A MULTISCALE APPROACH TO SHOT CHANGE DETECTION

Hugh Denman
Electronic Engineering Department,
Trinity College Dublin
email: hdenman@cantab.net

Anil Kokaram
Electronic Engineering Department
Trinity College Dublin
email: anil.kokaram@tcd.ie

12 May 2004

Abstract

We describe a multistage approach to shot cut detection based on image descriptor differencing at a coarse temporal scale, followed by identification of shot cuts and fades at frame-level accuracy based on explicit modelling of image data evolution during fades.
Keywords*: Video parsing, cut detection, media discontinuity*

1 Introduction

Effective shot change detection for video sequences is an essential prerequisite for the automated and computer-assisted manipulation of digital visual media.

We propose here a framework for shot change detection based on analysis of image descriptors, such as luminance histograms. Our implementation focuses on shot change detection as the first stage of media processing for the addition of footage to a large media library, so we are interested in exploiting as far as possible image descriptors that will be of use generally in the media library. These descriptors can then be computed in a separate feature extraction pass and stored alongside the media files. Thus, that portion of the execution time of this algorithm that involves feature extraction should properly be amortized over the lifetime of the media asset in the library, taking into account use of the features in subsequent stages.

In the subsequent sections of this paper, we will first describe a general method for data fusion across multiple media descriptors for discontinuity detection, and an application of this method to shot change detection for video sequences. We then outline a new approach to dissolve modelling and describe its effectiveness.

2 Change detection in frame descriptors

Consider a difference features n between two frames, and let $P(T)$ be the probability that the two frames span a shot change boundary. We have

$$P(T|n) = \frac{P(n|T)P(T)}{P(n)}$$

2.1 Prior probabilities

In order that new frame features can be added to the shot detection framework, and to make the system as generally applicable as possible, we will avoid using specific prior distributions. Thus, we assign

a uniform probability distribution to the likelihood, $P(n|T)$. This is a working assumption which reflects that across shot transitions, a feature change may take on any value - for example, global motion estimation will be degenerate across hugely disimilar frames, and colour histograms may be arbitrarily different or similar. This likelihood could alternatively be computed for a particular feature using an existing corpus for which the ground truth is known, but there is then an attendant risk of specialising to the characteristics of that corpus.

Many researchers have assigned prior distributions to T parameterised on shot length, introducing a bias against very short shots, but shot length distribution is a characteristic of genre, and in music videos, for example, shots may be shorter than one half second. Furthermore, glitches and special effects in video may result in shots of only one frame long, e.g. where significant dropout has occured, or in a faux 'subliminal image' effect common in music videos. We therefore also assign a uniform probability to $P(T)$, to reflect that shots may conceivably be of any positive length. Any more informative prior will necessarily be genre-specific (and could easily be incorporated where consideration is restricted to a specific genre).

We are then left with the problem of computing the prior for the feature in question, $P(n)$. Again, this can be explicitly evaluated by off-line analysis of a corpus, but again we feel that greater generality is achieved by computing this distribution from the data itself. In this implementation, we assume that the distribution will be approximately normal within a shot for any given difference feature, and that the value across a shot transition will be a large outlier of the normal distribution. Then, evaluation of whether a given value is an outlier can proceed based on two windows, one to either side of the point under consideration. As estimation of the parameters for a normal distribution is sensitive to outliers, we cannot include the present point in the window, and would ideally exclude previously identified shot cuts as well.

The principal parameter to be determined is the appropriate window size for estimation of the statistics, and it is here that our prior conception of shot length must be taken into account. At present, we use a 60 frame window for frame-to-frame metrics, and aindow size of 6 samples for metrics spanning 10 frames. In future, some form of adaptive window sizing procedure, alongside a more sophisticated estimation process for the metric statistics inside the window, will be incorporated.

2.2 Video Features

The present implementation makes use of three principal frame-level features. The first is an estimate of the translation global motion, computed using integral projection based on an image model of

$$I_n(\mathbf{x}) = \mathbf{I_{n-1}}(\mathbf{x} + \mathbf{d})$$

where $\mathbf{d}$ is the global motion, i.e. not varying with pixel site. This measure can be used directly as a frame-to-frame difference, as generally large estimated displacements correspond to large frame differences. More sophisticated global estimation techniques can be used, as, for example, in the paper by Kokaram [4].

Histograms have also been recognised as suitable for shot change detection, for example as described by Han *et al* [3]. We employ here a the bin-to-bin histogram difference. Each frame of the sequence is converted into the L, U, V colourspace, and a 101-bin histogram is computed of the L plane:

$$h(i) = \sum_{L(\mathbf{x})=\mathbf{i}} 1$$

This full histogram h is then downsampled to a ten bin histogram H, and our difference measure between frames is the sum of the absolut bin-to-bin differences of their downsampled histograms:

$$D_{n_1,n_2} = \sum_{i=1:10} |H(i)_{n_1} - H(i)_{n_2}|$$

The final feature used is the frame-to-frame edge moment differential. An edge map of each of the two frames to be compared is found, using the Canny edge detector. These edge maps are then dilated using a five-by-five disk-shaped structuring element, to improve robustness under motion. We denote this dilated edge map E, taking on values $E(\mathbf{x}) = \mathbf{1}$ if there is an edge at site $\mathbf{x}$ and zero otherwise. The second order moment M of the dilated edge map is then found, where

$$M = \sum_{\mathbf{x}} |\mathbf{x}|\mathbf{E}(\mathbf{x})$$

This second order moment is strongly correlated with the distribution of edges in the image. Thus, the frame-to-frame difference of this moment is an indicator of the difference between frames. A variety of other edge-related features can be used for shot detection and characterisation, for example the Hough transform [2] and a disappearing edge count [1].

The global motion and histogram differences are computed off line, and used for first-pass cut detection: if any difference value exceeds 50 standard deviations, the corresponding frame is immediately assumed to mark the start of a new shot. The standard deviation value used is the lesser of two calculated from windows to either side of the frame under consideration; this results in more stable sequence statistics being automatically selected. The use of such a locally estimated measure is greatly preferable to a prior fixed threshold value. For example, a shot of a single frame in length can be detected, and a shot transition between an ordinary shot and a shot consisting of a succession of unrelated frames can be detected, but a shot consisting of a succession of unrelated frames is not artificially partitioned.

Using the assumption that each difference feature is independent of the others, we can also combine local deviations to find more subtle shot cuts. Adding local deviations is conceptually equivalent to multiplying and scaling the associated probabilities. Where a combined local deviation exceeds 50, we flag a shot transition.

Local deviations between 10 and 50 in any single difference feature we consider to be possible shot cuts, for further examination. At present the only subsequent feature in use is the frame-to-frame difference of the second order moment of the dilated edge distribution. We evaluate this difference vector around the possible shot cut and compute local deviations as before. These local deviations are added to those previously computed using the other features, and if the sum local deviation exceeds 50, we assume that a shot cut has been detected.

This process can be augmented naturally to add in more sophisticated frame difference techniques until the confidence (local deviation) at each frame has move outside the thresholds of uncertainty.

The framework as developed here uses differencing between adjacent frames. We expect that gradual shot transitions can be more easily detected at a coarse temporal scale. In the following section, we outline how a possible shot transition region is examined to see whether it is likely to be a fade.

3 Fades

Fades, also known as dissolves, are a common transition in many video genres, including motion pictures, sports footage, and music videos. While analysis of frame features at a coarse temporal scale is generally sufficient to localise fades and other gradual shot transitions, this method by itself will result in very low precision, as video regions with high motion content will also be found. Some researchers have used edge information to analyse possible fade regions, but this is a computationally expensive approach, especially as dilation of edge maps is crucial for robsutness to motion. We introduce here an efficient scheme for modelling fades in which possible dissolve regions are characterised by an *alpha curve*, where alpha is a parameter varying from 1 to 0 as the fade progresses. Examination of this curve then informs classification of the video region as being a fade, or otherwise.

3.1 Fade model

Our model assumes that a frame occuring during a fade is made up of a linear combination of two *template frames*, designated I_{T_0} and I_{T_1}, at positions preceding and succeeding the fade region. The image predicted by this model, for a given crossfade strength α, is designated $I_{M(\alpha)}$, and calulated by:

$$I_{M(\alpha)} = \alpha I_{T_0} + (1-\alpha) I_{T_1}$$

The likelihood of a given value of alpha is proportional to the agreement between the image predicted by the model and the observed data, which is the image at time t, designated I_t. Specifically,

$$p(\alpha|I_t) \propto \exp(-\sum_{\mathbf{x}}[(I_t(\mathbf{x}) - \mathbf{I_{M(\alpha)}(x)})^{\mathbf{2}}])$$

For a given image, we can estimate the MAP value of alpha by differentiation with respect to alpha. It transpires that the optimal value is given by

$$\alpha_{opt} = \frac{\sum I_t \nabla_{T_0,T_1} - \sum I_{T_1} \nabla_{T_0,T_1}}{\sum \nabla^2_{T_0,T_1}}$$

where ∇_{T0,T_1} is simply the difference image $I_{T_0} - I_{T_1}$.

3.2 Global motion

The model as presented makes no account of the motion content of the image sequence, which will result in probable failure to accurately estimate alpha in dissolves that occur between sequences with significant motion. As a first step to improving robustness in this instance, we introduce global motion compensation. When computing α for frame I_t, we first apply cumulative global motion parameters to frame I_{T_0} and inverse parameters to frame I_{T_1}, to compensate each template to time t. After this compensation, only a partial region of each template frame will contain valid data, and estimation of α is performed on the overlapping area of the valid regions. Naturally, this process introduces a dependency on the accuracy of the global motion estimator; the results described herein were based on an implentation using a fast, projection-based estimator, estimating translation motion only, and difficulties can be expected in sequences featuring fast zooms. Compounding this disadvantage is that we require global motion estimates in precisely the region where they are most difficult to compute, viz. within the dissolve itself; we intend to investigate a more sophisticated implementation which will extrapolate global motion parameters computed prior to the suspected dissolve region where possible. A further difficulty is that analysis of regions undergoing rapid global motion may be impossible, where the intersection of the regions valid after global motion compensation is the null set.

3.3 Local motion

Local motion in the dissolve region will introduce a large, localised discrepancy between the template image and the current frame. These discrepancies will then confound the alpha estimation process. To account for this, we employ an iterative reweighting scheme based on a Cauchy weighting function. In our first estimate, the weight at every site is 1. We then examine the residual image, $I_e = I_{M(\alpha)} - I_t$, and update the weight at each site according to:

$$w(\mathbf{x}) = \frac{\mathbf{1}}{(\mathbf{1} + \mathbf{r^2})}, \quad \mathbf{r} = \frac{\mathbf{I_e(x)}}{(\mathbf{2.385})(\mathbf{s})\sqrt{(\mathbf{1}-\mathbf{h})}}$$

In the above formula, the residuals $I_e(\mathbf{x})$ are being scaled to take into account the leverage of the point h (distance from data centroid), and s is related to the median absolute distance of the residuals from their median (a measure of the overall spread of the data).

This process is repeated until the number of residuals exceeding a certain threshold is zero, or the sum of the residuals begins to increase, or the number of iterations exceeds a certain limit. None of these halting conditions is entirely satisfactory, as correct estimation of alpha may indeed involve increasing the number of pixels assigned to local motion after some iterations, and choice of the appropriate thresholds is difficult (currently alpha estimation is discontinued if less than 2% of the image has an error of more than 20 graylevels). While the present, somewhat ad-hoc approach does produce satisfactory results, it is frequently apparent that the algorithm is performing more iterations than necessary.

3.4 Fade curve analysis

Having calculated the alpha values for each frame in the region of interest, we then examine the resulting curve to see whether it has the characteristics of a fade. We have adopted a simple approach in which the alpha curve is partitioned into three sets of adjacent values, and fit a line to each partition. We iterate over every possible choice of two changepoints in the alpha curve, and lines are fitted to each of the three resulting segments. A confidence measure is associated with each line, based on the mean squared distance from each point in the segement to the line. The partition that gives the highest average confidence over the three fitted lines is then selected. As the number of points in an alpha curve is only of the order of forty, this exhaustive search strategy is by no means computationally prohibitive.

Having found the lines of best fit, the slope of each of the three lines can then be examined to determine if a fade has occured: we expect a flat first line, followed by a line with a negative slope of moderate magnitude (corresponding to the transition region), followed by a final flat region. This method generally determines the start and endpoints of the fade to an accuracy of within +- one frame, depending on the motion characteristics of the shots involved.

We also examine the alpha curve for sudden discontinuities; if successive values differ by more than 0.7, we assume that a shot cut has occured.

3.5 Examples

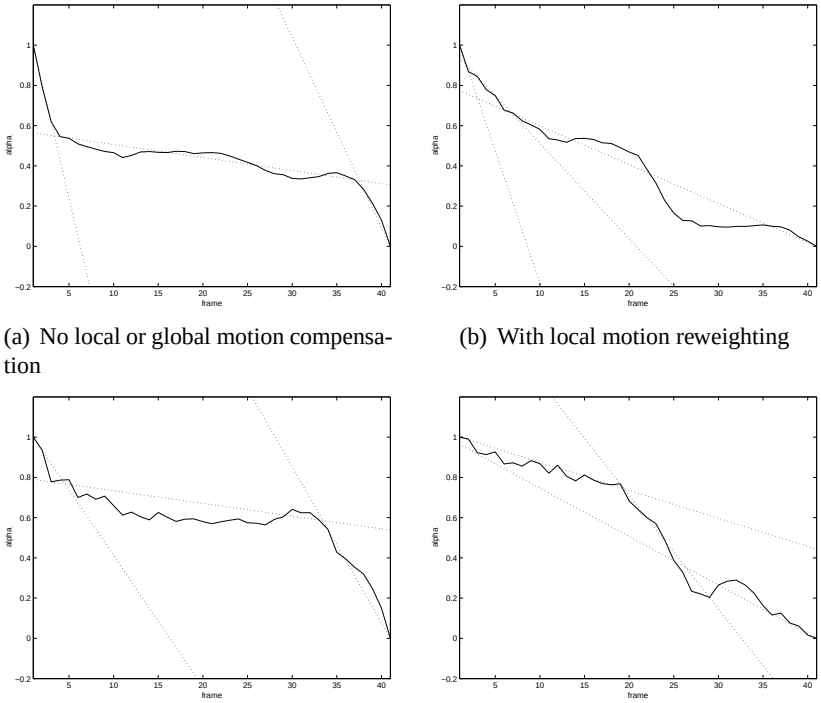
Figure 1 shows a fast dissolve in a cricket sequence, and figure 2 shows the extracted alpha curves. It can be seen that without either global motion compensation or reweighting local motion regions, fade detection has failed. The fade is detected, however, when both compensation strategies are employed.

We encounter again the issue of selection of an appropriate window size; where the window is too small, the flat areas corresponding to the unmixed shots will not be readily apparent, whereas with an overlarge window, cumulative motion effects can be expected to degrade the quality of the curve greatly. Furthermore, the slower the dissolve, the larger the window will be necessary. At present, a fixed window size of forty frames is employed.

4 Results

We have applied the shot transition detection framework described in section 2 to a variety of test sequences, though at the time of description the algorithm is under continual refinement and elaboration. The first is a simple 'proof of concept' sequence, referred to here as *News*, with 645 frames, 5 cuts, and 4 frames. For this sequence, we attempt to detect frame fades across regions even if the regions are already known to contain a cut, and discard cuts that are subsequently found to be within fade regions. Here we achieve 100% recall for both cuts and fades, and 100% precision for fades, with all fade start and end points detected to within one frame of the observed values. However, two spurious shot cuts are detected, bringing the cut detection precision down to 83.3%. These spurious shots correspond to sudden small differences from image to image in regions that are otherwise perfectly stable, so in a sense this kind of failure is intrinsic to the algorithm as presented. However, these false alarms could easily and cheaply

Figure 1: A cricket sequence containing fast local and global motion on both sides of a fast dissolve. The frames shown correspond with offsets 5, 10, 15, 20, 25, and 30 in figure 2.



(a) No local or global motion compensation

(b) With local motion reweighting

(c) With global motion compensation

(d) With global motion compensation and residual reweighting (fade region identified, between frames 19 and 30)

Figure 2: Alpha estimation across a fast dissolve with significant global and local motion. The dissolve starts at frame 19 and ends at frame 27. The dotted lines show the region partitioning.

be suppressed by imposing a minimum on the norm of the frame difference image, to insure that cuts are only detected when changes are over a significant region of the image.

We also analysed a 14,000 frame video of cricket play. This sequence contains 62 cuts and 20 fades. It is characterised by much fast global motion, including fast zooms, and quick crossfades, typically over 6 to 10 frames. Here we achieve 92% recall and 86% precision in cut detection using the media discontinuity scheme of section 2. Fades are identified with 70% recall and 80% precision. When we take into account the detection of cuts via discontinuities in the alpha curve, we score 94% recall and 86$ precision.

The accuracy of the results in analysing the cricket sequence suffer due to the motion characteristics of the sequence. We expect that these results can be improved upon through improved global motion estimation.

References

[1] Paul Browne, Alan F Smeaton, Noel Murphy, Noel O'Connor, Sean Marlow, and Catherine Berrut. Evaluating and combining digital video shot boundary detection algorithms. In *Irish Machine Vision and Image Processing Conference*, 2000.

[2] Hugh Denman, Niall Rea, and Anil Kokaram. Content-based analysis for video from snooker broadcasts. *Journal of Computer Vision and Image Understanding*, 92:141–306, 2003.

[3] Seung-Hoon Han, Kuk-Jin Yoon, and In-So Kweon. A new technique for shot detection and key frames selection in histogram space. *Image Processing and Image Understanding*, 2000.

[4] Anil Kokaram and Perrine Delacourt. A new global estimation algorithm and its application to retrieval in sport events. In *IEEE International Workshop on Multimedia Signal Processing*, Cannes, France, October 2001.