RMIT University at TRECVID 2004

Timo Volkmer S.M.M. Tahaghoghi Hugh E. Williams

School of Computer Science and Information Technology
RMIT University, GPO Box 2476V, Melbourne, Australia, 3001

{tvolkmer,saied,hugh}@cs.rmit.edu.au

Run overview

We participated in Shot Boundary Detection and Story Segmentation. This page provides a summary of: (1) the approaches tested in the submitted runs; (2) differences in results between the runs; (3) the overall relative contribution of the techniques; and, (4) our overall conclusions.

Shot Boundary Detection

Our approach uses the moving query window technique [12, 13, 14].

1. We submitted ten runs using only visual features and varied parameters for gradual transition detection as follows:

   rmit1: Threshold level `1.6`, threshold history `1`
   rmit2: Threshold level `1.6`, threshold history `4`
   rmit3: Threshold level `1.7`, threshold history `1`
   rmit4: Threshold level `1.7`, threshold history `4`
   rmit5: Threshold level `1.8`, threshold history `1`
   rmit6: Threshold level `1.8`, threshold history `4`
   rmit7: Threshold level `1.9`, threshold history `1`
   rmit8: Threshold level `1.9`, threshold history `4`
   rmit9: Threshold level `2.0`, threshold history `1`
   rmit10: Threshold level `2.0`, threshold history `4`

   We added a new postprocessing step to our gradual transition detection scheme that was not applied in TREC 2003. We submitted ten additional runs — numbered **rmit11** through **rmit20** — using ASR and identical parameters to those above. The ASR information is used in a postprocessing step to remove gradual transitions that coincide with spoken words.

2. The threshold level trades between recall and precision: low thresholds favour recall, high thresholds favour precision. Compared to a long history length, a short history length generally improves gradual transition recall with no or little negative effect on precision. Our ASR filtering was ineffective.

3. Threshold level has a significant effect on our results, varying recall between 86% and 91.5%, and precision between 82.9% and 90% overall. Increasing history length decreases overall recall and precision by 1%–2%. Postprocessing with ASR data dramatically reduces recall while precision practically remains.

4. Our new postprocessing step for gradual transitions — discussed in detail in the paper — is highly effective. Our exploration of parameters has identified settings that are optimal for television news. We have concluded that threshold settings of 1.7 or 1.8, and a history length of 1 works best.

Story Segmentation

1. We submitted six runs, with variations of a moving window size:

   rmit1: No window, Condition 1
   rmit2: Small window, Condition 2
   rmit3: Small window, Condition 3
   rmit4: No window, Condition 1
   rmit5: Larger window, Condition 2
   rmit6: Larger window, Condition 3

2. Runs for conditions 1 and 3 have reasonable recall but low precision, while runs in condition 2 have better precision but low recall.

3. Using conditions 1 and 3, we truncate stories: the result is reasonable recall but high false positives. Under condition 2, our results are better.

4. Our technique does not combine evidence well, but is a reasonable first attempt. We are continuing our work for TREC 2005.

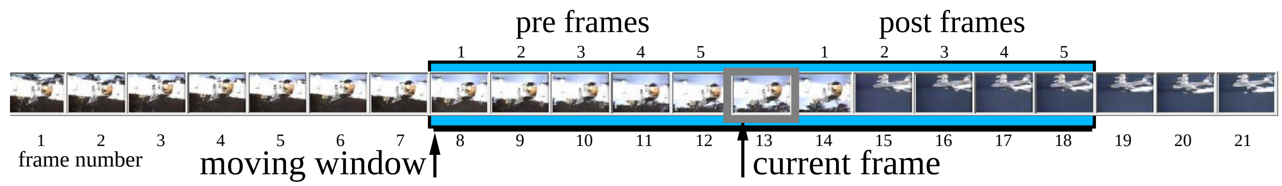

Figure 1: Moving query window with a half-window size (HWS) of 5. The five frames before and the five frames after the current frame form a collection on which the current frame is used as a query example.

1 Introduction

Segmenting video into sections of interest is crucial for search and retrieval. Identifying the basic semantic entities — the *shots* [5] — is a common first step and usually achieved through finding the transitions that constitute the boundaries between adjacent shots. It has been observed that for both cuts and gradual transitions, frames are similar when they are all from within a shot, and that this situation changes when a transition is encountered. The majority of transition detection techniques make use of this observation.

Automatic cut detection systems have proven to work effectively [1, 10, 13], while the detection of gradual transitions has been more difficult [7, 8]. This is supported by current and past results from TRECVID participants. In this paper, we present our latest techniques for shot boundary detection that focus on effective gradual transition detection. Using our moving query window from previous TRECVID workshops and one-dimensional histograms for frame comparison, we show that our scheme is highly effective.

Segmenting video into stories, that is, identifying coherent sections concerning a specific topic coverage is another key task. Effective approaches have been presented at TRECVID in the past [2, 3] but the task of segmenting video into stories remains problematic. In this paper, we present our first attempt at story segmentation that makes use of the fundamental discoveries we have made in shot boundary detection. Our results show our approach is still ineffective, but likely to be the basis of a useful technique after further refinement.

2 Shot boundary detection

Our approach remains largely the same as our moving query window approach presented previously at TRECVID [12, 15]. Figure 1 shows a moving query window: the window has an equal number of frames on either side of a current frame, and the current frame is advanced or *moved* as the video is processed.

Each frame is represented by one-dimensional histograms in the HSV colour space. A major difference to last year is, however, that we use localised histograms. Each frame is separated into 16 equal-sized regions. For each region, we extract a separate histogram with 16 bins per colour component. The quantised pixel information for all components are evaluated within a single vector dimension.

Our past research has shown that the problem of detecting gradual transitions is rather different to detecting abrupt transitions. Despite using identical histogram representations, the evaluation process that we apply for each is different. Our implementation allows us to accomplish cut detection and gradual transition detection simultaneously during one single pass. The details of our approach are discussed next.

2.1 Abrupt transitions

For cut detection, we use our ranking-based method [13]. This method has proven to work very effectively [12] with features derived from the Daubechies wavelet transform [4]; however, computation of wavelets is expensive. In 2003, to reduce computational cost, we used the ranking-based method in combination with one-dimensional, global histograms in the HSV colour space [15]. Results were strong but not as good as those obtained with the wavelet feature. One goal for this year was to improve cut detection quality in combination with the relatively simple HSV histogram feature.

We have observed that scenes with rapid object movement are difficult to correctly delineate, and sometimes lead to false detections. Other groups apply motion compensation techniques to handle such cases [9, 16] but this adds additional computational overhead. We have observed that it is common that the main activity typically occurs in the focus area — usually in the centre — of frames. This observation lead us to investigate the effect of assigning less weight to the centre of each frame when comparing inter-frame differences.

Consider Figure 2. We propose dividing each frame into 16 regions and extracting a histogram for each region. When comparing frames in the moving query

Figure 2: For cut detection, we use only information outside the frame focus area to reduce the effect of rapid scene activity.

window, we assign a weight to each region, allowing fine-grain control over the significance attached to an area of the frame; as discussed previously, this allows lowering of weight associated with the middle of the frame, that is, the region typically affected by rapid object movement.

During our training experiments, we observed that this technique works best when assigning no weight to the four central frame regions, that is, disregarding the focus area of each frame. Our preliminary results showed an improvement of up to 5% in the quality-index over the results that we achieved at TRECVID in 2003. An additional benefit is that a system can be implemented such that it only extracts and compares the histograms for the frame regions that are used in comparisons. This allows faster execution times for both the decoding and the evaluation phase. However, our current implementation does not take advantage of this efficiency gain. In the cut detection experiments described in this paper, we use above approach, disregarding the central four frame regions.

2.2 Gradual transitions

Our main focus this year was on improved detection of gradual transitions. Our approach is based on the moving query window. However, in contrast to our cut detection stage, we do not rank frames. Instead, the similarity to the current frame is computed for each frame in the moving window. Frames on each side of the current frame are then combined into two sets of pre- and post-frames. For both sets, the average similarity to the current frame is computed. We then monitor the ratio between the average similarities, allowing us to detect gradual transitions by observing peaks in the ratio curve [14]. We observed promising results with this technique in 2003, but it suffered from many false detections. Our focus this year was reducing this effect.

We use the same one-dimensional, localised HSV histograms as used in cut detection, again divided into 16 regions per frame. For gradual transitions, we compare frames using the average distance between all corresponding frame regions between two frames, using identical weights for each frame region. Our previous experiments suggest that assigning different weights to some regions does not improve gradual transition detection. However, using localised histograms with the average distance of corresponding regions does improve precision.

A major improvement over the technique presented last year [15] was achieved when we introduced an additional threshold to reduce false detections. In cut detection, we require the last frame of the previous shot and the first frame of the next shot have a minimum dissimilarity of 25% of the maximum possible inter-frame difference [13]. Given that the frames of one shot are usually similar — independent of the type of transition — it seems reasonable to apply this requirement to gradual transitions. Therefore, after detecting a possible gradual transition, we compare the frame immediately before the start of the possible transition to the frame directly after the end of the possible transition. A gradual transition is only reported if these frames are dissimilar; this new step dramatically improves performance, as shown later.

The implementation presented last year had a limitation that prevented us from detecting gradual transitions longer than 60 frames. This limitation is now eliminated and the system detects gradual transitions of arbitrary length. Additionally, we have made threshold computation more dynamic, which improves detection with modern television footage where transitions often follow in rapid succession.

2.3 Algorithm details

An important parameter of our system is the size of the moving window: we refer to this as the Half-Window Size (HWS), that is, the number of frames on either side of the current frame. We have experimented with different sizes for cut detection and gradual transition detection. We determined an optimum size for cut detection and use this setting since [13]. Therefore, we regard our cut detection as parameter free.

Our experiments suggest that it is difficult to determine an optimal window size for all gradual transitions in different footage types. Rather, the length is dependent on the average length of the transitions in the footage. In TRECVID 2004, the footage type is limited to television news; we have found that we achieve best results for this footage when using HWS=14.

The dynamic threshold for peak detection in our gradual transition detection stage is calculated using a number of past frames that we store in a history buffer. The buffer size is specified by the Threshold History Size (THS) factor. The number of frames in the history is the number of frames in the entire query window, multiplied by THS. Specifically, we store the ratio between the pre-frames difference and post-frames difference for each frame in this buffer. We compute the average ratio over all frames of the buffer and determine the current threshold value based on the standard deviation to the actual curve. We multiply the computed threshold by an Upper Threshold Factor (UTF). Both THS and UTF can then be used to fine-tune the technique to different video material.

A larger history results in a less dynamic threshold curve. Varying UTF has a direct impact on recall but can help to reduce false detections in low quality, noisy footage. The goal of the parameter variation in our run submissions was to find optimal settings for modern television news.

3 Story Segmentation

This is our first attempt at story segmentation, and we regard our techniques as preliminary and experimental. Our goal was to investigate whether our techniques from shot boundary segmentation can be applied to a semantically higher-level task. The TRECVID story segmentation task requires to submit at least one run in each of three required conditions. These conditions are:

Condition 1: Only audiovisual features can be used, no transcripts or ASR is allowed.

Condition 2: Audiovisual features and ASR are allowed but no transcripts.

Condition 3: Only ASR can be used.

The Spoken Language Processing Group at LIMSI [6] has donated the Automatic Speech Recognition (ASR) output for the test collection that we have used.

For condition 1, where only audiovisual features are used, we segment the video into shots using shot boundary reference data (and not our technique described previously). After this step, we used a query-by-example approach to identify shots that show news commentators, that is, *anchor shots*. We further used the audio signal to detect shots that start with low noise, with the rationale that pauses in speech or sound may indicate a topic switch. Shot boundaries that are either an anchor shot or have low noise were taken as story beginnings.

Condition 2 allows the use of audiovisual features as well as text extracted through automatic speech recognition. We again segmented shots using the reference data and assigned the spoken words from the ASR output to each shot, based on the word timing information. We treated the resulting text fragments as individual documents. We then indexed the documents using our `zettair`[1] text search engine.

We made the assumption that a story consists of several shots, that is, a story consists of consecutive, contiguous documents indexed by our search engine. We also assume that a story begins at the start of each video. We then carry out and repeat the following steps:

1. Extract the first n documents from the video stream to form a set N
2. Extract the next m documents beginning at $s = n + 1$ to form a set M
3. For each document in M, run its text as a query on the document collection. Then, count the fraction of the top l ranked answers that are in the set M and record the result
4. Sum the results of the previous step, identifying the total t, non-unique answers from N that appear as results to the queries from M
5. Report the end of a story if t is less than a threshold value p. If so, begin again from step 1 for the next story. If not, continue from step 2 for $s = s + 1$

For run condition 3, only the text is used. We separate the text stream into documents based on long pauses between spoken words. We then apply the same algorithm as for run condition 2. In contrast to condition 2, this can allow story changes within a shot.

3.1 Algorithm Details

The anchor shot detection for run conditions 1 and 2 is based on histogram comparison. We extract one-dimensional, localised histograms with 16 regions as shown in Figure 2. Each region is represented through a 48-bin histogram in the HSV colour space and frames are compared using the average Manhattan distance

[1] `http://www.seg.rmit.edu.au/zettair/`

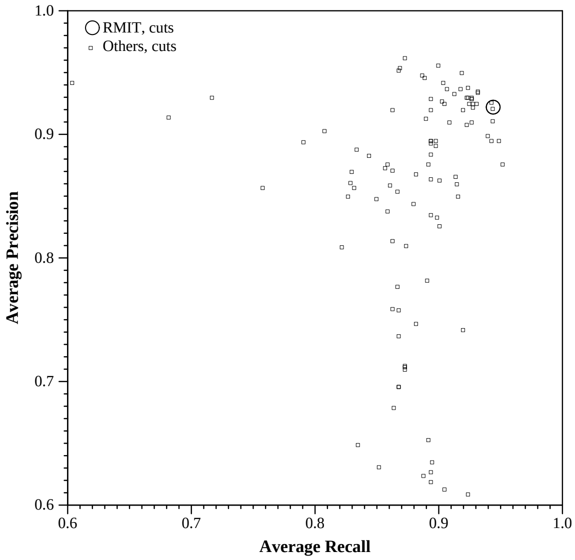

Figure 3: Performance of our system for cut detection on the TRECVID 2004 *shot boundary detection task, measured by Recall and Precision.*

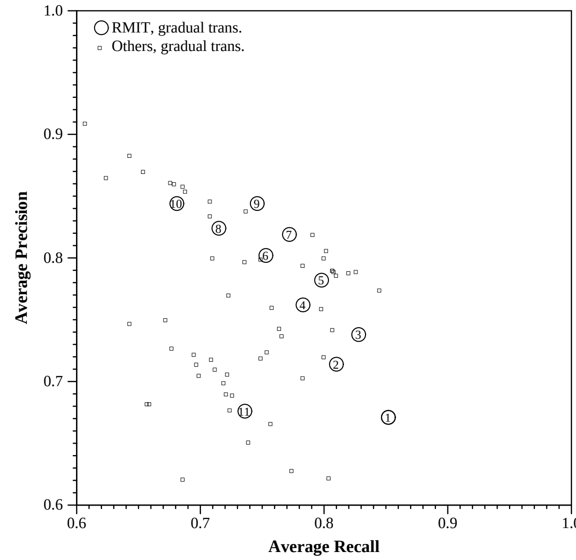

Figure 4: Performance of our system for gradual transition detection on the TRECVID 2004 *shot boundary detection task, measured by Recall and Precision.*

between corresponding regions. We used 39 example anchor person frames taken from the TRECVID 2003 keyframe collection as query examples. An anchor shot is reported when the accumulated difference to all example frames within the first 20 frames of a shot falls below a set threshold. We use a fixed threshold that we have heuristically determined in preliminary experiments.

The audio stream is downsampled to exactly one sample per frame, that is 29.97 samples per second. We accumulate the absolute amplitude levels within the first 20 frames of a shot to evaluate the noise level at the beginning of the shot. Our early experiments have suggested that this can indicate a shot that introduces new story. The decision to use the first 20 frames of a shot was made through empirical observation.

Our technique relies heavily on processing text obtained from automatic speech recognition. In the algorithm described in Section 3, we use a form of moving query window; however, at this stage, it only considers text documents. Crucial parameters in this algorithm are the sizes of sets N and M, the l top ranked documents considered, and the threshold p. Surprisingly, our preliminary experiments suggest that $n = 1$ works best, that is, only one document should seed each story. We observed that a variation of the threshold level did not change results significantly, as long as the threshold is not lower than $p = 0.5$. We decided to use $p = 0.85$. The value of m has the most significant effect on our results and is the focus of our submission, as shown in Table 3.

4 Results and Discussion

In this section we discuss results of our systems for shot boundary detection and story segmentation when applied to the TRECVID 2004 test sets.

4.1 Shot Boundary Detection

The test set for shot boundary detection consisted of 12 video files with total duration of approximately 5 hours and 44 minutes. There were 4,797 transitions, labelled as cut, dissolve, fade, or other, where the latter three types are gradual.

Figure 3 shows the performance of our system for cut detection, measured in recall and precision and compared to all other submissions. We have not varied parameters that affect cut detection. Our algorithm was set to produce an optimum trade-off between recall and precision according to the TRECVID quality measure [11]. This measure slightly favours recall over precision. Our results are good and have improved by almost 3% in recall and 9% in precision over those of last year.

In Figure 4, the recall and precision of our technique for gradual transitions is shown compared to the results of the other submissions. We have varied parameters

			All Transitions		Cuts		Gradual Transitions			
SysID	UTF	THS	Recall	Precision	Recall	Precision	Recall	Precision	F-Recall	F-Precision
rmit1	1.6	1	0.915	0.829	0.944	0.922	0.852	0.671	0.694	0.833
rmit2	1.6	4	0.901	0.850	0.944	0.921	0.810	0.714	0.796	0.735
rmit3	1.7	1	0.907	0.859	0.944	0.921	0.828	0.738	0.715	0.832
rmit4	1.7	4	0.893	0.870	0.944	0.921	0.783	0.762	0.796	0.714
rmit5	1.8	1	0.897	0.877	0.944	0.921	0.798	0.782	0.730	0.831
rmit6	1.8	4	0.883	0.885	0.944	0.921	0.753	0.802	0.794	0.710
rmit7	1.9	1	0.889	0.890	0.944	0.921	0.772	0.819	0.743	0.830
rmit8	1.9	4	0.871	0.893	0.944	0.921	0.715	0.824	0.793	0.697
rmit9	2.0	1	0.881	0.899	0.944	0.921	0.746	0.844	0.762	0.830
rmit10	2.0	4	0.860	0.900	0.944	0.921	0.681	0.844	0.789	0.694

*Table 1: Detailed results for all runs in shot boundary detection for our system, along with the varied parameters upper threshold factor (*UTF*) and threshold history size (*THS*).*

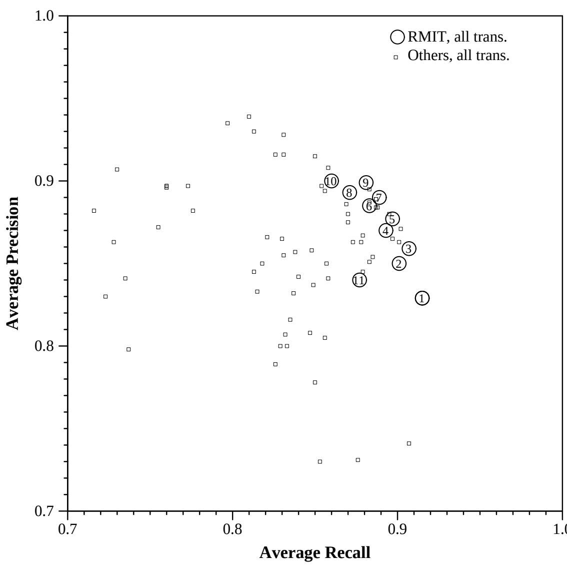

Figure 5: Performance of our system for all transitions on the TRECVID 2004 *shot boundary detection task, measured by Recall and Precision.*

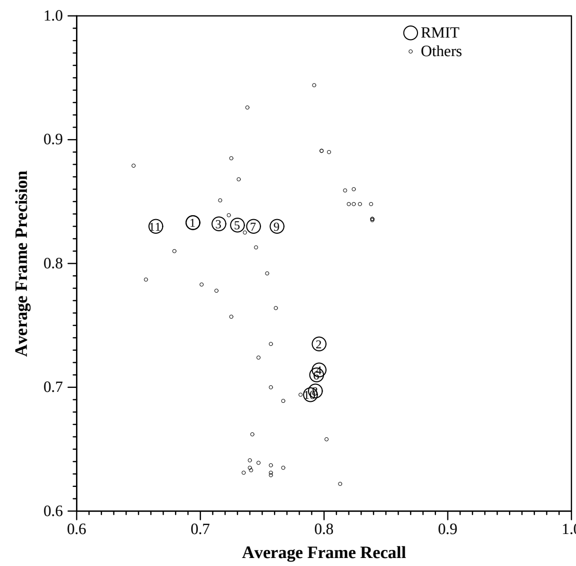

Figure 6: Frame Recall and Frame Precision of our system for gradual transitions on the TRECVID 2004 *shot boundary detection task.*

to explore optimal quality, and our runs show the variation in trade-off between recall and precision. Run 11 illustrates the effect of applying ASR in a postprocessing step to run 1: this lowers recall substantially with only a small gain in precision, leading to the conclusion that our text-based step is ineffective. (Runs 12 to 20 are not shown in the graph, but have the same effect on the results from runs 2 to 10.) For gradual transitions, we now have very competitive results: most importantly, compared to last year, we improved our recall by 77% while maintaining our high precision.

For the combined results in cut and gradual transition detection — as shown in Figure 5 — our technique outperforms most others. Figure 6 shows frame recall and frame precision to measure how accurately start and end of gradual transitions are detected. We observe good results, similar to last year. Table 1 shows detailed results of all runs with parameter details.

Timing results are shown in Table 2. We measured elapsed time, repeated each run ten times and presented the average. The timing experiments were performed on a single CPU Intel-based machine with a 3GHz Pentium 4 processor, 1,024 MB of main-memory running SuSE Linux 9.1 with the SuSE standard kernel 2.6.5.

We separate decoding and histogram extraction from the evaluation step; as we use only one feature, times for decoding are identical for each run. Overall, our

SysID	Decoding	Evaluation	Total
rmit1	15,316.3	195.9	15,512.2
rmit2	15,316.3	174.8	15,491.1
rmit3	15,316.3	195.9	15,512.2
rmit4	15,316.3	174.8	15,491.1
rmit5	15,316.3	195.9	15,512.2
rmit6	15,316.3	174.8	15,491.1
rmit7	15,316.3	195.9	15,512.2
rmit8	15,316.3	174.8	15,491.1
rmit9	15,316.3	195.9	15,512.2
rmit10	15,316.3	174.8	15,491.1

Table 2: Timing results in seconds (real time) for all runs of our shot boundary segmentation system. The times are averages of 10 runs over the complete test set.

schemes take around 4 hours and 18 minutes to process the TRECVID 2004 shot boundary test set. This approximates to 75% real time. The only parameter variation that causes differences in evaluation time is the history buffer size; other parameters have no effect on evaluation times. Our current implementation, especially the decoding stage, is not optimised for efficiency.

4.2 Story segmentation

The story segmentation test set in TRECVID 2004 consisted of 128 video files with a total duration of nearly 70 hours.

Table 3 shows recall and precision for all submitted runs along with the run parameters used. Figure 7 shows the results of our system compared to all other submissions. Our recall is acceptable, but our precision is unacceptably low. It is clear that our assumptions need investigation: for example, we assume that stories begin with anchor shot, and this appears unreasonable. We also believe that our moving window approach with documents is highly susceptible to topic drift, that is, a single false positive causes later story detections to fail. We plan significant further investigation and refinement.

5 Conclusions and Future Work

We have presented our approaches to shot boundary detection and story segmentation. Our results show that our latest refinement of the moving query window is highly effective for cut and gradual transition detection. In particular, considering only certain regions in a frame improves cut detection and reduces false positives. The introduction of an additional threshold

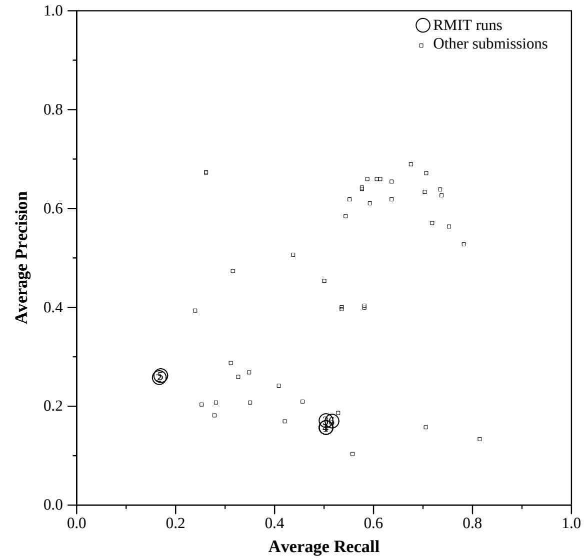

Figure 7: Recall and precision of our system in the TRECVID 2004 *story segmentation task, compared to other participants. Incomplete submissions are not shown.*

SysID	Condition	Win.Size (`m`)	Rec.	Prec.
rmit1	1	—	0.504	0.157
rmit2	2	1	0.167	0.258
rmit3	3	1	0.504	0.171
rmit4	1	—	0.504	0.157
rmit5	2	2	0.170	0.262
rmit6	3	2	0.516	0.170

Table 3: Results of our system for the submitted runs in the story segmentation task.

— that compares frames of previous and next shots directly when detecting gradual transitions — significantly improves quality. In a first, basic attempt, we were unable to use ASR output to improve shot boundary detection.

We have applied our ideas from shot boundary detection, such the moving query window, to story segmentation. A moving query window may be the basis of a successful technique, but much more work is needed. For example, our current technique does not combine evidence well; this is a major goal of our future work.

6 Acknowledgements

We thank Nick Lester of the Search Engines Group (SEG) for advice on using `zettair`.

References

[1] B. Adams, A. Amir, C. Dorai, S. Ghosal, G. Iyengar, A. Jaimes, C. Lang, C.-Y. Lin, A. Natsev, M. Naphade, C. Neti, H. J. Nock, H. H. Permuter, R. Singh, J. R. Smith, S. Srinivasan, B. L. Tseng, T. V. Ashwin, and D. Zhang. IBM Research TREC-2002 video retrieval system. In E. M. Voorhees and L. P. Buckland, editors, *NIST Special Publication 500-251: Proceedings of the Eleventh Text REtrieval Conference (TREC 2002)*, pages 289–298, Gaithersburg, MD, USA, 19–22 November 2002.

[2] A. Amir, M. Berg, S. Chang, W. Hsu, G. Iyengar, C. Lin, M. Naphade, A. P. Natsev, C. Neti, H. N. J. R. Smith, B. Tseng, Y. Wu, and D. Zhang. IBM Research TREC-2003 video retrieval system. In E. M. Voorhees and L. P. Buckland, editors, *TRECVID 2003 Workshop Notebook Papers*, Gaithersburg, MD, USA, 18–21 November 2003.

[3] L. Chaisorn, C. Koh, Y. Zhao, H. Xu, T.-S. Chua, and T. Qi. Two-level multi-modal framework for news story segmentation of large video corpus. In E. M. Voorhees and L. P. Buckland, editors, *TRECVID 2003 Workshop Notebook Papers*, Gaithersburg, MD, USA, 18–21 November 2003.

[4] I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992.

[5] A. Del Bimbo. *Visual Information Retrieval*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

[6] J. L. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Journal of Speech Communication*, 37(1-2):89–108, 2002.

[7] R. W. Lienhart. Reliable transition detection in videos: A survey and practitioner's guide. *International Journal of Image and Graphics (IJIG)*, 1(3):469–486, July 2001.

[8] S. Marchand-Maillet. Content-based video retrieval: An overview. Technical Report 00.06, CUI - University of Geneva, Geneva, Switzerland, 2000.

[9] G. M. Quénot. TREC-10 shot boundary detection task: CLIPS system description and evaluation. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-250: Proceedings of the Tenth Text REtrieval Conference (TREC 2001)*, pages 142–151, Gaithersburg, MD, USA, 13–16 November 2001.

[10] G. M. Quénot, D. Moraru, and L. Besacier. CLIPS at TRECVID: Shot boundary detection and feature detection. In E. M. Voorhees and L. P. Buckland, editors, *TRECVID 2003 Workshop Notebook Papers*, Gaithersburg, MD, USA, 18–21 November 2003.

[11] R. Ruiloba, P. Joly, S. Marchand-Maillet, and G. M. Quénot. Towards a standard protocol for the evaluation of video-to-shots segmentation algorithms. In *Proceedings of the European Workshop on Content-Based Multimedia Indexing (CBMI'99)*, pages 41–48, Toulouse, France, 25–27 October 1999.

[12] S. M. M. Tahaghoghi, J. A. Thom, and H. E. Williams. Shot boundary detection using the moving query window. In E. M. Voorhees and L. P. Buckland, editors, *NIST Special Publication 500-251: Proceedings of the Eleventh Text REtrieval Conference (TREC 2002)*, pages 529–538, Gaithersburg, MD, USA, 19–22 November 2002.

[13] S. M. M. Tahaghoghi, J. A. Thom, H. E. Williams, and T. Volkmer. Video cut detection using frame windows. In V. Estivill-Castro, editor, *Proceedings of the Twenty-Eighth Australasian Computer Science Conference (ACSC 2005)*, volume 38, Newcastle, NSW, Australia, 31 January – 3 February 2005. Australian Computer Society. To appear.

[14] T. Volkmer, S. M. M. Tahaghoghi, and H. E. Williams. Gradual transition detection using average frame similarity. In S. Guler, A. G. Hauptmann, and A. Henrich, editors, *Proceedings of the 4th International Workshop on Multimedia Data and Document Engineering*, Washington, DC, USA, 2 July 2004. IEEE Computer Society.

[15] T. Volkmer, S. M. M. Tahaghoghi, H. E. Williams, and J. A. Thom. The moving query window for shot boundary detection at TREC-12. In E. M. Voorhees and L. P. Buckland, editors, *TRECVID 2003 Workshop Notebook Papers*, Gaithersburg, MD, USA, 17–18 November 2003.

[16] L. Wu, X. Huang, J. Niu, Y. Xia, Z. Feng, and Y. Zhou. FDU at TREC 2002: Filtering, Q&A, web and video tracks. In E. M. Voorhees and L. P. Buckland, editors, *NIST Special Publication 500-251: Proceedings of the Eleventh Text REtrieval Conference (TREC 2002)*, Gaithersburg, MD, USA, 19–22 November 2002.