Shot-Boundary Detection: Unraveled and Resolved?

Alan Hanjalic, *Member, IEEE*

***Abstract*—Partitioning a video sequence into *shots* is the first step toward video-content analysis and content-based video browsing and retrieval. A video shot is defined as a series of interrelated consecutive frames taken contiguously by a single camera and representing a continuous action in time and space. As such, shots are considered to be the primitives for higher level content analysis, indexing, and classification. The objective of this paper is twofold. First, we analyze the shot-boundary detection problem in detail and identify major issues that need to be considered in order to solve this problem successfully. Then, we present a conceptual solution to the shot-boundary detection problem in which all issues identified in the previous step are considered. This solution is provided in the form of a statistical detector that is based on minimization of the average detection-error probability. We model the required statistical functions using a robust metric for visual content discontinuities (based on motion compensation) and take into account all (*a priori*) knowledge that we found relevant to shot-boundary detection. This knowledge includes the shot-length distribution, visual discontinuity patterns at shot boundaries, and characteristic temporal changes of visual features around a boundary. Major advantages of the proposed detector are its robust and sequence-independent performance, while there is also the possibility to detect different types of shot boundaries simultaneously. We demonstrate the performance of our detector regarding two most widely used types of shot boundaries: hard cuts and dissolves.**

***Index Terms*—Shot-boundary detection, video analysis, video databases, video retrieval.**

I. Introduction

THE DEVELOPMENT of shot-boundary detection algorithms has the longest and richest history in the area of content-based video analysis and retrieval—longest, because this area was actually initiated some decade ago by the attempts to detect hard cuts in a video, and richest, because a vast majority of all works published in this area so far address in one way or another the problem of shot-boundary detection. This is not surprising, since detection of shot boundaries provides a base for nearly all video abstraction and high-level video segmentation approaches. Therefore, solving the problem of shot-boundary detection is one of the major prerequisites for revealing higher level video content structure. Moreover, other research areas can profit considerably from successful automation of shot-boundary detection processes as well. A good example is the area of video restoration. There, the restoration efficiency can be improved by comparing each shot with previous ones and—if a similar shot in terms of visual characteristics is found in the past—by adopting the restoration settings already used before. Further, in the process of coloring black-and-white movies, the knowledge about shot boundaries provides time stamps where switch to a different gray-to-color look-up table should take place.

However, despite countless proposed approaches and techniques so far, robust algorithms for detecting various types of shot boundaries have not been found yet. We relate here the attribute “robust” to the following major criteria:

1) excellent detection performance for all types of shot boundaries (hard cuts and gradual transitions);
2) constant quality of the detection performance for any arbitrary sequence, with minimized need for manual fine-tuning of detection parameters in different sequences.

Regarding the usage of shot-boundary detection algorithms in the processes of video restoration and coloring, fulfilling the two aforementioned criteria is the major prerequisite to a successful automation of these processes. If the detection performance is poor, substantial involvement of the operator is required in order to correct wrong restoration settings or gray-to-color look-up table. Moreover, if the detection performance is sequence dependent, it can be difficult for the operator to find optimal detector settings for each sequence to be restored or colored. For the processes of high-level video content analysis, fulfilling of the aforementioned criteria by the shot-boundary detector has even a larger importance. First, bad detection performance may negatively influence the performance of subsequent high-level video analysis modules (e.g., movie segmentation into episodes, movie abstraction, broadcast news segmentation into reports). Second, if we cannot expect a video restoration/coloring operator (expert) to adjust the shot-boundary detector settings to different sequences, this can be expected even less from a nonprofessional user of commercial video-retrieval equipment.

The objective of this paper is twofold. We first analyze the problem of shot-boundary detection in detail and identify all issues that need to be considered in order to solve this problem in view of the two criteria listed above. Then we present a conceptual solution to the shot-boundary detection problem in which all issues identified in the previous step are considered and using which we aim at fulfilling the two aforementioned robustness criteria. This solution is provided in the form of a statistical detector that is based on minimization of the average detection-error probability.

The paper is structured as follows. Section II gives a detailed analysis of the shot-change detection problem, while Section III provides an extensive overview of the solutions to this problem, proposed so far. The main purpose of Sections II and III is to unravel the shot-boundary detection problem and so to explain our motivation for developing our statistical detector in the first place and also to justify the choices made in the process of detector development. We present our statistical detector in detail

Manuscript received February 7, 2000; revised November 23, 2001. This paper was recommended by Associate Editor S.-F. Chang.

The author is with the Faculty of Information Technology and Systems, Department of Mediamatics, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: A.Hanjalic@its.tudelft.nl).

Publisher Item Identifier S 1051-8215(02)02015-3.

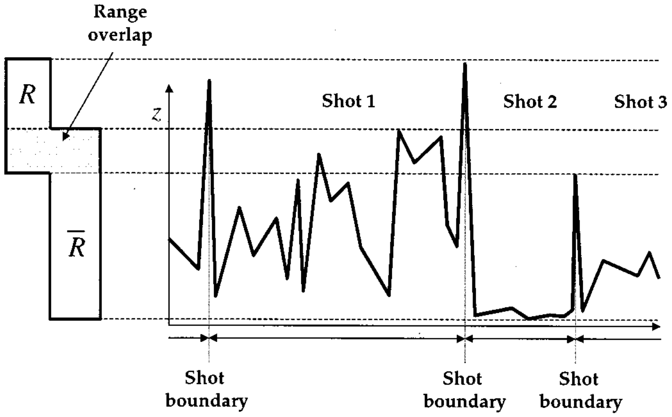


Fig. 1. The problem of unseparated ranges $\overline{R}$ and R.

in Section IV, while in Section V we demonstrate its performance for the two most widely used types of shot boundaries: hard cuts and dissolves. We conclude this paper with a discussion in Section VI.

II. Shot-Boundary Detection: A Problem Analysis

The basis of detecting shot boundaries in video sequences is the fact that frames surrounding a boundary generally display a significant change in their visual contents. The detection process is then the recognition of considerable *discontinuities* in the visual-content flow of a video sequence. In the first step of this process, *feature extraction* is performed, where the features depict various aspects of the visual content of a video. Then, a *metric* is used to quantify the feature variation from frame k to frame $k+l$, with l being the inter-frame distance (skip) and $l \geq 1$. The discontinuity value $z(k, k+l)$ is the magnitude of this variation and serves as an input into the *detector*. There, it is compared against a *threshold* T. If the threshold is exceeded, a shot boundary between frames k and $k+l$ is detected.

To be able to draw reliable conclusions about the presence or absence of a shot boundary between frames k and $k+l$, we need to use the features and metrics for computing the discontinuity values $z(k, k+l)$ that are as discriminating as possible. This means that a clear separation should exist between discontinuity-value ranges for measurements performed *within shots* and *at shot boundaries*. In the following, we will refer to these ranges as $\overline{R}$ and R, respectively. The problem of having unseparated ranges $\overline{R}$ and R is illustrated in Fig. 1, where some discontinuity values within shot 1 belong to the overlap area. Such values $z(k, k+l)$ make it difficult to decide about the presence or absence of a shot boundary between frames k and $k+l$ without avoiding detection mistakes, i.e., *missed* or *falsely detected* boundaries.

We realistically assume that the visual-content differences between consecutive frames within the same shot are mainly caused by two factors: *object/camera motion* and *lighting changes*. Depending on the magnitude of these factors, the computed discontinuity values within shots vary and sometimes lie in the overlap area, as shown in Fig. 1. Thus, the easiest way of obtaining good discrimination between ranges $\overline{R}$ and R is to use features and metrics that are insensitive to motion and lighting changes. Even more, since different types of sequences can globally be characterized by their average rates and magnitudes of object/camera motion and lighting changes (e.g., high-action movies versus stationary dramas), eliminating these distinguishing factors also provides a high level of *consistency* of ranges $\overline{R}$ and R across different sequences. If the ranges $\overline{R}$ and R are consistent, the parameters of the detection system (e.g., the threshold T) can first be optimized on a set of training sequences to maximize the detection reliability, and then the system can be used to detect shot boundaries in an arbitrary sequence without any human supervision, while retaining a high detection reliability. In this way, selecting features and metrics as described above would automatically lead to a shot-boundary detector conform to the criteria defined in the introduction to this paper.

However, while features and metrics can be found such that the influence of motion on discontinuity values is strongly reduced, the influence of strong and abrupt lighting changes on discontinuity values and thus also on the detection performance cannot be reduced that easily. For instance, one could try working only with chromatic color components, since common lighting changes can mostly be captured by luminance variations. But this is not an effective solution in extreme cases, where all color components are changed. Strong and abrupt lighting changes can result in a series of high discontinuity values, which can be mistaken for the actual shot boundaries. In the remainder of this paper, we refer to possible causes for high discontinuity values within shots as *extreme factors*. These factors basically include strong and abrupt lighting changes, but also some extreme motion cases that cannot be captured effectively by selecting features and metrics as mentioned above.

An effective way to reduce the influence of extreme factors on the detection performance is to embed additional information in the shot-boundary detector. The main characteristic of this information is that it is not based on the range of discontinuity values but on some other measurements performed on a video, that—each in its own way—indicate the presence or absence of a shot boundary between frames k and $k+l$. As a first example, we introduce the information resulting from a comparison of a temporal pattern created by consecutive discontinuity values (measured pattern) and known temporal patterns that are specific for different types of shot boundaries (template patterns). In general, we can distinguish *hard cuts*, which are the most common boundaries and occur between two consecutive frames, from *gradual transitions*, such as fades, wipes, and dissolves, which are spread over several frames. Then, the decision about the presence or absence of a shot boundary between frames k and $k+l$ made by the detector is not only based on range information, that is, on the comparison of the discontinuity value $z(k, k+l)$ and the threshold T, but also—as shown in Fig. 2—on the match between the measured pattern formed by discontinuity values surrounding $z(k, k+l)$ and a template pattern of a shot boundary.

Another type of additional information that can be useful in supporting the decision process in the shot-boundary detector results from observation of the characteristic behavior of some visual features along frames surrounding a shot boundary for the cases of gradual transitions. Let us for this purpose consider one specific boundary type—a dissolve—and observe the temporal

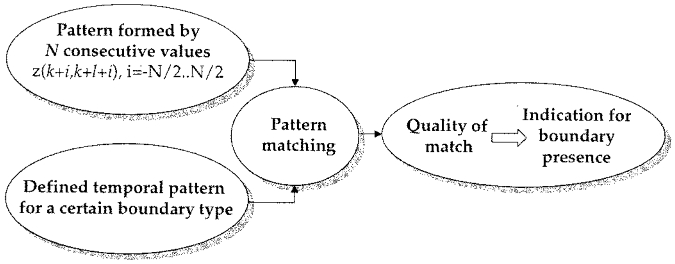

Pattern formed by
N consecutive values
z(k+i,k+l+i), i=-N/2..N/2
Pattern matching
Quality of match
Indication for boundary presence
Defined temporal pattern for a certain boundary type

Fig. 2. Matching of the temporal pattern formed by N consecutive discontinuity values and a temporal pattern characteristic for a shot boundary. The quality of match between two patterns provides an indication for boundary presence between frames k and $k+l$ that can be used as additional information in the detector.

behavior of intensity variance that is measured for every frame within a dissolve. Since a dissolve is the result of mixing the visual material from two neighboring shots, it can be expected that variance values measured per frame along a dissolve ideally reveal a downwards-parabolic pattern [2], [15], [17]. Hence, the decision about the presence of a dissolve can be supported by investigating the behavior of the intensity variance in the "suspected" series of frames (e.g., those where pattern matching from Fig. 2 shows good results) and by checking how well this behavior fits the downwards-parabolic pattern.

Further improvement of the detection performance can be obtained by taking into account *a priori* information about the presence or absence of a shot boundary at a certain time stamp along a video. We differentiate here between additional and *a priori* information because the latter is not based on any measurement performed on a video sequence. An example of *a priori* information is the dependence of the probability for shot boundary occurrence on the number of elapsed frames since the last detected shot boundary. While it can be assumed zero at the beginning of a shot, this probability grows and converges to the value 0.5 with increasing number of frames in the shot. The main purpose of this probability is to make the detection of one shot boundary immediately after another one practically impossible and so to contribute to a reduction of false detection rate. Therefore, by properly modeling *a priori* probability and by securing its convergence to 0.5, the influence of this probability on the detection performance should be minimized as soon as a reasonable shot length is reached.

In view of the discussion in previous paragraphs, combining motion compensating features and metrics for computing the discontinuity values with additional information that can help reducing the influence of extreme factors and with *a priori* information about the shot-boundary presence or absence at a certain time stamp, we are likely to provide a solid base for creating a detector that is optimal with respect to the criteria defined in Section I. Such a detector is illustrated in Fig. 3.

Variation of the detection threshold for each frame k is a consequence of embedding additional and *a priori* information into the detector. This information regulates the detection process by continuously adapting the threshold e.g., to the quality of the boundary or variance pattern match for each new series of consecutive discontinuity values and to the time elapsed since the last detected shot boundary. The remaining task is to define a detector where the above components are integrated such that the resulting threshold function $T(k)$ provides optimal detection performance. We will proceed with the development of such detector in Section IV after investigating the advantages and disadvantages of shot-boundary detection methods published in recent literature.

III. Previous Work on Shot-Boundary Detection

Developing techniques for detecting shot boundaries in a video has been the subject of substantial research over the last decade. In this section, we give an overview of the relevant literature. The overview concentrates, on the one hand, on the capability of features and metrics to reduce the motion influence on discontinuity values. On the other hand, it investigates existing approaches to shot-boundary detection, involving the threshold specification, treatment of different boundary types, and usage of additional and *a priori* information to improve the detection performance.

A. From Features and Metrics to Discontinuity Values

Different methods exist for computing discontinuity values, employing various features related to the visual content of a video. For each selected feature, a number of suitable metrics can be applied. Good comparisons of features and metrics used for shot-boundary detection with respect to the quality of the obtained discontinuity values can be found in [1], [6], [9], [13], [15].

The simplest way of measuring the visual-content discontinuity between two frames is to compute the mean absolute change of intensity $I(x, y)$ between the frames k and $k+l$ for all frame pixels, i.e., for $1 \leq x \leq X$ and $1 \leq y \leq Y$, where X and Y are the frame dimensions [12]. A modification of this technique is only counting the pixels that change considerably from one frame to another [20]. Here, the absolute intensity change is compared with the pre-specified threshold T_1, and is only considerable if it exceeds that threshold, that is

$$z(k, k+l) = \frac{1}{XY} \sum_{x=1}^{X} \sum_{y=1}^{Y} D_{k, k+l}(x, y)$$

with

$$D_{k, k+l}(x, y) = \begin{cases} 1, & \text{if } |I_k(x, y) - I_{k+l}(x, y)| > T_1 \\ 0, & \text{else.} \end{cases} \tag{1}$$

An important problem of the two approaches presented above is the sensitivity of discontinuity values $z(k, k+l)$ to camera and object motion. To reduce the motion influence, a modification of the described techniques was presented in [30], where a 3 × 3 averaging filter was applied to frames before performing the pixel comparison.

Much higher motion independence show the approaches based on motion compensation. There, a *block matching* procedure is applied to find for each block $b_i(k)$ in frame k a corresponding block $b_{i,m}(k+l)$ in frame $k+l$, such that it is most similar to the block $b_i(k)$ according to a chosen criterion (difference formula) D, that is

$$\begin{aligned} D_{k, k+l}(i) &= D(b_i(k), b_{i,m}(k+l)) \\ &= \min_{j=1 \cdots N_{\text{Candidates}}} D(b_i(k), b_{i,j}(k+l)). \end{aligned} \tag{2}$$

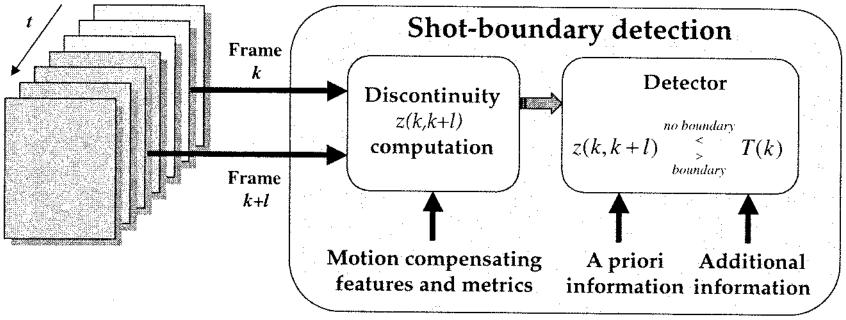


Fig. 3. Shot-boundary detector where all issues are taken into account that are relevant for optimizing the detection performance.

Here, $N_{\text{Candidates}}$ is the number of candidate blocks $b_{i,j}(k+l)$ considered in the procedure to find the best match for a block $b_i(k)$. If k and $k+l$ are neighboring frames of the same shot, the values $D_{k,k+l}(i)$ can generally be assumed low. This is because, for a block $b_i(k)$, almost the identical block $b_{i,m}(k+l)$ can be found due to a global constancy of the visual content along consecutive frames of a shot. This is not the case if frames k and $k+l$ surround a shot boundary because, in general, the difference between corresponding blocks in the two frames will be large due to a radical change in visual content across a boundary. Thus, computing the discontinuity value $z(k, k+l)$ as a function of differences $D_{k,k+l}(i)$ is likely to provide a reliable base for detecting shot boundaries.

An example of computing the discontinuity values based on the results of block-matching procedure is given in [23]. There, a frame k is divided into $N_{\text{Blocks}} = 12$ nonoverlapping blocks and the differences $D(b_i(k), b_{i,j}(k+l))$ are computed by comparing pixel-intensity values within blocks. Then, the obtained differences $D_{k,k+l}(i)$ are sorted and normalized between 0 and 1 (where 0 indicates a perfect match), giving the values $d^s_{k,k+l}(i)$. These values are multiplied with weighting factors c_i and summarized over the entire frame to give the discontinuity values, that is

$$z(k, k+l) = \sum_{i=1}^{N_{\text{Blocks}}} c_i d^s_{k,k+l}(i). \tag{3}$$

A popular alternative to pixel-based approaches is using histograms as features. Consecutive frames within a shot containing similar global visual material will show little difference in their histograms, compared to frames on both sides of a shot boundary. Although it can be argued that frames having completely different visual contents can still have similar histograms, the probability of such a case is small. Since histograms ignore spatial changes within a frame, histogram differences are considerably more insensitive to *object* motion with a constant background than pixel-wise comparisons are. However, a histogram difference remains sensitive to *camera* motion, such as panning, tilting, or zooming.

If histograms are used as features, the discontinuity value can be obtained by bin-wise computing the difference between frame histograms. Both grey-level and color histograms are used in literature, and their differences are computed by a number of metrics. Some mostly used ones are the sum of absolute differences of corresponding bins [29] and the so-called χ^2-test [18]. Further, a metric involving histograms in the HVC color space [9] (*Hue*—color type, *Value*—intensity, luminance; *Chroma*—saturation, the degree to which color is present) exploits the advantage of the invariance of Hue under different lighting conditions. This is useful in reducing the influence of common (weak) lighting changes on discontinuity values. Such an approach is proposed in [4], where only histograms of H and C components are used. These 1-D histograms are combined into a 2-D surface, serving as a feature. Based on this, the discontinuity is computed as

$$z(k, k+l) = \sum_{x=1}^{X}\sum_{y=1}^{Y}\{|\delta_{k,k+l}(x, y)|\times\Delta_{\text{Hue}}\times\Delta_{\text{Chroma}}\} \tag{4}$$

where $\delta_{k,k+l}(x, y)$ is the difference between the bins at coordinates (x, y) in HC-surfaces of frames k and $k+l$, and Δ_{Hue} and Δ_{Chroma} are the resolutions of Hue and Chroma components used to form the 2-D histogram surface.

Also the histograms computed block-wise can be used for shot-boundary detection, as shown in [18]. There, both the images k and $k+l$ are divided into 16 blocks, histograms $H_{k,i}$ and $H_{k+l,i}$ are computed for blocks $b_i(k)$and $b_i(k+l)$ and the χ^2-test is used to compare corresponding block histograms. When computing the discontinuity as a sum of region-histogram differences, eight largest differences were discarded to efficiently reduce the influence of motion and noise. An alternative to this approach can be found in [27], where first the number of blocks is increased to 48, and then the discontinuity value is computed as the total number of blocks within a frame, for which the block-wise histogram difference exceeds a pre-specified threshold T_1, that is

$$z(k, k+l) = \sum_{i=1}^{48} D(b_i(k), b_i(k+l)) \tag{5}$$

with

$$D(b_i(k), b_i(k+l)) = \begin{cases} 1, & \text{if } \dfrac{1}{N_{\text{Bins}}}\displaystyle\sum_{j=1}^{N_{\text{Bins}}}\dfrac{(H_{k,i}(j)-H_{k+l,i}(j))^2}{H_{k,i}(j)} > T_1 \\ 0, & \text{else.}\end{cases} \tag{6}$$

According to [19], the approach from [27] is much more sensitive to hard cuts than the one proposed in [18]. However, since emphasis is put on blocks, which change most from one frame to another, the approach from [27] also becomes highly sensitive to motion.

Another characteristic feature that proved to be useful in detecting shot boundaries is edges. As described in [16], first the overall motion between frames is computed. Based on the motion information, two frames are registered and the number and position of edges detected in both frames are compared. The total difference is then expressed as the total edge change percentage, i.e., the percentage of edges that enter and exit from one frame to another. Due to registration of frames prior to edge comparison, this feature is robust against motion. However, the complexity of computing the discontinuity values is also high. Let p_k be the percentage of edge pixels in frame k for which the distance to the closest edge pixel in frame $k+l$ is larger than the pre-specified threshold T_1. In the same way, let p_{k+l} be the percentage of edge pixels in frame $k+l$, for which the distance to the closest edge pixel in frame k is larger than the pre-specified threshold T_1. Then, the discontinuity value between these frames is computed as

$$z(k,\, k+l) = \max(p_k,\, p_{k+l}). \tag{7}$$

At last, we mention here the computation of the discontinuity value $z(k,\, k+l)$ using the analysis of the motion field measured between two frames. An example for this is the approach proposed in [3], where the discontinuity value $z(k,\, k+1)$ between two consecutive frames is computed as the inverse of *motion smoothness*.

B. Detection Approaches

Threshold Specification for Detecting Hard Cuts: The problem of choosing the right threshold for evaluating the computed discontinuity values has not been addressed extensively in literature. Most authors work with heuristically chosen global thresholds [4], [18], [20]. An alternative is given in [30], where first the statistical distribution of discontinuity values within a shot is measured. Then the obtained distribution is modeled by a Gaussian function with parameters μ and σ, and the threshold value is computed as

$$T = \mu + r\sigma. \tag{8}$$

Here, r is the parameter related to the prespecified tolerated probability for false detections. For instance, when $r = 3$, the probability of having falsely detected shot boundaries is 0.1%. The specification of the parameter r can only explicitly control the rate of false detections. The rate of missed detections is implicit and cannot be regulated, since the distribution of discontinuity values measured on boundaries is not taken into account. However, even if they can be specified in a nonheuristic way, the crucial problem related to the global threshold still remains, as illustrated in Fig. 4. If the prespecified global threshold is too low, many false detections will appear in the shot, where high discontinuity values are caused by extreme factors, as defined in Section II. If the threshold is made higher to avoid falsely detected boundaries, then the high discontinuity value corresponding to the shot boundary close to frame 500 (in Fig. 4) will not be detected.

A much better alternative is to work with adaptive thresholds, i.e., with thresholds computed locally. The improved detection performance that results from using adaptive threshold function $T(k)$ instead of the global threshold T is also illustrated in Fig. 4. If the value of the function $T(k)$ is computed at each frame k based on the extra information embedded in the detector (Fig. 3), high discontinuity values computed within shots can be distinguished from those computed at shot boundaries.

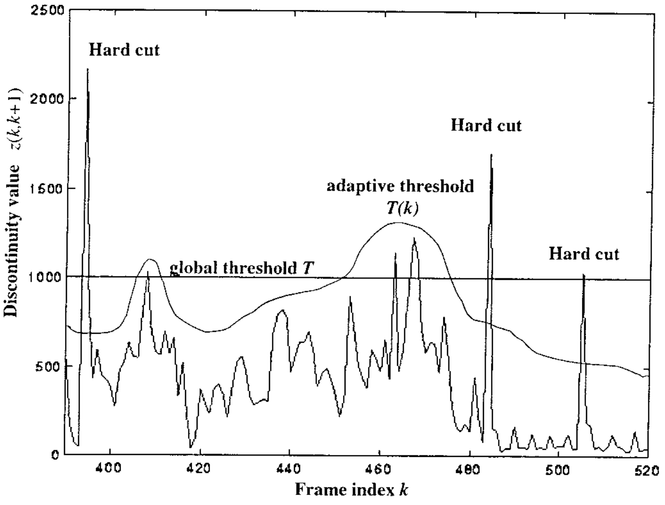


Fig. 4. Improved detection performance when using an adaptive threshold function $T(k)$ instead of a global threshold T.

A method for detecting hard cuts using an adaptive threshold is presented in [29]. There, the values $T(k)$ are computed using the information about the temporal pattern that is characteristic for hard cuts. The authors compute the discontinuity values with the inter-frame distance $l = 1$. As shown in Fig. 5, the N last computed consecutive discontinuity values are considered, forming a sliding window. The presence of a shot boundary is checked at each window position, in the middle of the window, according to the following criterion:

$$\begin{aligned} &if \quad z(k,\, k+1) = \max_{i=-N/2,\, \ldots,\, N/2} (\forall\, z(k+i,\, k+1+i)) \\ &\qquad \wedge\, z(k,\, k+1) \geq \alpha z_{sm} \Rightarrow \mathit{abrupt\ shot\ boundary}. \end{aligned} \tag{9}$$

In other words, a hard cut is detected between frames k and $k+1$ if the discontinuity value $z(k,\, k+1)$ is the window maximum and α times larger than the second largest discontinuity value z_{sm}within the window. The parameter α can be understood as the *shape parameter* of the boundary pattern. This pattern is characterized by an isolated sharp peak in a series of discontinuity values. Applying (9) to such a series at each position of a sliding window is nothing else than matching the ideal pattern shape and the actual behavior of discontinuity values found within the window. The major weakness of this approach is the heuristically chosen and fixed parameter α. Because α is fixed, the detection procedure is too coarse and too inflexible, and because it is chosen heuristically, one cannot make statements about the scope of its validity.

In order to make the threshold specification in [29] less heuristic, a detection approach was proposed in [11], which combines the sliding window methodology with the Gaussian distribution of discontinuity values proposed in [30]. Instead of choosing the form parameter α heuristically, this parameter

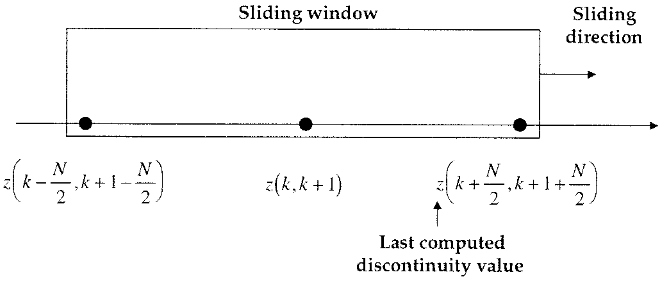

Sliding window
Sliding direction
$z\left(k-\frac{N}{2},k+1-\frac{N}{2}\right)$
$z(k,k+1)$
$z\left(k+\frac{N}{2},k+1+\frac{N}{2}\right)$
Last computed discontinuity value

Fig. 5. Illustration of a sliding window approach from [29].

is determined indirectly, based on the pre-specified tolerable probability for falsely detected boundaries. However, similar to [30], the rate of missed detections cannot be regulated since the distribution of discontinuity values measured on boundaries is not taken into account.

One way in which the additional and *a priori* information embedded in the detector can influence the process of shot-boundary detection much more effectively is using the *statistical detection theory*. One of the first applications of the statistical detection theory to signal analysis can be traced back to the work of Curran and Ross in [8]. A characteristic example of recent works in this area can be found in [28]. There, the proposed statistical method for detecting hard cuts includes *a priori* information based on shot-length distributions, which can be assumed consistent for a wide range of sequences. Besides, the detection rule based on comparing the likelihoods of two hypotheses (“boundary,” “no boundary”) is obtained as a result of statistical minimization of the detection error, which makes the resulting threshold function $T(k)$ statistically optimal for given likelihoods and *a priori* probability function. In view of this, the method proposed by [28] has come closest to the optimal solution of the shot-boundary detection problem so far, in view of the discussion in Section II. However, there are three major imperfections in this method, which negatively influence its detection performance. First, the authors use motion-sensitive features and metrics, which makes the detection performance sequence dependent and vulnerable by extreme factors. Second, no mechanism is present in the detector that can reduce the influence of the extreme factors on the detection performance. Third, it is not clear how the detector can be extended to deal with other types of shot boundaries.

Detection of Gradual Transitions: Different boundary types were considered in most of the approaches presented in recent literature, although the emphasis was mostly put on the detection of hard cuts. This preference can be explained by the fact that there is no strictly defined behavior of discontinuity values around and within gradual transitions. While hard cuts are always represented by an isolated high discontinuity value, the behavior of these values around and within a gradual transition is not unique, not even for one and the same type of transition. In the following, we will present some recent approaches to detecting gradual transitions.

One of the first attempts for detecting gradual transitions can be found in [30], where a so-called twin-comparison approach is described. The method requires two thresholds: a higher one, T_h, for detecting hard cuts, and a lower one, T_l, for detecting gradual transitions. First the threshold T_h is used to detect high discontinuity values corresponding to hard cuts, and then the threshold T_l is applied to the rest of the discontinuity values. If a discontinuity value is higher than T_l, it is considered to be the start of a gradual transition. At that point, the summation of consecutive discontinuity values starts and goes on until the cumulative sum exceeds the threshold T_h. Then, the end of the gradual transition is set at the last discontinuity value included in the sum.

In [10], a model-driven approach to shot-boundary detection can be found. There, different types of shot boundaries are considered to be editing effects, and are modeled based on the video production process. Especially for dissolves and fades, different chromatic scaling models are defined. Based on these models feature detectors are designed and used in a feature-based classification approach to segment the video. The described approach takes into account all types of shot boundaries defined by the models.

One further method for detecting one specific class of gradual transitions, dissolves, investigates the temporal behavior of the intensity variance of the frame pixels. This variance-based approach was first proposed by Alattar in [2], but has been used and modified by other authors as well (e.g., [17]). Since, within a dissolve, different visual material is mixed, a characteristic downwards-parabolic pattern revealed by variance values measured along frames of a dissolve is reported. The intensity variance starts to decrease at the beginning of the transition, reaches its minimum in the middle of the transition and then starts to increase again. The detection of the transition is then reduced to detecting the parabolic curve pattern in a series of measured variances. In order to be recognized as a dissolve, the potential pattern has to have a width and the depth that exceeds the pre-specified thresholds.

One of the major problems in the above approach is that the two large negative spikes of the parabolic pattern are not sufficiently pronounced due to noise and motion in a video. This has been addressed by Truong *et al.*, who proposed an improved version of the variance-based detector of Alattar [2]. Truong *et al.* proposed to exploit the facts: 1) that the first derivative of the pattern should be monotonically increasing from a negative to a positive value; 2) that intensity variances of both shots involved should be larger than a given threshold; and 3) that the dissolve duration usually falls between two well-defined thresholds [25], [26].

In [24], a chromatic video edit model for gradual transitions is built based on the assumption that discontinuity values belonging to such a transition form a pattern consisting of two piece-wise linear functions of time, one decreasing and one increasing. Such linearity does not apply outside the transition area. Therefore, the authors search for close-to-linear segments in the series of discontinuity values by investigating the first and the second derivative of the slope in time. A close-to-linear segment is found if the second derivative is less than a pre-specified percentage of the first derivative.

Although each of the described models is reported to perform well in most cases, strong assumptions are made about the behavior of discontinuity values within a transition. Furthermore, several (threshold) parameters need to be set heuristically. The fact that patterns which are formed by consecutive discontinuity values and correspond to a gradual transition can strongly vary over different sequences has moved Lienhart [14] to propose a conceptually different approach to detecting gradual transitions, in this case dissolves. The approach is less concerned about actual features used for dissolve detection, but more with a general framework for recognizing gradual transitions. First, a huge number of dissolve examples are created from a given video database using a dissolve synthesizer. Then these examples are used to train a "heuristically optimal" classifier which is then employed in a multi-resolution search for dissolves of various durations.

Although relatively good results are reported in [14], [25], and [26], these results are still far from those obtained for hard cuts. As will be shown in the following sections, further improvements in detecting gradual transitions are possible.

IV. A Robust Statistical Shot-Boundary Detector

In this section, we develop a statistical shot-boundary detector which is required as a last component in the optimized shot-boundary detection scheme in Fig. 3. This detector should integrate the range information as well as the additional and *a priori* information, resulting in an adaptive threshold $T(k)$ that provides optimal detection performance. Since we employ statistical detection theory for this purpose, our approach can best be compared with the one of Vasconcelos and Lippman in [28]. Statistical detection theory provides tools to integrate all information relevant to the detection process and to obtain the threshold function $T(k)$ based on the criterion that the average probability for detection mistakes is minimized. In other words, the detection performance can be made statistically optimal for given input information.

In order to provide optimal input information for the detection process, we introduce three major modifications in the approach from [28], inspired by the problem analysis in Section II. First, in order to provide a high level of discrimination between ranges $\overline{R}$ and R, as well as sequence independence of these ranges, we compute the discontinuity values using motion compensating features and metrics. Second, we embed the additional information on temporal boundary patterns and on the behavior of the intensity variance into our detector in order to minimize the effect of extreme factors on the detection performance. Finally, in the development of our detector, we take into account all types of shot boundaries, and not only hard cuts.

A. Detector Development

In terms of the statistical detection theory, shot-boundary detection can be formulated as the problem of deciding between the following two hypotheses.

1) *Hypothesis* S: Boundary present between frames k and $k+l$.
2) *Hypothesis* $\overline{S}$: No boundary present between frames k and $k+l$.

When making the above decision, two types of errors can occur. A *false detection* occurs if hypothesis S is selected while the hypothesis $\overline{S}$ is the right one. Analogously, if the hypothesis $\overline{S}$ is selected while S is the right one, the error corresponding to a *missed detection* occurs. If we assume that all detection errors (e.g., both missed and false detections) are treated equally, the quality of a statistical detector is determined by the average probability P_E that any of the errors occurs. Before we express P_E analytically, we first define the probability for the occurrence of each error type separately. The probability P_M for a missed detection and the probability P_F for a false detection can be computed using (10) and (11), respectively

$$P_M = \int_{Z_{\overline{S}}} p(z|S)\,dz \tag{10}$$

$$P_F = \int_{Z_S} p\left(z|\overline{S}\right)\,dz. \tag{11}$$

We call $Z_{\overline{S}}$ and Z_S the *discontinuity-value ranges* belonging to the hypothesis $\overline{S}$ and S, respectively. The range $Z_{\overline{S}}$ contains all discontinuity values $z(k,\,k+l)$, for which the detector chooses for the hypothesis $\overline{S}$, and vice-versa. The *likelihood functions* $p(z|S)$ and $p(z|\overline{S})$ are precomputed using training data, and represent the likelihood that an arbitrary discontinuity value $z(k,\,k+l)$ is obtained at time stamps where a shot boundary occurs and where no shot boundary occurs, respectively. Likelihood functions can be considered analogous to previously used ranges of discontinuity values $\overline{R}$ and R. Consequently, the requirements for a good discrimination between ranges can now be transferred to the likelihood functions $p(z|S)$ and $p(z|\overline{S})$.

Average probability for a detection error can now be formulated as follows:

$$\begin{aligned} P_E &= P_E(k) \\ &= P_k(S)P_M + P_k\left(\overline{S}\right)P_F \\ &= P_k(S)\int_{Z_{\overline{S}}} p(z|S)\,dz + P_k\left(\overline{S}\right)\int_{Z_S} p\left(z|\overline{S}\right)\,dz. \end{aligned} \tag{12}$$

Here, $P_k(S)$ and $P_k(\overline{S})$ are the probabilities for the validity of the hypothesis S and $\overline{S}$, respectively, between frames k and $k+l$. Since we have only two hypotheses to choose from, we can eliminate $P_k(\overline{S})$ using the expression

$$P_k\left(\overline{S}\right) = 1 - P_k(S). \tag{13}$$

Now we can further concentrate on the probability $P_k(S)$ only. We define $P_k(S)$as the product of *a priori* probability $P_k^a(S)$ for the hypothesis S and the conditional probability $P_k(S|\psi(k))$ for the same hypothesis between frames k and $k+l$ that depends on additional information $\psi(k)$ collected from a video, that is

$$P_k(S) = P_k^a(S)P_k(S|\psi(k)). \tag{14}$$

As already explained in Section II, we differentiate between $P_k^a(S)$ and $P_k(S|\psi(k))$, since *a priori* probability $P_k^a(S)$ is independent on any results obtained by analyzing a video. On the one hand, we base the computation of $P_k^a(S)$ solely on the number of frames elapsed since the last detected shot boundary. On the other hand, we make $P_k(S|\psi(k))$ dependent on a specific boundary type and, therefore, on measurements performed on a video that are indicative for the presence of that boundary type between frames k and $k+l$. In standard literature on statistical detection theory, the probability $P_k(S)$ is mostly considered equal to *a priori* probability $P_k^a(S)$. In view of this, the conditional probability $P_k(S|\psi(k))$ can be understood as a modifier for *a priori* probability. This modification becomes apparent in situations where both *a priori* probability and the likelihood functions are in favor of the hypothesis S, whereby $\overline{S}$ is the proper hypothesis (e.g., when a large discontinuity value appears long after the last detected boundary). In this way, boundaries detected falsely due to extreme factors can be prevented using the additional information that is embedded in $\psi(k)$ and that can normally not be considered in another way during the detection process: *a priori* probability takes into account only the shot length information while probability density functions consider only the range in which the observed discontinuity value $z(k,\, k+l)$ falls.

In order to maximize the quality of our detector, the development of the detector should be based on minimization of the average error probability P_E. A simple analysis of relations between likelihood and probability functions in (12) for which P_E is minimized leads to the following decision rule at the value $z(k,\, k+l)$:

$$\left.\frac{p(z|S)}{p\left(z|\overline{S}\right)}\right|_{z=z(k,\,k+l)} \overset{\overline{S}}{\underset{S}{\lessgtr}} \frac{1-P_k(S)}{P_k(S)} = \frac{1-P_k^a(S)P_k(S|\psi(k))}{P_k^a(S)P_k(S|\psi(k))}. \tag{15}$$

In view of the above development steps, it can be said for the detector (15) that it is statistically optimal given the knowledge embedded in the function $P_k(S)$ and given the probabilistic relation between discontinuity values $z(k,\, k+l)$ and the two hypotheses, embedded in likelihood functions $p(z|S)$ and $p(z|\overline{S})$. The last expression can be transformed into

$$z(k,\, k+l) \overset{\overline{S}}{\underset{S}{\lessgtr}} T(k) \tag{16}$$

which corresponds to the detector in Fig. 3.

Since the conditional probability $P_k(S|\psi(k))$ is computed differently for each boundary type, we need first to develop separate detectors (15), each being in charge for one specific boundary type. So we proceed in this section with completing the development of the detector (15) for two most widely used boundary types, hard cuts, and dissolves. Later, however, we show in Section IV-F how detectors developed for various boundary types can be combined for the purpose of detecting all boundaries simultaneously.

The specificity of the detectors regarding different boundary types can be seen only in the way the additional information relevant for the detection is embedded in the function $\psi(k)$. We show in Section IV-E how this function is computed for hard cuts and dissolves. While we consider the basic analytical model for the conditional probability $P_k(S|\psi(k))$ the same for all boundary types, certain parameters of the model change per boundary type. In Section IV-E, we also introduce the model for $P_k(S|\psi(k))$ and give optimal parameter settings for hard cuts and dissolves. All other components of the detector (15), that is, the likelihood functions $p(z|S)$ and $p(z|\overline{S})$ (Section IV-C), and *a priori* probability $P_k^a(S)$ for shot boundary (Section IV-D), we consider independent of boundary type. While this is understandable for *a priori* probability, we explain in Section IV-C why we consider likelihood functions constant as well. Since modeling of likelihood functions is based on the way the discontinuity values $z(k,\, k+l)$ are computed along a sequence, we first select in the following section suitable features and metrics for computing the discontinuity values.

B. Discontinuity Values

As discussed in Section II, we aim at measuring the discontinuity values in a way that the discrimination between ranges $\overline{R}$ and R, and the sequence independence of these ranges are maximized. For this purpose we compute the discontinuity values by compensating the motion between video frames using a block-matching procedure similar to the one proposed in [23].

We divide frame k into N_{Blocks} nonoverlapping blocks $b_i(k)$ and search for their corresponding blocks $b_{i,\,m}(k+l)$ in frame $k+l$. The block-matching criterion used here is the sum of absolute differences of block-wise average values of all three color components Y_{ave}, U_{ave} and V_{ave} of blocks $b_i(k)$ and $b_{i,\,m}(k+l)$, that is

$$\begin{aligned} D(b_l(k),\, b_{l,\,m}(k+l)) = {} & |Y_{\text{ave}}(b_l(k)) - Y_{\text{ave}}(b_{l,\,m}(k+l))| \\ & + |U_{\text{ave}}(b_l(k)) - U_{\text{ave}}(b_{l,\,m}(k+l))| \\ & + |V_{\text{ave}}(b_l(k)) - V_{\text{ave}}(b_{l,\,m}(k+l))|. \end{aligned} \tag{17}$$

After the corresponding blocks $b_{i,\,m}(k+l)$ have been found using the formula (2), we obtain the discontinuity value $z(k,\, k+l)$ by summarizing the differences (17) between blocks $b_i(k)$ and $b_{i,\,m}(k+l)$ over all blocks; that is

$$z(k,\, k+l) = \frac{1}{N_{\text{Blocks}}} \sum_{i=1}^{N_{\text{Blocks}}} D(b_i(k),\, b_{i,\,m}(k+l)). \tag{18}$$

In view of the fact that most of the video resources are available in the MPEG compressed format, we assume that the input into a video analysis system will most likely be a partially decoded sequence, also called a DC sequence [29]. We consider in this paper the sequences compressed using MPEG-1 standard. Due to small frame dimensions in the resulting DC sequence, we selected the dimensions of the blocks used in the block-matching procedure as 4 × 4 pixels. Maximum block displacement (the size of the search area) is set to 4 pixels, i.e., 1 block length/width.

To detect both hard cuts and gradual transitions, two parallel computations of discontinuity values are needed. A hard cut is found between two consecutive frames of a sequence. To detect such boundary, it is handy to work with discontinuity values that are computed using inter-frame distance $l=1$. Opposed to this, a gradual transition is stretched along several frames so that a distinction between two shots separated by a gradual transition is possible only by comparing the frames from the beginning and from the end of a boundary. For this reason, it is also necessary to compute the second discontinuity value curve by comparing distant frames using inter-frame distance $l>1$. The skip l should ideally be large enough to capture any gradual transition appearing in a video. However, if l is larger than the minimum shot length, the skip between frames being compared will

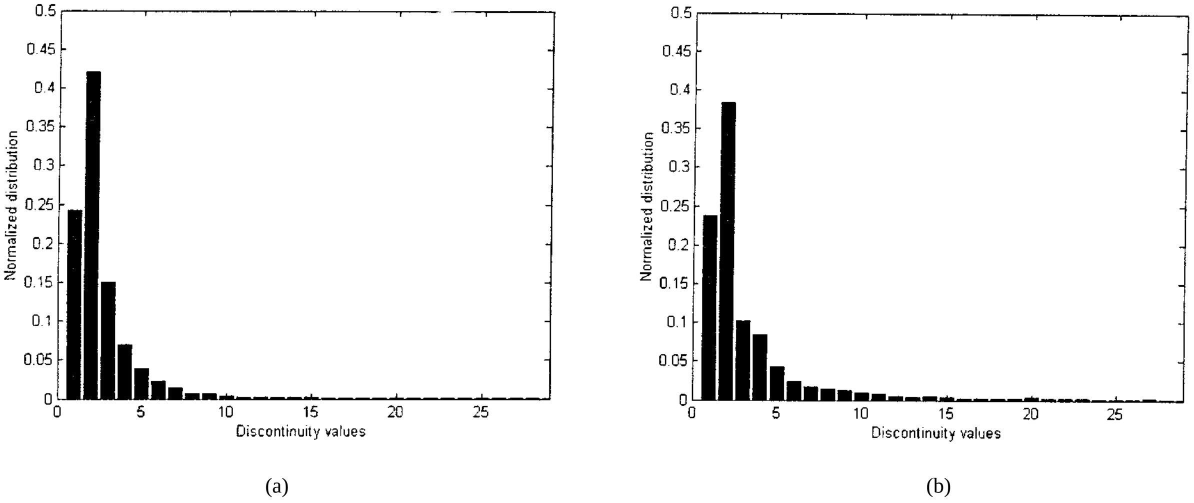


Fig. 6. Normalized distribution of intra-shot discontinuity values obtained for: (a) $l = 1$ and (b) $l = 22$.

in many cases stretch over three or more shots, which could result in missed shot boundaries [29]. Therefore, we set the value of l close to the minimum shot length. Although this minimum is not strictly defined, it can realistically be assumed that no shot lasts for less than a second. Then, the orientation value for the minimum shot length can simply be taken as a number of frames per second. In our approach we selected l as equal to 22 frames.

Larger skip l between frames when computing the second discontinuity curve requires a larger search area in the block-matching procedure. In this case we allow a maximum displacement of three blocks in each direction.

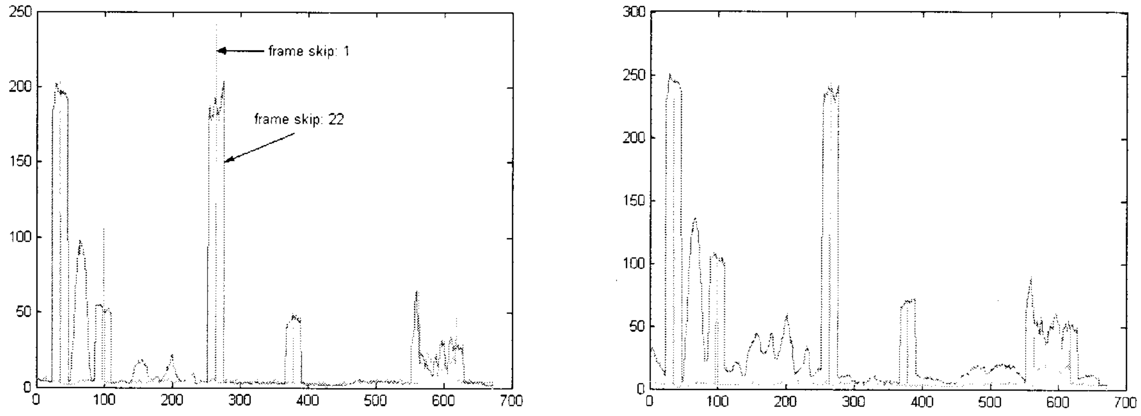
C. Likelihood Functions

We now perform a parametric estimation of likelihood functions $p(z|S)$ and $p(z|\overline{S})$ which will be used in the detection rule (15). In order to get an idea about the most suitable analytical functions used for such estimation, the normalized distributions of discontinuity values $z(k, k+l)$ computed within shots and at shot boundaries (inter- and intra-shot values) are obtained first, using several representative training sequences. We used four training sequences in the total length of about 10 000 frames. These sequences were excerpts from the movie “Jurassic Park,” the “Seinfeld” television comedy, and two documentaries. These excerpts were selected such that they do not contain any strong motion or strong lighting changes. This is because we wanted to obtain reliable likelihood functions for both hypotheses. This reliability would be reduced if the discontinuity values are included that are out of their proper range due to the effects of some extreme factors.

In general, when (18) is used to measure inter- and intra-shot values for two different values of inter-frame skip l, four different distributions can be expected—two for each hypothesis. We, however, choose to use only one distribution per hypothesis. Namely, since (18) measures correspondences of pixel blocks in terms of average colors, well matched within a search area adjusted according to the skip l, the skip between frames has very little influence on the distribution of discontinuity values for the intra-shot case. Therefore, the two intra-shot distributions can be considered similar, as also indicated by experimental results shown in Fig. 6(a) and (b). The first diagram [Fig. 6(a)] is obtained using the aforementioned training material and by computing the values (18) between consecutive frames. For the second diagram, the values (18) were computed for frame pairs with $l = 22$ frames in-between. For the purpose of obtaining the parametric model for the likelihood function for the “no boundary” hypothesis we combine the two distributions from Fig. 6(a) and (b) in order to base our model on as much test data as possible. Resulting normalized distribution of intra-shot discontinuity values is shown in Fig. 8(a).

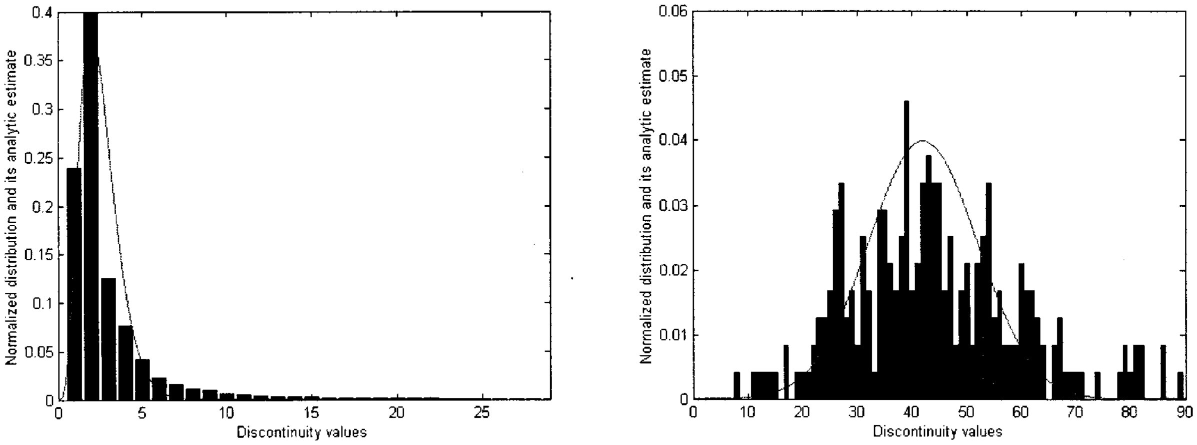
We can also assume a high similarity of the two measured inter-shot distributions if the size of the search area remains constant for both values of the inter-frame skip l. This can be observed from the diagrams in Fig. 7(a) and (b). The first diagram [Fig. 7(a)] is obtained by using a variable search area, as selected in Section IV-B (1 block for $l = 1$, 3 blocks for $l = 22$). It can be seen that while intra-shot values in both curves remain in the same range, the inter-shot values are smaller in the curve obtained for the skip $l = 22$: the larger the search area, the higher the probability for each block of finding a “good” match in a frame of another shot. For the diagram in Fig. 7(b) we used the constant (smaller) search area for computing discontinuity values with different skip factors. Compared to the case in Fig. 7(a), the discontinuity values measured at shot boundaries (both hard cuts and gradual transitions) now remain in the same range in both curves, while there is an increase in intra-shot discontinuity values in the curve with the larger skip value. The latter is understandable since the search area is too small to find appropriate block matches.

Thus, if we adjust the search area to the skip l in order to obtain constant ranges for intra-shot discontinuity values [Fig. 7(a)], there is a possibility that the ranges of intra-shot discontinuity values are different. This means that, in principle, we should use two different likelihood functions for the hypothesis S in the detectors (15) for hard cuts and gradual transitions. This would not be a problem if the distribution of inter-shot values obtained for the skip $l > 1$ can be obtained



(a) (b)

Fig. 7. Inter-shot and intra-shot discontinuity values computed using: (a) different search areas and (b) one and the same search area.



(a) (b)

Fig. 8. (a) Normalized distribution of values $z(k,\, k+l)$ computed within shots for $l = 1$ (discrete bins) and its analytic estimate (continuous curve). (b) Normalized distribution of values $z(k,\, k+l)$ computed at shot boundaries for $l = 1$ (discrete bins) and its analytic estimate (continuous curve).

in a reliable way. Finding good block matches in a sufficiently large search area is, namely, in many cases so successful that inter-shot discontinuity values sometimes fall into the range of intra-shot values. For this reason, no clear distinction between ranges $\overline{R}$ and R exists, which negatively influences the detection performance regarding gradual transitions. We, therefore, work only with inter-shot distribution of peaks corresponding to hard cuts, measured for the skip $l = 1$. As can be seen from Figs. 7(a) and 8(b), the normalized inter-shot distribution obtained for the skip $l = 1$ can be considered wide enough to include also the inter-shot values measured for the skip $l > 1$.

The shape of the distribution in Fig. 8(a) indicates that a good analytic estimate for this distribution and so for the likelihood function $p(z|\overline{S})$ can be found in the family of functions given as

$$p\left(z|\overline{S}\right) = h_1 z^{h_2} e^{-h_3 z}. \tag{19}$$

Using the similar principle of global shape matching, the distribution in Fig. 8(b) and so the likelihood function $p(z|S)$ can best be modeled using a Gaussian function

$$p(z|S) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-(1/2)((z-\mu)/\sigma)^2}. \tag{20}$$

The most suitable parameter combinations $(h_1,\, h_2,\, h_3)$ and $(\mu,\, \sigma)$ are then found experimentally, such that the rate of detection mistakes for the training sequences is minimized. In other words, we applied the models (19) and (20) in the detection rule (15) for various parameter combinations and selected those combinations for which the detection performance on our training sequences was optimal. The optimal parameter triplet found for the model (19) is (1.33, 4, 2) and the optimal parameter pair for the model (20) was found as (42, 10). The resulting analytic functions serving as parametric estimate of the likelihood function $p(z|\overline{S})$ and $p(z|S)$ are also shown in Fig. 8(a) and (b), respectively.

D. A Priori Probability Function

In this section we derive an analytical expression for *a priori* probability $P_k^a(S)$ for the presence of a shot boundary between frames k and $k + l$. As explained before, we make this probability dependent on the number of frames elapsed since the last detected shot boundary. In order to fulfill its purpose, the probability function $P_k^a(S)$ has to satisfy the following two criteria.

1) $P_k^a(S)$ must be a monotonously increasing function of time, that is, of frame index k.

2) While starting at the value 0 immediately after the last detected shot boundary, that is, $P_k^a(S) = 0$ for $k = 0$, the function $P_k^a(S)$ will converge toward the value 0.5 for $k \rightarrow \infty$. The value 0.5 indicates that for a sufficiently large number of elapsed frames the function $P_k^a(S)$ should have no influence on the detection performance any more.

The first item indicates that $P_k^a(S)$ can best be modeled as cumulative probability. In order to do so, suitable probability distribution needs to be found first. Studies reported in [22] and [7], involving statistical measurements of shot lengths for a large number of motion pictures, have shown that the distribution of shot lengths for all the films considered matches the Poisson function well [21]. Therefore, we adopt the Poisson function as the base for computing the cumulative probability $P_k^a(S)$. Since the second item requires that the function $P_k^a(S)$ converges toward the value 0.5, cumulative probability values need to be halved in order to ensure proper convergence, that is

$$P_k^a(S) = \frac{1}{2} \cdot \sum_{w=0}^{\lambda(k)} \frac{\mu^w}{w!} e^{-\mu}. \tag{21}$$

The parameter μ of the Poisson distribution in (21) represents the average shot length of a video sequence, w is the frame counter, which is reset each time a shot boundary is detected, and $\lambda(k)$ is the current shot length at the frame k.

The Poisson function was obtained in [7] and [22] as most suitable for motion pictures. However, we assume that the conclusions on shot-length distributions made by Salt and Coll can be extended further to all other types of video programs. Although it is expected that the parameter μ should be adapted to different program types (movies, documentaries, music video clips, etc.) and possibly also to sub-types (e.g., an action movie vs. drama) in order to compensate for possible variations in program characteristics, our experiments have shown that the μ value can be set constant for a broad scope of different program types. For instance, when detecting shot boundaries in our training sequences, we let the value of μ change in the entire range between 50 and 150. However, there was no considerable change in detection results. Wide validity and flexibility of the parameter μ becomes understandable if one recalls that the influence of the function $P_k^a(S)$ in the detector (15) is high only in the beginning of a shot (e.g., for relatively small values of the frame index k) and has the task to prevent the appearance of unrealistically short shots. An adjustment of this value is required only in some extreme cases, such as music TV clips or commercials, where shots are generally much shorter than in “usual” programs and where, therefore, a higher rate of shot changes needs to be taken into account *a priori*.

In general, a suitable range for μ per program type can be obtained by analyzing the distributions of shot lengths of various genres in large video collections. Once this range information is available, the adjustment of the μ value can be performed fully automatically if the program type (genre) information is available in the shot-boundary detection system. An example of such a system is the one operating directly on DVB [31] streams. Here, each transmitted program compliant to DVB standard also contains a header (so-called DVB Service Information), which also contains the information on program type (movie, documentary, music TV clip, etc.). Then, μ can be set easily by means of a simple look-up table.

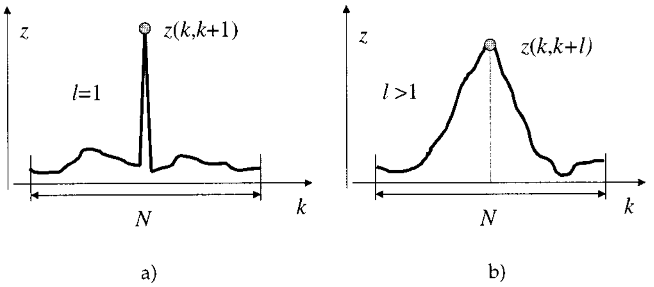


Fig. 9. Discontinuity values in a sliding window of the length N with expected behavior: (a) at a hard cut and (b) within a dissolve.

E. Conditional Probability for Boundary Presence

Function $\psi(k)$ for Hard Cuts: As discussed in Section II, the information on the pattern created by several consecutive discontinuity values surrounding the value $z(k,\, k+l)$ can be highly valuable for detecting the presence of a shot boundary between frames k and $k+l$. This information is especially important for detecting boundary types characterized by very distinct patterns. One such boundary type is the abrupt boundary or hard cut. Yeo and Liu showed in [29] that the presence of an isolated sharp peak surrounded by low discontinuity values—as illustrated in Fig. 9(a)—is a reliable indication for the presence of a hard cut at the position of the peak. According to the procedure presented in [29], the presence of a sharp peak between frames k and $k+1$ can be detected by investigating the series of discontinuity values computed for the inter-frame skip $l = 1$ by finding the ratio between $z(k,\, k+1)$ and the second largest discontinuity value z_{sm} in the close surrounding of $z(k,\, k+1)$ and comparing the obtained result with a given threshold. If the threshold is exceeded, a hard cut is found. However, we saw from the discussion in Section III that the above process is far too threshold dependent.

Instead of coarsely comparing the ratio between $z(k,\, k+1)$ and z_{sm} with a fixed threshold, we choose to interpret this ratio as a measure for matching between the measured and the template pattern. The larger the ratio, the better is the match; that is, the larger is the probability of a hard cut presence. In view of this, the ratio between $z(k,\, k+1)$ and z_{sm} could serve as function $\psi(k)$ which is the argument of the conditional probability $P_k(S|\psi(k))$ in the detector (15) for hard cuts.

We, however, found that computing the relative distance between $z(k,\, k{+}1)$ and z_{sm} performs much better in practice than the simple ratio between $z(k,\, k{+}1)$ and z_{sm}. The reason for this is that the ratio is dependent on the range of values $z(k,\, k+1)$ and z_{sm} while the relative distance is not. Finally, we formulate the function $\psi(k)$ for the detector (15) of hard cuts as in (22), shown at the bottom of the next page. The value $\psi(k)$ indicates the degree of pattern matching in the middle of the sliding window of the length N and centered at the discontinuity value $z(k,\, k+1)$. Since the necessary condition for the presence of the hard cut between frames k and $k+1$ is that a sharp, isolated peak is found at $z(k,\, k+1)$, no boundary can be found there if

$z(k,\, k+1)$ is not the maximum of the window. The length N of the window should be as large as possible in order to obtain reliable pattern matching results. However, based on the same argumentation as the one for selecting the inter-frame skip l, a maximum value of N is rather limited. We therefore choose to link the value of N to the value of l. However, since an uneven value for N is more practical due to symmetry of the sliding window with respect to its center $z(k,\, k+1)$, we choose N equal to 21.

Function $\psi(k)$ for Dissolves: Defining the $\psi(k)$ function for the detector of gradual transitions is considerably more difficult. Considering the quality of pattern matching only, as for the case of hard cuts, is here not likely to lead to a good detection performance. This is mainly due to the fact that the pattern of consecutive discontinuity values at places of gradual transitions may vary in both shape and length. Also, it is dependent on the features and metrics used to compute the discontinuity values as well as on the type of gradual transition. For instance, if motion-compensating features and metrics are selected as described in Section IV-B, and if we concentrate on detecting the dissolves only, the pattern observed in the series of consecutive discontinuity values computed with inter-frame skip $l > 1$ is expected to have a close-to-triangular shape as shown in Fig. 9(b). The problem is, however, that, first, the triangle width will change from one dissolve to another due to varying dissolve length. Second, the sides of the triangle will not be that pronounced due to noise and extreme factors. Therefore, the boundary pattern of a gradual transition is not as unique as the sharp, isolated peak of a hard cut. Consequently, patterns created by consecutive discontinuity values that are due to extreme factors may be falsely classified as those belonging to gradual transitions. For this reason, the function $\psi(k)$ for the detector of gradual transitions needs to be based not only on pattern matching but also on other information that can help improve the detection robustness. This information is strongly related to the specific boundary type considered in the detector: the more exclusive the information for the boundary type considered, the better performance of the detector (15) for that boundary type can be expected. This is simply because this exclusive information can best distinguish the boundary from any other effect in a video. We demonstrate here how the function $\psi(k)$ can be computed for a dissolve detector.

We consider the triangular pattern of a dissolve, that is created by consecutive discontinuity values belonging to the sliding window of the length N and centered in $z(k,\, k+l)$ with $l = 22$. Similarly as in the case of hard cuts, we select the value of N equal to 21. Since the triangular pattern changes from one dissolve to another, there is little sense in trying to model the pattern precisely and measure the degree of pattern matching based on that model. We, therefore, formulate the criterion for pattern matching based only on two basic characteristics of this pattern:

1) the middle window value $z(k,\, k+l)$ is the maximum value of the window;
2) window's maxima on each side of $z(k,\, k+l)$ have to be as close to the middle of the window as possible.

As can be concluded from these characteristics, we require that pattern created by discontinuity values matches the "ideal" pattern of a dissolve only regarding the "top" of the triangular shape in Fig. 9(b) and do not consider the shape of triangle sides. We now introduce the function $\psi_1(k)$ as

$$\psi_1(k) = \begin{cases} 1, & \text{if } z(k,\, k+l) = \max\Bigg(\forall\, z(i,\, i+l) \\ & \quad i \in \left(k - \dfrac{N-1}{2},\, k + \dfrac{N-1}{2} - 1\right)\Bigg) \\ & \wedge\, \Delta_{\max}^{l} + \Delta_{\max}^{r} < \dfrac{N}{2} \\ 0, & \text{else.} \end{cases} \tag{23}$$

Here, $\Delta_{\max}^{l}$ and $\Delta_{\max}^{l}$ are the distances of maximum discontinuity values to the left and to the right of $z(k,\, k+l)$ from the window middle point. As can be recognized from the condition in (23), the value of $\psi_1(k)$ is set equal to 1 at the frame k if the pattern created by discontinuity values of the sliding window centered at $z(k,\, k+l)$ fulfills the two matching criteria listed above. Otherwise, $\psi_1(k)$ is set to 0. So, each series of consecutive discontinuity values for which $\psi_1(k)$ is set equal to 1 is further considered as a dissolve candidate.

We now investigate each dissolve candidate more thoroughly by analyzing the behavior of intensity variance along the corresponding series of frames. This analysis concentrates on matching the variance behavior with the downwards-parabolic pattern that is characteristic for a dissolve. However, since this parabolic pattern is never perfect, we choose to match only some of its characteristic global properties. Namely, if the sliding window captures a dissolve, the variance measured for frames in the middle of the window will be considerably lower than the variance of frames positioned near window's edges. Opposite to this, if no dissolve occurs in a sliding window, variance is expected to remain close to one stable value; that is, its rate of change in the window will be very small. Therefore, we compute the function $\psi_2(k)$ as the relative change of frame variance in the middle of the sliding window, i.e., at the frame $k + (l/2)$ compared to frames close to window edges. Analytically, this can be written as in (24), shown at the bottom of the next page. Here, $\sigma(k')$ is the variance of the frame in the middle of the window with $k' = k + (l/2)$, while $\sigma(k' - (N/2))$ and $\sigma(k' + (N/2))$ are the variances of frames at both window edges. The third option in (24), where $\psi_2(k)$ is by definition equal to 0, corresponds to the case where variance in the middle of the window is larger than those at window edges. Since this cannot occur in a downwards parabolic pattern, such

$$\psi(k) = \begin{cases} 100 \cdot \dfrac{z(k,\, k{+}1) - z_{sm}}{z(k,\, k{+}1)}\ \%, & \text{if } z(k,\, k{+}1) = \max\left(\forall\, z(i,\, i{+}1),\ i \in \left(k - \dfrac{N-1}{2},\, k + \dfrac{N-1}{2} - 1\right)\right) \\ 0, & \text{else} \end{cases} \tag{22}$$

a relation among three variances cannot reveal a dissolve. The obtained value for $\psi_2(k)$ is, therefore, a reliable indication for the presence of a dissolve in a candidate series of discontinuity values selected using function $\psi_1(k)$. Multiplying $\psi_1(k)$ and $\psi_2(k)$ provides therefore the function $\psi(k)$ which can be used in the detector (15) for dissolves; that is

$$\psi(k) = \psi_1(k) \cdot \psi_2(k). \tag{25}$$

Modeling the Conditional Probability Function: The value of the function $\psi(k)$ serves as the argument of the conditional probability function $P_k(S|\psi(k))$ the main task of which is to project $\psi(k)$ onto a corresponding probability that the sliding window captures a shot boundary. Since $P_k(S|\psi(k))$ is a non-decreasing function of $\psi(k)$, and the range of $\psi(k)$ is [0, 100], the function $P_k(S|\psi(k))$ has to fulfill the following basic criteria:

1) $\psi(k) = 0 \rightarrow P_k(S|\psi(k)) = 0$;
2) $\psi(k) = 100 \rightarrow P_k(S|\psi(k)) = 1$.

Besides, our analysis has shown that $P_k(S|\psi(k))$ should not be too sensitive to the values of $\psi(k)$ being close to borders of the interval [0, 100]. For these values of $\psi(k)$ the value of $P_k(S|\psi(k))$ should be very close to 0 or 1, respectively. The actual transition from 0 to 1 should take place in the middle range of the interval [0, 100], that is, for values of $\psi(k)$ for which the boundary characteristics become sufficiently recognizable. This transition should, however, not be abrupt but flexible enough in order not to reject any reasonable boundary candidate. After experimenting with different functions that fulfill the aforementioned criteria, we find the optimal shape of the function $P_k(S|\psi(k))$ as illustrated in Fig. 10.

The function from Fig. 10 can be formulated analytically as follows:

$$P_k(S|\psi(k)) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{\psi(k) - d}{\sigma_{\text{erf}}}\right)\right) \tag{26}$$

with

$$\operatorname{erf}(x) = \frac{2}{\pi}\int_0^x e^{-t^2}\, dt. \tag{27}$$

The parameters d and σ_{erf} are the "delay" from the origin and the spreading factor determining the steepness of the middle curve segment, respectively. In a way similar to the parameter sets in Section IV-C, the optimal parameter combination (d, σ_{erf}) is found experimentally such that the detection performance for the training sequences is optimized. Since $\psi(k)$ is computed differently for each boundary type, the parameter combination (d, σ_{erf}) also needs to be determined for each boundary type separately. For hard cuts and dissolves we found the optimal pairs of parameters as (60, 2) and (21, 2), respectively.

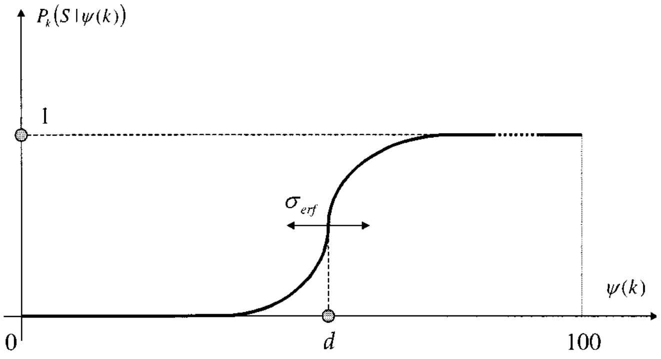


Fig. 10. Conditional probability function $P_k(S|\psi(k))$.

We like to emphasize that the basic shape of the conditional probability $P_k(S|\psi(k))$, as defined in (26), can be considered the same for all boundary types. What remains boundary dependent are the function $\psi(k)$ and the parameter set (d, σ_{erf}).

F. Detecting Different Boundary Types Simultaneously

In the first part of Section IV, we introduced general principles for developing a statistically optimal shot-boundary detector and explained that, due to specific characteristics of different boundary types, this detector needs to be developed for each boundary type separately. However, by linking all separate detectors in a cascade, as shown in Fig. 11, it is also possible to detect all different boundaries simultaneously.

Each block in the cascade corresponds to one detector (15). The first block of the cascade is responsible for detecting hard cuts. As such, it takes as input the discontinuity values measured between consecutive video frames; i.e., for $l = 1$. All other blocks detect various types of gradual transitions and therefore take as input the discontinuity values that are obtained with inter-frame skip $l > 1$.

$$\psi_2(k) = \begin{cases} 100\cdot\left(1 - \dfrac{\min\left(\sigma(k'),\, \sigma\left(k' + \frac{N}{2}\right)\right)}{\max\left(\sigma(k'),\, \sigma\left(k' + \frac{N}{2}\right)\right)}\right)\%, & \text{if } \left|\left(\sigma(k') - \sigma\left(k' + \frac{N}{2}\right)\right)\right| \le \left|\left(\sigma(k') - \sigma\left(k' - \frac{N}{2}\right)\right)\right| \\ 100\cdot\left(1 - \dfrac{\min\left(\sigma(k'),\, \sigma\left(k' - \frac{N}{2}\right)\right)}{\max\left(\sigma(k'),\, \sigma\left(k' - \frac{N}{2}\right)\right)}\right)\%, & \text{if } \left|\left(\sigma(k') - \sigma\left(k' - \frac{N}{2}\right)\right)\right| < \left|\left(\sigma(k') - \sigma\left(k' + \frac{N}{2}\right)\right)\right| \\ 0, & \text{if } \sigma(k') > \sigma\left(k' - \frac{N}{2}\right) \wedge \sigma(k') > \sigma\left(k' + \frac{N}{2}\right). \end{cases} \tag{24}$$

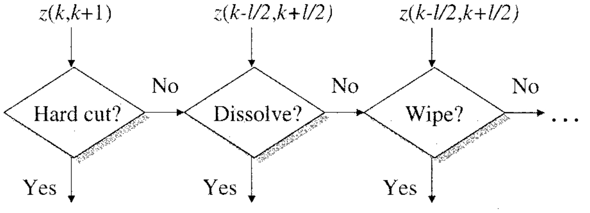


Fig. 11. Detector cascade for simultaneously detecting various shot boundaries.

Linking the detectors as described above can also be beneficiary for improving the total detection performance of the cascade. We explain this on the example of two series of discontinuity values measured for $l = 1$ and $l = 22$ and aligned in time as shown in Fig. 7. Due to "plateaus" circumventing each of the hard-cut peaks, a falsely detected gradual transition can be reported at a certain plateau point, before or after a hard cut is detected. This is because some very high plateau values appearing before or after the actual boundary peak can "confuse" one or more detectors of gradual transitions. For this reason, all detected gradual transitions detected within the interval $(k - l/2,\, k + l/2)$ of a hard cut can be assumed a consequence of a plateau and can therefore be eliminated *a posteriori*. The probability to eliminate a valid gradual transition hereby is almost neglectable since, on the one hand, the detection of hard cuts using detector (15) is very reliable and, on the other hand, the presence of a hard cut and a gradual transition on a distance of only a fraction of a second is highly improbable. Consequently, proper detection of hard cuts can reduce the number of falsely detected gradual transitions.

V. Performance Evaluation

In this section, we evaluate our detector in view of the criteria posed in the introduction to this paper. These criteria were originally formulated as:

1) excellent detection performance for all types of shot boundaries;
2) constant quality of the detection performance for any arbitrary sequence, with minimized need for fine tuning of detector parameters per sequence.

Since we concentrate here on two types of shot boundaries—hard cuts and dissolves—we tested the performance of the cascade in Fig. 11 consisting of two first blocks only. In our tests we used five test sequences that belong to four different program categories: movies, soccer game, news, and commercial documentary. It is important to note that these sequences were carefully selected as those containing many effects that are known to often cause detection errors. For instance, the commercial documentary used in our tests was made to promote a high-tech company. There, several editing tricks are applied, many of which can "confuse" a shot boundary detector. Also, many parts of a sequence show fireworks, launching of Space Shuttle, explosions, strong zoom-ins, zoom-outs, and camera panning. Many editing effects other than gradual transitions were also present in the Dutch news sequence that was included in our test set. Our soccer sequences are live broadcasts with many segments characterized by highly complex motion activity, strong zooms, and fast object motion. Similar effects were also present in the part of the movie "Saving Private Ryan" which was included in our test set. None of these sequences were previously employed for training the detection procedure, that is, for obtaining the likelihood functions and detector parameters, as explained in Section IV.

TABLE I
Detection Results for Hard Cuts (A) and Dissolves (G)

Test material	Total		Detected		False	
	A	G	A	G	A	G
Ryan	17	0	17	0	0	1
Soccer1	10	14	10	12	0	0
Ajax	6	1	6	1	0	0
News	26	7	26	5	0	2
Documentary	45	1	45	1	0	2
Total	104	23	104	19	0	5

The results presented in Table I illustrate a high precision and recall obtained using our proposed detector. While being 100% for hard cuts, precision and recall for dissolve detection were obtained as 79% and 83%, respectively. Formulated differently, for dissolve detection we reached a detection rate of 83% and a false alarm rate of 22%. In order to provide an idea about the quality of these results, we refer to the survey on methods for shot-boundary detection published by Lienhart in [15]. Best results in dissolve detection were reached using the method of [25], [26] that is based on measuring intensity variance in the dissolve region, and the method of Lienhart proposed in [14] that uses a large dissolve training set created by a dissolve synthesizer. Truong *et al.* report the precision of 75.1% and recall of 82.2%. Lienhart reaches a precision of 82.4% and recall of 75%. Although our test set was not as large as in the two aforementioned approaches, the results in Table I can be considered fully reliable due to a careful selection of test sequences.

Although the rates of proper and false detections are not precisely the same for all sequences, there are no extreme outliers in the performance. In this sense, we feel free to claim that the performance of our detector remains relatively consistent over all sequences. This claim is also supported by the fact that all sequences were tested using the same parameter settings, that is, those introduced in the process of detector development in Section IV. In addition to this, one and the same value of the parameter μ (equal to 70) was used for all sequences.

VI. Discussion

Most existing approaches for shot-boundary detection are based on explicitly given thresholds or relevant threshold parameters, which directly determine the detection performance. Due to such a direct mutual dependence, the detection performance is highly sensitive to specified parameter values. For instance, a threshold set to 2.3 will interpret a discontinuity value 2.31 as a shot boundary, and a value 2.29 as a regular value within a shot. Beside the sensitivity, the problem of specifying such a precise threshold remains. And, consequently, the scope of the validity of such a precise threshold is highly questionable.

Manual parameter specification clearly cannot be avoided in any of the detection approaches. However, the influence of these parameters on the detection performance can be diminished and the detection can be made more robust if the parameters are used at lower levels of the detector system hierarchy, so only for the purpose of globally defining the detector components. Each component then provides the detector with nothing more than an indication of the presence of a boundary based on a specific criterion. The decision making about the presence of a shot boundary is then left solely to the parameter-free detector, where all the indications coming from different sources are evaluated and combined. In this way, the importance of a single manually specified parameter is not as great as when that parameter is directly a threshold. This parameter can therefore be assumed valid in a considerably broader scope of sequences. In the statistical detector presented in this paper, this is the case with parameter sets (h_1, h_2, h_3) and (μ, σ), which are used to define the likelihood functions (19) and (20), as well as with parameters d and σ_{erf} used to formulate the conditional probability function (26). The only parameter requiring an adjustment is μ, which is used in (21) to define *a priori* probability for shot-boundary presence. The adjustment of the parameter μ is, however, easy and can be performed automatically, if the ranges for μ are available for different program types (e.g., by performing statistical measurements) and if the program type is known at the input into the video analysis system, e.g., when working with DVB signals.

Since the parameters used in our detector can either be assumed generally valid or be adjusted automatically, no human supervision is required during the detection procedure. At the same time, since the parameters are optimized for a general case, similar high detection performance can be expected for any input sequence. Both of these aspects make the developed detector suitable for an implementation in a fully automated sequence analysis system. The facts that the detection method presented in this paper can operate on a wide range of video sequences without human supervision, and keep the constant high detection quality for each of them, are the major advantages the proposed detector has over the methods from recent literature.

Acknowledgment

The author acknowledges the involvement of Prof. I. Lagendijk of Delft University of Technology, The Netherlands, and Dr. H. Zhang of Microsoft Research China in the initial phase of the work reported here.

References

[1] G. Ahanger and T. D. C. Little, "A survey of technologies for parsing and indexing digital video," *J. Vis. Commun. and Image Repres.*, vol. 7, no. 1, pp. 28–43, Mar. 1996.

[2] A. M. Alattar, "Detecting and compressing dissolve regions in video sequences with a DVI multimedia image compression algorithm," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, vol. 1, May 1993, pp. 13–16.

[3] A. Akutsu, Y. Tonomura, H. Hashimoto, and Y. Ohba, "Video indexing using motion vectors," in *Proc. VCIP'92*, Boston, MA, 1992, pp. 1522–1530.

[4] F. Arman, A. Hsu, and M. Chiu, "Feature management for large video databases," in *Proc. IS&T/SPIE Storage and Retrieval for Image and Video Databases*, Feb. 1993, pp. 2–12.

[5] D. Bordwell and K. Thompson, *Film Art: An Introduction*. New York: McGraw-Hill, 1993.

[6] J. S. Boreczky and L. Rowe, "Comparison of video shot boundary detection techniques," in *Proc. IS&T/SPIE Storage and Retrieval for Still Image and Video Databases IV*, vol. 2670, Feb. 1996, pp. 170–179.

[7] D. C. Coll and G. K. Choma, "Image activity characteristics in broadcast television," *IEEE Trans. Commun.*, vol. COM-24, pp. 1201–1206, Oct. 1976.

[8] T. F. Curran and M. Ross, "Optimum detection thresholds in optical communications," *Proc. IEEE*, pp. 1770–1771, Nov. 1965.

[9] B. Furht, S. W. Smoliar, and H. Zhang, *Video and Image Processing in Multimedia Systems*. Norwell, MA: Kluwer, 1995.

[10] A. Hampapur, R. Jain, and T. Weymouth, "Digital video segmentation," in *Proc. ACM Multimedia'94*, 1994, pp. 357–364.

[11] A. Hanjalic, M. Ceccarelli, R. L. Lagendijk, and J. Biemond, "Automation of systems enabling search on stored video data," in *Proc. IS&T/SPIE Storage and Retrieval for Image and Video Databases V*, vol. 3022, Feb. 1997, pp. 427–438.

[12] T. Kikukawa and S. Kawafuchi, "Development of an automatic summary editing system for the audio visual resources," *Trans. Inst. Electron., Inform., Commun. Eng.*, vol. J75-A, no. 2, pp. 204–212, 1992.

[13] R. Lienhart, "Comparison of automatic shot boundary detection algorithms," in *Proc. IS&T/SPIE Storage and Retrieval for Image and Video Databases VII*, vol. 3656, Jan. 1999, pp. 290–301.

[14] ——, "Reliable dissolve detection," in *Proc. IS&T/SPIE Storage and Retrieval for Media Databases 2001*, vol. 4315, Jan. 2001, pp. 219–230.

[15] ——, "Reliable transition detection in videos: A survey and practitioner's guide," *Int. J. Image Graph.*, vol. 1, no. 3, pp. 469–486, Aug. 2001.

[16] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting cuts and classifying scene breaks," in *Proc. ACM Multimedia '95*, San Francisco, CA, 1995, pp. 189–200.

[17] J. Meng, Y. Juan, and S. Chang, "Scene change detection in a MPEG compressed video sequence," in *Proc. IS&T/SPIE*, vol. 2419, Feb. 1995, pp. 14–25.

[18] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in *Visual Database Systems II*, E. Knuth and L. M. Wegner, Eds. Amsterdam, The Netherlands: North-Holland, 1992, pp. 113–127.

[19] K. Otsuji and Y. Tonomura, "Projection detecting filter for video cut detection," in *Proc. ACM Multimedia '93*, 1993, pp. 251–257.

[20] K. Otsuji, Y. Tonomura, and Y. Ohba, "Video browsing using brightness data," in *Proc. SPIE/IS&T VCIP'91*, vol. 1606, 1991, pp. 980–989.

[21] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, International Editions ed. New York: McGraw-Hill, 1984.

[22] B. Salt, "Statistical style analysis of motion pictures," *Film Quarterly*, vol. 28, pp. 13–22, 1973.

[23] B. Shahraray, "Scene change detection and content-based sampling of video sequences," in *Proc. IS&T/SPIE*, vol. 2419, Feb. 1995, pp. 2–13.

[24] S. Song, T. Kwon, and W. Kim, "Detection of gradual scene changes for parsing of video data," in *Proc. IS&T/SPIE*, vol. 3312, 1998, pp. 404–413.

[25] B. T. Truong, C. Dorai, and S. Venkatesh, "Improved fade and dissolve detection for reliable video segmentation," in *Proc. IEEE Int. Conf. Image Processing (ICIP 2000)*, vol. 3, 2000, pp. 961–964.

[26] ——, "New enhancements to cut, fade and dissolve detection processes in video segmentation," in *Proc. ACM Multimedia 2000*, Nov. 2000, pp. 219–227.

[27] H. Ueda, T. Miyatake, and S. Yoshizawa, "IMPACT: An interactive natural-motion picture dedicated multimedia authoring system," in *Proc. CHI'91*, 1991, pp. 343–350.

[28] N. Vasconcelos and A. Lippman, "Statistical models of video structure for content analysis and characterization," *IEEE Trans. Image Processing*, vol. 9, pp. 3–19, Jan. 2000.

[29] B.-L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 533–544, Dec. 1995.

[30] H. Zhang, A. Kankanhalli, and S. W. Smoliar, "Automatic partitioning of full-motion video," *Multimedia Syst.*, vol. 1, pp. 10–28, 1993.

[31] ETS 300 421, "Digital broadcasting systems for television, sound and data services; framing structure, channel coding and modulation for 11/12 GHz satellite services," *EBU/ETSI JTC*, Dec. 1994.

Alan Hanjalic (M'00) received the Dipl.-Ing. degree from the Friedrich-Alexander University, Erlangen, Germany, in 1995, and the Ph.D. degree from Delft University of Technology, Delft, The Netherlands, in 1999, both in electrical engineering.

He is an Assistant Professor with the Department of Mediamatics, Delft University of Technology. Previously, he was a Research Assistant in the Department of Mediamatics from 1995 to 1999, a Visiting Scientist at Hewlett-Packard Labs, Palo Alto, CA, in 1998, and a Research Fellow at British Telecom Labs, Ipswich, U.K., during December 2000–January 2001. His main research interest is in the area of visual content management—image and video content analysis for browsing and retrieval (query) applications—which included his participation in the European ACTS (AC018) project SMASH (Storage for Multimedia Applications Systems in the Home) from 1995 to 1998. He also works on semantics extraction from video and images, researches new image- and video-compression methodologies that will enable efficient performing of content-based operations on compressed images and video directly, and investigates the subjective aspects of image and video analysis and retrieval by using the techniques of affective computing. He is the author or co-author of the book *Image and Video Databases: Restoration, Watermarking and Retrieval* (Amsterdam, The Netherlands: Elsevier, 2000) and a large number of publications, some of which already belong to standard references.

Dr. Hanjalic was Guest Editor of the *International Journal of Image and Graphics* (Special Issue on Content-based Image and Video Retrieval) in July 2001. He has served as a Program Committee member, Session Chair, and panel member for several IEEE and IS&T/SPIE conferences. He is an expert in the area of information technology and systems of the Belgian Science Foundation (IWT) and is a member of IEEE Signal Processing Society.