# Evaluation of Automatic Shot Boundary Detection on a Large Video Test Suite

**Colin O'Toole[1], Alan Smeaton[1], Noel Murphy[2] and Sean Marlow[2]**

**School of Computer Applications[1] & School of Electronic Engineering[2]**
**Dublin City University**
**Glasnevin, Dublin, Ireland.**

## Abstract

The challenge facing the indexing of digital video information in order to support browsing and retrieval by users, is to design systems that can accurately and automatically process large amounts of heterogeneous video. The segmentation of video material into shots and scenes is the basic operation in the analysis of video content. This paper presents a detailed evaluation of a histogram-based shot cut detector based on eight hours of TV broadcast video. Our observations are that the selection of similarity thresholds for determining shot boundaries in such broadcast video is difficult and necessitates the development of systems that employ adaptive thresholding in order to address the huge variation of characteristics prevalent in TV broadcast video.

## 1.0 Introduction

The indexing and retrieval of digital video is an active research area in computer science. The increasing availability and use of on-line video has led to a demand for efficient and accurate automated video analysis techniques. As a basic, atomic operation on digital video, much research has focused on segmenting video by detecting the boundaries between camera shots.

A *shot* may be defined as a sequence of frames captured by "a single camera in a single continuous action in time and space" [1]. For example, a video sequence showing two people having a conversation may be composed of several close-up shots of their faces which are interleaved and make up a scene. Shots define the low-level, syntactical building blocks of a video sequence.

A large number of different types of boundaries can exist between shots [8]. A cut is an abrupt transition between two shots that occurs between two adjacent frames. A fade is a gradual change in brightness, either starting or ending with a black frame. A dissolve is similar to a fade except that it occurs between two shots. The images of the first shot get dimmer and those of the second shot get brighter until the second replaces the first. Other types of shot transitions include wipes and computer generated effects such as morphing.

A *scene* is a logical grouping of shots into a semantic unit. A single scene focuses on a certain object or objects of interest, but the shots constituting a scene can be from different angles. In the example above the sequence of shots showing the conversation would comprise one logical scene with the focus being the two people and their conversation.

The segmentation of video into scenes is far more desirable than simple shot boundary detection. This is because people generally visualise video as a sequence of scenes not of shots, just like a play on a stage, and so shots are really a phenomenon peculiar to only video. Scene boundary detection requires a high level semantic understanding of the video sequence and such an understanding must take cues from, amongst other things, the associated audio track and the encoded data stream itself. Shot boundary detection, however, still plays a vital role in any video segmentation system, as it provides the basic syntactic units for higher level processes to build upon.

Although many published methods of detecting shot boundaries exist, it is difficult to compare and contrast the available techniques. This is due to several reasons. Firstly, full system implementation details are not always published and this can make recreation of the systems difficult. Secondly, most systems are evaluated on small, homogeneous sequences of video. These results give little indication how such systems would perform on a broader range of video content types, or indeed how differing content types can affect system performance.

As part of an ongoing video indexing and browsing project, our recent research has focused on the application of different methods of video segmentation to a large and diverse digital video collection. The aim is to examine how different segmentation methods perform on different video content types. With this information, it is hoped to develop a system capable of accurately segmenting a wide range of broadcast video.

This paper focuses on the preliminary results obtained using a work in progress system based on colour histogram comparison. We are also investigating other methods of video segmentation based on motion vectors, edge detection, macroblock counting and, more importantly, combinations of the above techniques.

## 2.0 Related Work

Much research has been done on automatic content analysis and segmentation of video. Attention has mainly been focused on different methods of shot boundary detection, with most techniques analysing consecutive frames to decide if they belong to the same shot. Zhang et al [2] use a pixel-based difference method, which, although slow, produced good results once the threshold was manually tailored to the video sequence.

Another, more common method is to use histograms to compare consecutive video frames. Nagasaka and Tanaka [3] compared several statistical techniques using grey level and colour histograms. Zhang et al [2] used a running histograms method to detect gradual as well as abrupt shot boundaries. Cabedo and Bhattacharjee [1] used the cosine measure for detecting histogram changes in successive frames and found it more accurate than other, similar methods. Gong et al [4] used a combination of global and local histograms to represent the spatial locations of colour regions.

Other systems use information encoded in the compression format to detect shot boundaries. Meng et al [5] examine the ratio of intracoded and predicted macroblocks in MPEG P-frames to decide if a transition has taken place, with a large number of intracoded macroblocks indicating a change. Cabedo and Bhattacharjee [1] use a variety of methods to process I, B, and P frames in an MPEG-2 video stream.

The comparison of intensity edges is another source of information in detecting shot boundaries. Zabih et al [6] compared the number and position of edges in successive video frames, allowing for global camera motion by aligning edges between frames. Transitions can be detected and classified by examining the percentages of entering and exiting pixels. Canny [7] suggested the replacement of Sobel filtering with more robust methods, with the aim of defining edges more clearly, particularly in very bright or dark scenes.

Borezcky and Rowe [8] compared several different methods of shot boundary detection using a variety of video content types. They concluded that histogram-based methods were more successful than others, but that shot boundary detection thresholds must be guided by the target application. A feature of this paper is the unusually large amount of video test data used for evaluation of the various techniques. This large test set, which is lacking in the evaluation of many other systems, allowed for a fuller analysis of the algorithms investigated.

Tague [9] described formal Information Retrieval evaluation methods and their use in the analysis of experimentation results. Of particular interest are the evaluation criteria of recall, precision, and fallout, elements of which we employ in section 5.

## 3.0 Description of system

### 3.1 Histogram Creation

We modelled our colour histogram segmentation system on those described in [1] and [2]. The technique used compares successive frames based upon three 64-bin histograms (one of luminance, and two of chrominance). These three histograms are then concatenated to form a single a N-dimensional vector, where N is the total number of bins in all three histograms (in our case N=192).

### 3.2 Cosine Similarity Measure

We use the dissimilarity analogue of the cosine measure [1] for comparing the histograms of adjacent frames. The two N-dimensional vectors, A and B, represent the colour signatures of the frames. The distance $D_{cos}(A,B)$ between vectors A and B is given by:

$$D_{\cos}(A,B)=1-\frac{\sum_{i=i}^{N}(a_i \bullet b_i)}{\sum_{i=1}^{N} a_i^2 \bullet \sum_{i=1}^{N} b_i^2}$$

where $a_i$ is one bin in $A$ and $b_i$ is the corresponding bin in $B$. As can be seen the cosine measure is basically the dot product of two unit vectors. The result is the cosine of the angle between the two vectors subtracted from one. Therefore a small value for $D_{cos}$ indicates that the frames being considered are similar, while a large $D_{cos}$ value indicates dissimilarity.

A high cosine value can indicate one of two things. Firstly, it can (and should) signal that a shot boundary has occurred. Secondly, it can be the result of ‘noise’ in the video sequence, which may be caused by fast camera motion, a change in lighting conditions, computer-generated effects, or anything that causes a perceptual change in the video sequence without being an actual shot boundary.

As can be seen, the algorithm used is quite simple compared to some previously published examples. This is representative of the fact that the system is currently a work-in-progress. Also, previous studies [8] have shown that simpler algorithms often outperform more complex ones on large heterogeneous video test sets, due to the absence of “hidden” variables and simpler relationships between threshold settings and results.

## 3.3 Shot Boundary Detection

Figure 1 shows the results obtained from a short 2000-frame segment (1 min, 20 sec) of video, taken from an episode of the soap “Home and Away”. The cosine values are plotted on the Y-axis. The peaks indicate high difference values and therefore denote shot boundaries. In this particular segment all the shot boundaries are cuts, i.e. no gradual transitions occur. As can be seen, no shot boundaries occur until around frame 550. The small peaks and bumps represent the ‘noise’ mentioned above.

In this particular sequence it can be seen that the noise levels are quite low. This makes it easy to detect a real shot boundary using a fixed threshold, shown by the horizontal line at cosine value 0.05. The transitions themselves are also very distinct. Thus, these results represent the ideal conditions for correctly identifying shot cuts using histogram-based detection.

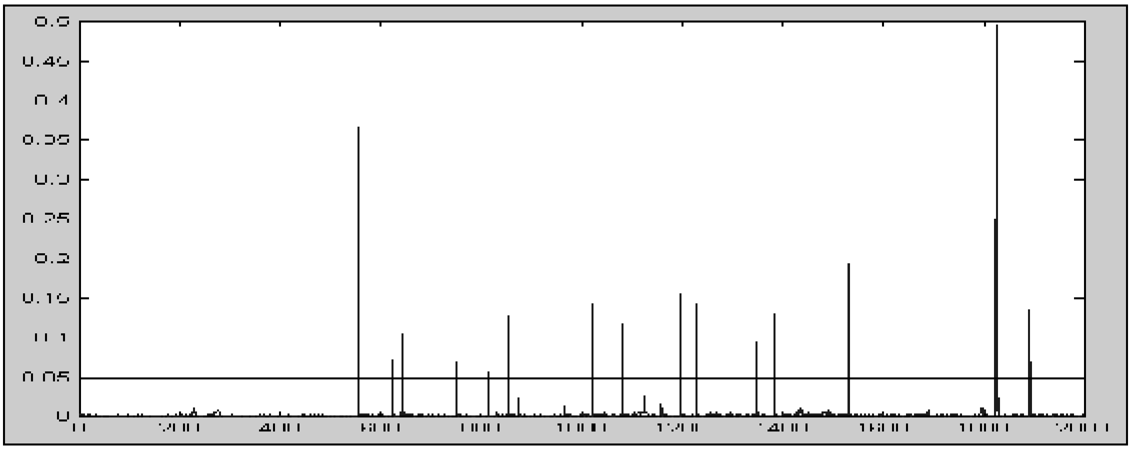


**Figure 1. Cosine similarity results for 2000 frames of video from “Home and Away”.**

Unfortunately, these ideal conditions rarely exist in real-world television broadcasts, which is our target application environment. Modern television productions make extensive use of effects, including:

- Fades, dissolves and other gradual transitions.
- Computer-generated effects (e.g. morphing of one shot into another, especially in adverts).
- Split-screen techniques (e.g. ticker-tape, interviews, etc. where 2 or more "screens" appear on-screen).
- Global camera motion (e.g. zooming and panning shots which are used in almost all productions).

All of these techniques introduce noise into the video sequence, which may be either falsely identified as a shot boundary, or serve to mask the presence of real shot boundaries.

An example of the former case is a split-screen interview, as are common on TV news programs. In such cases the anchorperson remains constant in one window, with the second window switching between different reporters, and shots of the news event. The changes in the second window may indicate that a transition has occurred where in reality it is all one single logical video shot.

An example of the other effect of noise, where effects mask a shot cut, is the use of slow dissolves or morphs between scenes. In this case the change may be so gradual that the difference between consecutive frames is too low to detect.

Figure 2 is another 2000-frame sample. This piece of video is the end of a commercial break, returning to a program at around frame 1400. As can be seen, commercial breaks are usually noisy and hectic sequences. This is because, in comparison to programs, commercials typically have a huge number of cuts in a short space of time. Commercials also frequently include much more advanced visual effects than programs, frequently using computer generated techniques to distort, transform, and merge images. These facts make commercials some of the most difficult types of video to segment accurately. In contrast to figure 1, the same threshold (cosine value of 0.05) results in a large number of false positives.

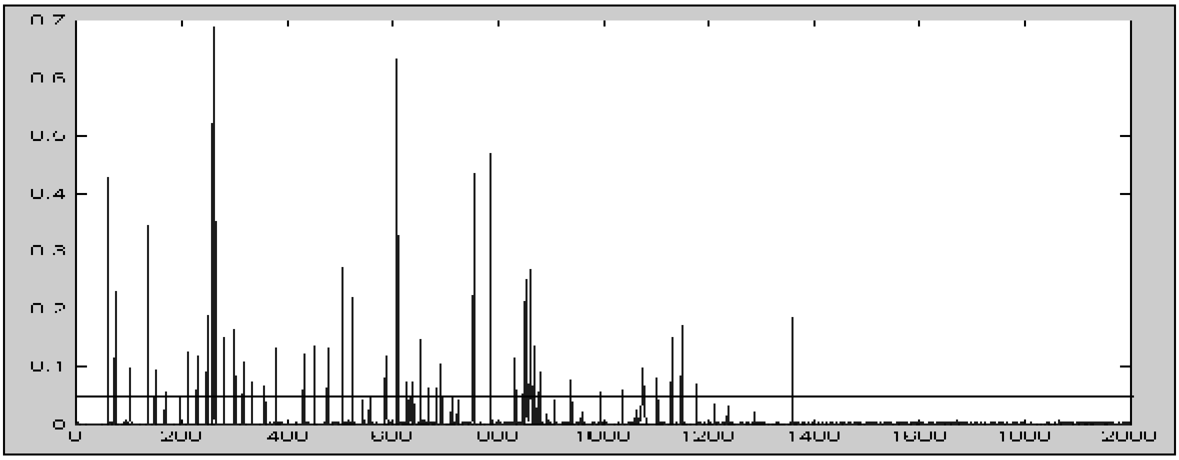


**Figure 2. Cosine similarity results for a noisy segment of video.**

### 3.4 Thresholds

To decide whether a shot boundary has occurred, it is necessary to set a threshold, or thresholds for the similarity between adjacent frames. Cosine similarity values above this threshold are logged as real shot boundaries, while values below this threshold are ignored.

To accurately segment broadcast video, it is necessary to balance the following two - apparently conflicting - points:

- The need to prevent detection of false shot boundaries, by setting a sufficiently high threshold level so as to insulate the detector from noise.
- The need to detect subtle shot transitions such as dissolves, by making the detector sensitive enough to recognise gradual change.

## 4.0 Description of Test Data

The test data we used in our work consisted of eight hours of broadcast television from a national TV station, comprising of all material broadcast from 1pm to 9pm on the 12th June 1998. The video was digitised in MPEG-1 format at a frame rate of 25 fps (total of 720,000 frames) and a resolution of 352*288 pixels (commonly known as the SIF standard). This was accomplished using a Pentium PC with a “Sphinx Pro” video capture board. For ease of manipulation, and to keep file sizes manageable, the video was digitised in 24 segments of 20 minutes each. Once captured, the video segments were transferred to a Sun Enterprise Server for further processing.

The test data incorporated a broad variety of program types, as well as a large number of commercials. Rather than sort the different content types into discrete test sets, the video was captured and stored “as is”. This ensures that any given 20-minute segment may contain a variety of video content types. Thus the test set replicates the type of heterogeneous video most commonly seen on broadcast television.

To provide an authoritative guide to the test set, the locations and types of shot, scene, and program boundaries were manually analysed to give a series of detailed log files, each representing a 20-minute video segment. This collection of log files is referred to as the *baseline***,** and represents a huge investment in time. The baseline allows us to compare the results generated by our detection algorithms to a ground truth. It also enables us to calculate statistics such as the number of frames and shot boundaries found in each content type. As noted above, the baseline contains extremely detailed semantic information. Although this paper focuses only on shot detection, the richness of the baseline will enable more complex methods to be evaluated successfully.

While our paper focuses on similar topics to that of Borezcky and Rowe[8], we employ a substantially larger test set, which is not pre-sorted into specific content types, but is rather representative of the complexity and variety of television broadcast video. We focus exclusively on one algorithm (the Cosine Similarity Measure), which we have tested extensively. We have also produced a more content-rich baseline with which to compare our results.

Below are the video types contained in the eight hours of test data, divided by segment. Also listed are detailed descriptions of the video test set, with figures extracted from the manually generated baseline files. Table 1 shows the test set analysed by video segment.

| **Video Segment** | # of cuts | # of gradual transitions | Ratio | **Video Segment** | # of cuts | # of gradual transitions | Ratio |
|---|---|---|---|---|---|---|---|
| **1** | 194 | 45 | 4:1 | **13** | 263 | 47 | 6:1 |
| **2** | 230 | 28 | 8:1 | **14** | 304 | 29 | 10:1 |
| **3** | 210 | 79 | 3:1 | **15** | 153 | 29 | 5:1 |
| **4** | 207 | 52 | 4:1 | **16** | 172 | 28 | 6:1 |
| **5** | 253 | 39 | 6:1 | **17** | 209 | 22 | 9:1 |
| **6** | 139 | 2 | 69:1 | **18** | 198 | 48 | 4:1 |
| **7** | 191 | 17 | 11:1 | **19** | 242 | 49 | 5:1 |
| **8** | 204 | 47 | 4:1 | **20** | 302 | 30 | 10:1 |
| **9** | 159 | 6 | 26:1 | **21** | 277 | 25 | 11:1 |
| **10** | 145 | 29 | 5:1 | **22** | 244 | 52 | 5:1 |
| **11** | 227 | 22 | 10:1 | **23** | 252 | 2 | 126 |
| **12** | 323 | 20 | 16:1 | **24** | 258 | 33 | 8:1 |
| **Note**: Each segment is 30000 frames (20 minutes). | | | | | | | |

**Table 1. Video test set analysed by segment**

Table 2 shows the test set analysed by video content type.

1. News & weather: This includes two news broadcasts, one of 25 minutes and one of an hour. Also included was a 10-minute episode of Nuacht, the Irish language news.
2. Soaps: Included are four complete episodes of soaps. They are “Home and Away”, “Emmerdale”, “Fair City”, and “Shortland Street”. Each episode was 30 minutes long.
3. Cooking: This consisted of one half-hour cookery program. Surprisingly, this segment included many subtle shot transitions.
4. Magazine/Chat show: This was one 110-minute episode of a popular magazine show. Included are fitness, music, gardening, and film features, as well as interviews. This program contains a good mix of content types and shot transitions.
5. Quiz show: One half-hour episode of a popular local quiz show.
6. Documentary: A short (15 minute) documentary charting the lives of some of the famous people of the 20th century. Includes lots of black and white footage.
7. Comedy/Drama: One full episode of “Touched by an Angel” (55 minutes) and one of “Keeping up Appearances” (35 minutes).
8. Commercials: Mixed among the above are a large number of commercials. As always, these provide varied and challenging material for segmentation.

| Video Type | # of Frames | # of Cuts | # of Gradual Transitions | Ratio of Cuts to Gradual Transitions |
|---|---|---|---|---|
| News and weather | 134540 | 598 | 69 | 9:1 |
| Soaps | 144958 | 909 | 94 | 10:1 |
| Cookery programs | 37370 | 188 | 42 | 4:1 |
| Magazine/chat shows | 134985 | 759 | 64 | 12:1 |
| Quiz shows | 29093 | 269 | 4 | 67:1 |
| Documentary | 7494 | 47 | 23 | 2:1 |
| Comedy/Drama | 110618 | 839 | 72 | 12:1 |
| Commercials | 106976 | 1771 | 415 | 4:1 |
| **Total** | **706034** | **5380** | **779** | **(average) : 15:1** |

**Table 2. Video test set analysed by video content type**

# 5.0 Results

## 5.1 Aims and Methods

Before beginning the experiments proper, our segmentation algorithm was tuned on a number of small (5-10 minute) video segments extracted from the test set. These training runs enabled us to determine useful threshold levels.

In reporting our experimental results, we use recall and precision to evaluate system performance. Recall is the proportion of shot boundaries correctly identified by the system to the total number of shot boundaries present. Precision is the proportion of correct shot boundaries identified by the system to the total number of shot boundaries identified by the system. We express recall and precision as:

$$\textit{Recall} = \frac{\textit{Number of shot boundaries correctly identified by system}}{\textit{Total number of shot boundaries}} \qquad \textit{Precision} = \frac{\textit{Number of shot boundaries correctly identified by system}}{\textit{Total number of shot boundaries identified by system}}$$

Ideally, both recall and precision should equal 1. This would indicate that we have identified all existing shot boundaries correctly, without identifying any false boundaries.

Although precision and recall are well established in traditional text-based information retrieval, there is as yet no standard measure for evaluating video retrieval systems. Other possible measures, which may be utilised in future experiments, include fallout and the E-measure [10].

Recall and precision are useful evaluation tools. However, by expressing results as a simple percentage they can give a misleading indication of system performance. For this reason we have chosen to include a summary of the actual figures obtained during the experiments. In reporting our results we chose a representative sample from the thresholds tested for inclusion in each graph. These samples include threshold levels that resulted in good results for all segments, and also samples from each extreme of the recall/precision spectrum.

Thresholds are not considered if they result in recall or precision figures of less that 0.5 for a majority of segments or content types. Although low recall or precision may be acceptable in some specialised applications, segmentation of large amounts of varied video requires reasonable levels to be useful.

In conducting the experiments we addressed specific questions with regard to shot boundary detection thresholds for broadcast video. We focused on the selection of correct thresholds for a mixture of video content types, as well as tailoring specific thresholds towards specific types. In particular, we were interested to see if pre-set, fixed thresholds were suitable for such a varied test set. The experiments conducted and results obtained are described below.

## 5.2 Do Fixed Threshold Values Perform Adequately on Video Containing Multiple Content Types?

To address the question of whether we can hard-code threshold values, we ran the algorithm, using a range of threshold values, on all 24 segments of the test set. A boundary detected by the algorithm was said to be correct if it was within one frame of a boundary listed in the baseline.

Recall and precision graphs are presented as figures 3 and 4 respectively. A summary of results for the full video test set is also shown in table 3. The following points can be noted from these results:

1. On the middle threshold, the algorithm averages 85% recall, and 88% precision. However there is noticeable variation between the segments, as the algorithm performs better on different segments with different thresholds.
2. The algorithm performed poorly on segment 3 when compared to the rest of the test set. Even at the lowest threshold level, recall was only 75%, with a precision of 46%. This segment includes several commercial breaks. It also includes a lot of black and white footage from a documentary program. The colour-based method obviously has its discriminatory power reduced here, leading to poorer results.
3. The algorithm performed best on segment 6, typically achieving 98% precision and recall. This segment contains part of an episode of a magazine/chat show. Significantly there were no commercial breaks during this sequence. As commercials are generally the most difficult type of video to segment, this helps to explain the good results. Also, as noted in table 2, this segment has a huge ratio (69:1) of cuts to gradual transitions. The lack of difficult transitions makes for quite easy segmentation.
4. Lowering the threshold below a certain level does not guarantee better recall. Typically, once this level is reached (about 0.015 for our system), the increase in recall for a given threshold reduction is quite small, and is accompanied by a much larger loss of precision.
5. The opposite is also true, in that raising the threshold beyond a certain point gives decreasing precision results with rapidly falling recall.
6. For a majority (65%) of missed shot boundaries, examination of the raw data revealed a significant (>0.0075) cosine value for the appropriate frame pair. In these cases the fault of non-detection lies with the threshold selection, and not the detection ability of the algorithm itself. Attempting to employ a lower fixed threshold to detect these shot boundaries would result in a drastic decrease in precision. This suggests that an intelligent means of adaptive thresholding, perhaps using a known and reasonable threshold level as a starting point, could significantly improve upon the results obtained here. The need to improve methods of eliminating noise in the video stream is also a vital step if improvements are to be made in this area.

| | Total # of shot boundaries | # correctly identified | # falsely identified | # missed | Recall | Precision |
|---|---|---|---|---|---|---|
| Threshold 1 (0.010) | 6159 | 5689 | 3775 | 470 | 92 | 60 |
| Threshold 2 (0.020) | | 5472 | 1504 | 687 | 89 | 78 |
| Threshold 3 (0.035) | | 5163 | 731 | 996 | 85 | 88 |
| Threshold 4 (0.060) | | 4508 | 431 | 1651 | 74 | 92 |
| Threshold 5 (0.15) | | 2789 | 195 | 3370 | 45 | 94 |

**Table 3 – Total figures for the entire test set of 720000 frames.**

Figure 3: Recall for 20 Video Segments with 5 Sample Thresholds

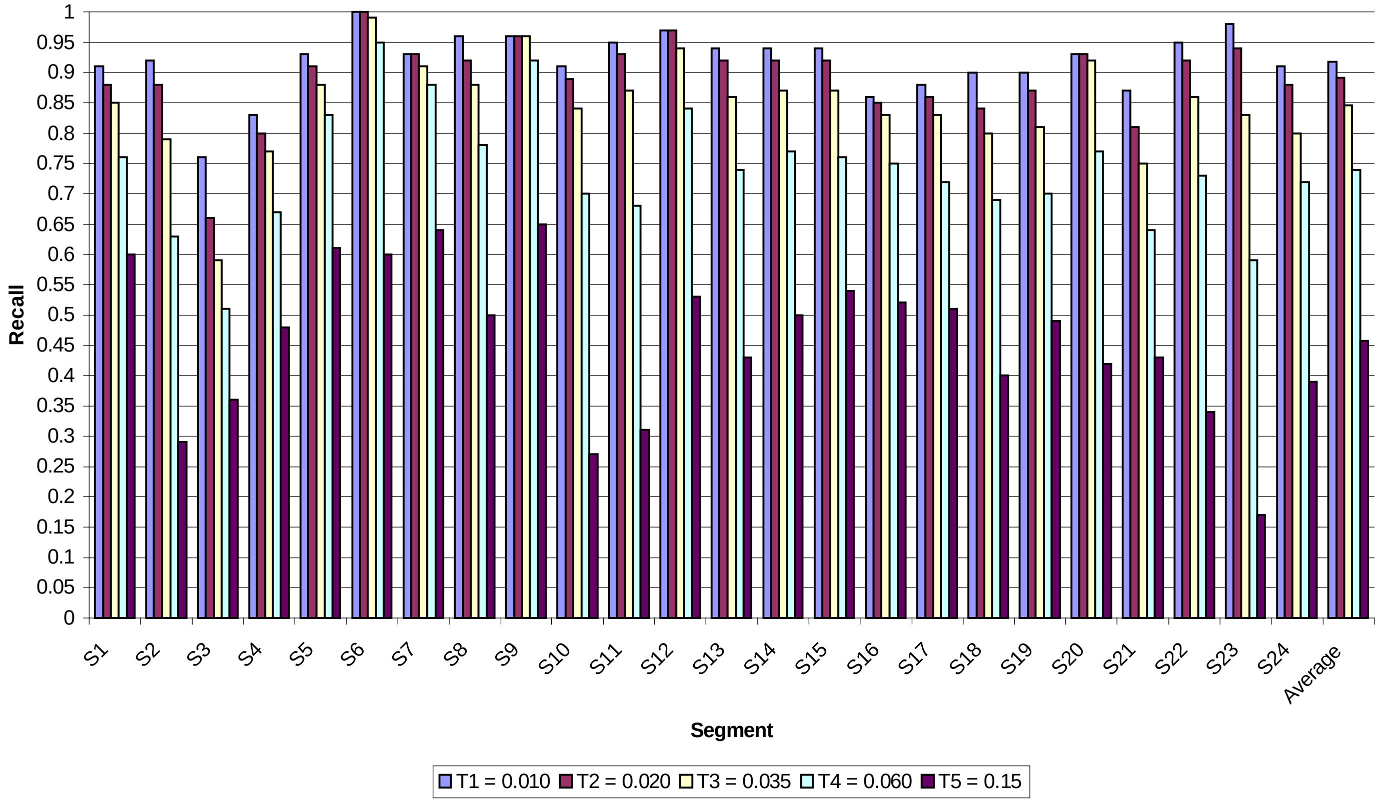
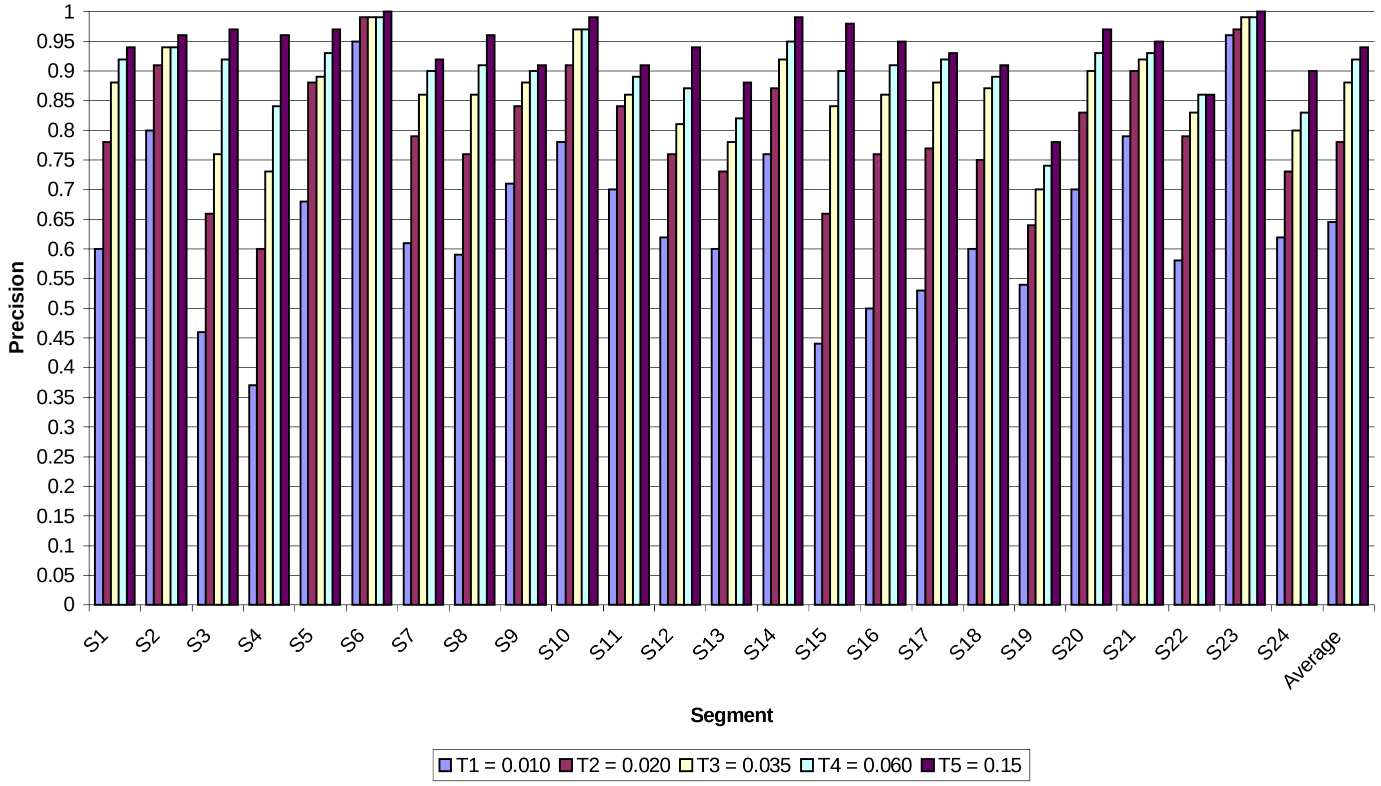


Figure 4: Precision for 20 Video Segments with 5 Sample Thresholds



## 5.2 Do Varied Video Content Types Affect the Results Obtained from Different Fixed Thresholds?

We have seen the results obtained by a selection of shot boundary detection thresholds on the 24 segments of the video test set. However, these results tell us little about why a particular segment/threshold combination is producing a particular result. Our second set of experiments explored how effective the system was at segmenting specific content types. This would show how different content type/threshold settings interacted and affected the overall result.

This second experiment requires that we examine the video test set by video content type, rather than by segment, as each segment contains a mix of content types. We employed the same five threshold settings as for section 5.1. Figures 5 and 6 show the recall and precision graphs for the eight video content types contained in the test set. The following general points can be noted:

1. Threshold levels can affect different video content types in markedly different ways. In some cases (for example between the "news" and "soaps" content types), the results are close enough to consider a single threshold value. However, the results for even these similar content types can vary by 20% for the same threshold.
2. In the case of dissimilar content types (commercials, documentary, cookery), the same threshold can produce completely different results. For example, a threshold of 0.035 results in 94% recall for the magazine/chat show content type, 79% for the commercials content type, and 9% for the documentary content type. Although this threshold setting performed best overall in section 5.1, these results show that it is totally inadequate for the mix of dissolves and black and white footage found in the documentary content type.
3. Again, examination of the missed shot boundaries revealed a majority (65%) that had significant cosine values. Had a reliable form of intelligent thresholding been employed in the algorithm, recall scores, which are currently quite poor, could be greatly improved.

We can also comment on the different video content types:

1. Commercials: This algorithm performed reasonably well when segmenting this content type, considering the complexity of some of the shot transitions present. Using a threshold setting of 0.035 (Threshold 3), 79% recall and 74% precision was achieved. However, moving to either end of the recall/precision spectrum quickly led to unbalanced results, which would prove unacceptable in our target application.

2. Soaps: This content type generally presented no difficulties to the system. On the middle threshold setting a precision of 92% was achieved. The low recall score of 76% was traced to the starting and ending credits of "Home and Away". This sequence contains some very difficult gradual transitions, which even our human baseline-creators found difficult to segment accurately.

3. News: As for soaps, the result for the news content type was generally good. Again, moving to extremes of the recall/precision spectrum led to poor results. When using a balanced threshold, recall and precision values averaged about 86%-87%.

4. Cookery: This content type proved difficult to accurately segment due to a large number of slow scene dissolves. Although low threshold settings (<0.030) afforded good recall, (85%-90%), the corresponding precision scores were poor (35%-50%). At the medium threshold settings (0.30-0.40) precision values are still quite poor (71%) although recall has improved to 83%. High thresholds, as expected, led to poor recall values (<50%). This content type demands an improved detection system before it can be segmented with confidence.

5. Magazine/chat show: Despite the varied content of this video type, the system performed quite well, probably due to the relative low ratio (12:1) of cuts to gradual transitions. Low and medium threshold values returned reasonable results with recall and precision ranging from 78%-98%. Higher threshold levels returned poor (<50%) recall scores, but gave little improvements in precision. This indicates that some proportion of the shot boundaries is being masked by noise in the video sequence.

6. Quiz show: The system performed well on this content type, which included few gradual shot boundaries. Low (<0.020) threshold values led to high (98%) recall scores with acceptable precision (78%). A more balanced threshold led to recall and precision scores of 97%. High threshold values are not suited to this content type, resulting in unacceptable (<55%) recall values.

7. Comedy/Drama: All threshold levels delivered good precision (>85%) on this content type, indicating that a high percentage of the shot boundaries are well defined. Recall dropped sharply from around 88% to 50% as the threshold was raised above 0.070, making such a setting unacceptable, even though doing so gave a precision of 100%.

8. Documentary: The system performed very poorly on this content type. This was due to the low ratio (2:1) of cuts to gradual shot boundaries and the large amounts of poor-quality black and white footage used as part of the documentary. At the medium threshold, which returned good results on all of the other content types, recall was only 9% and precision was 60%. In contrast to some of the other content types, best results were achieved with low (<0.015) threshold values, which gave around 64% recall and 52% precision. This content type highlights the difficulties of selecting one global threshold for broadcast video. Although the low scores achieved here were balanced in the overall system graphs (section 5.1) by the results obtained elsewhere, it is obvious that this content type demands more advanced shot boundary detection methods than our system currently offers.

**Figure 5: Recall for 8 Video Content Types with 5 Thresholds**

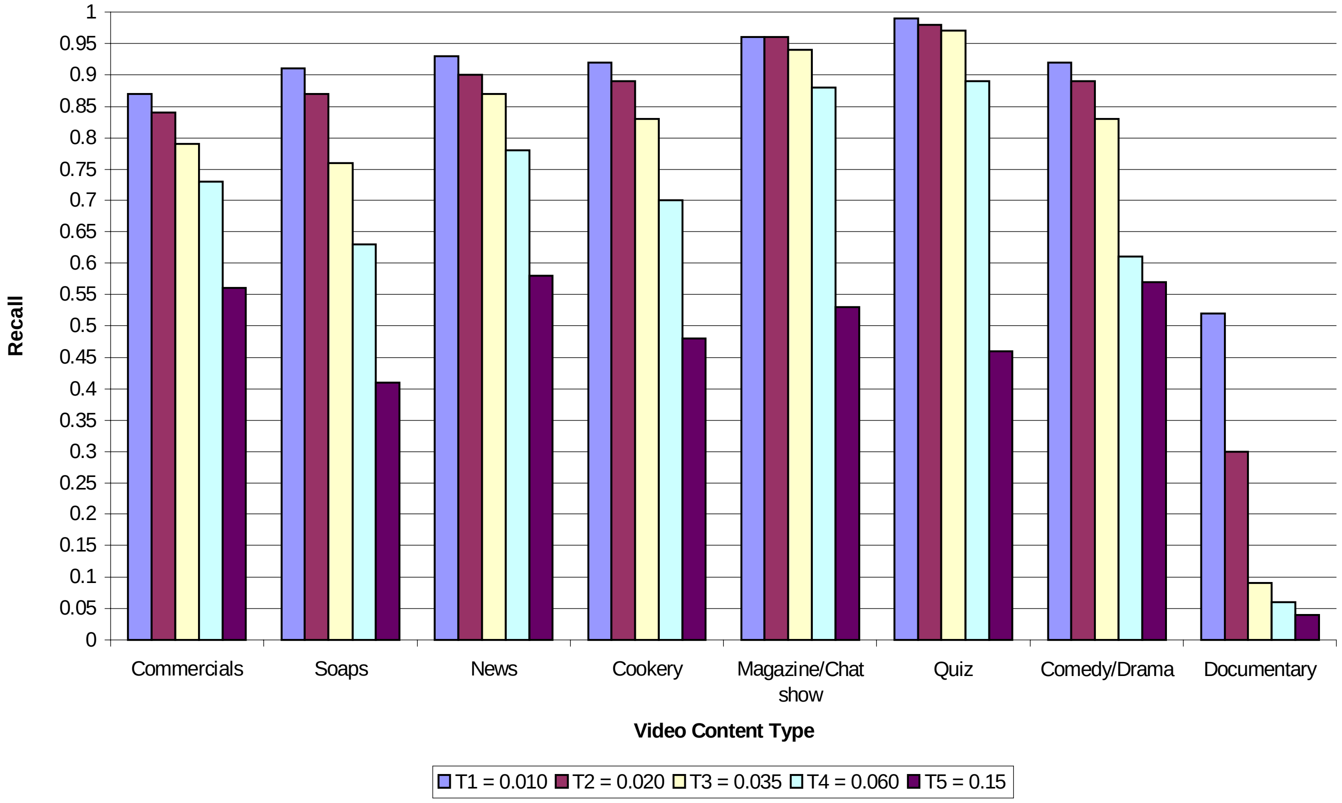
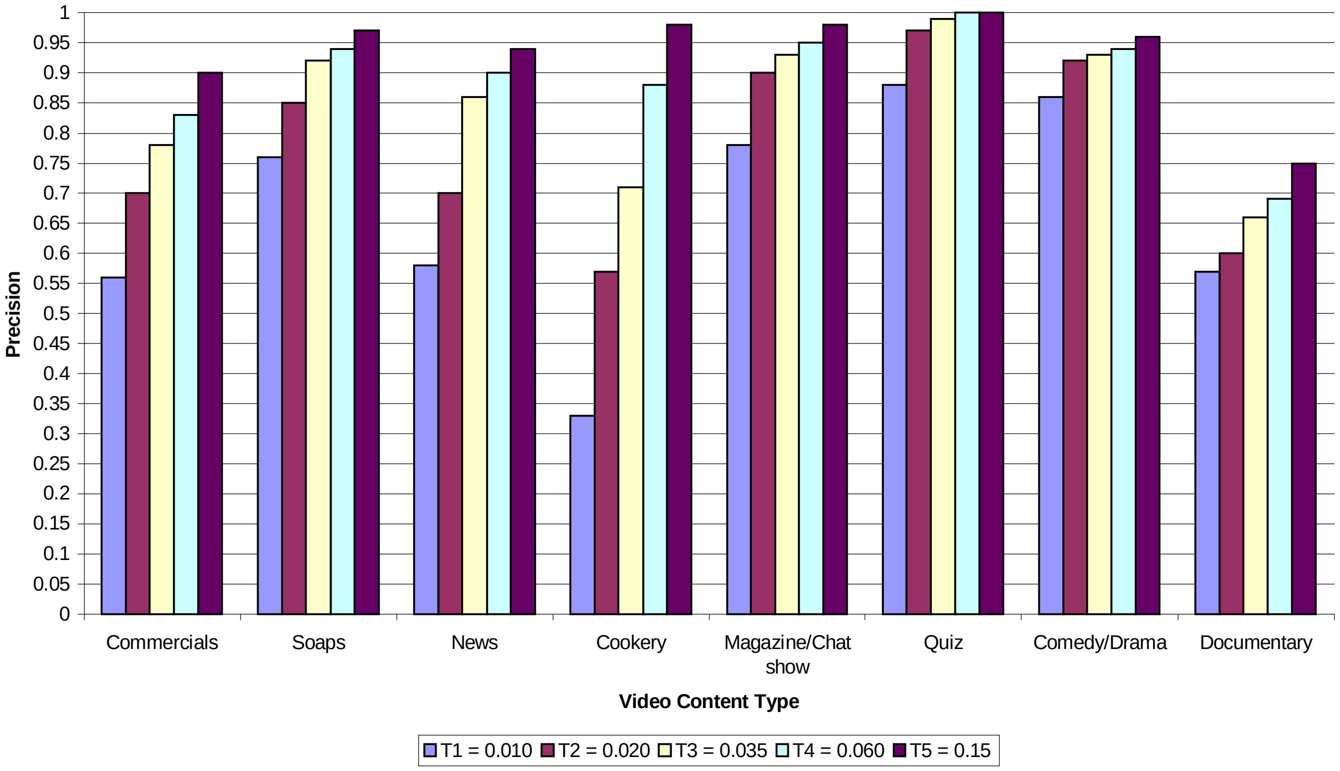


**Figure 6: Precision for 8 Video Content Types with 5 Thresholds**



## 6.0 Conclusions

Although our system for segmenting video into shots performs quite well when using a reasonable threshold, there is clearly room for improvement. By examining the results obtained for segment 3 (see section 5.21), it is clear that the presence, or absence, of certain video content types can have a large effect on the accuracy of shot boundary detection systems.

Also, a different test set may well respond in a radically different way to the thresholds employed here. The huge diversity of broadcast video ensures that any attempt to define one definitive boundary detection threshold will be futile. For our intended application, the automatic indexing of TV originated broadcast video, the manual selection of a threshold for particular video sets is also inappropriate.

One solution to this dilemma is to allow semi-automatic selection of thresholds depending on the program type as taken from a television schedule. Thus a range of thresholds for various program types (for example news, drama and documentary) would be available depending to the current content type. However, we believe that even a threshold tailored to some instances of a content type may not perform well on other instances of the same type.

Based on the results obtained, particularly with reference to the experimental results shown in section 5.2, we believe that fixed thresholds are inadequate to deal with the variety of different video content types found in broadcast television.

The challenge facing automatic video indexing and retrieval is to design systems that can accurately segment large amounts of heterogeneous video. In our opinion, this requires the development of systems that employ adaptive thresholding methods, perhaps using the television schedule method discussed above to generate a starting content-specific value. It is our hope that such systems will help to solve the problem of detecting subtle gradual transitions without unacceptably lowering precision. We plan to apply such methods to our existing system, amongst other improvements suggested by the results presented here, and evaluate the results. Other areas where we will continue to work include the selection of single and multiple representative frames for shots, the automatic combination of constituent shots into scenes, and the development of alternate means of shot and scene boundary detection.

## 7.0 Acknowledgements

The authors would like to thank the National Software Directorate for financial support, under contract number SP05-1997. We would also like to thank the referees for their helpful comments.

## 8.0 References

1. X. U. Cabedo and S. K. Bhattacharjee: Shot detection tools in digital video, *in Proceedings Non-linear Model Based Image Analysis 1998*, Springer Verlag, pp 121-126, Glasgow, July 1998.

2. H. J. Zhang, A. Kankanhalli and S. W. Smoliar, Automatic partitioning of full-motion video, in *Multimedia Systems*, volume 1, pages 10-28, 1993.

3. A. Nagasaka and Y. Tanaka, Automatic video indexing and full-video search for object appearances, in *Visual Database Systems II*, Elsevier Science Publishers, pages 113-117, 1992.

4. Y. Gong, C. H. Chuan, and G. Xiaoyi, Image indexing and retrieval based on colour histograms, in *Multimedia Tools and Applications*, volume 2, pages 133-156, 1996.

5. J. Meng, Y. Juan, S.-F. Chang, Scene change detection in an MPEG compressed video sequence, in *IS&T/SPIE Symposium Proceedings*, volume 2419, February 1995.

6. R. Zabih, J. Miller, and K. Mai, A feature-based algorithm for detecting and classifying scene breaks, in *Proceedings ACM Multimedia 95*, pages 189-200, November 1993.

7. J. Canny, A computational approach to edge detection, in *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8(6), pages 679-698, 1986.

8. J. Boreczky and L.A. Rowe, Comparison of video shot boundary detection techniques, in *IS&T/SPIE proceedings: Storage and Retrieval for Images and Video Databases IV*, volume 2670, pages 170-179, February 1996.

9. M. Tague, The pragmatics of Information Retrieval experimentation, in *Information Retrieval Experiment*, Karen Sparck Jones Ed., Buttersworth, pages 59-102, 1981.

10. C. J. Van Rijsbergen, Information Retrieval, Buttersworth, 1979.