TU DELFT at TRECVID 2005: Shot Boundary Detection

Umut Naci and Alan Hanjalic

Delft University of Technology
Faculty of EEMCS, Department of Mediamatics
Delft, The Netherlands
U.Naci@ewi.tudelft.nl

ABSTRACT

In TRECVID 2005 we tested a simple but effective method for simultaneously detecting video shot transitions of various types by means of an analysis of spatiotemporal video data blocks. This method differs from the existing approaches in the way that it takes volumetric data cubes in the video as the fundamental processing unit (sysID: TUDelft1 and TUDelft2). TUDelft1 also involves an illumination and flash light normalization component, which resulted in considerable amount of increase in overall performance figures. Due to its simplicity, the proposed method is highly computationally efficient. We observed that the most important problem of the system is its fragility against motion in gradual transition detection. This stems from the fact that the analysis currently takes the gradient information only in time direction. We plan to correct for this problem in 2006 when we updated the system also for camera motion detection.

1. INTRODUCTION

Video is a three-dimensional signal and its properties can best be revealed by simultaneously exploiting all three axes of its information flow, two of them revealing the visual content flow in horizontal and vertical frame directions, and the third one revealing the variations in this flow over time. While this approach is likely to improve the performance of video content analysis at all (semantic) levels, in this paper we show its applicability at the lowest analysis level, that is, the level of shot transition detection. In particular, we address the problem of detecting *gradual transitions* (e.g. dissolves, fades and various graphical effects). This problem – compared to a simpler problem of detecting abrupt transitions, or *cuts* - has not successfully been solved yet by the shot transition detection approaches developed so far. Namely, although a vast diversity of methods for detecting various types of shot transitions exist, as can be seen from surveys and representative methods [1-12], and one could tend to assume that there is a sufficient potential for definitely solving this problem, there are critical deficiencies that prevent the effective usage of the existing methods in the practice of video content analysis. The first major deficiency is that the existing systems are insufficiently capable of coping with a great diversity of the measurable signal behavior around and within gradual shot transitions. The origin of this diversity is threefold:

- *practically unlimited number of transitions types* (due to a vast variety of possible editing effects),
- *varying video-directing styles* (this is particularly related to the length/speed of a transition),
- *multiple superimposed effects* (e.g. the case of a gradual transition accompanied by an object or camera motion).

Another important deficiency of existing methods is that the desired high efficiency of shot transition detection can hardly be matched with a high reliability of detection performance. This results either in fast but unreliable methods or in the methods where severe concessions are made with respect to the efficiency in order to improve the reliability.

In this paper we propose a method that is likely to contribute to neutralizing the two deficiencies mentioned above, and so to lead to an improvement of the overall transition detection performance, both in terms of efficiency and reliability. Our proposed method is based on the extraction of the relevant features from spatiotemporal image blocks and modeling those features to detect and identify a vast range of transition types including cuts, dissolves, fades, and an abundance of graphical effects. The extracted features are mainly related to the behavior of luminance values of pixels in the blocks and form the basis of the unified framework for detecting various transition types. The detection performance is independent of the variations in the form and length (speed) of a transition. Further, as the features used and the processing steps performed are rather simple, our proposed method is computationally inexpensive. Finally,

Figure 2: Examples of graphical effects (wipes)

we are able to detect the beginning and ending time stamps of the transitions.

The scheme of our proposed method is shown in Figure 1. The features extracted from spatiotemporal video data blocks serve to provide elementary evidence on the presence of a shot transition in the observed time interval. We search for this evidence by investigating local properties of the visual content flow that can help differentiate between the shot transitions and other phenomena in this flow, like those caused by camera and object motion or lighting changes.

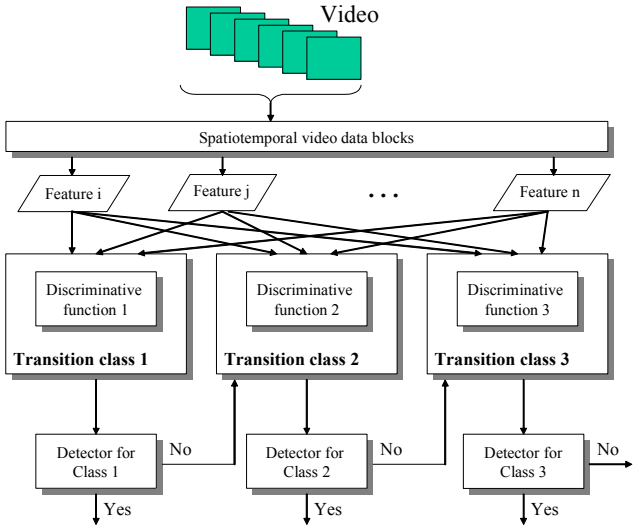

Figure 1: The scheme of the proposed method

The feature values collected from a number of neighboring blocks are used to compute the values of *discriminative functions* [1] for three major classes in which we group all transition types. The discriminative function value serves as an indication for the occurrence of a shot transition from the corresponding class within the observed time interval. The transition class 1 contains cuts. Dissolves and fades are, due to the similar underlying principle, grouped into the same transition class 3, while graphical effects covering most of the remaining transition types belong to the transition class 2. As opposed to a dissolve or fade, which are characterized by a gradual content change in every pixel of a video frame, a graphical effect introduces local abrupt changes in the frame content distributed over time. The effect starts by replacing the old content by the new one in some frame regions and continues until the entire frame contains the material of the new shot. We will further refer to these effects as *wipes*, some examples of which are shown in Figure 2.

Based on the values of the discriminative functions we compute the probability values for finding a shot transition from a particular class in the observed time interval. The probability values serve as input into the cascade of detectors using which shot transitions are detected at all places where their probabilities are sufficiently high.

We start the technical part of the paper with the detailed explanation of the feature extraction process in Section 2. The actual detection of different transition types based on the computation of discriminative function values and their mapping onto probabilities is explained in Section 3. In Section 4, we elaborate on the performance of our method based on TRECVID 2005 experiments. We complete the paper by Section 5 where the concluding remarks on the proposed method and some ideas for its further improvement and evaluation can be found.

2. FEATURE EXTRACTION

Let video data be defined as a three dimensional discrete function of luminance (intensity) values $I(x,y,t)$ where $0 \le x < X$, $0 \le y < Y$ and $0 \le t < T$. Here, X, Y and T represent the

horizontal and vertical frame dimensions and the length of the video, respectively. To perform a 3D analysis on the data, we define overlapping spatiotemporal data blocks of dimensions C_x, C_y and C_t and the temporal overlap factor α. An illustration of these blocks is given in Figure 3. We represent each block by the set of luminance values $I_{i,j,k}(m,n,f)$ of its pixels, that is

$$I_{i,j,k}(m,n,f) = I(m+i\cdot C_x,\, n+j\cdot C_y,\, f+k\cdot\alpha\cdot C_t) \qquad (1)$$

Here, $0\le m<C_x$, $0\le n<C_y$, $0\le f<C_t$, and $0<\alpha\le 1$, while the triplet (i,j,k) serves to index a block in the totality of video data.

We observed that within a single data block it is sufficient to analyze changes in luminance along the time dimension to be able to detect various shot transitions types. In case of a cut, in a block comprising the data from two consecutive shots the majority of pixel luminance tracks will show a large discontinuity at the time stamp of a cut. As partly visible from the examples in Figure 2, a wipe is characterized by a series of local abrupt content changes in different frame regions and at different discrete time stamps because of a limited temporal resolution of video. Therefore, the same types of discontinuities in the pixel luminance tracks can be expected in individual data blocks as in the case of a cut. The major difference between a cut and a wipe is that in the case of a wipe the discontinuities are spread over a time interval (wipe duration), as opposed to cuts, where the pixel luminance discontinuities are aligned in time, that is, they share the same time index t. Compared to cuts and wipes, dissolves and fades are characterized by monotonously changing luminance values in spatiotemporal data blocks over a period of time (dissolve/fade length).

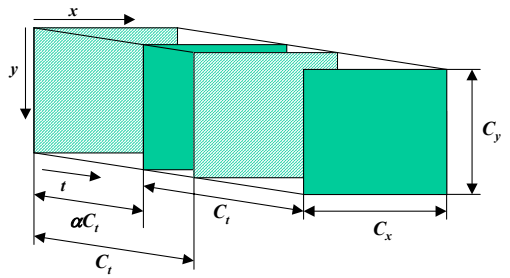

Figure 3: Illustration of two overlapping spatiotemporal blocks of video data

We now translate the above observations related to the behavior of luminance within a block (i,j,k) into a quantitative evidence of shot transition occurrence per block by defining the following feature set:

- $F_1(i,j,k)$, which evaluates the monotonousness of the luminance flow in the block (i,j,k) along the time dimension,
- $F_2(i,j,k)$, which is the measure of abruptness (gradualness) of a change in the luminance flow in the block (i,j,k) along the time dimension,
- $\boldsymbol{F}_3(i,j,k)$, which evaluates how simultaneous the changes in the luminance flow in the block (i,j,k) are at different video frames f of the block. This feature is obtained as a vector

$$\boldsymbol{F}_3(i,j,k) = \{F_3^f(i,j,k) \mid 0 \le f < C_t\}$$

Features $F_1(i,j,k)$ and $F_2(i,j,k)$ will be used for detecting dissolves and fades while $\boldsymbol{F}_3(i,j,k)$ will serve for detection of all other transition types (class 1 and 2).

To compute the above features, we first search for the derivative values of the function $I_{i,j,k}(m,n,f)$ along the time dimension. This derivative is defined as

$$\nabla_{\mathbf{k}} I_{i,j,k}(m,n,f) = I_{i,j,k}(m,n,f+1) - I_{i,j,k}(m,n,f) \qquad (2)$$

where $\mathbf{k}$ is the unit vector in time direction. Then, we calculate two different measures from this derivative information per block, namely *the absolute cumulative luminance change:*

$$\nabla^a_{\mathbf{k}} I_{i,j,k} = \frac{1}{C_x\cdot C_y\cdot(C_t-1)}\cdot\sum_{m=0}^{C_x-1}\sum_{n=0}^{C_y-1}\sum_{f=0}^{C_t-2}\left|\nabla_{\mathbf{k}} I_{i,j,k}(m,n,f)\right| \qquad (3)$$

and the *average luminance change:*

$$\nabla^d_{\mathbf{k}} I_{i,j,k} = \frac{1}{C_x\cdot C_y\cdot(C_t-1)}\cdot\sum_{m=0}^{C_x-1}\sum_{n=0}^{C_y-1}\sum_{f=0}^{C_t-2}\left(\nabla_{\mathbf{k}} I_{i,j,k}(m,n,f)\right) \qquad (4)$$

Besides calculating the values (3) and (4), we keep track of the maximum time derivative value per pixel track of a block. For each spatial location (m, n) in the block (i,j,k), we search for the frame $f^{\max}_{i,j,k}(m,n)$, at which the maximum luminance change takes place, that is

$$f^{\max}_{i,j,k}(m,n) = \arg\max_f\left(\left|\nabla_{\mathbf{k}} I_{i,j,k}(m,n,f)\right|\right) \qquad (5)$$

After the frames (5) are determined for each pair (m, n), we average the maximum time derivative values found at these frames for all pairs (m, n), that is

$$\nabla^{\max}_{\mathbf{k}} I_{i,j,k} = \frac{1}{C_x\cdot C_y}\sum_{m=0}^{C_x-1}\sum_{n=0}^{C_y-1}\left|\nabla_{\mathbf{k}} I_{i,j,k}(m,n,f^{\max}_{i,j,k}(m,n))\right| \qquad (6)$$

The first two of the features we introduced above can now be defined as follows:

$$F_1(i,j,k) = \left| \frac{\nabla^d_{\mathbf{k}} I_{i,j,k}}{\nabla^a_{\mathbf{k}} I_{i,j,k}} \right| \tag{7}$$

and

$$F_2(i,j,k) = 1 - \frac{\nabla^{\max}_{\mathbf{k}} I_{i,j,k}}{\nabla^a_{\mathbf{k}} I_{i,j,k}} \tag{8}$$

The value of $F_1(i,j,k)$ equals to 1 if the function $I_{i,j,k}(m,n,f)$ is monotonically increasing or decreasing, and gets closer to zero as the fluctuations in the function values increase. The higher the value of $F_2(i,j,k)$ (i.e. close to 1), the more gradual (smooth) are the variations in the function $I_{i,j,k}(m,n,f)$ over time.

The block points $(m,n, f^{\max}_{i,j,k}(m,n))$ marking the maximum time derivative values per pixel track in a spatiotemporal video data block are also useful for detecting cuts and wipes. To do this, we calculate the feature $\boldsymbol{F}_3(i,j,k)$, which is the measure of whether the dominant changes in the luminance flow occur simultaneously for all pixel tracks, that is, whether the points $(m,n, f^{\max}_{i,j,k}(m,n))$ form a plane vertical to the time direction. For this reason a component $F_3^f(i,j,k)$ of the vector $\boldsymbol{F}_3(i,j,k)$ corresponds to a plane approximation error at the frame f of a block:

$$F_3^f(i,j,k) = \frac{1}{C_x \cdot C_y} \cdot \sum_{m=0}^{C_x-1} \sum_{n=0}^{C_y-1} \frac{\left(f^{\max}_{i,j,k}(m,n) - t_{\max dist} \right)^2}{\left(f^{\max}_{i,j,k}(m,n) - f \right)^2 + \varepsilon} \tag{9}$$

for $0 \le f < C_t$ and

$$t_{\max dist} = \begin{cases} 0 & \textbf{\textit{if}}\ f < C_t/2 \\ C_t - 1 & \textit{otherwise} \end{cases}$$

Here, ε is a small number, introduced to avoid division by zero in case of a perfectly planar distribution of maximum time derivative points.

We emphasize here that in the case of an overlap between consecutive blocks (defined by the factor α) the formula (9) may be used several times for one and the same frame $t = k\alpha C_t + f$ of a video, as this frame may correspond to different value pairs (f, k). In such cases, the value of the feature component $F_3^f(i,j,k)$ is computed as the mean of all values (9) computed for the same frame t.

The matrix in Figure 4 depicts the $F_3^f(i,j,k)$ values for an eight-second sports video that contains two cuts and two wipes. Each column contains the values of $F_3^f(i,j,k)$ collected row by row from all blocks sharing the same time index k. The brightness level of matrix elements directly reveals the values of $F_3^f(i,j,k)$. We observe that in case of a cut, high values of this feature are time-aligned, that is, they form a plane vertical to the time axis. On the other hand, a wipe is characterized by high feature values, which are not time-aligned, but distributed over a limited time interval. The characteristic regular patterns found for the wipes in Figure 4 correspond to the specific wipe type illustrated by the second example in Figure 2. One can also observe accidental high feature values between the transitions. These values mainly result from object or camera motion. For instance, the "cloud" of high feature values between two cuts in Figure 4 corresponds to a camera following a running player after scoring a goal. In the following section we define criteria for successfully distinguishing between such "clouds" and the patterns corresponding to cuts and wipes.

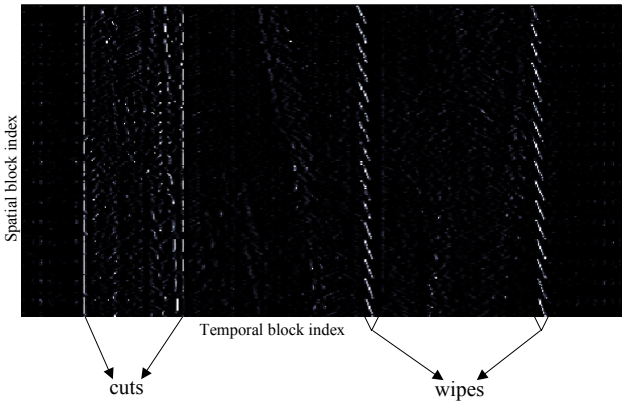

Figure 4: An illustration of $F_3^f(i,j,k)$ values along the time dimension

3. SHOT TRANSITION DETECTION

3.1 Cut detection

To detect cuts, we first integrate the elementary evidence found in the individual blocks and represented by the values $F_3^f(i,j,k)$, into the discriminative function $\psi_1(t)$, for $0 \le t < T$, which serves as an indicator of a cut at the frame t:

$$\psi_1(t) = \psi_1(k \cdot \alpha \cdot C_t + f) = \frac{1}{\left\lfloor \frac{X}{C_x} \right\rfloor \cdot \left\lfloor \frac{Y}{C_y} \right\rfloor} \sum_{i=0}^{\left\lfloor \frac{X}{C_x} \right\rfloor} \sum_{j=0}^{\left\lfloor \frac{Y}{C_y} \right\rfloor} F_3^f(i,j,k) \tag{10}$$

In the next step we apply a piecewise linear mapping of $\psi_1(t)$ values to the interval [0, 1] to obtain the probability of finding a cut at the observed frame t:

$$p^{abrupt}(t)=\begin{cases}0 & ,\textbf{\textit{if}}\ \psi_1(t)\le A\\ \dfrac{1}{B-A}\cdot\psi_1(t)-\dfrac{A}{B-A} & ,\textbf{\textit{if}}\ \boldsymbol{A}<\psi_1(t)<B\\ 1 & ,\textbf{\textit{if}}\ \psi_1(t)\ge B\end{cases} \quad (11)$$

Here, the parameters A and B are selected based on observing the distribution of the function $\psi_1(t)$ for cut and non-cut regions in a number of representative video sequences. Due to a rather clear separation of these regions, the selection of the parameters is not critical for the performance and can be kept constant for an arbitrary video being analyzed. For the same reason, a simple fixed threshold can be applied to filter out the cuts.

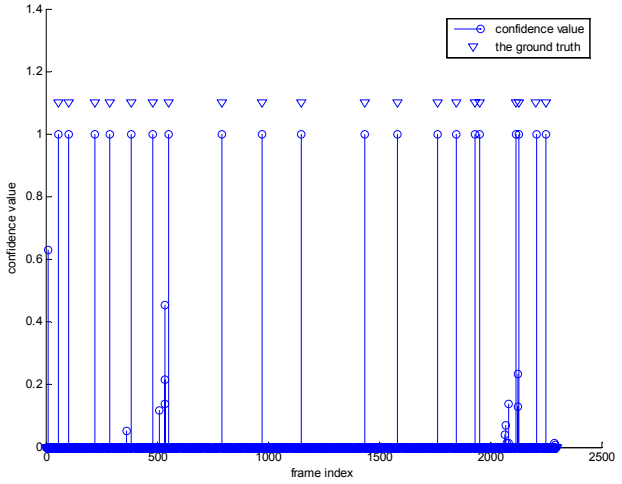

Figure 5. Probability values computed for a sample video sequence and aligned with ground truth positions of cuts

The mapping (11) is useful for enabling more intuitive selection of the detection threshold than when working with the function $\psi_1(t)$ directly. This threshold can namely be interpreted as the minimum acceptable probability that a detected cut will not be false. Although this thresholding mechanism is relatively simple, it proves sufficient to obtain a detection performance, which is more than satisfactory compared to the state-of-the-art. This is mainly due to a high discriminative power of the used features. Figure 5 illustrates this power on the example of a sample sequence from our test set. Clearly, all cuts and no false cuts are detected for any threshold ranging from 0.65 to 1. More information about the performance and a discussion on problematic cases are given in Section 4.

3.2 Fade/dissolve detection

Referring to our discussion in Section 2, the elementary evidence within blocks for detecting dissolves/fades is contained in the values of the features $F_1(i,j,k)$ and $F_2(i,j,k)$. As opposed to cuts, the locations of which are checked per frame t, we investigate here whether the blocks sharing the same temporal block index k belong to a transition or not. To do this we combine the available evidence from all time-aligned blocks for a given k into a discriminative function $\psi_3(k)$ indicating that the observed temporal video "slice" is a part of a dissolve/fade. We define this function as the average of the feature-based evidence values from all blocks belonging to the observed video slice:

$$\psi_3(k)=\frac{1}{\left\lfloor\frac{X}{C_x}\right\rfloor\cdot\left\lfloor\frac{Y}{C_y}\right\rfloor}\sum_{i=0}^{\left\lfloor\frac{X}{C_x}\right\rfloor}\sum_{j=0}^{\left\lfloor\frac{Y}{C_y}\right\rfloor}\left(F_1(i,j,k)\cdot F_2(i,j,k)\right) \quad (12)$$

Ideally, the function (12) shows high values for all consecutive video slices belonging to a dissolve, and low values elsewhere. However, to maximize the reliability of function values, we apply median filtering to the function (12) to eliminate its accidental (noisy) value fluctuations. We adopt the result of this operation as the probability that the time interval given by the index k is captured by a dissolve/fade, that is

$$p^{dissolve/fade}(k)=\textbf{\textit{median}}(\psi_3(k)) \quad (13)$$

Figure 6 illustrates the ranges of probability values corresponding to both detection hypotheses. Similarly as for the cuts, a simple thresholding mechanism can be applied for reliable dissolve/fade detection.

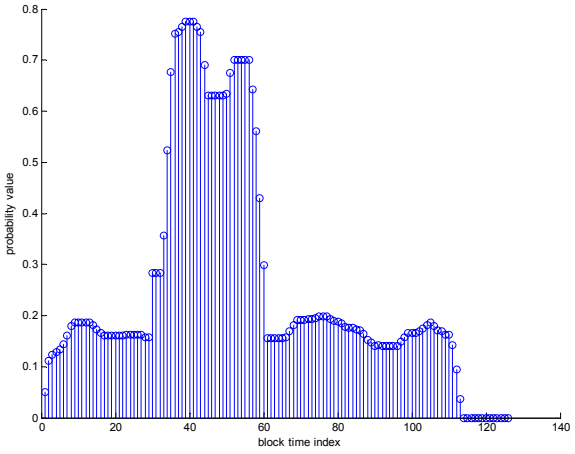

Figure 6. Probability values computed for a sample video sequence around and within a dissolve region

Finally, we find the starting and ending video slices, defined as u and v, respectively, of a series of detected consecutive fade/dissolve intervals and choose the frames in the middle of the blocks surrounding the detected block series as the approximate starting and ending frames of a transition:

$$\begin{aligned} t_{start}^{dissolve/fade} &= u \cdot \alpha \cdot C_t - (C_t/2) \\ t_{end}^{dissolve/fade} &= v \cdot \alpha \cdot C_t + (C_t/2) \end{aligned} \tag{14}$$

By artificially extending the dissolve/fade length to the surrounding blocks we generate more confidence that the entire transition is indeed captured by the borders (14).

3.3 Wipe detection

The detection of wipes in our system is fundamentally the same as the cut detection. Because of the limitations of the video frame rate, the wipes correspond to consecutive abrupt changes in different frame regions that are captured by spatially non-overlapping blocks. We detect a wipe if the blocks at different spatial locations contain abrupt changes in their pixel luminance tracks at different time points, but within a limited time interval.

We first apply (9) to calculate the significance of an abrupt change in the pixel luminance track at a frame f in block (i, j, k). Since we assume that the blocks change abruptly only once along a wipe, we relate the obtained result to the sum of the values (9) computed at the neighboring $2N$ frames surrounding the frame f, requiring that this sum can not exceed the value $F_3^f(i,j,k)$. Finally, we normalize the result of the comparison with respect to the neighboring frames, as defined in (15), to calculate the probability of a wipe-related discontinuity at the frame f:

$$p_f^{wipe}(i,j,k) = \frac{\max\left(0, F_3^f(i,j,k) - \sum_{\substack{q=-N \\ q\neq 0}}^{+N} F_3^{f+q}(i,j,k)\right)}{\sum_{q=-N}^{+N} F_3^{f+q}(i,j,k)} \tag{15}$$

Here, $p_f^{wipe}(i,j,k)$ has a high value (close to 1) only if the value (9) at frame f is considerably higher than at all other frames in the neighborhood. As an implementation detail, if the value of $f+q$ exceeds the block margins, the frames should be taken from the previous or the next block. For example, if $f+q>C_t$ then

$$F_3^{f+q}(i,j,k) = F_3^{f+q-C_t}(i,j,k+1) \tag{16}$$

Just like in the case of cut detection, we now define the discriminative function which indicates whether the frame t is a part of a wipe. This function integrates the elementary evidence contained in the probability (15) as follows:

$$\psi_2(t) = \psi_2(k \cdot \alpha \cdot C_t + f) = \frac{\sum_{i=0}^{\left\lfloor \frac{X}{C_x} \right\rfloor} \sum_{j=0}^{\left\lfloor \frac{Y}{C_y} \right\rfloor} p_f^{wipe}(i,j,k)}{\left\lfloor \frac{X}{C_x} \right\rfloor \cdot \left\lfloor \frac{Y}{C_y} \right\rfloor} \tag{17}$$

Following the same reasoning as in Section 3.1, we map the discriminative function onto probability of having a wipe at frame t:

$$p^{wipe}(t) = \begin{cases} 0 & , \text{if } \psi_2(t) \le D \\ \dfrac{1}{E-D} \cdot \psi_2(t) - \dfrac{D}{E-D} & , \text{if } D < \psi_2(t) < E \\ 1 & , \text{if } \psi_2(t) \ge E \end{cases} \tag{18}$$

Again, the parameters D and E are selected based on observing the distribution of the function $\psi_2(t)$ for wipe and non-wipe regions in a number of representative video sequences.

For a series of high probability values (18), we determine the starting and ending time stamps of a wipe in the similar way as in (14). Here, however, u and v are the frame indices, marking the beginning and ending frame of the detected wipe frame series:

$$\begin{aligned} t_{start}^{wipe} &= u - (C_t/2) \\ t_{end}^{wipe} &= v + (C_t/2) \end{aligned} \tag{19}$$

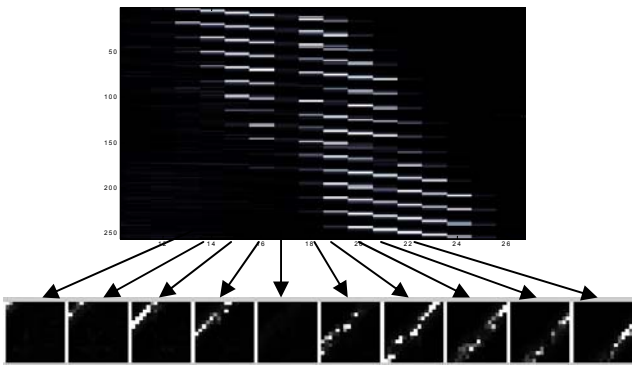

Figure 7. A zoom in on a wipe region from Figure 4. When we recombine the column data into 2-D images, we can clearly see the local abrupt changes that are propagating in the direction of a wipe.

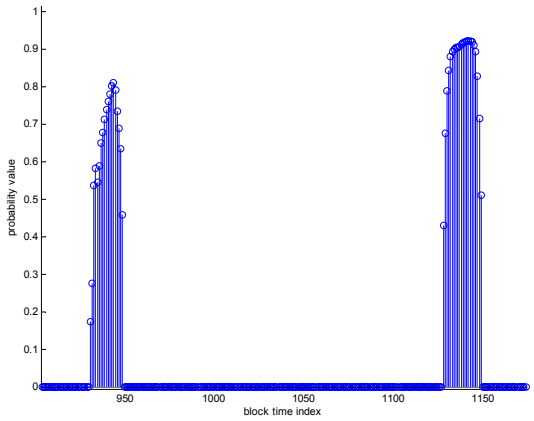

Figure 8. Probability values computed for a sample video sequence around and within two wipe regions

4. TRECVID 2005 RESULTS

TRECVID 2005 experiments were quite useful not only for juxtaposing our method against the state of the art in a fair manner but also for seeing the problems and rooms for improvement after the evaluation by an unbiased test platform. We had three observations out of the tests: the method has the potential of handling a few common problems that the frame based methods continuously suffer; the gradual transition detection suffers from the single directional analysis in 3D data blocks (this will be improved in 2006 and multidirectional feature extraction routines will be shared by shot detection and camera motion detection units); we need better compatibility with TRECVID rules, lack of which created some artificial deterioration in performance figures.

Our first observation is about the analysis of our detections and false alarms and misses. This newly developed method is able to handle naturally the fast motion and complicated graphical effects in general, although no specific action has been taken for individual potential problems (except for a simple flash detector).

The second observation is about the low recall rate in gradual transition detection (Table 1). The system currently takes into account the evolution of data in the spatiotemporal blocks only in the time direction. This causes a problem when the dissolves are combined with motion. Most of the misses in the gradual transition detection (especially in dissolves) stems from this fact. In the following version of the system, we use a full gradient based analysis to overcome this problem. Since our motion analysis unit already extracts full gradient information, shot detection unit will be able to borrow this extra piece of information.

And finally, we observed that there exist excessive amount of short gradual transitions in the test data (i.e. 3 frames in length) and we detect these short transitions constantly with one frame lag. As an example if there is a transition between frames 21-23 we detect it as a transition between frames 22-24. Because this is considered as one false detection and one miss, our cut detection results are decreased by 5 to 10% both in recall and precision (Table 2). Another source of error was that we consider fade in-outs as two separate transitions. More precisely if the screen turns to black and then dissolves into the following shot, we consider the black screen (or the graphical effect in between) as a separate shot and announce 2 transitions.

We believe that this 3D block based approach is more suitable for low level analysis than frame by frame analysis/comparison based methods and in its this first TRECVID experience showed some interesting and promising results.

Table 1. The obtained performance figures for the proposed shot transition detection algorithm

	Recall (%)	Precision (%)
Abrupt	91.8	82.3
Gradual	39.8	81.1

Table 2. The performance figures after the corrections in short gradual transition detection and fade in-outs.

	Recall (%)	Precision (%)
Abrupt	94.2	92.3 (94.5)
Gradual	49.4	82.0

5. CONCLUSIONS AND FUTURE WORK

In this work we explored the possibilities of utilizing the spatiotemporal block based analysis of video for constructing a unified framework for detecting and identifying different types of shot transitions. Our proposed method generally performed well. It showed some weaknesses under very fast object and camera motion and sudden illumination changes. It is however, the question to what extent these weaknesses can be improved without a higher-level (semantic) analysis of a video.

The biggest contribution of this paper we see in the availability of a unified framework for detecting a vast diversity of shot transition with a reasonably high performance. Further, as no complex, specialized video or image processing operation is employed, the method is also highly computationally efficient.

Finally, the methods and the concepts presented here are also directly applicable for other purposes as well, such as, for instance, local analysis of camera and object motion and scene organization. By extending the scope of our method in this way, we can also raise the performance of shot transition detection as the information on motion within spatiotemporal blocks can help better distinguish the pixel luminance behavior related to motion from those resulting from shot transitions.

REFERENCES

[1] Hanjalic, A., Content-Based Analysis of Digital Video, Kluwer Academic Publishers, 2004.

[2] Lienhart, R., "Reliable Transition Detection in Videos: A Survey and Practitioner's Guide", *in International Journal ofImage and Graphics, Vol. 3, pp. 469-486,* 2001.

[3] Gargi, U., R. Kasturi, S. H. Strayer, "Performance Characterization of Video-Shot-Change Detection Methods", *IEEE Transactions on Circuits and Systems for Video Technology, Vol. 10, No. 1, pp. 1- 13,* February 2000.

[4] Browne, P., A. F. Smeaton, N. Murphy, N. O'Connor, S. Marlow, and C. Berrut "Evaluation and combining digital video shot boundary detection algorithms", *in Proc. of the 4th Irish Machine Vision and Information Processing Conference*, Queens University Belfast, 2000.

[5] Lienhart, R., "Comparison of Automatic Shot Boundary Detection Algorithms", *in Proc. SPIE Vol. 3656 Storage and Retrieval for Image and Video Databases VII, pages 290–301,* San Jose, CA, USA, 1999.

[6] Ngo, C.-W., T.-C. Pong, R. T. Chin, "Video Partitioning by Temporal Slice Coherency", *IEEE Transactions on Circuits and Systems for Video Technology*, *vol. 11, no. 8,* August 2001.

[7] Kim, H., S.-J. Park, J. Lee, W. M. Kim, S. M.-H. Song, "Processing of Partial Video Data for Detection of Wipes", *IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases VII, vol. 3656,* San Jose, California, January 1999.

[8] Bescos J, Cisneros G, Martinez JM, et al. "A unified model for techniques on video-shot transition detection", *IEEE Transactions on Multimedia 7 (2): 293-307,* April 2005

[9] Boccignone G, Chianese A, Moscato V, et al., "Foveated shot detection for video segmentation" *IEEE Transactions on Circuits and Systems for Video Technology 15 (3): 365-377,* March 2005

[10] R. Ewerth, B. Freisleben: "Video Cut Detection without Thresholds", *11th Workshop on Signals, Systems and Image Processing, pp. 227-230,* Poznan, Poland, PTETiS, 2004

[11] Koprinska, I. and Carrato, S., "Video Segmentation: A Survey"*, Signal Processing: Image Communication*, *16(5), pp.477-500,* Elsevier Science, 2001.

[12] Hanjalic A.: "Shot-Boundary Detection: Unraveled and Resolved?", *IEEE Transactions on Circuits and Systems for Video Technology 12 (2): 90-105,* February 2002