

Summarizing BBC Rushes the Informedia Way

Alexander G. Hauptmann, Michael G. Christel, Wei-Hao Lin, Bryan Maher, Jun Yang,
Robert V. Baron, and Guang Xiang

School of Computer Science
Carnegie Mellon University

Pittsburgh, PA 15213 USA

{hauptmann, christel, whlin, bsm, juny, rvb, guangx}@cs.cmu.edu

ABSTRACT

For the first time in 2007, TRECVID considered structured evaluation of automated video summarization, utilizing BBC rushes video. This paper discusses in detail our approaches for producing the submitted summaries to TRECVID, including the two baseline methods. The *cluster* method performed well in terms of coverage, and adequately in terms of user satisfaction, but did take longer to review. We conducted additional evaluations using the same TRECVID assessment interface to judge 2 additional methods for summary generation: *25x* (simple speed-up by 25 times), and *pz* (emphasizing pans and zooms). Data from 4 human assessors shows significant differences between the *cluster*, *pz*, and *25x* approaches. The best coverage (text inclusion performance) is obtained by *25x*, but at the expense of taking the most time to evaluate and perceived as the most redundant. Method *pz* was easier to use than *cluster* and had better performance on pan/zoom recall tasks, leading into discussions on how summaries can be improved with more knowledge of the anticipated users and tasks.

1. INTRODUCTION

This paper describes our submission to the TRECVID 2007 summary track, giving details of the approaches, some of the rationale behind them and a few related experiments. Our Carnegie Mellon University Informedia research group has investigated the utility of automated video summarizations for news and documentaries, i.e., for produced materials, since the mid-1990s [1]. However, most of the Informedia summaries (called video skims [1]) were based on broadcast news and documentaries, with redundancies edited out, and good automatic speech recognition transcripts available. In contrast, the BBC rushes contained little editing beyond camera switching on and off, and poor quality audio.

2. SUMMARY CREATION APPROACHES

2.1 Automatic Evaluation

One of the difficulties in developing video summarization systems is that video summaries need to be manually re-evaluated with every change. Human evaluation is not only costly and slow, but summary evaluation results (e.g., satisfaction or redundancy judgments), unlike relevance judgments in conventional information retrieval, can not be easily reused in assessing a different version of the summarization system. Any slight change one makes requires a new run of human evaluation, which makes systematic exploration of the design space almost impossible.

To reduce the amount of human evaluation in the development of a video summary system, we designed an automatic way to approximate one of the evaluation metrics, Inclusion (IN). The IN score is the percentage of events in an original that a viewer recognizes from a summary. If the occurrences of all events (in time) were known, then a program can automatically determine how many events were included in a summary video. To achieve this, we first marked the starting and ending offsets of all events in a rush manually (a one-time effort). An event was scored as *included* if a segment of a summary overlaps with one of the occurrences of the event in the original. An IN score is the percentage of unique events that a summary video covers. Automatically calculated IN scores thus can approximate manual IN scores by replacing human event detection with a time segment overlap test. An excerpt from an annotated truth file is shown in Figure 6.

One may, however, easily inflate automatic IN scores by composing a summary video from many extremely short segments sampled from the original long video (see the discussion of *25x* below). Our automatic IN scoring considers an event included as long as the event overlaps however briefly with some frames a summary video. The weakness of the automatic IN calculation results from the assumption that a human can observe an event no matter how briefly the event is shown on the screen, which is clearly not true. A summary using many extremely short clips will likely overlap most events from the original, but people may not recognize the events from just a few frames. To prevent automatic scoring from misleading our video summary systems to favor extremely short clips, we imposed a constraint that all ‘shots’ of a summary must last at least one second. One second is close to the lower limit that humans can comfortably recognize non-trivial visual content on the screen, e.g. text on the screen is always shown for at least that long. Research on automatic shot detection makes use of the empirical observation and also chooses one second as minimal shot duration [3].

Does the 1-second rule makes the automatically estimated IN scores similar to manually evaluated IN scores reported by NIST assessors? We verified if our automatic IN scoring with the 1-second rule approximates manual IN scores in the following experiment: We randomly chose eight rushes from the testing set, each of which have three versions of summaries (**CMUBASE1**, **CMUBASE2**, and **CMU Submission**), resulting in a total of 24 summaries ($n = 24 = 3 * 8$.) For each event listed in the ground truth file provided by NIST, we manually annotated the starting and ending offset of all occurrences of the event in the video. We then calculated automatic IN scores and compared them to the official, manual derived, scores by assessors from NIST.

The comparison showed a strong positive correlation between automatic and manual IN scores (see Figure 1). The Pearson’s correlation coefficient was 0.67 ($t = 4.29$, $p < 0.01$). The strong correlation justified (in retrospect) our approximation of IN scores. Note that automatic IN scores tend to overestimate the actual scores (points above the dashed line.) In addition to imperfect human judgment, a possible explanation is that one second is still too short for human assessors to recognize some events, such as pans/zooms.

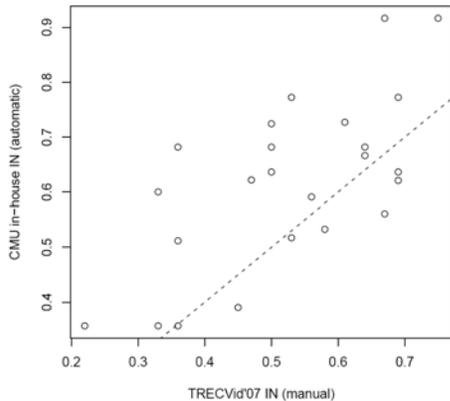


Figure 1. Automatically estimated IN scores of eight rushes of the testing set vs. manually evaluated IN scores from NIST for the same summaries. The dashed line is slope 1 and intercepts at 0, i.e., equivalent automatic and manual IN scores.

2.2 Submitted Summaries

- CMUBASE1: Uniform Sampling Baseline.** CMU submitted an initial baseline summary using a uniform sampling approach. We decided on 1 second as the standard duration of a summary excerpt. The video was divided into segments of 25 seconds each and the middle second (1/25th i.e. 4%) from each segment was included in the summary. Video and audio remained synchronized. This very simple baseline turned out to be a tough standard to beat.
- CMUBASE2: Simple Clustering Baseline.** Encouraged by the organizers, we also tried a more sophisticated baseline using simple color clustering. Using our own shot boundary detector, we lowered the threshold of sufficient differences between adjacent frames to detect a shot compared to broadcast news, allowing any dramatic motion to create a shot change. Hence there were more shots than normally seen in edited broadcast video, with 26268 shots in the development set. From the start of each shot (near the dramatic change) we extracted a keyframe, and partitioned this into a 5x5 grid. In each grid cell, we extracted the mean and standard deviation of hue, saturation and value (HSV color space). Video and audio again remained synchronized.

One keyframe from each shot was used in *per-video* K-means clustering, with the number of clusters set to the number of seconds (rounded down) in the 4% summary. I.e. for a 10 minute video (600 seconds) we would have a target summary length of 24 seconds (4%), and therefore cluster the data into 24 clusters. From each cluster, one second from the middle of the shot closest to the centroid was included in the summary. We did not consider merely displaying the keyframe for one

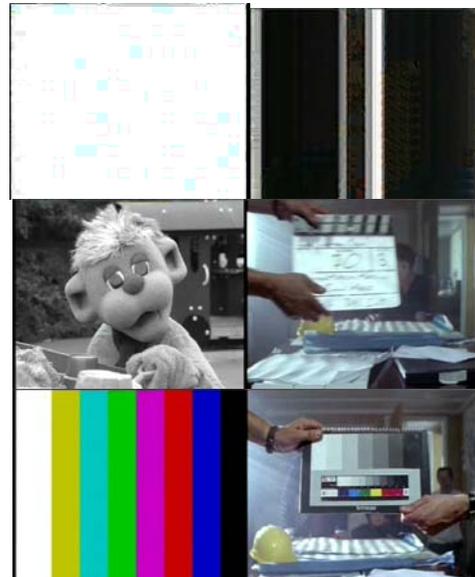


Figure 2. Examples of clearly irrelevant shots to be detected and filtered. (a) white frame, (b) black frame, (c) gray-scale image, (d) clapper, (e) color bar, (f) color calibration chart.

second, as the events frequently involve different motions, which would be lost in any static representation.

- CMU submission.** Our “real” submission tried to improve on the above approaches, using iterative clustering, filtering of clearly irrelevant material, using as much of the available 4% as possible and improved audio. It is described in the next section.

3. ITERATIVE CLUSTERING WITH FILTERING FOR SUMMARIES

Our official submission was created based on iterative color clustering with noise filtering, backfilling of unused space and audio coherence.

Table 1. Accuracy of the detection of irrelevant frames

Irrelevant Shot Type	MAP	F1
White frames	0.38	0.49
Black frames (31 positives labeled)	0.19	0.06
Mostly gray frames (3 positive instances)	0.09	0.00
“Clapper” shots	0.49	0.46
Color bars	0.95	0.96
Color calibration chart	0.72	0.77

We had noticed that the development summaries included a number of shots that were clearly irrelevant, such as color bars, predominantly white or black frames, shots that were all gray-scale, which seemed to be an artifact of the camera switching

on/off, a color calibration chart that was occasionally held up in front of the camera, as well as the “clapper” which indicates *scene* and *take* numbers. For each of these we built automatic detectors that tried to identify these classes of frames as “semantic feature concepts.” Figure 2 shows examples of such frames.

The performance of the classifier for the different types of irrelevant frames varied: we were able to detect *white frames* with an MAP of 0.38 (F1 of 0.49), *black frames* which were labeled only 31 times with MAP of 0.19 (F1 of 0.06), largely gray images which were labeled only 3 times with MAP of 0.09 (F1 of 0.00). The clapper shots were detected with an MAP of 0.49 (F1 of 0.46), while the color bars were easy to classify at MAP 0.95 (F1 of 0.96) and the small color chart was detected with an accuracy MAP of 0.72 (f1 of 0.77).

Table 2. Official Results for the CMU Submission and the CMU Baselines (Means) on the 7 evaluation criteria [1]

The next step after k-means clustering was to eliminate all clusters which were predominantly composed of a clearly irrelevant class. We were now left with fewer clusters, and so we clustered the data again to end up with the targeted number of clusters, with one cluster for each second of video in the target summary. From these clusters, we selected the first second of the shot whose keyframe was closest to the cluster centroid.

Since some shots were shorter than one second, we were again left with a little extra room in the summary (below our target of 4%). To maximize the IN score, we wanted to include as much and as diverse information as possible. The leftover space was ‘backfilled’ by selecting one second from a shot that was *furthest* from a cluster centroid, effectively an outlier. The procedure was repeated until the resulting summary was just below 4%. To

	CMUBASE1	CMUBASE2	CMU Submission
IN	0.59	0.58	0.60
TT	105.66	100.48	101.83
DU	62.11	60.84	58.09
XD	-2.25	-0.97	1.78
VT	60.88	58.68	57.44
EA	3.44	3.41	3.37
RE	3.52	3.50	3.62

facilitate viewing of these disjoint one-second clips, we created the final videos with a 1-frame fade between every segment, as these appeared a little smoother and less jarring.

Audio was associated with this video since we felt that understanding the acoustic context would help to more quickly understand the visual events: Given an edit list of segments based on visual characteristics, we selected the corresponding time boundaries in the ASR transcripts, and determined which edits contain speech and where silences separated the speech transcripts using Signal-to-Noise Ratio calculation. Earlier research on skims [1] has shown that choppy audio is very distracting, and in that research we had successfully used the SNR segmentation to obtain reasonable acoustic phrases in news skims.

We initialized an audio edit list with the mid-point of each visual edit instruction, found the nearest SNR boundaries to each audio edit segment and extended the currently shorted audio edit segment to this boundary. The process stopped when the total duration of the summary (4%) was reached. This simple approach favors playing coherent, recognizable audio segments, related to the visual segments, but loses audio/video synchronization.

We generated summary video files as follows. Each of our approaches took either raw videos or clustering results, and outputted a summary composed of a list of (start time, end time) tuples. We wrote a script that turned the list into a AviSynth script¹. We chosen AviSynth because a) AviSynth scripts are plain text files that are easy to manipulate b) AviSynth scripts are playable media files, which allows quick visual sanity check c) There are a rich set of AviSynth filters that provide useful functions for video summarization, including video/audio synchronization, video concatenation, fade in/fade out transitions, etc. We converted the AviSynth scripts into MPEG files using ffmpeg²

4. RESULTS

There were seven key criteria for analysis used in the NIST TRECVID summary evaluations:

- IN - Inclusion meaning the fraction of the target events found in the summary
- DU - duration of the summary (secs)
- XD - difference between target and actual summary size in seconds
- TT - total time spent judging the inclusions (secs.)
- VT - total video play time (versus pause) judging the inclusions (secs,)
- EA - Was the summary easy to understand:1 strongly disagree - 5 strongly agree
- RE - Was there a lot of duplicate video:1 strongly agree - 5 strongly disagree

Table 1. Official Results for the CMU Submission and the CMU Baselines (Means) on the 7 evaluation criteria [1]

Our results are shown in Table 1. There is a marginal improvement in our submission over the baseline in terms of inclusion (IN) and ease of viewing (EA), as well as the video play time (VT). None of these results indicate that our submission was significantly “better” than the baselines. This was disappointing, as we had tried to optimize for inclusion (IN), but failed to achieve a major gain.

5. ANALYSIS

Figure 3 shows that based on automatic evaluation of INclusion, the iterative clustering slightly outperforms the baseline uniform result as well as the baseline clustering result at 4%. This difference between approaches shrinks at lower summary ‘compression’ rates, but increases as the target summaries become shorter. For this data, a roughly 2% summary based on iterative

¹ <http://avisynth.org/>

² <http://ffmpeg.mplayerhq.hu/>

clustering would be a good choice, since it represents a good tradeoff point between summary length and content represented.

Using the automated evaluation of inclusion (IN), a number of internal experiments on the development data yielded results within 2% of the submitted and judged run. In these experiments, using the automatic INclusion score, we studied variations on the basic approach.

On the road to deciding what the CMU submission should be, we experimented with a number of different techniques, and for each technique, a set of options, based on the 20 movies set aside as evaluation data. Inclusion (IN) was the primary metric used for comparison, although some attention was given to the length of summary. These techniques are discussed below.

“Representative Second”: For each shot to be included in the summary, we needed to decide which “1 second” within the shot to be included. We tried selecting the first second, middle second, and the last second of each shot, and compared the IN of these methods. We found that selecting the first second of the shot resulted in the highest automatically calculated IN (75.26%), which was slightly better than using the last second (74.84%) and using the middle second (73.60%). We also tried using the first half second and the last half second of the shot as well as sampling a second worth of frames within the shot (to have better coverage of the shot and sampling a second worth of frames around the shot boundary to capture more fast motions between shot transitions). The variations yielded slightly better results for some movies but didn't consistently perform better than baselines. Effectively, only the first portion of a shot can be considered representative of a shot's content for the purposes of these automatic IN score evaluations.

“Backfilling” methods: Because the length of some shots is less than 1 second, the summary is often shorter than 4% of the movie length if only 1 second is selected from each cluster. We wanted to fill this extra room with some video snippets that were most likely to increase the IN score. That means some clusters may have more than 1 second of data included in the summary. Since we assume that 1 second of data is sufficient to represent a shot, we proceeded by selecting more than one shot from a single cluster. To increase diversity, we selected the second shot from a cluster that is *farthest* from the centroid shot which was already selected. We call such shots “outlier” shots. However, since there is not enough space for including the outlier shot from every cluster, we wanted to prioritize the clusters and select outliers from higher ranked clusters until the space was filled. We experimented with two ways of ranking the clusters, either by the number of shots in each cluster (large to small), or by the maximum distance between any two shots in each cluster (large to small). We found that the second method achieved higher IN (77.43%) than the first method (75.88%), which was higher than no backfilling (75.26%). Therefore, in the submission we included the outliers of the clusters with the largest inter-shot distance, i.e., clusters with the largest diversity.

Clustering Features: Our baseline submissions use 5x5 HSV color feature for clustering. We also tried to add motion feature and time feature into our feature set in order to improve the clustering results. The motion feature is the average pixel-difference between consecutive frames in a shot. The time feature is represented as the duration from the start of a movie. However,

we found that the inclusion of both features or either of them in fact hurt the clustering results based only on color features, measured in terms of the IN of the summary.

Number of Clusters: Setting the number of clusters to the maximum number allowed by the summary length is not necessarily optimal. If the content of the movie is not very diversified, a smaller number of clusters are desired. Therefore, we used a BIC-based stopping criterion in the clustering algorithm so that it will select the optimal number of clusters (with the upper bound set to the maximal number). However, we found that the IN score dropped quite significantly in this case with only a small decrease of summary length. Therefore, this dynamic clustering method was not used in our submission.

Temporal constraints. Adding a temporal constraint into the clustering algorithm, with the time distance to between two shots [4] in addition to color differences, also failed to improve results.

Using **shot motion** as determined by pixel differencing over the frames of a shot as an additional feature to also resulted in marginally worse performances, as measured in automatically estimated IN scores.

Learned shot selection. Finally we tried to learn to select the best shot for a summary from a given color cluster based on the training data using features such as shot-length, face presence, automatic pan/zoom detection, distance from the cluster centroid, and amount of motion. This approach only resulted in inclusion rates identical to the much simpler strategy of selecting the first second of the centroid shot of the CMU submission, which is independent of any specific training data.

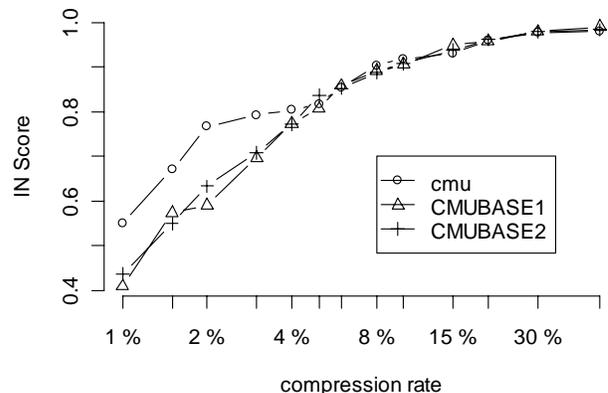


Figure 3. Comparison of different summary compression rates vs. automatically determined INclusion scores of labeled events from the training data. Note the x axis is in log scale.

Self-similarity. Another approach which seemed promising, but did not improve on the result was the use of self similarity to detect repetitions of scenes within takes, and boundaries between takes. In making video summaries, identifying the scene boundaries in the videos in advance is potentially helpful in improving the quality of the summaries. The rationale for this is that scenes are unit with high intra-similarity and low inter-similarity and once scenes are found with relatively high accuracy, we could reduce redundancy and extract only highly informative frames within the scenes.

The self-similarity color matrix provides a convenient manner of gauging the shot-wise distance of the whole movie. Visualizing the content of the videos via these matrices reveals similar shots in areas with darker color and dissimilar shots in lighter color. Generally, the distance values of the matrix on two sides of the scene boundaries differ significantly and those within the scene are relatively homogeneous. Exploiting this inherent characteristic of the videos offers an alternative method to the clustering mechanism for scene identification, i.e., utilizing color similarity as the observations to infer hidden scene boundaries.

The classic hidden Markov model (HMM) has the potential to capture these properties seamlessly and we can recast scene identification as an inference problem in the HMM framework. The observed quantities consist of the frame similarity values in the matrix and the binary states of the HMM are modeled to be BOUNDARY and NON-BOUNDARY. The color distances between one keyframe and all the other keyframes (a row in the matrix) are taken to be an observation sequence and the HMM is trained via a supervised learning method rather than the Baum-Welch algorithm. To automatically extract scenes from the HMM output, we introduced heuristics to exploit the recognized boundaries along the diagonal of the matrix and pruning as much noise as possible. An example is shown in Figure 4.

6. SUMMARIES VIA SIMPLE SPEED-UP OR DOMAIN HEURISTICS

Our research group debated intensely over which one of our automated methods should be submitted to NIST for evaluation. Should we emphasize aesthetics over INclusion, how much time does a viewer need to identify a pan/zoom, should detected faces or people be given a priority, is there a role for audio, does the audio need to be synchronized as earlier work showed that news summarizes with asynchronous audio were jarring. Among the most heated discussion was whether a simple 25x summary, which merely speeds up the playback by selecting every 25th frame, was too trivial and therefore embarrassing to submit to evaluation, even though our informal tests revealed it would likely score very high on the INclusion metric, but also required much effort to watch.

We kept the other video summarizations generated on the 42 videos in the test set, and made use of them in subsequent testing to determine relative differences between our approaches. Once NIST published the evaluation protocol, interfaces, and lists of text inclusions for the 42 test videos, we then ran that same evaluation using 4 recruited Carnegie Mellon testers. As with the NIST assessment, if an assessor judged one type of summary of a video, then s/he judged all summaries of that video. Each video assessment started with playing the full video at about 5x real time at least once while familiarizing themselves with the text list of things they were going to be checking. They then evaluated three summary types: cluster (discussed earlier), 25x (a simple speedup of 25 times normal playback resulting in a 4% duration compared to the original, with a summary audio track added as normal-playback-speed narration), and pz (pan-zoom emphasis in addition to cluster approach). The order of the 3 was counterbalanced and with only 3 types to measure, the summaries were not repeated to the same assessor for the same video. [2] gives more details of the evaluation procedure.

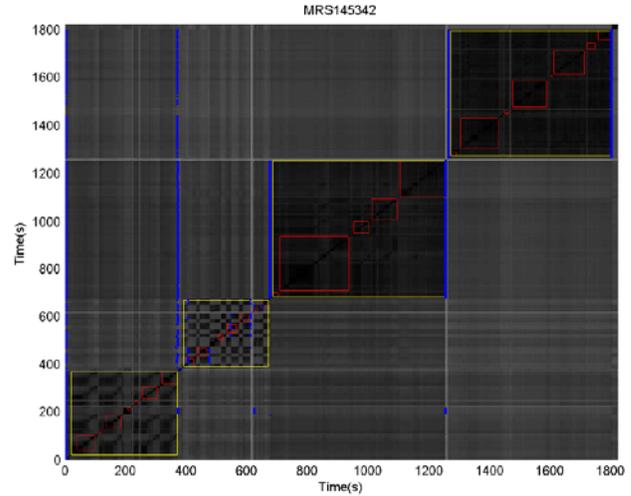


Figure 4. Detected Scene boundaries (blue) of the HMM process from video MRS145342. Yellow boxes indicate the true scenes, while red boxes denote true takes within scenes.

Our internal debates brought up numerous issues, including the role of audio, the ease of use and great coverage provided by a simple speed-up strategy (e.g., 25x), the tuning of summaries to what is defined as being “important” (pans and zooms), and the relative importance of skim effectiveness (information coverage) vs. skim efficiency (time needed to review) and subjectiveness measures. We conducted additional assessment to collect insights supported by statistical findings with respect to these issues.

By simply speeding up the playback of the video 25 times, you create a 4% video summary. The audio is incomprehensible at 25x playback, but some of the BBC rushes dialogue seemed to hold value based on casual inspection of the development data. So, we wanted to augment the 25x video with a regular speed narration. We chose 4% audio content based on the algorithm used create the audio associated with the submitted summaries.

We noted in the instructions to the task that pans and zooms were emphasized as being important. This serves as a form of domain expertise: for future users skimming through summaries of BBC rushes, they will likely want to identify pans and zooms. Rather than hope that our cluster method somehow captures pans and zooms well enough, we created a pz method as follows that still makes use of the clusters discussed in Section 2:

1. All pans and zooms longer than 1 second are automatically tagged. All clusters are identified as in Section 2.
2. Each cluster is represented in time order in the summary. If a cluster has a pan or zoom, the longest one is used to represent the cluster. Otherwise, the representation is chosen based on having video with faces (we assumed faces to be important to humans) and not noise video, where noise video includes color bars, white shots, and clapper shots.
3. If no face video and no pan/zoom exists for the cluster, the cluster representation is as done for Section 2.
4. Pans/zooms are kept in up to 6 second runs, using the central 6 seconds if the identified run was longer. To save time in the summary, however, pan/zoom sequences longer than 2 seconds were cut in half by playing back the video at 2x but using the first half of the audio (so audio playback is normal rate).

5. If the resulting summary is too long, pans/zooms are shortened down to 1 second in length as needed until we reach 4%.

The cluster, pz and 25x skims were all less than the upper bound of 4% of the original video's duration for each of the 42 test set videos. The XD measure [2], the difference between target and actual summary size in seconds, averaged 2.18, 2.82, and 2.06 for the cluster, pz, and 25x skims respectively.

7. ADDITIONAL RESULTS: 25x, pz, cluster

84 summary assessments were collected using the NIST protocol, 2 each for the test videos. The announced pairwise agreement in judging which of the (up to 12) desired items from the full video were included in the summary was on average 78% [2]. The agreement between our CMU assessors was 80.6%. We tested our cluster again to see how well CMU assessors agree with NIST assessment, and the numbers correlate well for IN and EA, correlation coefficient $r=0.8$ and 0.86 , NIST IN means for IN and EA 0.6 and 3.37 , CMU assessors' means 0.61 and 3.06 respectively. For TT and RE ($r=0.43$ and 0.24), CMU assessors took a bit more time (likely because they only had 3 summaries per video to grade) and were more lenient on redundancy: NIST TT and RE means 101.8 and 3.67 ; CMU assessors 109.9 and 4.17 respectively. These are for the same exact cluster summaries on the 42 test videos graded at NIST and then later at CMU.

The point of the exercise was not to check NIST's grading, but to see relative differences between cluster, 25x, and pz. Figure 5 overviews the differences on the TT, IN, EA, and RE measures.

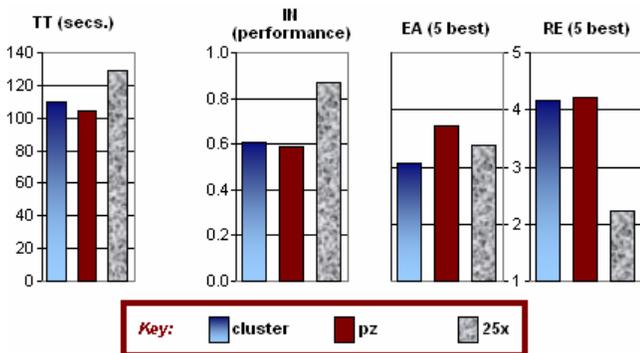


Figure 5. Mean TT, IN, EA, and RE collected from 84 evaluations for each of cluster, 25x, and pz summaries using NIST protocol (conducted twice across 42 test set videos).

Significant differences were found using ANOVA $p < 0.002$ across all four measures, with the Tukey HSD test confirming the following significant differences at $p < 0.01$: for TT, 25x is slower than the others; for IN, 25x produces better performance; for EA, cluster is worse than pz; for RE, 25x is worse than the others.

If the main objective of the summary is to maximize recall of text inclusions, i.e., produce the highest IN score, then 25x is an excellent method, with its 0.87 mean (0.92 median) far outstripping these other two runs and all other NIST submitted runs whose IN means ranged from 0.25 to 0.68 as graded at NIST. Such excellent performance comes at a cost: the TT metric was higher (but still exceeded by some of the NIST graded runs), and the acknowledged redundancy in the 25x summary was quite high (the RE measure). RE and EA were included as metrics to help

with assessing utility and end-user satisfaction, but while 25x was acknowledged as redundant, its ease of use measure (EA) was actually better than that for the clustering approach. We believe the inclusion of an audio narrative made the 25x skim more playable by end users.

If the main objective is to produce a video summary type that users would not mind playing over and over, then of course additional satisfaction metrics and longitudinal studies could be employed to better address that objective. Even with just EA and RE, though, pz shows itself to be an improved summary type than cluster by bringing in some domain knowledge. Namely, for video like BBC rushes where color bars, all white shots, and clapper bars are noise, people are important, and pans and zooms are likely to be looked for later, then emphasizing pans and zooms first, then faces, and dropping out noise works well for EA and RE as a strategy. The EA measure for pz was significantly better than that for the cluster method which did not emphasize pans or zooms, and its RE mean was the highest as well for the 3 tested methods. One reason for little separation on TT and IN between pz and cluster is the large overlap in the automated methods to produce each, and especially the step 5 (Section 4) for pz where pans/zooms are dropped rather than clusters being dropped when the assembled edit list to produce the summary is too long in duration. Future work includes testing more aggressive pz methods that preserve pans and zooms at the expense of clusters and anticipated coverage, i.e., rather than drop pans/zooms, drop clusters.

8. CONCLUSIONS

In general, the task proved to be harder than anticipated, with many of the attempts at improving clustering resulting in no gains of the baseline systems. One reason is that the 4% summary shows several approaches converging in terms of their coverage, due to the large redundancy of the data. We feel our automated evaluation approach to allow us to optimize "inclusion: (IN) rates has proved to be useful, judging by the correlation in the test data, as well as the fact that the rankings in the development data mirror the results in the test data. The cost of annotating the "truth" given the list of events for each video consisted of finding the start and end times of each segment at each repetition in the video. We were disappointed, that we did not develop clear improvements in inclusion rates over the baseline, which was already evident in our experiments on the development data.

The assessment framework provided by NIST and the TRECVID organizers for 2007 allows the international research community to systematically address video summarization for a given genre of video, with this year's test genre being BBC rushes materials. By taking the assessment framework and text inclusions listings, one can conduct follow-up investigations as we did here comparing the relative merits of 3 summarization methods: cluster, pz, and 25x. The duration of the summary is controlled to be nearly the same (XD measures close). Without such control, such as with trying to reach conclusions across the broad set of submitted summaries graded by NIST, it is difficult to state what video summary features leads to what sort of utility. The obvious can be stated: a verbatim extraction of a few seconds from the fill video will have very easy playability (EA), little redundancy (RE), very fast playback (TT), but very poor coverage (IN performance). A simple speed-up approach, coupled with audio

narrative, in our 25x skim produced great IN but weaker TT and RE. Further investigations include looking to the merits of audio by running a comparative study of 25x with and without audio, and looking at the utility of pan-zoom inclusion by more aggressive means of keeping pans/zooms over clusters in pz-style summaries.

9. ACKNOWLEDGMENTS

Our thanks to NIST and the TRECVID organizers for enabling this video summarization evaluation. This work was supported in part by the National Science Foundation under Grant No. IIS-0205219.

10. REFERENCES

- [1] Christel, M.G., Smith, M.A., Taylor, C.R., & Winkler, D.B. Evolving Video Skims into Useful Multimedia Abstractions. In *Proc. ACM CHI '98* (Los Angeles, April 1998), 171-178.
- [2] P. 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, 1-15
- [3] Hanjalic, A. Shot-Boundary Detection: Unraveled or Resolved? In *IEEE Transactions on Circuits and Systems for Video Technology*. 12(2). 2002. 90-105.
- [4] Lin, W.H. and Hauptmann, A., "Structuring Continuous Video Recordings of Everyday Life Using Time-Constrained Clustering", *IS&T/SPIE Symposium on Electronic Imaging*, San Jose, CA, January 15-19, 2006.

```

1 A closeup of man's head and shoulders,
  dozing
2 B man lights up a cigarette,
3 B man drinks
4 A man talks
5 A closeup of man's face, neck
6 A view from above of couple at table
7 A view from above of table with 6 people
8 A close view of table with 4 people and
  boom microphone
9 A closeup of two women and man at table
10 B woman with white top with vertical
  grey stripes stands and open parasol.
11 B woman with purple dress leaves
  birthday party in tea shop
12 A closeup of man and woman in dark
  jackets
13 A closeup of woman in grey stripes

E 0 50
1 56 80
2 84 88
3 101 104
4 99 201
3 145 147
1 215 248
2 256 260
.
.
.
```

Figure 6. Example time annotation of events for file MRS042545. The events are listed at the top and numbered. The lower part shows the event number, the offset in seconds when the event started and ended. Letters A, B, C, D, E, F instead of event numbers at the beginning of a line were used to refer to the irrelevant frames as in Figure 2.