Dublin City University at the TRECVid 2008 BBC Rushes Summarisation Task

Hervé Bredin, Daragh Byrne, Hyowon Lee, Noel E. O'Connor and Gareth J.F. Jones
Centre for Digital Video Processing and Adaptive Information Cluster
Dublin City University, Ireland
herve@eeng.dcu.ie

ABSTRACT

We describe the video summarisation systems submitted by Dublin City University to the TRECVid 2008 BBC Rushes Summarisation task. We introduce a new approach to redundant video summarisation based on principal component analysis and linear discriminant analysis. The resulting low dimensional representation of each shot offers a simple way to compare and select representative shots of the original video. The final summary is constructed as a dynamic storyboard. Both types of summaries were evaluated and the results are discussed.

Categories and Subject Descriptors

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

General Terms

Algorithms, Performance

1. INTRODUCTION

The TRECVid Rushes Summarisation Task 2008 seeks to summarise raw un-edited rushes content (extra video, B-rolls footage) provided by the BBC [3]. Rushes contain many frames or sequences of frames that are highly repetitive and often contain much redundant/junk content. The target for the end summary was 2% of the original content which reduces 30-60 minute content to 20-40 seconds approximately. This paper outlines our approach to rushes summarisation within the 2008 task [3].

From last year's activity [4], we know that summarisation of raw rushes content is particularly difficult with challenges unique to this type of content. Generally, a good video summary should enable a user to quickly and efficiently interrogate the video content. As such, it should include segments of the original which are highly representative of its content and convey its core concepts. For rushes, summaries must additionally seek to remove redundant content such as colour bars and blank frames while detecting and reducing repeated content such as retakes. Surprisingly, the most effective approach in last year's activity was the highly accelerated baseline and this motivates the exploration of how to best achieve balance between content coverage and content reduction. This is the focus of the approach outlined in this paper. Section 2 describes the extraction of features (shot boundary detection, junk detection), consequently used as input for the summarisation algorithm presented in Section 3 and based on binary low-dimensional shot footprints. The construction of the final summary is described in Section 4. Finally, results from the benchmarking activity are outlined and discussed, respectively in Sections 5 and 6. The paper concludes in Section 7.

2. FEATURE EXTRACTION

2.1 Colour Histograms

A 3-dimensional RGB colour histogram h is computed for each frame f of the video. With 8 bins per channel, it can be considered as a ($D = 8 \times 8 \times 8 =$) 512-dimensional feature vector.

2.2 Useless Frame Detection

Some monochromatic or colour bar frames might appear at the beginning or the end of the BBC rushes video, or even between two shots. These useless frames contain a limited set of colours. The number of colours N_{colour} of a given frame f is estimated using its colour histogram h, with the following equation:

$$N_{\text{colour}}(f) \simeq \min \left\{ K \in [\![1, D]\!] \middle| \sum_{i=1}^{K} b_i(h) > 80\% \right\}$$

where $b_i(h)$ is the value of the i^{th} bins of h (ranked in descending order). Frame f is classified as useless if $N_{\text{colour}}(f) < 10$.

2.3 Shot Boundary Detection

Shot boundary detection is performed using an adaptive thresholding of the *Bhattacharyya* distance between the histograms of every two consecutive frames. Let us denote d_i the distance between frames i and $i + 1$. The adaptive threshold is set to

$$t_i = \max \left(\alpha \cdot \text{median} \left\{ d_{i-\frac{\Delta}{2}} \dots d_{i+\frac{\Delta}{2}} \right\}, T_{\text{noise}} \right)$$

A shot boundary is detected between frames i and $i + 1$ if $d_i > t_i$. The value of T_{noise} and α were set heuristically

TVS'08, October 31, 2008, Vancouver, British Columbia, Canada.
Copyright 2008 ACM 978-1-60558-309-9/08/10 ...$5.00.

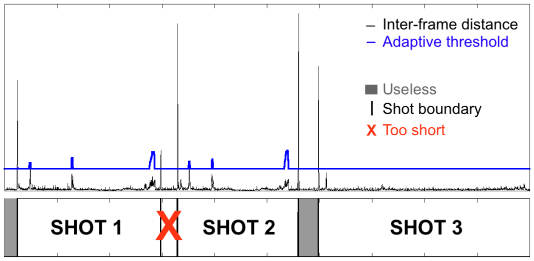

Figure 1: Creating list of shots. Top: Shot boundary detection (inter-frame distance d_i vs. adaptive threshold t_i). Bottom: Detected boundaries and useless frames.

to respectively minimise false detection due to video encoding noise in the case of segments with no (or little) motion, and maximise the correct detection rate. Chosen values are $T_{\text{noise}} = 0.1$ and $\alpha = 5$, with $\Delta = 25$ frames.

2.4 Shot List

Figure 1 illustrates useless frames and shot boundary detection on a 3 min 20 sec segment of video *MS215830*. A list of shots is straightforwardly composed by selecting non-useless video segments between two consecutive shot boundaries, with an additional constraint of a minimum duration of 10 seconds.

3. CONTENT ANALYSIS

At this point, a set $\mathcal{S} = \left\{\mathcal{S}_i\right\}_{i\in[\![1,N_s]\!]}$ of N_s shots is available. Each shot $\mathcal{S}_i$ contains L_i frames $\left\{f_k^i\right\}_{k\in[\![1,L_i]\!]}$, with their corresponding D-dimensional colour histogram h_k^i. In other words, each shot $\mathcal{S}_i$ is represented by a $L_i \times D$ matrix with dimensions therefore depending on the duration of the shot. We denote $\mathcal{H}$ the overall set of feature vectors:

$$\mathcal{H} = \bigcup_{i=1}^{N_s} \left\{ h_k^i \right\}_{k\in[\![1,L_i]\!]}$$

3.1 Dimensionality Reduction

Principal components and linear discriminant analyses are two linear transformation techniques allowing to greatly reduce feature space dimension while keeping most of the information relevant to our summarization approach (based on maximising the coverage of the original footage).

3.1.1 *Principal Components Analysis (PCA)*

PCA tries to find the directions (or components) in feature space that maximize the variance of the dataset [1]. It is applied on $\mathcal{H}$, only keeping the first 2 principal components:

$$\mathcal{V} = \mathrm{PCA}\left(\mathcal{H}\right) = \bigcup_{i=1}^{N_s} \left\{ v_k^i \right\}_{k\in[\![1,L_i]\!]} \quad \text{with } v_k^i \in \mathbb{R}^2$$

3.1.2 *Linear Discriminant Analysis (LDA)*

Unlike PCA, LDA also tries to minimise the overall intra-class variance [1]. Therefore to apply LDA, we modify the data set $\mathcal{H}$ into $\overline{\mathcal{H}}$ by adding class information to each sample: the class of a frame is the number of the shot to which it belongs:

$$\overline{\mathcal{H}} = \bigcup_{i=1}^{N_s} \left\{ \left(h_k^i, i\right) \right\}_{k\in[\![1,L_i]\!]}$$

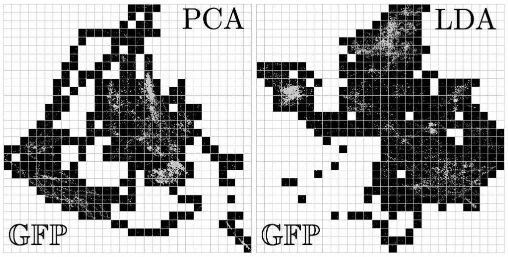

Figure 2: Global footprints $\mathbb{GFP}$ for video *MS215830*.

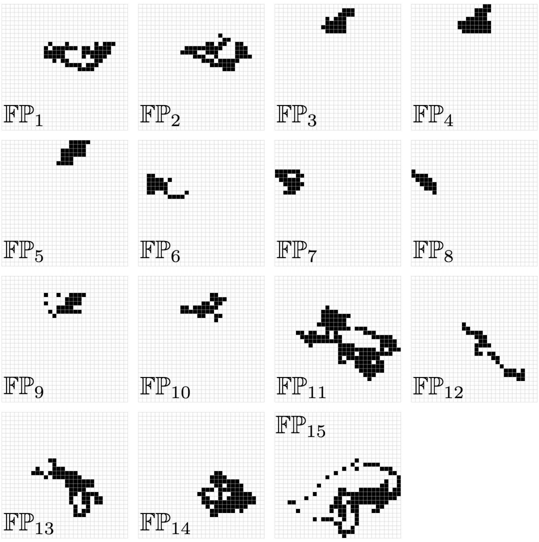

Figure 3: Shot footprints $\mathbb{FP}_1$ to $\mathbb{FP}_{15}$ for video *MS215830*. Two redundant shots have very similar footprints (shots 1 and 2 actually contain the same scene, as do shots 3, 4 and 5).

LDA is applied on $\overline{\mathcal{H}}$ and the first 2 components are kept:

$$\overline{\mathcal{V}} = \mathrm{LDA}\left(\overline{\mathcal{H}}\right) = \bigcup_{i=1}^{N_s} \left\{ \overline{v}_k^i \right\}_{k\in[\![1,L_i]\!]} \quad \text{with } \overline{v}_k^i \in \mathbb{R}^2$$

The resulting 2-dimensional vectors v_k^i and $\overline{v}_k^i$ are plotted in Figure 2 on the left and right respectively. Each grey dot corresponds to one frame of the video (x- and y-coordinates correspond to the first and second dimensions in the projection space). This visual representation of videos is inspired by the work described in [2].

3.2 Footprints

A 2-dimensional 30-bin histogram is then computed from $\mathcal{V}$ (or $\overline{\mathcal{V}}$). It is subsequently binarised into what we term the **global footprint of the video** – denoted $\mathbb{GFP}$ in the rest of the paper. As illustrated in Figure 2, the value of a bin is set to 0 (white) if the bin is empty, and 1 (black) otherwise.

Similarly, for each shot $\mathcal{S}_i$, a footprint $\mathbb{FP}_i$ is obtained based on their set of 2-dimensional vectors $\left\{v_k^i\right\}_{k\in[\![1,L_i]\!]}$ (or

$\left\{\overline{v}_k^i\right\}_k$). Shot footprints for video *MS215830* are shown in Figure 3. Then we define a set of footprint operations:

- $\mathbb{FP}_1 \cup \mathbb{FP}_2$ – a bin is set to 1 if it is 1 in $\mathbb{FP}_1$ **or** $\mathbb{FP}_2$.
- $\mathbb{FP}_1 \cap \mathbb{FP}_2$ – a bin is set to 1 if it is 1 in $\mathbb{FP}_1$ **and** $\mathbb{FP}_2$.
- $\#\{\mathbb{FP}\}$ is the number of bins whose value equals 1.

3.3 Video Abstraction

Our approach to video abstraction is divided into two steps. First, a set of representative shots is selected under two constraints: to maximise the content coverage of the original video and avoid as much as possible the selection of redundant shots in the final summary. Once "good" shots are selected, a second step of segment selection and acceleration is applied to reach the 2% duration ratio required by the TRECVid summarisation guidelines.

3.3.1 Selection of Representative Shots

The iterative selection of representative shots is performed using the algorithm below.

```
[1] Initialisation
1a. Selection of the shot with maximum coverage:
        $s(1) = \underset{i \in [\![1, N_s]\!]}{\operatorname{argmax}} \#\{\mathbb{FP}_i\}$
1b. Number of selected shots: $N = 1$
[2] Update current footprint: $\mathbb{CFP} = \bigcup_{i=1}^{N} \mathbb{FP}_{s(i)}$
[3] Stop if $\dfrac{\#\{\mathbb{CFP}\}}{\#\{\mathbb{GFP}\}} > r$
[4] Iteration
4a. Find shots whose footprint has minimum inclusion in the current footprint:
        $\mathcal{I}_{\text{min}} = \underset{i \in [\![1, N_s]\!]}{\operatorname{argmin}} \dfrac{\#\{\mathbb{FP}_i \cap \mathbb{CFP}\}}{\#\{\mathbb{FP}_i\}}$
4b. Among them, find the shot that results in the maximum increase in coverage:
        $s(N+1) = \underset{i \in \mathcal{I}_{\text{min}}}{\operatorname{argmax}} \#\{\mathbb{CFP} \cup \mathbb{FP}_i\}$
4c. Increment the number of selected shots: $N = N + 1$
4d. Go to step [3]
```

The idea is to iteratively select shots according to two quantities, until the expected coverage is achieved (step **3**, $r = 90\%$ in our case).

The first one (step 4**a**) measures how redundant a given shot is with the set of currently selected shots. The other one (step 4**b**) measures the increase in coverage resulting from the selection of a given shot.

At the end of this process, among the N_s original shots, only N of these (numbered $\mathcal{S}_{s(1)}$ to $\mathcal{S}_{s(N)}$) are selected to build the final summary. The other shots are considered redundant, and therefore rejected for full inclusion in the summary.

3.3.2 Selection of Representative Segments

Though only some of the shots are selected, their total duration is usually much longer than the requested 2% of the original video. Therefore, only shot segments are selected and shown in the final summary.

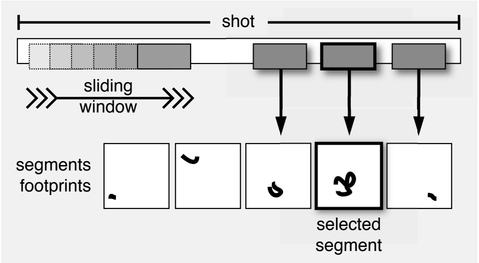

Figure 4: Selection of Representative Segments.

The expected 2% duration is shared between all shots, depending on their respective coverage – a shot $s(i)$ with a *larger* footprint is given a longer duration δ_i:

$$\delta_i = 2\% \cdot \delta \cdot \frac{\#\left\{\mathbb{FP}_{s(i)}\right\}}{\sum_{k=1}^{N} \#\left\{\mathbb{FP}_{s(k)}\right\}}$$

where δ is the duration of the original video.

Moreover, an additional step of play-back acceleration is applied to $\mathcal{S}_{s(i)}$, with acceleration factor $\alpha_{s(i)}$ depending on the measure $m_{s(i)}$ of the motion activity defined as:

$$m_{s(i)} = \frac{1}{L_{s(i)}} \sum_{k=1}^{L_{s(i)}} d_k^{s(i)}$$

where $d_k^{s(i)}$ is the inter-frame distance of k^{th} frame of shot $s(i)$ as defined in Section 2.3. $\alpha_{s(i)}$ is computed as an affine function of $m_{s(i)}$:

$$\alpha_{s(i)} = \mu - \phi \cdot m_{s(i)}$$

where ϕ and μ are chosen in a way that the acceleration factor will have a minimum value of 1 and a maximum of 5. In other words, a shot with low motion activity will be replayed faster than another shot with higher motion activity.

Finally, a video segment $\mathbb{S}_{s(i)}$ of duration $\left(\alpha_{s(i)} \cdot \delta_i\right)$ is extracted from each shot $S_{s(i)}$. To do so, as described in Figure 4, a 1-frame overlapping window of duration $\left(\alpha_{s(i)} \cdot \delta_i\right)$ slides over the whole shot. The corresponding footprint and coverage are extracted for each position and the selected segment is the one with maximum coverage. Since it will be played back in the summary at speed $\alpha_{s(i)}$, the final duration correctly equals δ_i.

3.3.3 Selection of Representative Keyframes

In order to extract keyframes for use in the the visual storyboard described below, one keyframe is extracted from each shot as the frame with minimum average distance d (as defined in Section 2.3) to all other frames of the shot. This simple approach avoids poor quality frames due to high camera motion for instance.

4. SUMMARY CONSTRUCTION AND PRESENTATION

The feature extraction and content analysis stages provided a set of shots for inclusion within the final summary. Each provided shot was defined to be one of two kinds depending on its importance: *played* (displayed as dynamic video content) or *fixed* (represented as a keyframe). The summaries were constructed using Processing [5], an open source programming language specifically designed for electronic arts and visual design.

4.1 Selected vs. rejected shots

Each *played shot* is included in the end summary. For each played shot its selected segment is played back often at a much accelerated rate, as described in Section 3.3.2. The approximate playback duration of each selected segment was between 2 and 3 seconds per segment. *Fixed shots* were also included within the summaries. These shots were not played back within the end summary but rather visually summarised through the display of a single representative keyframe image.

4.2 Storyboard

As video summaries attempt to *storyboard* the sequence of activities within original footage, we maintained this metaphor in our presentation. At both the start and end of each summary, a tiled multimedia storyboard of the shots (see Figure 5) was presented. This provides an at-a-glance visual overview of the original footage's content and progression. Played shots are highlighted with a blue halo to visually distinguish them from their fixed counterparts, allowing the viewer to anticipate where their attention should lie.

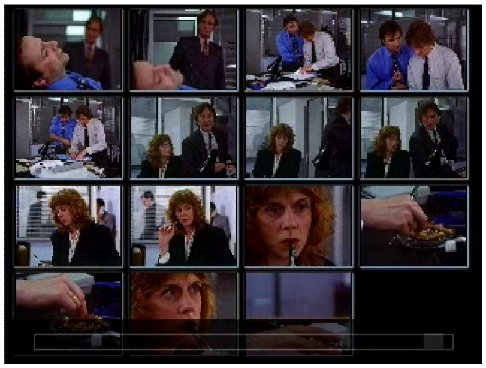

Figure 5: The overview "StoryBoard" presented at the start and end of a summary.

4.3 Audiovisual segment playback

Once the tiled overview has been presented, the storyboard smoothly zooms and moves to focus on the first selected segment (see Figure 6). The segment then occupies 80% of the 320×240 video screen. Portions of neighbouring segments can thus be seen by the viewer. Once the transition to the segment is complete both audio and video begin playing after which the summary transitions to the next playback segment. The summary additionally includes a timeline at the bottom of the screen. The timeline is highly transparent to prevent occlusion but is sufficiently visible to provide a useful cue to the location (position of the marker in the timeline) and duration (width of the marker in the timeline) of the shot being played within the original footage.

Audio was also included and aligned with the playback for the segment. An un-accelerated audio clip was favoured as the higher bound acceleration would noticeably distort the audio making it difficult to interpret. A snippet of audio from the middle of the segment was extracted for inclusion with each played segment.

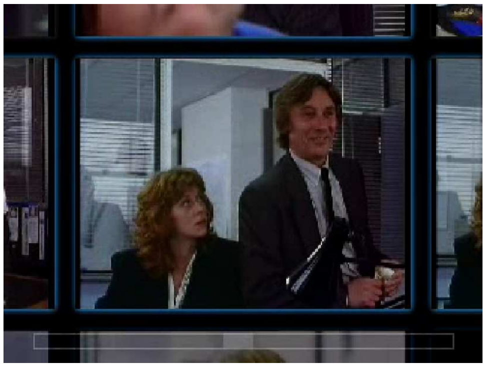

Figure 6: A segment zoomed for playback.

Approach	LDA	PCA
Fraction of inclusions found in the summary	**0.45**	**0.5**
Summary contains lots of duplicate video (1 = worst, 5 = best)	3.33	3.33
Summary contains lots of junk (1 = worst, 5 = best)	3	3
Duration of the summary (sec.)	33.1	33.3
Difference between target and actual summary size (target-actual) (sec.)	1.3	1.43
Total time spent judging the inclusions (sec.)	45	46.33
Total video play time (versus pause) judging the inclusions (sec.)	35.67	34.33
Summary had a pleasant tempo/rhythm (1 = worst, 5 = best)	2.67	2.67

Table 1: Scores for both summarisation approaches (median value on test videos).

5. RESULTS

The results from both runs of the DCU submissions are outlined in Table 1. The submission based on PCA (DCU2) performs better than the one based on LDA (DCU1). The performance for all measures is very similar with the notable exception of the measure of inclusion, for which DCU2 performs significantly better than DCU1. A full overview of the results for all participants is available in [3].

5.1 Dr Inclusion and Mr Redundancy

Considering the 2008 results, it is clear that there is a trade-off between the level of inclusions (and therefore the selection of *good* representative segments from the original) and the removal of redundant ones. The CMU baseline illustrates this point excellently. Since the baseline contains all of the original content (albeit highly accelerated), it is to be expected that it would cover the original content extremely well and have a high measure of inclusion, but perform poorly for redundancy. In the results, the baseline performs best and worst out of all 43 submissions for inclusions and redundancy respectively. This relationship between inclusion and redundancy is further highlighted in Figure 7.

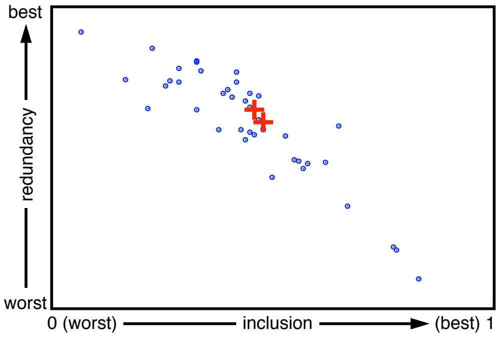

Figure 7: Comparison of Redundancy vs. Inclusions for all submitted runs. DCU Submissions marked with '+'

We thus propose to jointly consider inclusion and redundancy in evaluating performance. The best of our two submissions (DCU2) ranked $\mathrm{rank}_{\mathrm{IN}} = 12^{\mathrm{th}}$ for inclusion and $\mathrm{rank}_{\mathrm{RE}} = 19^{\mathrm{th}}$ for redundancy (both out of 43 submissions). When ranking systems using $\mathrm{rank}_{\mathrm{IN}} + \mathrm{rank}_{\mathrm{RE}}$, the CMU baseline submission ends up 21st and our systems are 7th (DCU2) and 17th (DCU1).

6. DISCUSSION

6.1 PCA vs. LDA

The only difference between the two runs we submitted was in the use of LDA (DCU1) or PCA (DCU2); all other parameters were the same. PCA appears to slightly outperform LDA: it has better inclusion ($\mathrm{IN}_{\mathrm{PCA}} = 0.5$ and $\mathrm{IN}_{\mathrm{LDA}} = 0.45$) and the same redundancy ($\mathrm{RE}_{\mathrm{PCA}} = \mathrm{RE}_{\mathrm{LDA}} = 0.33$). Though unexpected, it can be explained by the fact that two takes of the same scene are assigned to two different classes and thus LDA tends to accentuate the small differences between them.

6.2 Junk removal

The main weakness of our systems appears to be related to junk removal. We ranked 36th and 37th (out of 43) for the junk (JU) measure. Moreover, having a closer look at the individual video results, it appears that the junk detection step is also responsible for most of the summaries with low inclusion scores. For instance, our method incorrectly removes dark shots even though they should be considered for the summary.

6.3 User feedback

Evaluators were asked to judge how enjoyable the summary was to watch, rating the summary using a Likert scale to assess if it had a "pleasant tempo/rhythm". On this measure, the DCU runs were ranked 25th (DCU2) and 28th (DCU1). This reasonably poor performance could be attributed to the possible over-acceleration of the segments (4.42 times the normal speed on average). However, on investigation there was no obvious correlation between the level of acceleration, number of segments included (played or fixed) and/or the perceived pleasantness of the summary. While overall this metric was consistent from run 1 to run 2 (median 2.67 for both), there was much variation from summary to summary across the runs. The average variation across runs was 0.4 (median 0.33, maximum 1.33). As the runs varied in the number of included (played and fixed) segments, the content within those segments and the duration of that content, this suggests that the perceived pleasantness may have in some way been contingent on the content rather than the composition of the summary making it difficult to ascertain the impact of acceleration.

7. CONCLUSION

In this paper we described DCU's submissions to the TRECVid 2008 BBC Rushes Summarisation Task. Judging by the overall results, we managed to achieve a good compromise between inclusion of most of the important part of the original video and removal of redundant parts of the rushes.

Future work will focus on fine tuning our system parameters. The number of principal components used was selected heuristically, so we will consider using higher dimensions. Moreover, binarised footprints only contain some information related to the coverage of the shot and utterly ignore how much time is spent in a given bin. Using non-binarised footprints would bring this information back. Similarly, the influence of the number of footprint bins has to be investigated. Even though the end summaries did contain some audio, we did not explicitly analyse audio content and this will also be investigated.

Finally, we would like to outline the fact that our system for rushes video summarisation could easily be extended into an interactive version. It does indeed make sense to think of this task as an interactive one where the movie director would be looking for a particular shot (and its retakes). Our system could use the very same content analysis module and the same visual presentation except that the storyboard would be clickable, rather than driven by the duration constraints.

8. ACKNOWLEDGMENTS

Part of this work was sponsored by *BT Group plc* and the *Irish Research Council for Science, Engineering and Technology* (IRCSET): Funded by the *National Development Plan*.

9. REFERENCES

[1] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[2] Z. Li, A. Katsaggelos, and B. Gandhi. Fast Video Shot Retrieval based on Trace Geometry Matching. *IEE Proceedings on Vision, Image and Signal Processing*, 152(3):367–373, 3 June 2005.

[3] P. Over, A. F. Smeaton, and G. Awad. The TRECVid 2008 BBC Rushes Summarization Evaluation. In *TVS '08: Proceedings of the International Workshop on TRECVID Video Summarization*, pages 1–20, New York, NY, USA, 2008. ACM.

[4] P. Over, A. F. Smeaton, and P. Kelly. The TRECVid 2007 BBC Rushes Summarization Evaluation Pilot. In *TVS '07: Proceedings of the International Workshop on TRECVid Video Summarization*, pages 1–15, New York, NY, USA, 2007. ACM Press.

[5] C. Reas, B. Fry, and J. Maeda. *Processing: A Programming Handbook for Visual Designers and Artists*. The MIT Press, 2007.