A Simplified Approach to Rushes Summarization

Francine Chen
FX Palo Alto Laboratory
3400 Hillview Ave, Bldg 4
Palo Alto, CA USA
chen@fxpal.com

John Adcock
FX Palo Alto Laboratory
3400 Hillview Ave, Bldg 4
Palo Alto, CA USA
adcock@fxpal.com

Matthew Cooper
FX Palo Alto Laboratory
3400 Hillview Ave, Bldg 4
Palo Alto, CA USA
cooper@fxpal.com

ABSTRACT

In this paper we describe methods for video summarization in the context of the TRECVID 2008 BBC Rushes Summarization task. Color, motion, and audio features are used to segment, filter, and cluster the video. We experiment with varying the segment similarity measure to improve the joint clustering of segments with and without camera motion. Compared to our previous effort for TRECVID 2007 we have reduced the complexity of the summarization process as well as the visual complexity of the summaries themselves. We find our objective (inclusion) performance to be competitive with systems exhibiting similar subjective performance.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Abstracting methods*

General Terms

Algorithms, Experimentation, Performance

Keywords

video summarization, clustering, segmentation, presentation

1. INTRODUCTION

Video cameras have become ubiquitous as they are increasingly embedded in devices such as cell phones and digital still cameras. The explosion of user-generated video on the web bears testament to the fact that it has become easy for people to create and share video media. This increasing body of publicly shared video, which is largely unedited and unstructured, can be tedious and time consuming to search.

NIST has organized a track in TRECVID where research groups develop systems for summarizing unedited BBC footage. The TRECVID Rushes task and data are described in detail by Over et al. [9, 10]. In this paper, we describe the system we developed for the Rushes Summarization task.

TVS'08, October 31, 2008, Vancouver, British Columbia, CA.

Our system selects short excerpts of video, trying to identify non-redundant segments containing action. The action may be due to objects moving in the video or the camera panning or zooming across a scene. The system also attempts to eliminate uninteresting segments, including colorbars, clapboards, excessive motion, and inadvertent camera obstructions.

For our system this year, we simplified the approach we used in last year's system [2]. We kept the distinction between "dynamic" segments in which there is detectable camera motion from "static" segments where the camera is relatively stationary. Rather than treating the two types of segments differently and emphasizing the dynamic segments in our summaries as we did last year, we opted to process the two segment types similarly. This required devising a similarity measure which yields meaningful comparisons between the two different types of segments so that a single clustering step could identify redundant shots.

We submitted two variants of our system. One version is a simple baseline version that clusters the segments using a distance measure based on the average color histogram of frames within a segment. After clustering, a summary is built by randomly choosing a segment from each cluster and excerpting a fixed length clip from the middle of that segment. Our other submission tries to better capture the variability occurring within shots due to systematic camera motion, such as pans and zooms. The selection of a representative segment from each cluster favors those that both exhibit motion and are different from other selected segments.

This year we also used the metadata provided by NHK Science & Technical Research Laboratories to TRECVID Rushes participants. The metadata identifies the occurrences of up to 12 of the video event segments used for the 2007 Rushes evaluation. It also includes the time of clapboards, color bars, and plain black, white, or gray segments. We used this metadata to help debug and informally evaluate our system in identifying inclusions (events), clapboards, and junk frames.

2. SYSTEM DESCRIPTION

Figure 1 provides an overview of our video summarization system. Three types of features are computed from the decoded frames: color, image motion, and audio. Audio features are used to identify clapboard appearances. Colorbars, bluescreens, and other "junk" are identified from the color and motion features. The identified undesired "junk" and clapboard regions are filtered out from the independent

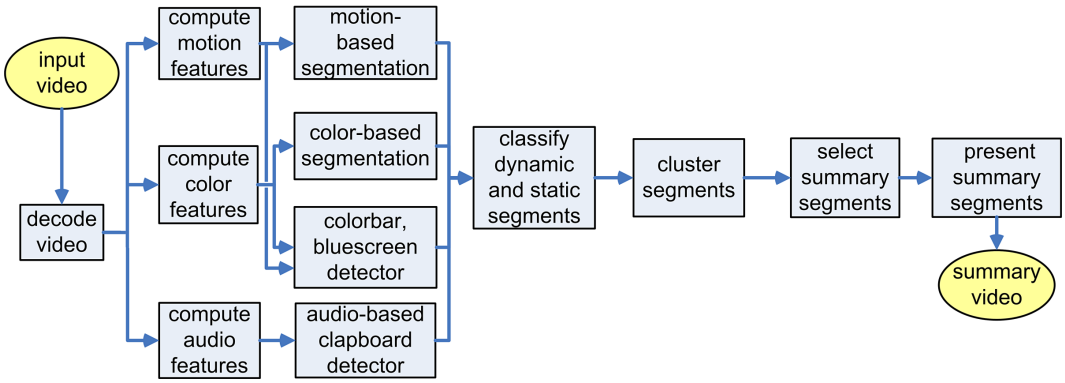

Figure 1: Block diagram of the summarization system.

segmentations estimated from color and motion features to yield a group of candidate segments which are clustered to identify similar segments. One excerpt is selected from each cluster, and the excerpts are concatenated to create a summary.

2.1 Segmentation

Segmentations based on color and motion are computed separately. The color-based shot segmentation uses YUV color histograms and the inter-frame self-similarity approach of [3]. The motion analysis uses the Lucas-Kanade [7, 1] point-based tracking functions included in the OpenCV [8, 1] image processing system. Colorbars and monochromatic "junk" frames are identified by finding regions where the color histogram has low entropy and there are few trackable feature points.

Horizontal pans, vertical pans, and zooms are identified in a separate step by analysis of the global motion statistics. The motion of feature points tracked between adjacent frames is filtered and then averaged to form per-frame estimates of global horizontal, vertical, and radial motion. These global estimates are then smoothed in time.

To detect systematic camera motion (pans and zooms) we compute a running average over a window of 200 frames (8 seconds) without overlap, producing the motion running average, $a[i]$, where i is the window index. Then for each overlapping window of 11 frames over $a[i]$, the 3^{rd} largest value of $a[i]$ within the window is selected. The overlapping windows over $a[i]$ are shifted by 1, creating a vector of "minimum running averages." The "minimum running standard deviation" is computed similarly. A weighted combination of the minimum running average and the minimum running standard deviation within a window is used to determine a "running" threshold for detecting pans and zooms within the window from the global motion estimates. This adaptive threshold helps in cases where the camera is shaky, while reducing the likelihood of missing camera movement by averaging over a very large window. Separate thresholds are computed for identifying horizontal, vertical, and radial motion. The horizontal and vertical motion segments are combined and labeled as pans.

2.2 Segment Classification

The system next removes junk segments, such as colorbars and bluescreens, from further consideration. It is also desirable to remove clapboards. Our audio-based clap detector [2] indicates when a clapboard sounded, but not the extent over which the clapboard is visible in the frame. When an audio clap is identified, we wish to remove not only the clap, but the entire duration in which a clapboard is present. Rather than trying to identify or track the clapboard visually, we chose to remove a fixed amount of video surrounding the time of the audio clap. For each detected clap we removed the 6 seconds before and the 2 seconds after the clap event, but stopping the window at a segment boundary if one occurs within that window. The duration of video removed before and after a clap was determined using the NHK metadata to empirically reduce the clapboard detection miss rate and false alarm rate.

After filtering junk (color bars and the like) video, the color-based and motion-based segmentations are used to identify two types of segments: those containing camera motion, or *dynamic segments*, and those segments where the camera is relatively steady, or *static segments*. The dynamic segments identified in the motion-based segmentation described above are filtered to remove those that are not suitable for inclusion in a summary. These include segments that are very short (we required a 20 frame minimum to be kept), and those with excessive motion. Everything remaining after identifying and filtering the dynamic segments is considered a static segment. The color-based shot segmentation is then used to further subdivide the dynamic and static segments.

We observed that the static segments tend to exhibit greater similarity to each other, presumably because the background is relatively stable, in contrast to the dynamic segments. We differentiate between dynamic segments and static segments so that in the clustering step, we can investigate the use of a similarity measure that incorporates the idea that matching against dynamic segments is better performed when the dynamic segment is represented by more than a single average feature value because the frames in dynamic segments can vary greatly.

3. SUMMARY CONSTRUCTION

This year, we submitted two systems which vary in the measures used for clip similarity for clustering and the criteria for selecting representative clips from clusters.

3.1 Baseline System

To identify similar segments, clustering is performed on the candidate dynamic and static segments. We compute

the mean block-histogram feature for each candidate segment and perform hierarchical agglomerative single-link clustering using the Euclidean distance measure over this feature. The tree is truncated at the level with the same number of leaves as the desired number of summary excerpts. Given the short maximum allowable duration of 2% of the original video length, we decided to extract as many one second excerpts as allowable for each summary; thus the number of leaves equals the number of seconds allowed in the summary.

The tree determines the cluster membership for each candidate segment. From each cluster, we randomly select a segment, and excerpt the central 1 second interval for inclusion in the summary.

3.2 Enhanced System

In the second, "enhanced", summarization system, we tried to address the observation that much more change occurs in the dynamic segments. Before computing the similarity between segments, we sampled frames from the dynamic segments. A sampling rate of one frame in every ten was used. Since computing the similarity of all the sample frames between two segments is expensive, we chose not to sample the static segments. Instead, the static segments were represented as a mean block histogram and thus were only one "frame" long. We had also experimented with computing a mean block histogram feature in non-overlapping windows to represent each dynamic segment, but observed better results when a histogram of a single frame was used.

In TrecVid 2007, a variety of methods was used to identify similar shots. Kleban et al. [6] used dynamic programming to match frames and used the score as a feature. In contrast, Detyniecki and Marsala [4] considered two shots to be the same if the distance between the beginning frames or the distance between the ending frames was less than a threshold. This does not allow for a shot being a subshot that occurs in the middle of another shot. We also avoid the use of a threshold as used in [4] or learning weights for using the distance as a feature as used in [6]. Like Hauptmann et al. [5], our method is based on clustering. In [5] their shot boundary detector computed a very fine segmentation and then compared one keyframe from each shot.

In contrast, we compute the similarity between two segments using sampled frames to represent a shot but ignoring the frames' ordering. The same method is used whether the comparison is between two static segments, two dynamic segments, or one dynamic and one static segment. We assume that at least part of the shorter segment, s_s, is a subset of the longer segment, s_l. For each frame f_s of the shortest segment, the best similarity of that frame, e.g., for frame i, $f_x(i)$, against all frames of the longer segment is computed:

$$sim(f_s(i), f_l) = \max_{j=1,\ldots,L} sim(f_s(i), f_l(j))$$

where there are L frames in the longer segment. Normalized cosine was used as the similarity measure.

The overall similarity between two segments f_s and f_l, $sim(f_s, f_l)$, is computed as the average of the N best similarities, where $N = \min(5, l_s)$, where l_s is the number of frames in the short segment. Thus the overall similarity is computed as:

$$sim(f_s, f_l) = \frac{1}{N} \sum_{k=1,N} sim(f_s(b(k)), f_l)$$

Figure 2: Video summary indicating the position of summary segments within the original video (light gray), length of the original video, and time and position (green) of the currently playing segment.

where $b(k)$ is the index of the frame with the k^{th} best similarity.

This inter-segment similarity was then used for hierarchical agglomerative clustering as in the baseline system, again with the number of clusters set to the desired number of excerpts. In a second enhancement, we select segments to represent each cluster using a discriminative approach. For this, we compute a ranking of each segment within a cluster that combines the average similarity to segments within the cluster and the average dissimilarity to segments already selected for inclusion in the summary:

$$s^* = \operatorname*{argmax}_s \left(\frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} sim(s, c) - \frac{1}{|\mathcal{S}|} \sum_{c \in \mathcal{S}} sim(s, c) \right) .$$

In the above, we denote the cluster to be excerpted by $\mathcal{C}$ and the set of previously selected segments by $\mathcal{S}$. Thus high ranking segments are both good cluster representatives as well as distinct from other excerpted segments in the video. One second excerpts are combined to form the summary as before.

4. SUMMARY RENDERING

The selected one second segment excerpts are ordered by the start time of the earliest segment in the cluster to which each selected segment belongs, which we hypothesized would make it easier for the evaluators to match the shot against a list of shots.

The summary video is rendered with a 5 frame (.2 second) overlapped fade transition between the summary clips which we judged to be more pleasing than cuts or the fade-through-blue transitions which we used last year. We informally observed that the fade transition between clips reduces the sense of repetition when nearly-identical clips are shown in sequence. The original audio for each clip is used in the summary. Visual cues are overlayed on the frame to provide information to the viewer about the context of the summary segments. This is shown in Figure 2. The time of the currently playing segment within the original video is shown alongside the total length of the original video. A timeline representing the original video is also shown with shading marking the portions of the original video which are included in the summary. The currently playing segment is

System	inclusion fraction	tempo rating	junk rating	redundancy rating
mean				
baseline	0.46	3.30	3.41	3.17
enhanced	0.48	3.24	3.26	3.22
median				
baseline	0.44	3.33	3.67	3.00
enhanced	0.47	3.33	3.33	3.33

Table 1: Table of means and medians for inclusions and subjective measures for the baseline and enhanced system.

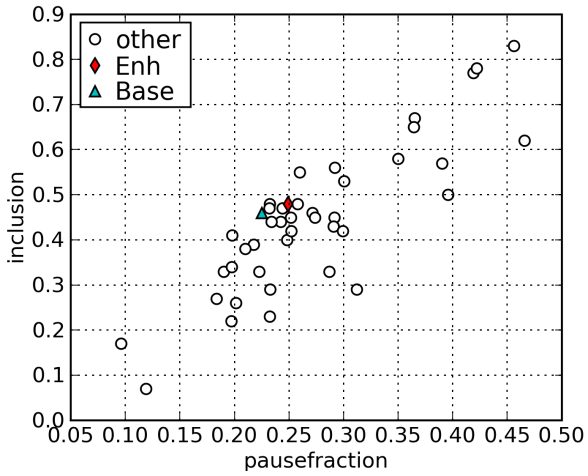

Figure 3: Scatter plot showing the strong correlation between inclusion fraction and the fraction of viewing time spent paused. Each point corresponds to a submission.

highlighted with a different color on the same timeline. In previous versions of our system [2] we provided more detailed feedback about the playback speed and the length and position within the current summary clip. Since we opted for a simpler fixed playback speed and excerpt length that extra visual overlay was deemed redundant and distracting and we eliminated it.

5. RESULTS

As described in [9, 10], the principal objective measure of evaluation for the summaries is the fraction of events from the original video also identified in the summary. A variety of subjective measures are also tabulated based on assessor feedback. This section summarizes the comparative results between our two submitted systems and the performance relative to the full set of submissions.

5.1 Comparative Performance

Table 1 shows the mean and median performance of our baseline and enhanced systems on the inclusions measure and the 3 subjective measures. This shows very small differences with a slight edge in inclusion rate to the enhanced system. The 2% margin in mean inclusion corresponds roughly to a difference of 9 events over the 461 events in the 39 test videos. On the subjective measures the enhanced system is rated marginally worse for junk and marginally better for redundancy. The redundancy outcome is expected given the

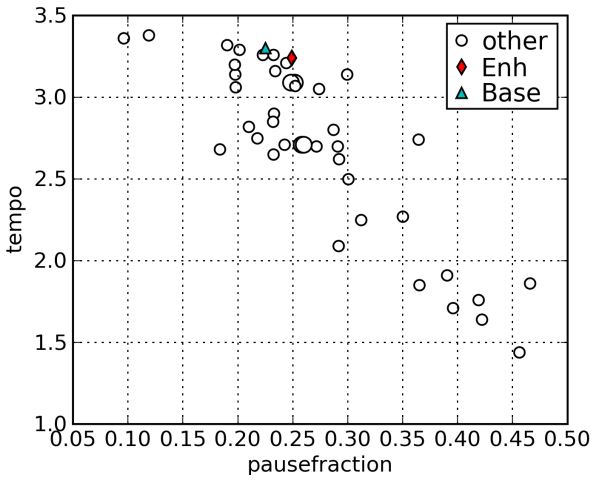

Figure 4: Scatter plot showing the strong correlation between fraction of time spent paused and tempo rating (higher is better). In general summaries rated positively for tempo (high values) had less evaluator pausing. Larger ◯ marks indicate multiple samples at the same position.

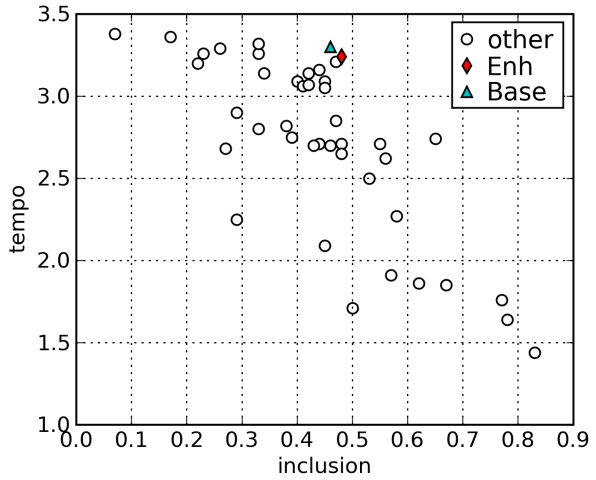

Figure 5: Scatter plot showing the strong correlation between inclusion fraction and tempo. Among the submissions rated highest for tempo ours are among the highest inclusion rates.

emphasis on diversification in the segment selection step of the enhanced system. The negative result on junk rating can also rationally be expected as we have seen that "junky" video is often quite visually distinctive from the rest. Per-video analysis indicates that the chance of an improvement or degradation of any measure on any given video was nearly equal and under a randomization test none of these differences rises to a .05 level of statistical significance.

5.2 Global Trends

Examination of the collective results shows some strong relationships between the objective performance and subjective performance measures.

Figure 3 shows a scatter plot of the fraction of mean viewing time spent paused versus mean inclusion rate. This

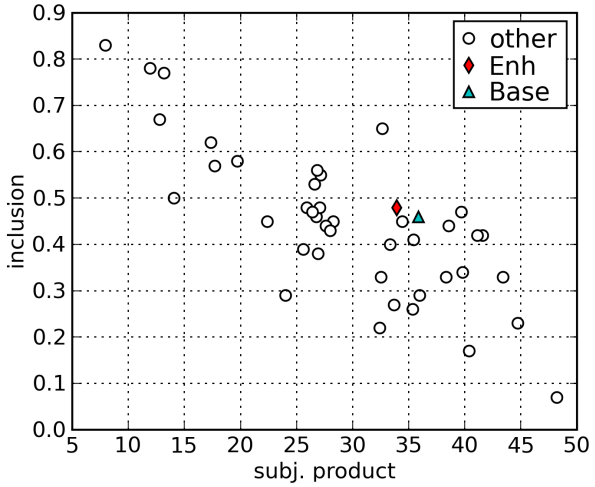

Figure 6: Scatter plot showing a correlation between inclusion fraction and the product of the 3 subjective measures (tempo, junk, redundancy). Once again the tradeoff between high subjective ratings and inclusions is evident. Our systems' inclusion rates are among the highest for their position on the subjective scale.

shows a very strong relationship between time spent paused and fraction of inclusions identified. A strong negative correlation between time spent paused and tempo rating is shown in Figure 4. That is, summaries with better tempo ratings had less pausing by the assessor. This is a somewhat intuitive relationship. The combination of the relationships shown in Figure 3 and Figure 4 suggests a negative relationship between tempo and inclusion and Figure 5 confirms this. Among the systems rated highest for tempo we fall among the best inclusion rates. Finally Figure 6 shows the relationship between inclusion and the product[1] of the subjective scores. The tradeoff between measured objective and subjective performance is evident with only a couple of outliers bucking the trend.

It is tempting to suggest that summaries with higher subjective ratings (tempo in particular) were more efficiently viewed by assessors, resulting in the correspondingly lower pause times in Figure 4, and that objective performance is the price one pays for building a system which scores well on the subjective measures. One alternative interpretation is that the summaries with good tempo ratings received those ratings at least in part *because* they had correspondingly poor inclusion rates, and that increased pausing is a natural consequence of higher inclusion as the assessor pauses the summary video while noting the presence of an event. It is unclear from this evaluation if the perception of the subjective dimensions, such as tempo, by the assessors was impacted by the presence of events, or if the high-inclusion systems were actually less subjectively pleasing in their sampling and presentation. That is, if the assessors were not tasked with finding event occurrences would they subjectively rate the summaries similarly? Or is there a "real" tradeoff between the objective and subjective measures?

[1]This plot looks very much the same if the mean is used to combine the subjective scores instead of the product.

6. SUMMARY

We submitted two systems to the TRECVID 2008 Rushes Summarization task which were simplified from our 2007 submission. Comparative performance of the two systems showed marginal improvements in the enhanced system, although the observed differences are not statistically significant. This is not very surprising since the two systems share the same overall framework and vary only in their clustering measures and clip selection methods. Overall we found a strong negative correlation between positive subjective ratings and positive objective (event inclusion) scores in the submissions overall. Given this tradeoff we found our submissions achieved relatively high performance among other systems with similarly positive subjective ratings.

7. REFERENCES

[1] J.-Y. Bouget. Pyramidal implementation of the lucas kanade feature tracker. description of the algorithm. Technical report, Intel Corporation Microprocessor Research Lab, 2000.

[2] F. Chen, M. Cooper, and J. Adcock. Video summarization preserving dynamic content. In *TVS '07: Proceedings of the international workshop on TRECVID video summarization*, pages 40–44, New York, NY, USA, 2007. ACM.

[3] M. Cooper and J. Foote. Scene boundary detection via video self-similarity analysis. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 378–81, 2001.

[4] M. Detyniecki and C. Marsala. Video rushes summarization by adaptive acceleration and stacking of shots. In *TVS '07: Proceedings of the international workshop on TRECVID video summarization*, pages 65–69, New York, NY, USA, 2007. ACM.

[5] A. G. Hauptmann, M. G. Christel, W.-H. Lin, B. Maher, J. Yang, R. V. Baron, and G. Xiang. Clever clustering vs. simple speed-up for summarizing rushes. In *TVS '07: Proceedings of the international workshop on TRECVID video summarization*, pages 20–24, New York, NY, USA, 2007. ACM.

[6] J. Kleban, A. Sarkar, E. Moxley, S. Mangiat, S. Joshi, T. Kuo, and B. S. Manjunath. Feature fusion and redundancy pruning for rush video summarization. In *TVS '07: Proceedings of the international workshop on TRECVID video summarization*, pages 84–88, New York, NY, USA, 2007. ACM.

[7] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679, 1981.

[8] Open source computer vision library. http://www.intel.com/technology/computing/-opencv/.

[9] P. Over, A. F. Smeaton, and P. Kelly. The TRECVID 2007 BBC rushes summarization evaluation pilot. In *Proceedings of the TRECVID Workshop on Video Summarization (TVS'07)*, pages 1–15, New York, NY, September 2007. ACM Press.

[10] P. Over, A. F. Smeaton, and P. Kelly. The TRECVID 2008 BBC rushes summarization evaluation pilot. In *Proceedings of the TRECVID Workshop on Video Summarization (TVS'08)*, pages 1–15, New York, NY, September 2008. ACM Press.