Cut Detection in Video Sequences Using Phase Correlation

Theodore Vlachos

***Abstract*—A novel algorithm for the detection of cuts in video sequences is proposed. The algorithm uses phase correlation to obtain a measure of content similarity for temporally adjacent frames and responds very well to scene cuts. The algorithm is insensitive to the presence of global illumination changes and noise and outperforms established methods for cut detection. As the proposed scheme is implemented in the frequency domain, the availability of fast hardware makes the scheme attractive for interactive and on-line applications.**

I. Introduction

THE identification of shot changes in video sequences is an important task toward automated content-based analysis and semantic description of video. It is usually regarded as a first step in the hierarchical partitioning of a scene into its constituent components and has important applications in video indexing, annotation, content-based retrieval, and compression [1].

A video sequence is a collection of camera shots concatenated using a variety of postproduction techniques. Each shot is an uninterrupted sequence of frames captured by the same camera at contiguous time instants and focusing on one or more objects of interest. Transitions from one shot to the next are achieved in one of many different ways. The most simple transition effect is a cut occurring within a single frame period.

Shot change detection is an emerging area of research, and several algorithms have already been proposed in the literature [2], [3]. Frequently quoted algorithms are based on pixel and statistical differencing, histogramming, and edge detection. Due to the proliferation of video compression in the digital domain, schemes based on the DCT and motion estimation have also been proposed. These schemes make extensive use of the readily available elements of a compressed video stream such as transform coefficients and motion vectors.

In this letter, a novel cut detection algorithm based on the phase correlation technique is proposed. The analytic framework is briefly discussed in Section II, and the proposed scheme is described in Section III. Numerical results obtained by simulating the proposed algorithm, comparisons with an established scheme are presented in Section IV, and conclusions are drawn in Section V.

Manuscript received February 2, 2000. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. G. Ramponi.

The author is with the Centre for Vision, Speech, and Signal Processing, University of Surrey, Guildford, U. K. (e-mail: t.vlachos@eim.surrey.ac.uk).

Publisher Item Identifier S 1070-9908(00)05866-1.

II. Phase Correlation

Phase correlation is a signal correlation technique that exploits fundamental properties of the discrete Fourier transform to provide a measure of similarity between two discrete signals [4]. It has been used in the past in image registration problems [5] and has recently become an important tool in the broadcasting domain for a wide range of applications including motion measurement for standards conversion, noise reduction, archive restoration, and video compression [6]–[8]. It has a number of well-documented advantages such as insensitivity to global illumination changes and noise. Of additional interest to the application under consideration, phase correlation is quite robust in the presence of camera and object motion in the sense that the magnitude of its response, which is central to the proposed detection scheme, is relatively insensitive to them [6].

For our application, phase correlation is the normalized circular cross-correlation of two regions of image data usually in the form of co-sited rectangular blocks in the current and the next frame of a video sequence. For two such blocks $\boldsymbol{b}_t(k, l)$ and $\boldsymbol{b}_{t+1}(k, l)$, related to frames acquired at times t and $t+1$, respectively, a correlation surface $\boldsymbol{S}_{t, t+1}$ is defined as follows

$$\boldsymbol{S}_{t, t+1}(k, l) = F^{-1}\left[\frac{F^*(\boldsymbol{b}_t)\cdot F(\boldsymbol{b}_{t+1})}{|F^*(\boldsymbol{b}_t)\cdot F(\boldsymbol{b}_{t+1})|}\right]^{\wedge} \tag{1}$$

where

F and F^{-1}	forward and inverse two-dimensional (2-D) discrete Fourier transforms, respectively;
(k, l)	horizontal and vertical block pixel coordinates on the image grid, respectively;
$*$	complex conjugate.

For identical blocks, the surface has a unique peak which is the two-dimensional (2-D) Dirac delta function. For nonidentical blocks (i.e., taken from consecutive fields or frames of moving sequences), several peaks can be simultaneously present. In this case, the locations of the largest peaks correspond to the dominant motion components, and the height of any such peak is an indication of confidence in the corresponding motion component.

It is worth noting that the denominator in (1) is a normalization or "whitening" operation on the power cross-spectrum of the two data blocks, which can be viewed as equivalent to the extraction of the relative phases of the corresponding discrete frequency components. Then the application of the inverse Fourier transform in (1) can be thought of as obtaining the solution to a set of simultaneous equations in order to determine the relative block displacement that would cause such phase shifts. Spectral whitening is essential in providing the desired immunity against global illumination changes, as discussed later on.

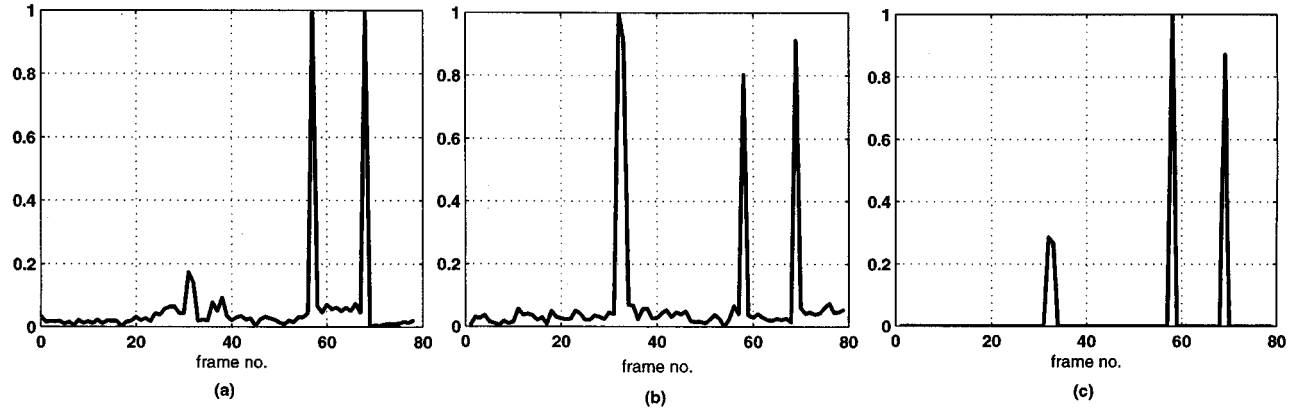

Fig. 1. Detector response comparison for the first 80 frames of test sequence "news" using (a) phase correlation, (b) histogram comparison, and (c) likelihood ratio.

III. Detection of Cuts

A. Block Partitioning and Windowing

Frames to be phase-correlated are partitioned to rectangular blocks. This renders the scheme more adaptive to local changes than would have been the case for a single block corresponding to the entire frame. Moreover, blocks are overlapping so that the windowing operation described below does not diminish the influence of picture material that happens to be situated close to block boundaries.

Windowing forces block elements to fade to a constant value (typically the midpoint of the available greyscale) around block edges, while at the same time preserving the integrity of the more centrally located elements. This is necessary due to the periodic nature of the Fourier transform, which introduces sharp transitions at the edges of the block, as the left and right (and the top and bottom) edges effectively join each other. Without windowing, such transitions would cause a spurious peak at zero displacement of the correlation surface. Another benefit of windowing is that new material that appears at the edges contributes less to the correlation process.

Windowing is typically performed by multiplication of a block $\boldsymbol{b}_t$ with a suitable windowing function w to yield a windowed block $\boldsymbol{b}_t'(k, l) = \boldsymbol{b}_t(k, l) \cdot \boldsymbol{w}(k, l)$. Pairs of windowed blocks are used instead of the original blocks in (1) for the reasons outlined above. 2-D widowing is typically implemented in variable-separable fashion as a cascade of two one-dimensional (1-D) operations $w(k)$ and $w(l)$ in the horizontal and vertical dimensions.

In practical applications, particularly useful choices of windowing functions can be obtained from the discrete approximation to the *raised cosine* function $w(k) = a + (1-a)\cos(k\pi/m)$ for $-m/2 \leq k \leq m/2$. Parameter a can be used to control the shape of the window. For example, $a = 1$ gives a rectangular window (no windowing), and $a = 0$ gives a pure half-period cosine, while $a = 0.54$ would give a Hamming type of window, which is widely used in practice.

B. Cut Detection

Cut detection is based on the response of the algorithm to content similarities between two frames, as measured by the height of the dominant peaks of the correlation surface $\boldsymbol{S}$. For simplicity, we shall only consider the highest such peak denoted by

$$p_{t,\,t+1} = \max_{k,\,l}\{\boldsymbol{S}_{t,\,t+1}(k,\,l)\} \tag{2}$$

The detection of cuts relies on successive phase correlation operations applied to pairs of consecutive block-partitioned frames of a video sequence. The heights of the dominant peaks are monitored, and when a sudden magnitude change is detected, then this is interpreted as a cut.

For a frame-pair, the total number of phase correlated blocks, and therefore peaks, is a function of the frame size, the block size, and the block overlapping ratio used. For an n-peak scheme, the detector response $R_{t,\,t-1}$ for consecutive frames t and $t+1$ is given by

$$R_{t,\,t+1} = -\sum_{i=1}^{n} \log p_{t,\,t+1}^{(i)} \tag{3}$$

where $p_{t,\,t+1}^{(i)}$ is the highest peak for the ith co-sited block pair given by (2). A scene cut produces a high value of R at the transition boundary and vice versa.

IV. Results

The proposed algorithm was tested on several arbitrary-content sequences obtained by recording satellite broadcasts of news and sports programs. The test sequences were first converted to CIF format by discarding even-parity fields and by spatial subsampling. Only the luminance components were used, and frames were partitioned to a total of 12 overlapping square blocks of 256 $\times$ 256 pixels each. This was achieved by positioning the first block at the top-left corner of the frame. Subsequent blocks were obtained by alternate horizontal and vertical shifts of the block coordinates by 16 pixels until the entire frame area was covered. The detection scheme of (3) was subsequently applied with Hamming windowing and a total of $n = 12$ peaks per frame. In Fig. 1(a), the detector response is plotted for the first 80 frames of test sequence news. This portion of the sequence contains two cuts at frames 57 and 68 and a sudden flash of light at frame 31. From the

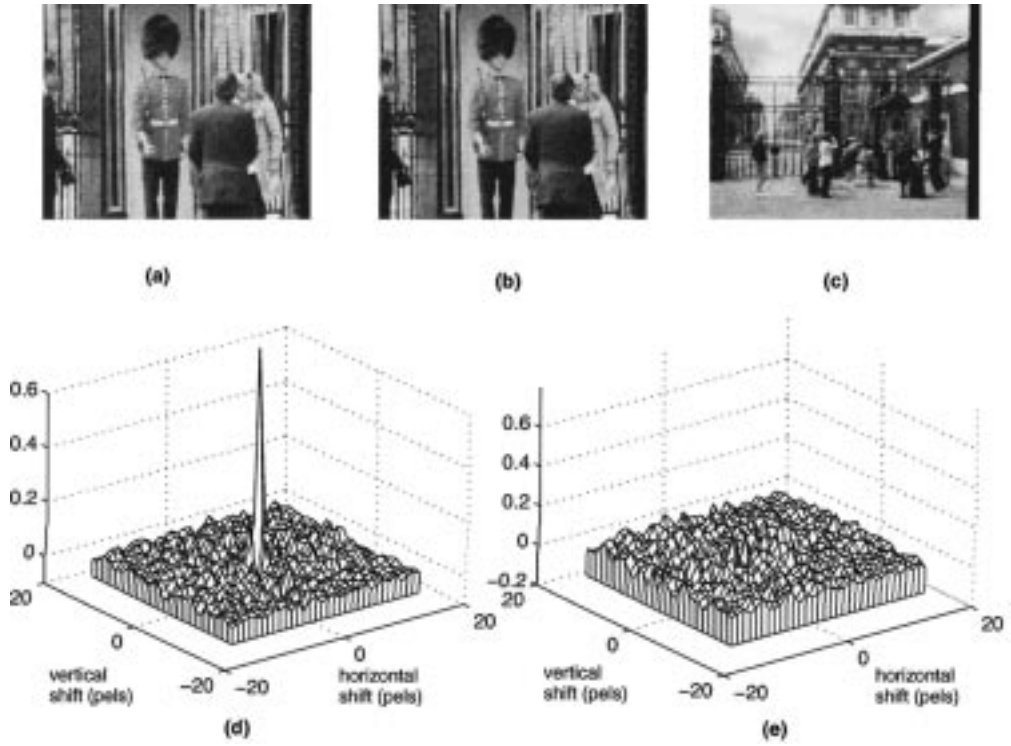

Fig. 2. Successive frames (a), (b), and (c) of test sequence news and associated phase correlation surfaces (d) and (e).

figure, it can be seen that the location of the cuts is correctly identified. It can also be seen that the algorithm does not confuse the flash for a scene cut. For comparison purposes, the response of an established algorithm based on global histogram comparison [1] is plotted in Fig. 1(b). Comparison with a global technique is arguably more appropriate given the block size for the proposed method. Nevertheless, the response of a purely local method based on likelihood ratio [1] computed for 16×16 blocks is also shown in Fig. 1(c). All three responses have been normalized in the range 0 to 1. Our results clearly demonstrate the sensitivity of the histogram scheme to global illumination changes such as flashes. In fact, it can be seen that this scheme responds stronger to the flash than the actual cuts. The likelihood ratio scheme performs better in this respect but has a weaker response to the second cut. Finally, Fig. 2 shows sample phase correlation surfaces in the vicinity of the cut between frames 68 and 69. Frames 67, 68, and 69 are depicted in Fig. 2(a)–(c), respectively. The surface resulting from correlating (a) with (b) is shown in Fig. 2(d), and the surface resulting from correlating (b) with (c) is shown in Fig. 2(e). As should be evident from (1), the surfaces shown are dimensionless and have an upper bound of unity. Overall, the results obtained by testing several other sequences were very promising.

V. Conclusions

A novel algorithm for cut detection in video sequences has been presented. The algorithm uses phase correlation to obtain a measure of content similarity for temporally adjacent frames and responds very well to simple cuts. Sudden illumination changes are not confused with true scene cuts, as would be the case with established cut detection algorithms such as those based on histogram comparison. The availability of fast, dedicated hardware [6] operating at broadcast video field rates (i.e., sub-20-ms processing time) makes the scheme attractive for interactive and on-line applications.

Acknowledgment

The author would like to thank Y. Yussof of the Centre for Vision, Speech, and Signal Processing (CVSSP), University of Surrey, Guildford, U.K., for helpful discussions.

References

[1] J. Ahanger and T. D. C. Little, "A survey of technologies for parsing and indexing digital video," *J. Vis. Commun. Image Rep.*, vol. 7, pp. 28–43, Mar. 1996.

[2] J. S. Boreczsky and L. A. Rowe, "Comparison of video shot boundary detection techniques," *Proc. SPIE*, vol. 2664, pp. 170–179, Jan. 1996.

[3] R. Lienhart, "Comparison of automated shot boundary detection algorithms," *Proc. SPIE*, vol. 3656, pp. 290–300, Jan. 1999.

[4] J. J. Pearson, D. C. Hines, S. Goldsman, and C. D. Kuglin, "Video rate image correlation processor," *Proc. SPIE*, vol. 119, 1977.

[5] S. Alliney and C. Morandi, "Digital image registration using projections," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 222–233, Mar. 1986.

[6] G. A. Thomas, "Television motion measurement for DATV and other applications," Res. Dept. Rep. 1987/11, British Broadcasting Corp., London, U.K., 1987.

[7] T. Vlachos and G. A. Thomas, "Motion estimation for the correction of twin-lens telecine flicker," *Proc. IEEE Int. Conf. Image Processing*, vol. 1, pp. 109–112, 1996.

[8] J.-H. Chenot, J. O. Drewery, and D. Lyon, "Restoration of archived television programs for digital broadcasting," in *Proc. Int. Broadcasting Convention '98*, Sep. 1998, pp. 26–31.