered. SVM classifiers are trained based on the TRECVID development data to detect color bars and monochromatic frames. Clapperboard clips are removed by an existing method for Near-Duplicate Keyframe (NDK) detection. After filtering the useless content, we reduce the inter-shot redundancy by removing repeated retake-

shots. Maximal matching based on the Hungarian algorithm is then adopted to measure the similarity between retake-shots at the level of key-frames. Finally, we reduce the intra-shot redundancy of the remaining shots in two steps:

1. We remove similar sub-shots by calculating the color similarity between key-frames that represent sub-shots;
2. We detect the important content including the presence of a face and motion intensity to score remaining key-frames and keep the key-frames with higher score according to the time limitation requirements of the final summary.

The key difference between the approach presented in this paper and our original described in [1], is the introduction of maximal matching in a bipartite graph to measure similarity between shots and this is the reason for the significantly improved performance reported in this paper. Figure 1 describes our overall approach to rushes summarization. First, a given rushes video is structured into shots and sub-shots and useless sub-shots are filtered (see Section 2 and Section 3). Then, inter-shot redundancy is reduced by removing repetitive re-take shots (see Section 4). Finally, a measure is proposed to score the presence of faces and motion for intra-shot redundancy removal (see Section 5). We present a summary of our experimental results in Section 6 and some conclusions in Section 7.

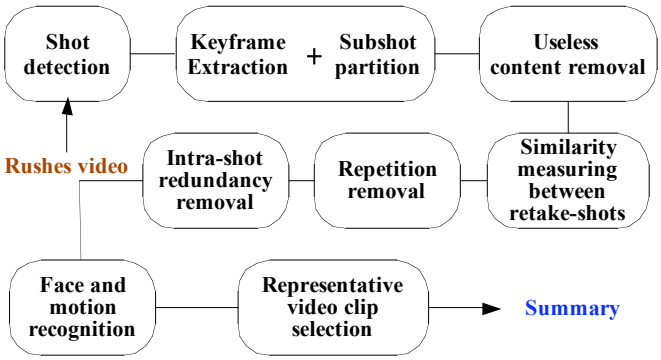


Figure 1: Our approach to rushes video summarization

2 Video Structuring

Given the raw nature of rushes video, we first structure it by detecting shots and sub-shots and extracting key-frames from each sub-shot. Since all rushes videos are unedited, hard cuts typically dominate the transitions used and so we focus only on detection of hard cuts. In our work we use a mutual information measure between two successive frames calculated separately for each RGB channel. The mutual information between two successive frames is calculated separately for each of the R, G and B channels. In the case of the R component, the element $C^{R}_{t,t+1}(i,j), 0 \le i,j \le N-1$, N being the number of gray levels in the image, corresponds to the probability that a pixel with gray level i in frame f_t has gray level j in frame f_{t+1}. The mutual information of frame f_k, f_l for the R component is expressed as:

$$I_{k,l}^{R} = -\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} C_{k,l}^{R}(i,j)\log\frac{C_{k,l}^{R}(i,j)}{C_{k}^{R}(i)C_{l}^{R}(j)}$$

The total mutual information between frames f_k and f_l is defined as:

$$I_{k,l} = I_{k,l}^{R} + I_{k,l}^{G} + I_{k,l}^{B}$$

A smaller value of the mutual information leads to a high probability of a large difference in the content between two frames. Local mutual information mean values on a temporal window W of size N_w for frame f_t are calculated as:

$$\bar{I}_t = \frac{\sum_{i=t}^{N_w+t} I_{i,i+1}}{N_w}$$

The standard deviation of mutual information on the window is calculated as:

$$\sigma_I = \sqrt{\frac{\sum_{i=t}^{N_w+t}\left(I_{i,i+1} - \bar{I}_t\right)^2}{N}}$$

The quantity $\frac{\left|\bar{I}_t - I_{t,t+1}\right|}{\sigma_I}$ is then compared to a threshold H, which represents the mutual information variation at frame f_t deviating from the mean value and determines a boundary frame. Assuming that the video sequence has a length of N frames, the shot boundary determination algorithm may be summarized as follows:

Step 1: calculate the mutual information time series $I_{t,t+1}$ with $0 \le t \le N - N_w$.

Step 2: calculate $\bar{I}_t$ and σ_I at each temporal window in which f_t is the first frame.

Step 3: if $\frac{\left|\bar{I}_t - I_{t,t+1}\right|}{\sigma_I} \ge H$, frame f_t is determined as a shot boundary.

We evaluated the effectiveness of this on the TRECVID development data that is provided for training and familiarization purposes and achieved an overall performance of 93.4% recall and 91.5% precision, which is acceptably close to the state of the art.

In rushes video, each shot usually contains not only the scripted action, but also other material that is not related to the story, such as camera adjustments, discussions between the director and actors, and unintentional camera motion. Further, the scripted action usually contains varied content because of camera and/or object movements. In video summarization, we aim to remove video segments not related to the story and to include only the other video segments. One key-frame for each shot, however, is not enough for this purpose and so we partition each shot into sub-shots corresponding to different content.

We split each frame into an 8x8 pixel grid and calculate the mean and variance of RGB color in each grid. The Euclidean distance is then used to measure the difference between neighboring frames. Usually, in one sub-shot the cumulative frame difference shows gradual change. High curvature points within the curve of the cumulative frame difference are very likely to indicate the sub-shot boundaries. Figure 2 explains this idea. After sub-shot partitioning, the key-frames are selected at the midpoints between two consecutive high curvature points.

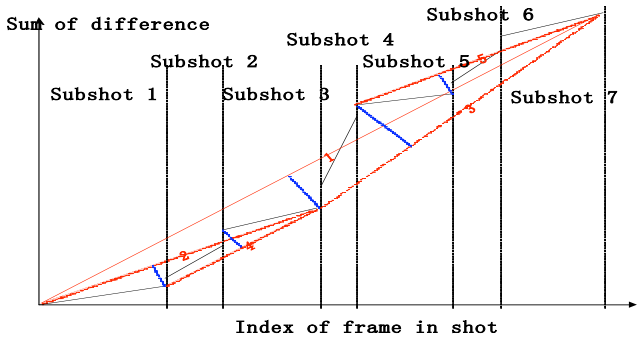
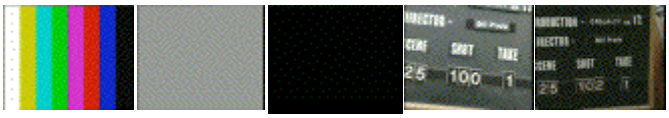


Figure 2: Sub-shots partition

3 Useless Content Removal

Samples of useless content contained in rushes video are illustrated in Figure 3. These include color bars, monochromatic shots, clapperboards and short shots. First, shots of duration less than 1 second are removed. For color bars and monochromatic shots, four features including color layout, scalable color, edge histogram and homogenous texture are extracted from the key-frames in the corresponding shots. SVM classifiers are trained to recognize color bars and monochromatic shots.



Color bars **Homogenous shots** **Clapboards**

Figure 3 Examples of useless content

We employ the algorithm for Near-Duplicate Keyframe (NDK) detection described in [6] to detect clapperboards. A set of 50 example keyframes of the clapboards were extracted from the TRECVID development set. The regions where clapperboards are present are manually annotated. Among the keyframes of each shot in the given rushes video, we detect the keypoints and match them with the example clapperboards. If enough matches are found that lie in the annotated regions, the keyframe is detected as a clapperboard and removed.

4 Re-take Shot Detection and Removal

In rushes video, the same shot can be re-taken many times in order to eliminate actor or filming mistakes. In this case, the re-take shots should be detected and the most satisfactory ones kept, removing the others from the final summarization. Rows 1, 2

and 3 in Figure 4 show the keyframes extracted from three retake-shots in one of the rushes test videos.

We assume that the similarity between shots can be measured according to the similarity of keyframes extracted from corresponding shots. Thus, the re-take shots are detected by modeling the continuity of similar keyframes. Motivated by maximal matching in bipartite graphs, an approach is proposed for similarity detection between video shots based on this matching theory.

Our similarity measure between video shots is divided into two phases: key frame similarity and shot similarity. In the key frame similarity component, a video shot is partitioned into several sub-shots and one key frame is extracted from each sub-shot. The similarity among sub-shots is used instead of the similarity between corresponding key frames. Key frame similarity is measured according to the spatial color histogram and texture features.

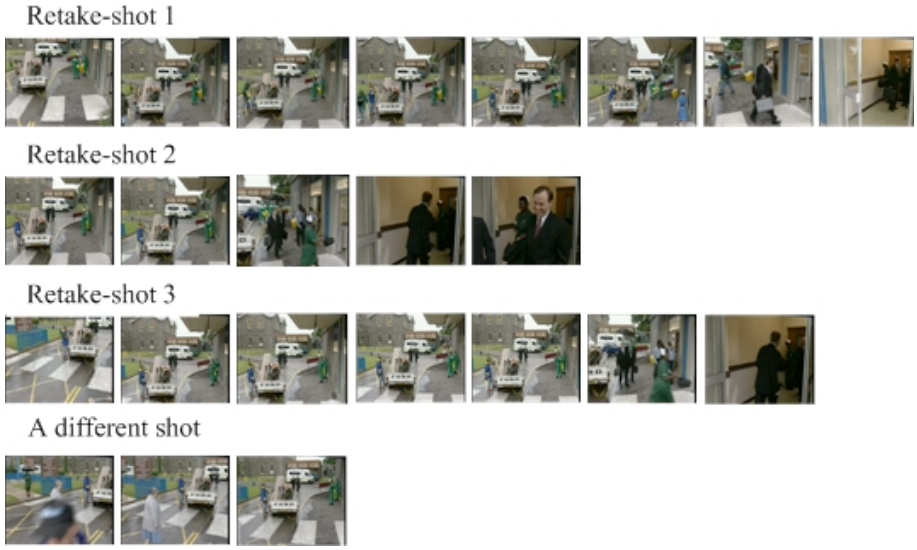
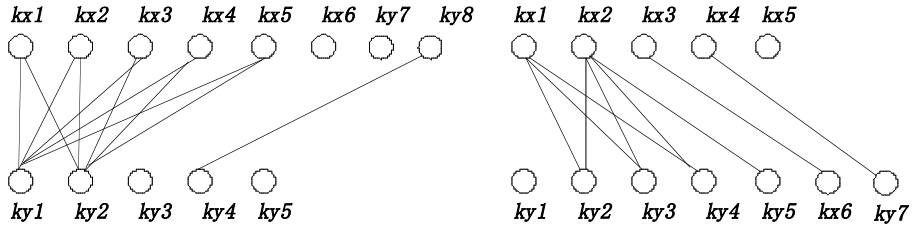


Figure 4: Examples of retake-shots

A shot can be expressed as: $S = \{k_1, k_2, ..., k_n\}$, where k_i represents the i^{th} keyframe. So, for two shots, $Sx = \{kx_1, kx_2, ..., kx_n\}$ and $Sy = \{ky_1, ky_2, ..., ky_m\}$, the similar keyframes between Sx and Sy can be expressed by a bipartite graph $G = \{Sx, Sy, E\}$, where $V = Sx \cup Sy$, $E = \{e_{ij}\}$, e_{ij} indicates kx_i is similar to ky_j. Figure 5 illustrates two examples of bipartite graphs for retake-shot 1, retake-shot 2 and retake-shot 3 shown in Figure 4.



retake-shot 1 and retake-shot 2 **Retake-shot 2 and retake-shot 3**

Figure 5: Two examples of bipartite graph for retake-shots

Clearly, there exist many similar pairs of keyframes between two retake-shots. But in our experiments we also find there often exist similar keyframes in one retake-shot. This results in one to many, many to one and many to many relations in a bipartite graph. In this case, there will be many similar keyframes pairs found between two dissimilar shots. The bipartite graph between retake-shot 3 and a different shot shown in Figure 4 illustrates this case in Figure 6.

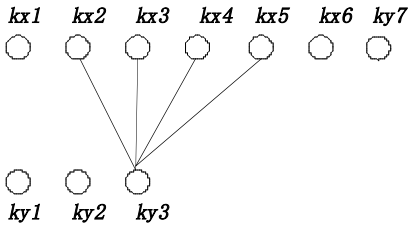


Figure 6 A bipartite graph between two dissimilar shots

If we use the number of similar keyframe pairs to determine the retake-shots, 4 similar keyframe pairs are found in the *Sx* shot shown in Figure 6 and this exceeds half of the keyframes in *Sx*. In this case, *Sx* is likely to be determined as similar to *Sy*, whilst this is not the case in practice.

In our approach, the similarity between two shots is measured by the maximal matching of similar keyframes in the bipartite graph model. The Hungarian algorithm [7] is used to calculate maxima matching M, $M \subseteq E$. If $M \geq \min\{\left\lceil \frac{2}{3}n \right\rceil, \left\lceil \frac{2}{3}m \right\rceil\}$ where n,m are the number of keyframes in these two shots. These thresholds were chosen based on experimental results in order to give the best similarity matches. Figure 7 shows the maximal matching results of the examples shown in Figure 5 and Figure 6.

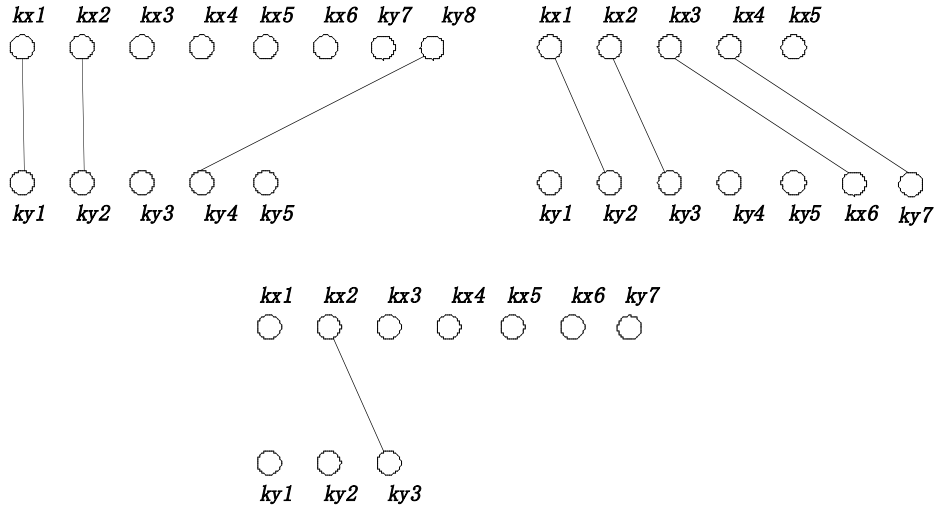


Figure 7: Examples of maximal matching results

From Figure 7, we can find the maximal matching of dissimilar shots is 1. From this, it should be clear that it is relatively straightforward to determine the true retake-shots according to the maximal matching.

The matching steps using the Hungarian algorithm are as follows:

Assumption: A given bipartite graph is $G_k = \{Sx, Sy_k, E_k\}$; "0" denotes a vertex that is not searched, "1" denotes a saturation vertex and "2" denotes a vertex that cannot increase the matching.

Step1: Given an initial matching *M*, mark the vertexes "1";

Step2: Check if every vertex in *Sx* has a non-"0" mark.

If yes, *M* is the maximal matching. End.

If no, find a vertex marked "0" $x_0 \in Sx$, let $A \leftarrow \{x_0\}, B \leftarrow \phi$.

Step3: Check if $N(A) = B(N(A) \subseteq Sy_k)$. $N(A)$ denotes the vertices belonging to Sy_k that neighbor with the vertices in *A*. $B(N(A) \subseteq Sy_k)$ denotes the vertices belonging to Sx_k that neighbor the vertices in $N(A)$.

If yes, x_0 cannot increase matching, mark x_0 "2", go to Step2;

If no, find a vertex Sy_i in $N(A) - B$, check if Sy_i is marked with "1".

If yes, there exists an edge $(Sy_i, z) \in M$, let $A \leftarrow A \cup \{z\}, B \leftarrow B \cup \{Sy_i\}$, go to Step3.

If no, exist an augmenting path from x_0 to Sy_i, let $M \leftarrow M \oplus P$, mark x_0 and Sy_i "1", go to Step2.

The complexity of this algorithm is *O(ne)*, where *n* is the number of vertices of *Sx* in the bipartite graph $G = \{Sx, Sy, E\}$ and *e* is the number of edges. After measuring the similarity of shots, re-take shots are detected. The last shot is retained and the others are removed since in rushes content the last retake shot is usually the one deemed most satisfactory.

5 Selecting Representative Shots and Summary Generation

After low value content and repetitive re-take shot removal, useful content is kept for summary generation. However the volume of the remaining content still typically exceeds the useful duration limit set by the TRECVID guideline — set at 4% duration of the original video in 2007. So, the most representative clips need to be selected to generate the final summary. In our work, we extract motion and face factors to rank how representative each remaining sub-shot is in the context of the overall video.

A three-stage process, achieved using the aceToolbox [8], is used to describe the level of motion activity in each sub-shot. First, MPEG-1 motion vector data is extracted from the video. Next, the percentage of non-zero blocks in the frame (where a high percentage indicates higher motion activity) is calculated for each frame in the video. Finally, this per-frame data is used along with the shot-boundary data calculated previously to compute an average motion measure for the entire sub-shot. As a result, each keyframe in a given sub-shot is assigned the same measure of motion activity.

Our face detection processing extends the Bayesian Discriminating Feature (BDF) originally proposed by Liu [9] for detecting frontal faces in grayscale images. Using a statistical skin color model [10], we can detect multiple faces at various sizes and orientations within color images. Ideally this processing would be carried out for each frame of the original footage, however, for efficiency we only perform this operation on the detected keyframes. While this potentially results in the loss of information, such as the prevalence of faces across shots, it ensures efficient processing while still providing enough information to reliably enhance summary construction.

Sub-shot duration is important for sub-shot selection so we use simple weighting to combine the factors.

$$\begin{aligned} Score = & \left(Number-of-faces / Maximum-faces-in-Footage \times 0.3\right) \\ & + \left(Amount-of-motion \times 0.3\right) \\ & + \left(Duration-of-subshot / Total-Duration-of-All \times 0.4\right) \end{aligned}$$

These weightings for the different components were chosen as discussed previously in [1]. Once the representative scores for sub-shots are calculated, the sub-shots with highest scores are selected according to the summary duration limitation. Finally, 1-second clips centred around the keyframe in each selected sub-shot are extracted for generating our final summary.

6 Experiments Results

Using our approach, we generated the summaries for all test rushes videos. The seven criteria set by the TRECVID guidelines for summarization evaluation are:

- EA: Easy to understand: (1 strongly disagree - 5 strongly agree);
- RE: Little duplicate video: (1 strongly disagree - 5 strong agree);
- IN: Fraction of inclusions found in the summary (0 - 1);
- DU: Duration of the summary (sec);
- XD: Difference between target and actual summary size (sec);
- TT: Total time spent judging the inclusions (sec);
- VT: Video play time (vs. pause) to judge the inclusions (sec).

IN, DU and XD are objective criteria that we can calculate directly from the TRECVID groundtruth to evaluate our summaries. However, EA, RE, TT and VT are criteria that depend on subjective judgments. Thus for a complete evaluation of our proposed approach it was necessary to re-run the evaluation performed by NIST with our own test subjects. Ten participants (all students in the School of Information System & Management, National University of Defense Technology) were selected to review the summaries under the exact same guidelines as provided by NIST and give their score for the four subjective criteria.

Of course, by running our own evaluation we could potentially introduce new subjective variations into the evaluation process. To investigate this, we first evaluated three sets of results: the two TRECVID baselines (see [5] for details) and our own original submission. The experimental results we obtained are compared to the official results reported from TRECVID in Table 1.

The results in Table 1 show that there exists a small difference in the subjective judgments between our participants and NIST assessors. This is understandable given that different people have different skills, intellects, powers of discernment, etc. However, from Table 1 we can see that the difference of judgments between our participants and NIST assessors is small. From this we conclude that our participants' evaluations on the subjective criteria are reasonable and credible. Given this, we proceeded to re-run the complete evaluation.

Table 1: Experimental results for the comparison between our participants and NIST assessors

Criterion		EA	RE	TT	VT
TrecBaseline1	Our Participants	3.12	3.26	115.45	73.20
	NIST	3.33	3.33	110.67	66.67
TrecBaseline2	Our Participants	3.35	3.30	118.10	70.38
	NIST	3.67	3.67	109.17	63.83
Our original [2]	Our Participants	2.29	3.33	76.78	48.49
	NIST	2.67	3.67	70.83	42.67

The experimental results for all of our summaries are shown in Table 2 and Table 3. The results in Table 2 show that our enhanced approach results in a big improvement in IN (0.40) with a slightly longer duration of summaries (0.71 sec) compared with our original approach. Of particular note is the fact that our enhanced approach's XD is 18.83, which is 8.5 sec longer than the mean of the other 22 teams.

This is because we tend to retain the valuable content in rushes as much as possible within the summary duration constraint. Table 3 shows the evaluation results for the four subjective criteria. Clearly we obtain very encouraging results for the EA and RE. These experimental results clearly show that our enhanced approach performs competitively compared with the other teams and the baselines.

Table 2: Experiment results for IN, DU and XU

Criterion	IN	DU	XD
TRECVID Baseline1	0.60	66.40	-2.28
TRECVID Baseline2	0.62	64.60	-0.89
Mean of all 22 teams	0.48	49.54	10.33
Our original [2]	0.38	40.90	8.65
Our enhanced	**0.78**	**41.61**	**18.83**

Table 3 Experiment results for EA, RE, TT and VT

Criterion	EA	RE	TT	VT
TRECVID Baseline1	3.12	3.26	115.45	73.20
TRECVID Baseline2	3.35	3.30	118.10	70.38
Our original [2]	2.29	3.33	76.78	48.49
Our enhanced	**3.74**	**3.88**	**89.21**	**44.50**

7 Conclusion and Discussion

This paper describes our approach to summarizing rushes video content. It focuses on the adaptation of an approach we used in the TRECVID summarization task. In this approach, we employ shot and sub-shot detections for video structuring and we train SVMs for removing useless content. We model the similarity of keyframes between two shots by bipartite graphs and we measure shot similarity by maximal matching for re-take shot detection. Based on consideration of motion, face and duration, sub-shots are ranked and the most representative clips are selected for inclusion in the final summary. This key different with respect to our original TRECVID submission is the inclusion of bipartite matching. To evaluate this new approach, we re-ran the evaluation procedure ourselves with our own assessors. Experimental results indicate that the subjective evaluation is in line with that originally carried out by NIST. Our improved approach clearly demonstrates improvements compared to our original approach, but more importantly compared to the TRECVID baselines and the other teams who participated.

Not withstanding this, the summarization problem clearly still remains challenging. Indeed, most submissions cannot significantly outperform the two baselines, which are simply based on fixed-length shot selection and visual clustering. This poses the key question as to whether a deeper semantic understanding of the content can help in this regard.

Acknowledgments. This work is supported by the National High Technology Development 863 Program of China (2006AA01Z316), the National Natural Science Foundation of China (60572137) and Science Foundation Ireland through grant numbers 03/IN.3/I361 and 07/CE/I1147.

References

1. Byrne D, Kehoe P, Lee H, O Conaire C, Smeaton A.F, O'Connor N and Jones G. A User-Centered Approach to Rushes Summarisation Via Highlight-Detected Keyframes In Proceedings of the TRECVID Workshop on Video Summarization (TVS'07), Augsburg, Germany, September 28, 2007, ACM Press, New York, NY, 2007, pp.35-39.
2. A.M. Ferman and A. M. Tekalp, Two-stage hierarchical video summary extraction to match low-level user browsing preferences", IEEE Trans. Multimedia, 5(2), 244-256, 2003.
3. Y. F. Ma, L. Lu, H. J. Zhang, and M. Li, A User Attention Model for Video Summarization. In Proc ACM Multimedia Conference, 2002.
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, pp.775-791, 2006.
5. Over, P., Smeaton, A.F. and Kelly, P. The TRECVID 2007 BBC rushes summarization evaluation pilot. In Proceedings of the TRECVID Workshop on Video Summarization (TVS'07), Augsburg, Germany, September 28, 2007, ACM Press, New York, NY, 2007, pp.1-15.
6. C. W. Ngo, W. L. Zhao, and Y. G. Jiang, Fast Tracking of Near-Duplicate Keyframes in Broadcast Domain with Transitivity Propagation, In Proc. ACM Multimedia Conference, Oct 2006.
7. Dai YQ, Hu GZ, Chen W. Graph Theory and Algebra Structure. Beijing: Tsinghua University Press, 1995, 89-91 (in Chinese).
8. O'Connor, N., Cooke, E., le Borgne, H., Blighe, M. and Adamek, T. The AceToolbox: Low-Level Audiovisual Feature Extraction for Retrieval and Classification, In Proceedings 2nd IEE European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies, 2005, London, U.K.
9. Liu, C. A Bayesian discriminating features method for face detection. IEEE Tran. on PAMI, 25:725-740, June 2003.
10. Cooray, S. and O'Connor, N. A Hybrid Technique for Face Detection in Color Images. In IEEE Conf. on Advanced Video Surveillance (AVSS'05), Italy, Sept 15-16, 2005.