Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2009, Article ID 509438, 12 pages
doi:10.1155/2009/509438

Research Article

Video Shot Boundary Detection Using QR-Decomposition and Gaussian Transition Detection

Ali Amiri and Mahmood Fathy

Computer Engineering Department, Iran University of Science and Technology, Narmak, 16846-13114 Tehran, Iran

Correspondence should be addressed to Ali Amiri, a_amiri@iust.ac.ir

Received 6 May 2009; Revised 6 September 2009; Accepted 26 November 2009

Recommended by Y.-P. Tan

This article explores the problem of video shot boundary detection and examines a novel shot boundary detection algorithm by using QR-decomposition and modeling of gradual transitions by Gaussian functions. Specifically, the authors attend to the challenges of detecting gradual shots and extracting appropriate spatiotemporal features that affect the ability of algorithms to efficiently detect shot boundaries. The algorithm utilizes the properties of QR-decomposition and extracts a block-wise probability function that illustrates the probability of video frames to be in shot transitions. The probability function has abrupt changes in hard cut transitions, and semi-Gaussian behavior in gradual transitions. The algorithm detects these transitions by analyzing the probability function. Finally, we will report the results of the experiments using large-scale test sets provided by the TRECVID 2006, which has assessments for hard cut and gradual shot boundary detection. These results confirm the high performance of the proposed algorithm.

1. Introduction

The latest developments in multimedia technology, combined with a considerable growth in computer performance and the expansion of the Internet, have provided people with access to a tremendous amount of video information. Video applications, currently expanding at a considerable rate, have initiated an increasing demand for innovative technologies and tools to index, browse, and retrieve video data efficiently.

Developed for automatic indexing, retrieval, and management of video, content-based video retrieval has become the subject of much research throughout the last decade [1, 2]. Structural analysis of video is a fundamental stage in analyzing video content and developing techniques for efficient access, classification, retrieval, and browsing of vast video databases. Among the several structural levels (i.e., frame, shot, scene, etc.), shot level organization has been deemed suitable for browsing and content-based retrieval [3].

A sequence of frames captured by one camera in a single continuous action in time and space is referred to as a video shot [4]. Normally, it is a group of frames that have constant visual attributes, (such as color, texture, and motion). Depending on whether the transition from one shot to another is abrupt or gradual, shot boundaries can be classified into two types: cut and gradual transitions. The cut transition is the typical abrupt change, where one frame is a part of the disappearing shot and the next one is a part of the appearing shot.

In contrast, gradual transitions can be categorized into dissolve, wipe, fade out/in, and so forth, based on the properties of various editing effects [5]. In the dissolve transition, the last few frames of the disappearing shot temporally overlap with the first few frames of the appearing shot. Amid the overlap transition, the intensity of the disappearing shot reduces from normal to zero (fade out), whereas the intensity of the appearing shot rises from zero to normal (fade in). In the fade transition, the disappearing shot fades out into a blank frame, and then the blank frame fades into the appearing shot. The wipe transition is in fact a group of techniques for changing the shot in which the appearing and disappearing shots exist at the same time in various spatial areas of the intermediary video frames, and the region taken up by the former develops until it completely takes the place of the latter. Many of the recent works have focused on abrupt transitions; gradual transitions are normally harder to identify, owing to camera and/or object motions in a shot.

Shot detection is not new to researchers in the content-based video analysis community. During the past decade, Shot Boundary Detection (SBD) has been actively studied in video retrieval, video summarization, pattern recognition, and multimedia communities [6]. Research on automatic shot detection has been increasing rapidly over the past few years. Since 2001, the TREC Video Retrieval (TRECVID) evaluation test bed has been established to carry out benchmark evaluations of video shot detection tasks [7], and has notably contributed to the development of SBD techniques. It shows that the detection of abrupt transitions has been to some extent addressed successfully, whereas the identification of gradual transitions is still a challenge [7].

In spite of recent improvements, SBD on large-scale video data is still a very difficult task, with many unsolved problems. Among them, the problem of how to devise an effective, unified approach that can detect various types of transitions and is less sensitive to the amount of camera panning and zooming, video object motions, color, and illumination variability within the shot. In order to achieve this goal, we put forward a QR-decomposition-based approach intended to be data content independent that will be used to detect different types of transitions, maximize the efficiency of SBD performance, and lessen the need for more complicated computations. We have carried out our solution and assessed it according to the TRECVID benchmark dataset. Our method produced very hopeful results, in comparison with the best results reported in the TRECVID assessments.

Section 2 considers some modern works regarding SBD which have been done over the past few years. Section 3 discusses the problems and demanding issues concerning shot transition detection and demonstrates the reason for which we have proposed this solution. In Section 4, a brief description of QR-decomposition is given. Section 5 presents our SBD approach. Section 6 illustrates empirical evaluations of our solutions and implementations on video SBD tasks using the TRECVID test bed. Section 7 presents our conclusions and considers some ideas that could enhance the performance of our present solution.

2. Related Works

In this section, we study some existing works concerning video shot transition detection and explain, in short, some common ranking approaches for shot detection, particularly for those works that are connected with our proposed approach. SBD, also known as temporal video segmentation, is the process of detecting the transitions between the adjoining shots [8]. Beginning in the early 1990s, several organizations had already initiated projects such as QBIC [9], Columbia VideoQ (object-oriented search engine) [10], and the Virage [11], that have linked to digital video libraries to intelligently manipulate video content. During that period, research attempts were mainly centered on video processing, such as SBD, video retrieval, video object detection, and video summarization. In recent times, SBD has become a more efficient component for all video retrieval and video summarization systems.

To date, numerous approaches have been put forward for the detection of shot boundaries, and have produced highly acceptable results. After studying the literature regarding these methods, we discovered that these methods could be classified into two categories: compressed domain methods and uncompressed domain methods. In compressed domain methods, the only data obtained from the videos are those directly accessible from the MPEG streams that are Discrete Cosine Transform (DCT) coefficients, motion vectors, and prediction directions for each block. Without the decoding process the computation will be much faster, but less reliable, particularly when high motion is at work. An example is the work of Pei and Chou [12].

Early research on shot detection was primarily centered on uncompressed domain methods. These methods are compared in [2, 13]. Many of these methods have been put forward for use in the detection of abrupt transitions. In some of these approaches, an abrupt transition is identified when a particular difference measure between successive frames surpasses a threshold. The difference measure is calculated at either a pixel level or the block level. Considering the limitation of pixel difference algorithms (high sensitivity to object and camera motions), a lot of researchers recommended using some alternative measures that were on basis of global information, such as intensity histograms or color histograms [14–17]. The standard color histogram-based algorithm and its variations are now vastly employed to identify abrupt transitions. The authors of [18] have studied the RGB color histograms for shot transition detection. They have applied the singular value decomposition to analyze the histograms.

Even these histograms do not clearly display the image difference produced due to large camera motion, and therefore are unable to distinguish between smooth camera motion/parameter changes and gradual scene transitions. Although using more intricate features, such as image edges, histograms, or motion vectors [19] makes the situation better, it will alleviate but not resolve this problem [20]. The authors of [4] have developed a solution to this issue by calculating information changes between adjoining images, quantized by mutual information (MI) in gray-scale areas of the images. They have also used affine image registration to compensate for camera panning and zooming. This results in an approach with much more complex computations. Also, an efficient approach that is based on measures of information theory has been proposed in [6]. The disadvantage of this method is that it is susceptible to large camera panning and zooming as well as flashlights.

Hence, the principal challenge in gradual transition detection is that the comparison which is on the basis of spatial features, such as color histogram, edge, motion vectors, is not suitable without modeling the temporal relation between frames. In order to overcome this problem, several approaches explore large windows of frames. Contrary to popular belief, these methods are not easy to do because the variation between two different shots can be mixed up with the object motion variation in those shots.

In spite of these extensive research efforts, the problem in other machine learning and pattern recognition tools has not received enough attention. In [21], Vasconcelos and Lippman evolved a Bayesian formulation for the problem and expanded the standard thresholding model in an adaptive and intuitive way. In [22], Lienhart determined a number of key techniques that are the basis of the various SBD schemes and studied their functions in identifying abrupt cuts, fades, and dissolves. In [23], Ling et al. utilized certain features, such as intensity pixel-wise difference, color histograms in HSV space, and edge histograms in vertical and horizontal directions as the input vectors to the support vector machine (SVM). The SVM is used to classify the frames into four categories: abrupt, dissolve, fade, and wipe transitions. However, due to their inconsideration of temporal features, their algorithms are sensitive to flashlights and object motions in real-world applications.

In [24], Xu et al. propose an SBD method for news video based on object segmentation and tracking. They combine three main techniques: partitioned histogram comparison, object segmentation, and tracking based on wavelet analysis. The authors of [25] have developed a neural network classifier for detecting transitions. The classifier is trained with a dissolve synthesizer that produces synthetic dissolves. The algorithm applies to contrast-based features, and color-based features, and has provided satisfactory results in comparison with standard techniques that are based on edge change ratios. In [3], Cooper et al. suggested an SBD method based on a supervised classification. They created some new intermediate features from low-level features via pairwise similarity. These features are used as input to an efficient supervised classifier to identify shot boundaries. In [26], the authors propose cohistograms to be used for video analysis, which is a statistic graph created by counting the matching pixel pairs of two images. However, their algorithm is insensitive to camera zooming. A training-based approach is also developed in [4], where a probabilistic-based algorithm is put forward to detect both abrupt and gradual transitions. After building priori likelihood functions through training experiments, they take into consideration all related knowledge to SBD, such as shot-length distribution and visual discontinuity patterns at shot boundaries.

Recently, researchers have come to notice the significance of the temporal modeling of features for video SBD tasks. In [8], Grana and Cucchiara design a linear transition model for SBD; their method is purely concentrated on gradual transitions with a linear behavior, as well as abrupt transitions. They utilized an accurate model which yields more discriminative power than with common methods. In [27], Yuan et al. conducted research on the SBD problem: they present a general, formal framework in terms of pattern recognition. They studied the major challenges posed by the frameworks. Meanwhile, they present a unified SBD system on the basis of the graph partition model. In [28], Zelnik-Manor and Irani focused on comparing temporal and spatial factorization. They discovered a dual approach to factorization. They showed that some of the latest SBD algorithms can be reformulated in terms of this factorization.

We suggest the QR-decomposition and Gaussian transition-based SBD method and demonstrate its efficiency through a theoretical and practical analysis. As opposed to the aforementioned approaches, our solution is capable of detecting various types of gradual shots and is insensitive to camera zooming and motion, object motions, and illumination changes. Finally, we test our algorithm on the TRECVID dataset.

3. Problems and Motivations

In this section, we discuss the problem of SBD and attend to a number of exacting issues. Next, we discuss the motives and philosophy of our approach for resolving these problems.

Recently, in [27], an SBD was defined as a pattern recognition task, which formed a classification system with three major modules: representation of visual content, construction of continuity signal, and classification of continuity values. The visual aspect of video signal was also studied. In this perspective, video can be viewed as a three-dimensional signal, in which two dimensions disclose the visual content in the horizontal and vertical frame directions, and the third dimension discloses variations of the visual content over the time axes. This formulation extracts certain kinds of visual features from each frame, obtains a compact content representation, and then calculates the continuity (similarity) values of adjacent features. In this fashion, the visual content flow is converted into a 1-D temporal signal. In the perfect condition, the continuity signal within the same shot always maintains large quantities, while decreases to low values around the positions of shot transitions. Finally, the researchers differentiate the boundaries from the nonboundaries and determine the types of transitions. This formal research on SBD makes shot detection a much more challenging task, especially compared to those of a traditional SBD task. Some of these challenges are as follows.

(1) Representation of visual content and extraction of appropriate features are the significant steps in SBD approaches and affects the efficiency of other modules. The values of visual features need to be constant values throughout a shot and must be irregular during a shot transition. This poses a challenge in the search of visual features that satisfy the previous limitations.

(2) Only certain spatial features (pixel-wise intensity, color histograms, edge, etc.) have recently been studied. By utilizing these features, abrupt illumination changes such as flashlights within shots, usually bring about considerable discontinuities of interframe features, which are usually confused with shot boundaries. A number of illumination-invariant features and similarity metrics have been put forward to address this issue. Nevertheless, these techniques are not usually successful, because temporal dependencies between the frames have not been considered.

Therefore, collecting temporal features from video sequences would be a challenge.

(3) The spatial features do not clearly represent the image difference produced by large camera or object motions, and are therefore unable to distinguish between smooth camera motion/parameter changes and gradual scene transitions. As of yet, there is no complete solution to this problem.

In order to address the above challenges, we advise a QR-decomposition and Gaussian transition-based approach in a unified solution, which can considerably increase the efficiency of shot detection tasks while simultaneously reducing the need for more complicated computations. The main ideas of our solution for dealing with these problems can be outlined as follows.

(1) To deal with the representation of visual content, we utilize three-dimensional histograms in the RGB color space of each frame as spatial features. The histogram reflects the overall perspective of each frame and has increased stability, but overlooks local information. In order to include spatial information of the color distribution, we divide each frame into 3×3 blocks, and create a three-dimensional histogram for each of those blocks. We then use these histograms as a feature vector of each frame in the video.

(2) To solve the loss of temporal features, by using the spatial feature vector of each frame as a column, we construct a feature matrix. We apply the QR-decomposition to this matrix and incorporate the QR components of this matrix as temporal features along the frames.

(3) To distinguish between the shot transitions and the image differences caused by large camera or object motions, we model each shot transition by using a Gaussian model. At first, we employ a QR-decomposition-based filter to identify the candidates of shot transition. We then develop an iterative algorithm which, given a frame of the candidate set, attempts to locate the best center position for the transition by minimizing an error function, which computes the fitness of data to the Gaussian model.

4. Review of QR-Decomposition Technique

It is evident that the study of singular values of a matrix represents valuable and useful information. The singular value decomposition (SVD) of a matrix is a factorization of the matrix into a product of three matrices. Given an $m \times n$ matrix A, where $m \geq n$, the SVD of A is defined as [29]

$$A = U\Sigma V^T, \tag{1}$$

where $U = [u_{i,j}]$ is an $m \times n$ column-orthogonal matrix whose columns are referred to as left singular vectors; $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ is an $n \times n$ diagonal matrix whose diagonal elements are nonnegative singular values arranged in descending order; $V = [v_{i,j}]$ is an $n \times n$ orthogonal matrix whose columns are referred to as right singular vectors. If $\operatorname{rank}(A) = r$, then Σ satisfies

$$\sigma_1 \geq \sigma_2 \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0. \tag{2}$$

The SVD has been utilized by a great number of researchers for rule base reduction [30, 31]. The main reason for using SVD in complexity reduction is that SVD decomposes a given system into different parts and specifies the level of the importance of each decomposed part. We can lower the size of the matrix by selecting its most important columns, known as the subset selection problem [9]. In order to do this, we can simply truncate the vectors that have the least level of importance in accordance with SVD.

The QR-Decomposition of a matrix A of order $m \times n$, where $m \geq n$ is given as [30]:

$$A.\Pi = Q.R, \tag{3}$$

where $\Pi = [\rho_{i,j}]$ is a permutation matrix; $Q = [q_{i,j}]$ is an $m \times n$ column-orthogonal matrix; and $R = [r_{i,j}]$ is an $n \times n$ upper triangular matrix whose diagonal elements, the R-values, are arranged in decreasing order and incline to track the singular values of A.

If there is a well-defined gap in the singular values of A, in other words if at an index r we have $\sigma_{r+1} \ll \sigma_r$, then the subset selection will tend to produce a subset containing the most important columns (rules) of A. However the singular values often decrease smoothly without any clear gap. In such cases, the truncation index r is determined by counting the number of (close to) zero singular values in the SVD of A since it has been claimed that the smaller are the singular values, the less important the associated rules will be. As for the singular values, the R-values also help to determine the number of to pick [20, 30].

In [18, 32, 33], the SVD has been utilized for SBD, and video retrieval and summarization. Although their method is claimed to be successful, the singular values usually reduce smoothly without any clear gap and as a result calculating the truncation index is not efficient. Also, the time complexity of computing the SVD of a matrix is more than its QR-Decomposition.

5. The Proposed SBD Method

In order to design an efficient SBD algorithm, two presumptions are required, the first of which is that a feature extraction method is both discriminating and nearly invariable within the shot. The second presumption is a distance function in the feature space that detects the transitions between shots.

In this section, we will present these two presumptions in the proposed SBD algorithm. In order to decrease the number of frames to be processed by QR-decomposition, the input video sequence was initially sampled with a fixed rate of 5 frames/second. Our experiments have demonstrated that this rate is sufficient for video programs that are devoid of dramatic motions.

Details of the feature extraction and distance function for SBD are presented in the following subsections.

5.1. Feature Extraction. We split each of the input frames into $N \times N$ small blocks. For each block $B^{(i)}$, $i = 1, 2, \dots, N^2$, in frame j, we created an m-dimensional feature vector $X_j^{(i)} = [X_{1,j}^{(i)}, X_{2,j}^{(i)}, \dots, X_{m,j}^{(i)}]^T$. Using $X_j^{(i)}$ as column vector j, we obtained feature matrix $X^{(i)}$ as follows:

$$X^{(i)} = \left[X_1^{(i)}, X_2^{(i)}, \dots, X_t^{(i)}\right] = \begin{bmatrix} X_{1,1}^{(i)} & X_{1,2}^{(i)} & \cdots & X_{1,t}^{(i)} \\ X_{2,1}^{(i)} & X_{2,2}^{(i)} & \cdots & X_{2,t}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m,1}^{(i)} & X_{m,2}^{(i)} & \cdots & X_{m,t}^{(i)} \end{bmatrix}. \tag{4}$$

In order to extract spatial features of each block, from a broad range of image features, we used color histograms which are essential features for signifying the overall spatial features of each block [34]. The combination of color histograms and QR captures the temporal color distribution for each shot.

It is essential to notice that according to the experimental results, the number of blocks affects directly on the efficiency of the proposed algorithm. Specially, we split each frame into 5×5 blocks. For each block i in frame j, we created a 216-dimensional feature vector $X_j^{(i)}$. To compute the feature vector in our system implementation, we made three-dimensional histograms in RGB color space with six bins for R, G, and B, respectively, leading to a total of 216 bins. These produced a 216-dimensional feature vector for the block. Finally, utilizing the feature vector of block i in frame j as the jth column, we generated the feature matrix $X^{(i)}$ for block i, in the video sequence.

5.2. QR-Based Candidate Selection for Gradual Shot Transitions. The SBD could be formulated as a binary classification problem. In other words, a probability distribution function such as $P(S \mid x_i)$ can model the type of each frame, for example, intershot, and intrashot, where x_i is the feature value extracted from frame i, and $S \in \{\text{intershot, intrashot}\}$ represents the class of the frame. Estimating $P(S \mid x_i)$ is one of the challenging aspects of this model in the literatures [35, 36]. In this section, we estimate this probability function using QR-Decomposition. In order to achieve this, we detect the truncation index through analyzing the behavior of the R-values of the feature matrix, and therefore based on this index, we set the probability value at 0 or 1. More details are given below.

Let $F = \{F_1, F_2, \dots, F_t\}$ be the sampling set of frames of an arbitrary video sequence. We divided each input image frame into $n \times n$ blocks, and following the proposed feature extraction method in the previous section, we extracted the feature matrix $X^{(i)}$ for block i, where $i = 1, 2, \dots, n^2$.

Next, by applying QR-Decomposition to matrix $X^{(i)}$, the Q and R matrices are computed. Each R-value which is taken from QR-Decomposition of matrix $X^{(i)}$ is connected with one of the columns of $X^{(i)}$. As those columns of $X^{(i)}$ that contain only intrashot data are nearly identical to each other, the R-values matching these columns will be smaller than the ones containing intershot data.

If we define the series $Y^{(i)} = \{X_{f_1}^{(i)}, X_{f_2}^{(i)}, \dots, X_{f_t}^{(i)}\}$ as an ordered list of $X^{(i)}$ according to the decreasing order of R-values, we can then calculate the intershot probability for the ith block at the jth frame as follows:

$$P\left(\text{intershot frame} \mid Y_j^{(i)}\right) = \begin{cases} 0 & \text{if } j > (1-\beta)t, \quad , j = 1, 2, \dots, t, \\ 1, & \text{otherwise,} \end{cases} \tag{5}$$

where β is a parameter computed using the training set. To compute β, the training set is classified into two groups of frames manually: intershot frames that belong to the shot transition and intrashot frames that belong to the shots. Therefore, β will be the percentage of intrashot frames.

Equation (5) shows the intershot probability for ith block at frame j, where $i = 1, 2, \dots, n^2$. Here, we define the total intershot probability for the jth frame:

$$P(j) = \frac{1}{n^2} \sum_{i=1}^{n^2} P\left(\text{inter shot frame} \mid Y_j^{(i)}\right). \tag{6}$$

The intrashot frames are very similar, so the above probability function will be constant and close to zero. Also, we expect the probability function to change gradually. Experimental results reveal that in the shot boundaries (intershot frames) the probability function has semi-Gaussian behavior (see Figure 1(a)). The proposed algorithm detects the types of shot boundary transitions through detail analysis of this function's behavior.

Experimental results further reveal that in the hard-cut transitions, where one frame is part of the disappearing shot and the next one is part of the appearing shot, the probability function of (6) alters abruptly and its value is close to one (see Figure 1(b)). Consequently, the hard-cut transitions could be detected by using simple thresholding.

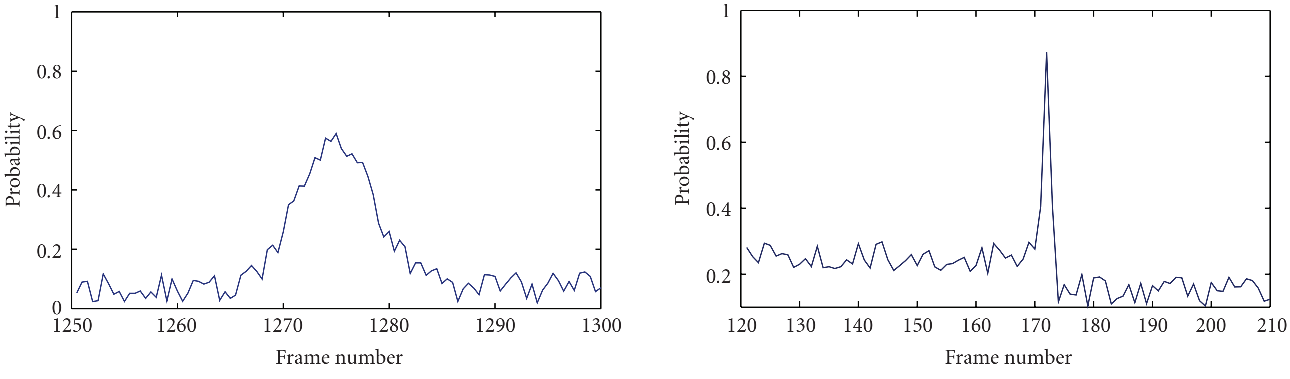
To detect the gradual shot transitions, we used an algorithm comprised of two steps: finding the candidates for the gradual transition centers and exploring the candidates to find the correct transition. In the first step, the candidates could be computed by applying a threshold to the probability function of (6). The following pseudocode clarifies the aforementioned method for hard-cut detection and candidate selection of gradual shot centers (see Algorithm 1).

The values of HardCut_Threshold and GradualTransition_Threshold are tuned using the training set. To compute the HardCut_Threshold, we consider all shots of the training set with hard cut transitions. We select, through exhaustive search, the largest real number in $[0, 1]$ that maximizes the sum of precision and recall as HardCut_Threshold. The GradualTransition_Threshold is a positive real number that is less than HardCut_Threshold and is determined similarly by using all gradual shots in the training set.

5.3. Finding the Correct Gradual Shot Transitions. In order to detect correct gradual transitions, we plot the probability function of (6) for all gradual transitions in the training set.

```
(1) for each frame $f_j$ in $F = \{f_1, f_2, \dots, f_t\}$ do
        (a) Compute $P(j)$ Using (6)
        (b) if ($P(j) >$ *HurdCutt_Threshold)* then
                Frame $f_j$ is a Hard Cut Transition
        else
                if ($P(j) >$ *GradualTransition_Threshold*) then
                        Frame $f_j$ is a candidate for Gradual Transition center
                end
        end
End
```

Algorithm 1



(a)

(b)

Figure 1: The probability function of (6). (a) For a gradual transition which is semi-Gaussian and (b) for a hard cut transition.

After considering them, we find that the function exhibits semi-Gaussian behavior during gradual transitions. Figure 2 shows some instances of these plots.

Ideally, an arbitrary Gaussian function is defined as.

$$G^{\eta}_{\sigma^2}(t) = \frac{1}{\sqrt{2.\pi.\sigma}} \cdot \exp\left(-\frac{(t-\eta)^2}{\sigma^2}\right), \tag{7}$$

where σ is a real number defined as the variance value of the function and $t = \eta$ is the mean or center of the function.

After finding the candidates of gradual transition centers, it is necessary to analyze them and to detect the correct gradual transitions. In an ideal case, a connection between the probability function of (6) and the Gaussian function of (7) could adequately indicate the correct transition presence, but realistically we must use an algorithm that explores the candidates of the shot transition center and automatically detects the correct transition.

For this reason, we designed an algorithm that uses the probability function of (6) to find the correct transition. It is imperative to observe that in the gradual shots, the transition is not detectable in all blocks of the shot boundary frames. Consequently, the probability function of (6) does not increase to an absolute maximum at the center of the shot transition. Due to the properties of QR-decomposition [29], increasing the size of the window of shot transition (variance in (6)) causes the values of the function to increase to an absolute maximum. In this case, the length of transition, defined as the number of frames in which the transition is visible, is $L = 2\sigma - 1$. Therefore, we will utilize these properties to detect correct transitions from the center of shot transitions candidates that were obtained in the previous section.

Let $\overline{n}$ be an obtained candidate for center of a gradual shot. Then, the median parameter of (7) is set to $\eta = \overline{n}$. Next, a value for variance parameter σ in (7) that makes a maximum adjustment with the probability function of (6) must be found. To do this, the following measure is defined:

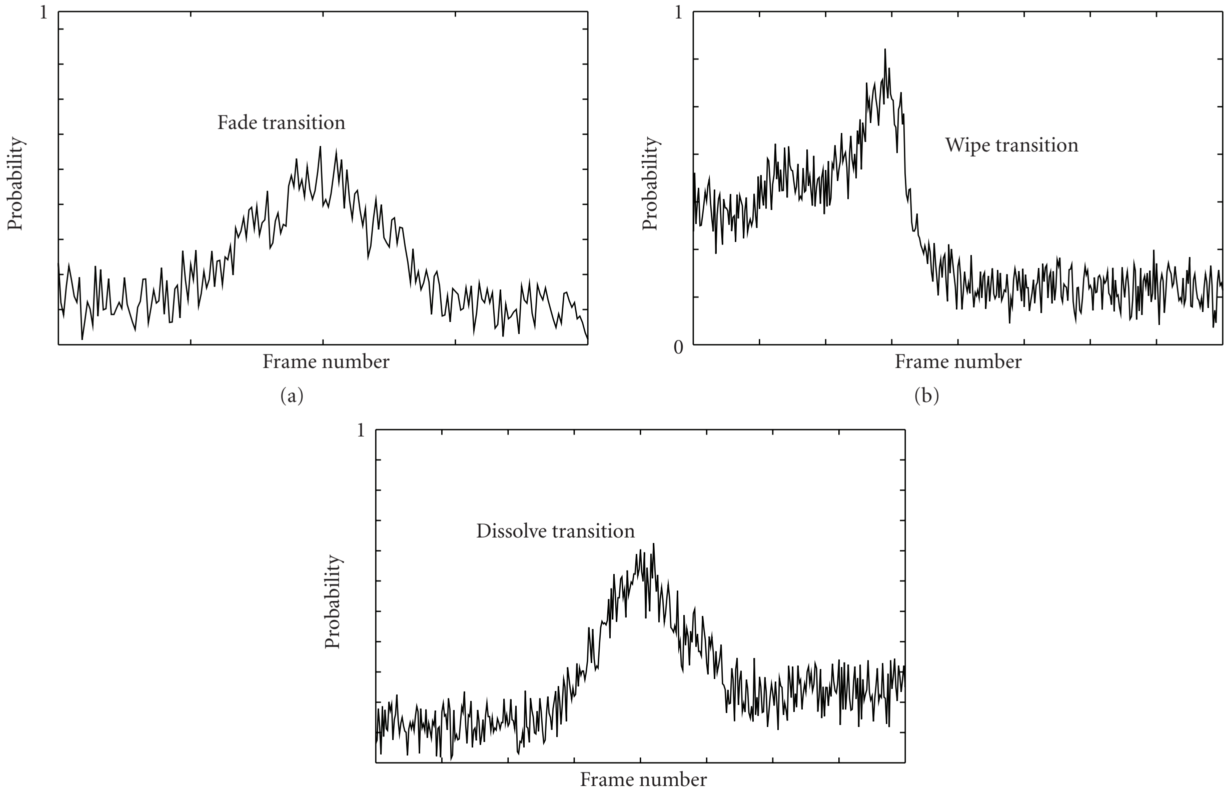
$$\begin{aligned} W^{\overline{n}}_{\sigma} = \sum_{i=0}^{2\sigma} \Big\{ \Big| G^{\overline{n}}_{\sigma^2}(\overline{n}-i) - P(\overline{n}-i) \Big| \\ + \Big| G^{\overline{n}}_{\sigma^2}(\overline{n}+i) - P(\overline{n}+i) \Big| \Big\}. \end{aligned} \tag{8}$$

The σ value that minimizes $W^{\overline{n}}_{\sigma}$ is defined as an optimum value for variance parameter:

$$\overline{\sigma} = \arg \min_{0 \le \sigma \le \Sigma} \{ W^{\overline{n}}_{\sigma} \}, \tag{9}$$

where Σ is the maximum size that a transition can presume. In our experiments, we consider $\Sigma = 24$.

In typical conditions, the algorithm searches the appropriate variance parameter which leads to the expected Gaussian shape and identifies the correct σ and hence the length of the transition. If Z is the total number of candidates of center of gradual shots, then because this part of the



(a)

(b)

(c)

Figure 2: The plots of the probability function of (6) for some gradual shot transitions of different types.

algorithm is run solely for the obtained candidate points, the total number of computation is certainly less than the product of Σ and Z.

Let the median and variance parameters be $\overline{n}$ and $\overline{\sigma}$, respectively, as identified by the algorithm; then the function $G_{\overline{\sigma}^2}^{\overline{n}}(t)$ is a Gaussian function that has maximum adjustment with the probability function $P(j)$ in (6). Now, we need to verify the importance of the transition and determine how well the actual data corresponds to the Gaussian function:

$$\text{Height}_{\overline{\sigma}}^{\overline{n}} = P(\overline{n}) - \min\{P(\overline{n} - 2\overline{\sigma}), P(\overline{n} + 2\overline{\sigma})\}. \tag{10}$$

This value is the height of the center value regarding the lower of two values of $P(j)$ in correspondence to the extremes of the Gaussian function, and gives information about the importance of the transition.

Experimental results illustrate that in real cases, object and camera motions cause some semitransition behavior in the probability function of (6). In order to address this effect, we must find the hypothesis of having an isosceles Gaussian function and define the fitting error measure as

$$\text{Error}_{\overline{\sigma}}^{\overline{n}} = \frac{1}{4\overline{\sigma}}\sum_{i=1}^{2\overline{\sigma}}\Big\{\Big|G_{\overline{\sigma}^2}^{\overline{n}}(\overline{n} - i) - P(\overline{n} - i)\Big| + \Big|G_{\overline{\sigma}^2}^{\overline{n}}(\overline{n} + i) - P(\overline{n} + i)\Big|\Big\}. \tag{11}$$

This error sum is divided by the interval's length to achieve a measure that is not dependent on the length of the transition. Also, a minimum threshold of the Height$_{\overline{\sigma}}^{\overline{n}}$ value, T_H, and a maximum threshold of the Error$_{\overline{\sigma}}^{\overline{n}}$ value, T_E, are used to differentiate real gradual shot changes from false ones. The final decision is made on the basis of two parameters. Finally, the analysis of the candidates and detection of the correct gradual transitions could be summarized as in Algorithm 2.

Table 1: Video set used in our experiments.

Video	Frames	cuts	gradual
News	95743	236	27
Cartoon	74384	142	39
Movie	85958	369	61
Sport	109381	187	17
Documentary	57491	93	9

Hence, the correct gradual transitions and the length of the transitions could be extracted by using this algorithm.

5.4. Computational Complexity. The computational complexity of the proposed SBD algorithm is reflected in the need to calculate three histograms for each color component R, G, B for each block of input frame. Let the size of input image frame be $M \times N$. We divide each input image frame into $n \times n$ blocks, then make MN/n^2 additions to calculate one histogram for each block. We must first calculate three

```
(1) for all $\overline{n}$ in the candidates for center of gradual shots do
        (1.1) for all $0 \le \sigma \le \Sigma$ do
                $\overline{\sigma} = \arg\min_{0\le\sigma\le\Sigma}\{W_{\sigma}^{\overline{n}}\}$ according to (9).
            end
        (1.2) Compute $\mathrm{Height}_{\overline{\sigma}}^{\overline{n}}$ according to (10).
        (1.3) Compute $\mathrm{Error}_{\overline{\sigma}}^{\overline{n}}$ according to (11).
        (1.4) if ($\mathrm{Height}_{\overline{\sigma}}^{\overline{n}} \ge T_H$ and $\mathrm{Error}_{\overline{\sigma}}^{\overline{n}} \le T_E$) then
                        Transition($\overline{n}, \overline{\sigma}$) = TRUE
                end
End
```

Algorithm 2

Table 2: Comparison of results of the different algorithms using our dataset.

	Our method		[37]		[38]	
Video	Precision	Recall	Precision	Recall	Precision	Recall
News	94	98	91	93	89	94
Cartoon	89	92	74	86	88	91
Movie	91	95	83	92	86	91
Sport	95	97	87	92	79	90
Documentary	90	92	92	94	93	95
Overall	91.8	94.8	85.4	91.4	87.0	92.2

histograms for each block, with $(3MN)/n^2$ additions for each block. Therefore, if we have t frames, we need a total of $(3MNt)/n^2$ additions to compute the feature matrix for each block in the video sequence. We also need mt^2 additional multiplication and addition operations [29] to compute the QR-Decomposition of the feature matrix of each block during the video sequence, where m is the number of dimensions of feature vector (number of histogram bins). Thus, for calculating (5) for each t frame, we need a total of $((3\text{MNt})/n^2 + mt^2 + t)$ addition and multiplication operations; therefore, calculation of (6) for each t frame with n^2 blocks will need $((3MNt)/n^2 + mt^2 + t)(n^2t)$. It is obvious that the complexity of the hard-cut detection algorithm in Section 5.2 needs t comparisons, and finding the correct gradual transition algorithm of Section 5.3 needs $\overline{n}(\Sigma^2+3\Sigma)$, where $\overline{n}$ is the number of candidates for gradual transitions and Σ is the maximum size of a transition. Consequently, the proposed SBD algorithm has polynomial time complexity. Thus, despite the existence of matrix computations in the algorithm, we have tolerable time complexity.

6. Experimental Results

The proposed video shot transition detection algorithm is evaluated by using a 4-hour video set. All videos have been segmented manually through identifying hard cuts and gradual transitions as well as their length. In all, 1180 shot transitions existed in these video sequences; of which, 1027 were hard-cut transitions, and 153 were gradual shot boundaries. The video clips were obtained mainly from the Internet and various television programs, and included various movie formats, such as AVI, MPEG-7, and SGI. The complete video database will be made available upon request. The details of each video are shown in Table 1.

We employed recall and precision as the measures for performance evaluation, which are defined below.

(i) The Recall measure, also known as the true positive function or sensitivity, equals the ratio of correct experimental detections over the number of all true detections:

$$\text{recall} = \frac{\text{number of correctly detected boundaries}}{\text{number of true boundaries}}. \tag{12}$$

(ii) The Precision measure is defined as the ratio of correct experimental detections over the number of all experimental detections:

$$\text{precision} = \frac{\text{number of correctly detected boundaries}}{\text{number of totally detected bounaries}}. \tag{13}$$

An excellent shot transition detector must possess both high precision and high recall. We compared our algorithm to the techniques proposed in [37, 38], which are shot detection software that can be downloaded freely and provide either MPEF-7 or XML formatted output. The results of these algorithms used on similar video sequences are shown in Table 2 (second and third columns). Because of camera flashes, a number of false shot cut detections were resulted. It is obvious that the proposed video SBD system has obtained reasonable performance. Also, our algorithm is efficient enough to detect shots with small length.

Table 3: The results of experiments on TRECVID 2006 data set.

	All		Cuts		Gradual		Frame	
	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
a	0.6898	0.7425	0.7065	0.7868	0.6446	0.6541	0.7243	0.7850
b	0.8210	0.8986	0.9216	0.8507	0.7416	0.8355	0.8739	0.9261
c	0.5953	0.8317	0.5926	0.8387	0.6030	0.8101	0.8275	0.7984
d	0.6403	0.5723	0.7284	0.5954	0.4031	0.5276	0.5639	0.7834
e	0.8317	0.8217	0.9070	0.8873	0.6420	0.6507	0.8527	0.5637
f	0.5377	0.6044	0.7311	0.6036	0.0159	0.7416	0.2540	0.7056
g	0.3278	0.1595	0.3703	0.1431	0.2126	0.3778	0.4269	0.7766
h	0.7617	0.8687	0.8215	0.8888	0.6013	0.8024	0.7716	0.8486
i	0.7848	0.7344	0.7949	0.8170	0.7565	0.5711	0.7726	0.7000
QR	0.9252	0.8912	0.9306	0.9044	0.9054	0.8715	0.9285	0.8993

Table 4: The results of the experiments on different types of gradual transitions.

Computing thresholds and parameters		Precision	Recall
Fade and dissolve separately	Fade	94.4	97.14
	Dissolve	89.1	94.2
	Average	91.75	95.67
All gradual transitions		89.04	91.4

In the experiments, the video set is sampled with a fixed rate of 5 frames/second. For feature extraction, each frame is divided into 5 × 5 blocks. Also, the HurdCutt_Threshold, GradualTransition_Threshold, and the T_E, T_H have been adjusted using the "news_1.avi" as a training set. The algorithm searches for hard cut transitions and candidates for gradual transitions via HurdCutt_Threshold and GradualTransition_Threshold, respectively. In order to detect correct gradual transitions, the method presented in Section 5.3 is used. The algorithm fits a Gaussian function with maximum adjustment to the probability function of these candidates, and then, computes the $\text{Height}_{\frac{\overline{n}}{\overline{\sigma}}}$ and $\text{Error}_{\frac{\overline{n}}{\overline{\sigma}}}$ for them. The correct gradual transitions are detected based on these parameters. To do this, as discussed in Section 5.3, it uses T_E and T_H thresholds. We chose the thresholds, via comprehensive search so as to maximize the sum of precision and recalls. Some of the examples of the detected candidates for gradual shots are shown in Figure 3. Also, in Figure 4, the $\text{Error}_{\frac{\overline{n}}{\overline{\sigma}}}$ diagram for all different types of gradual shot transitions is shown to justify the Semi-Gaussian assumption of the appearance of probability function of gradual transitions.

The newscasts from the reference video test set TRECVID 2006 were inserted into the testing set in order to make it possible, in the future, to compare these techniques with other SBD techniques. This set consists of over 6 hours of video sequences that have been digitalized with a frame rate of 29.97 fps and a resolution of 352 × 264 pixels. To increase the speed of our computations for our experiments, spatially downsampled frames with resolutions of 176 × 132 pixels were used. The ground truth given by TRECVID was utilized for these video sequences.

The results of experiments are displayed in Table 3. It is clear that the achieved results are better than those reported in the TRECVID 2006 competition [7]. The best reported hard cut detection results for recall and precision for TRECVID are 90% and 88%, respectively, while our approach yields 93% recall and 90% precision. Nearly all false detections appeared because of flashlights and high-speed camera motions. In some instances, false detections emerged where artistic camera edits were applied, such as in the case of commercials. The overlooked shot cut detections were mostly caused due to either shot changes between two images with highly similar spatial color distribution or shot changes that happened in only a section of the video frame.

In Figure 5, the overall results of detection from TRECVID 2006 participants are illustrated. The total number of groups that participated was 26, but only the best 20 are shown. The proposed algorithm has achieved great results in the recall with 90.5%–96.0% properly detected transitions using the same algorithm for all the various kinds of transitions. As the QR-based algorithm has no prediction for camera motions, light changes, or picture-in-picture changes, the results concerning its accuracy are not as satisfactory.

In our experiments, we examined the proposed algorithm with identical parameters and thresholds for all different types of gradual transitions. To evaluate the efficiency of the algorithm on different types of gradual transitions, we classify the data set of Table 1 into two types: fade and dissolve, and use 50 instances of each transition type which would be 100 transitions in total. Then, we use 30 transitions for training and 70 transitions for testing. We compute the parameters and thresholds of the algorithm separately for

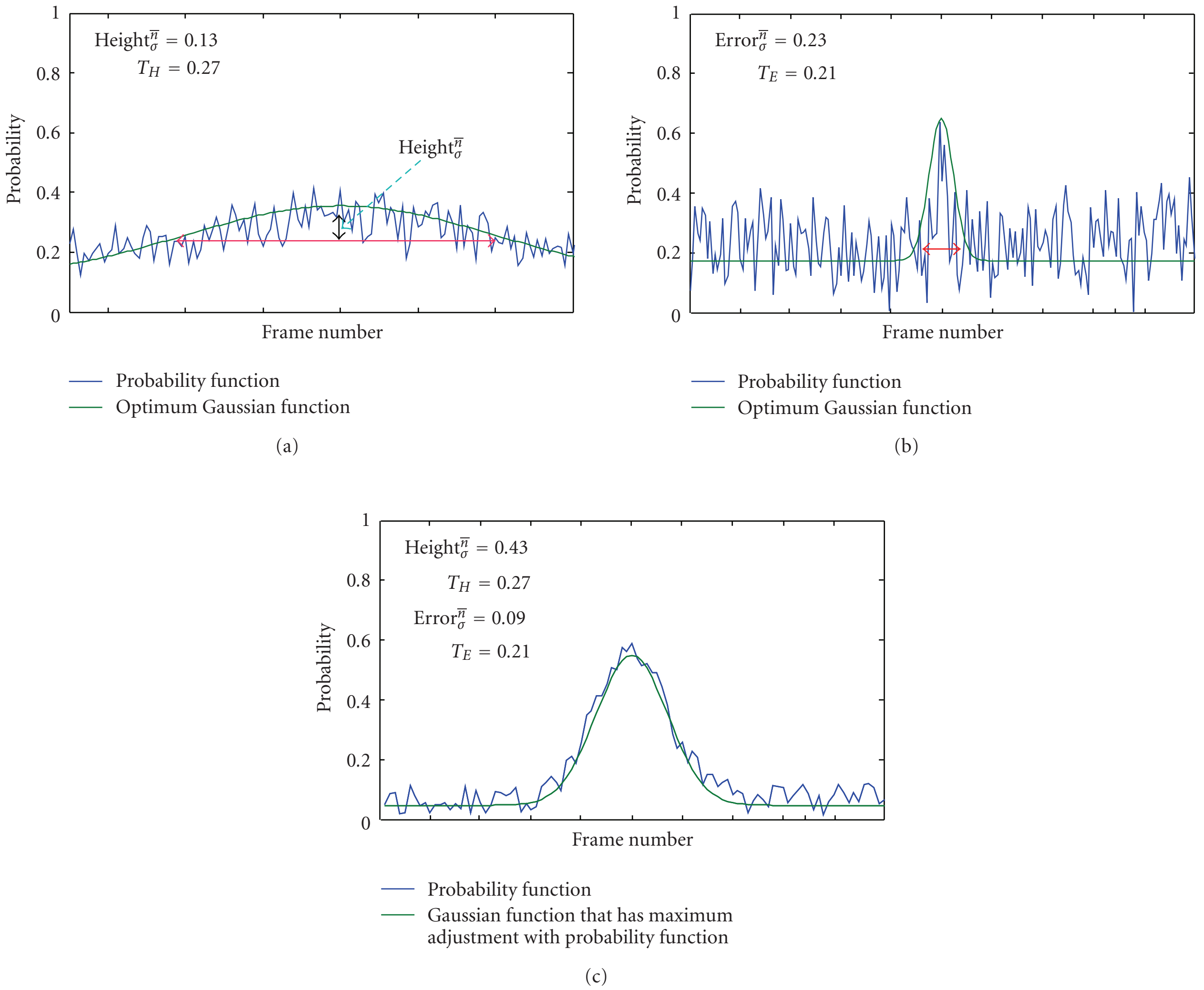


Figure 3: Examples of the detected candidates for gradual transitions, and finding the correct gradual transitions: (a) as $\text{Height}_{\overline{\sigma}}^{\overline{n}} < T_H$, it is an incorrect gradual transition. (b) As $\text{Error}_{\overline{\sigma}}^{\overline{n}} > T_H$, it is an incorrect transition. (c) A correct gradual transition.

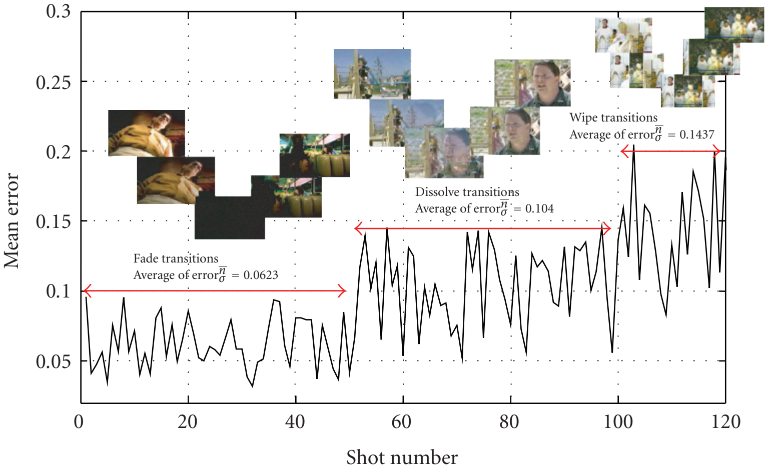


Figure 4: The diagram of $\text{Error}_{\overline{\sigma}}^{\overline{n}}$ for all different types of gradual shot transitions that have been extracted from TRECVID 2006 and the dataset is shown in Table 1.

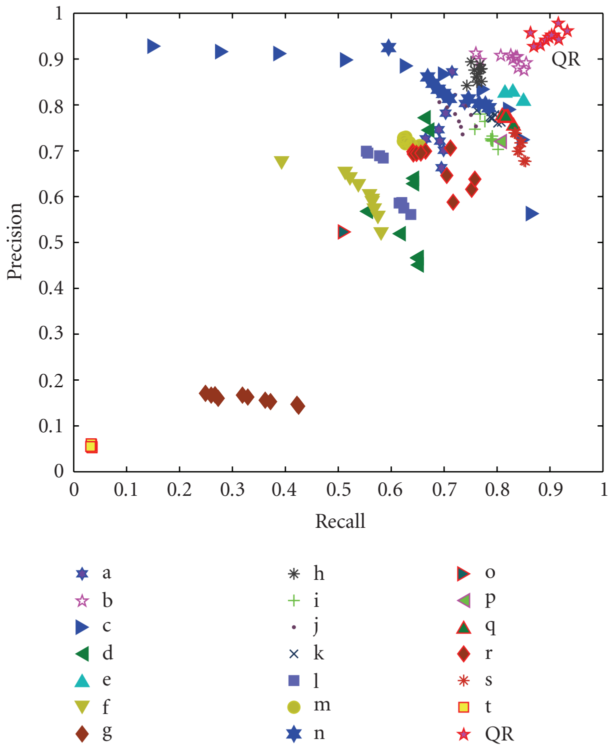


Figure 5: Results on the overall detection (cuts and transitions) based on the data provided by the organizers in TRECVID-2006. Our approach is labeled QR.

each transition type. The results of the experiments are displayed in Table 4. It is clear that when different types of transitions are processed distinctly, the results are better. Also, Figure 4 justifies these results. As shown in Figure 4, the mean error between probability function of (6) and the best fitted Gaussian function for dissolve transitions is greater than the mean error for fade transitions. Consequently, if we have previous knowledge about types of transitions in clips of dataset, it would be better to categorize them into different types, and then run the proposed algorithm separately.

7. Conclusion

In this paper, a new approach for shot boundary detection is introduced. We put forward a novel technique for detecting hard cut and gradual transitions through QR-decomposition and Gaussian functions. The algorithm utilizes the properties of QR-decomposition and extracts a block-wise probability function that shows the probability of video frames to be in shot transitions. The probability function has abrupt changes in hard cut transitions and semi-Gaussian behavior in gradual transitions. The algorithm detects the transitions by analyzing this probability function. Through the use of large-scale test sets provided by the TRECVID 2006, the precision of our algorithm was empirically proved to be very high. Also, our approach was successfully contrasted with other approaches that have been reported in previous literature.

References

[1] D. Brezeale and D. J. Cook, "Automatic video classification: a survey of the literature," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 38, no. 3, pp. 416–430, 2008.

[2] R. Lienhart, "Comparison of automatic shot boundary detection algorithms," in *Storage and Retrieval for Image and Video Databases VII*, vol. 3656 of *Proceedings of SPIE*, pp. 290–301, San Jose, Ca, USA, January 1999.

[3] M. Cooper, T. Liu, and E. Rieffel, "Video segmentation via temporal pattern classification," *IEEE Transactions on Multimedia*, vol. 9, no. 3, pp. 610–618, 2007.

[4] A. Hanjalic, "Shot-boundary detection: unraveled and resolved?" *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 2, pp. 90–105, 2002.

[5] C. Cotsaces, N. Nikolaidis, and I. Pitas, "Video shot detection and condensed representation: a review," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 28–37, 2006.

[6] Z. Cernekova, I. Pitas, and C. Nikou, "Information theory-based shot cut/fade detection and video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 82–91, 2006.

[7] NIST, Homepage of Trecvid Evaluation, http://www-nlpir.nist.gov/projects/trecvid/.

[8] C. Grana and R. Cucchiara, "Linear transition detection as a unified shot detection approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 4, pp. 483–489, 2007.

[9] M. Flickner, H. Sawhney, W. Niblack, et al., "Query by image and video content: the QBIC system," *Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[10] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 602–615, 1998.

[11] A. Hamrapur, A. Gupta, B. Horowitz, et al., "Virage video engine," in *Storage and Retrieval for Image and Video Databases V*, Proceedings of SPIE, pp. 188–197, San Jose, Calif, USA, February 1997.

[12] S.-C. Pei and Y.-Z. Chou, "Efficient MPEG compressed video analysis using macroblock type information," *IEEE Transactions on Multimedia*, vol. 1, no. 4, pp. 321–333, 1999.

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

[14] A. Dailianas, R. B. Allen, and P. England, "Comparison of automatic video segmentation algorithms," in *Integration Issues in Large Commercial Media Delivery Systems*, vol. 2615 of *Proceedings of SPIE*, pp. 2–16, Philadelphia, Pa, USA, October 1996.

[15] G. Ahanger and T. D. C. Little, "A survey of technologies for parsing and indexing digital video," *Journal of Visual Communication and Image Representation*, vol. 7, no. 1, pp. 28–43, 1996.

[16] N. V. Patel and I. K. Sethi, "Video shot detection and characterization for video databases," *Pattern Recognition*, vol. 30, no. 4, pp. 583–592, 1997.

[17] S. Tsekeridou and I. Pitas, "Content-based video parsing and indexing based on audio-visual interaction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 4, pp. 522–535, 2001.

[18] Y. Gong and X. Liu, “Video summarization and retrieval using singular value decomposition,” *Multimedia Systems*, vol. 9, no. 2, pp. 157–168, 2003.

[19] C.-L. Huang and B.-Y. Liao, “A robust scene-change detection method for video segmentation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1281–1288, 2001.

[20] M. Amintoosi, F. Farbiz, and M. Fathy, “A QR decomposition based mixture model algorithm for background modeling,” in *Proceedings of the 6th International Conference on Information, Communications and Signal Processing (ICICS ’07)*, Singapore, December 2007.

[21] N. Vasconcelos and A. Lippman, “Statistical models of video structure for content analysis and characterization,” *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 3–19, 2000.

[22] R. Lienhart, “Reliable transition detection in videos: a survey and practitioner’s guide,” *International Journal of Image and Graphics*, vol. 1, no. 3, pp. 469–486, 2001.

[23] X. Ling, O. Yuanxin, L. Huan, and X. Zhang, “A method for fast shot boundary detection based on SVM,” in *Proceedings of the 1st International Congress on Image and Signal Processing (CISP ’08)*, vol. 2, pp. 445–449, Sanya, China, May 2008.

[24] X.-W. Xu, G.-H. Li, and J. Yuan, “A shot boundary detection method for news video based on object segmentation and tracking,” in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics (ICMLC ’08)*, vol. 5, pp. 2470–2475, Kunming, China, July 2008.

[25] R. Lienhart and A. Zaccarin, “A system for reliable dissolve detection in videos,” in *Proceedings of IEEE International Conference on Image Processing (ICIP ’01)*, vol. 3, pp. 406–409, Thessaloniki, Greece, October 2001.

[26] P. Hao and Y. Chen, “Co-histogram and its application in video analysis,” in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME ’04)*, vol. 3, pp. 1543–1546, Taipei, Taiwan, June 2004.

[27] J. Yuan, H. Wang, L. Xiao, et al., “A formal study of shot boundary detection,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 2, pp. 168–186, 2007.

[28] L. Zelnik-Manor and M. Irani, “Temporal factorization vs. spatial factorization,” in *Proceedings of the 8th European Conference on Computer Vision (ECCV ’04)*, vol. 3022 of *Lecture Notes in Computer Science*, pp. 434–445, Prague, Czech Republic, May 2004.

[29] G. Golub and C. Loan, *Matrix Computations*, Johns-Hopkins, Baltimore, Md, USA, 2nd edition, 1989.

[30] M. Seines and R. Babuska, “Rule base reduction: some comments on the use of orthogonal transforms,” *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 31, no. 2, pp. 199–206, 2001.

[31] O. Kaynak, K. Jezernik, and A. Szeghegyi, “Complexity reduction of rule based models: a survey,” in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE ’02)*, vol. 2, pp. 1216–1221, Honolulu, Hawaii, USA, May 2002.

[32] Y. Gong and X. Liu, “Video shot segmentation and classification,” in *Proceedings of the 15th International Conference on Pattern Recognition*, vol. 1, pp. 860–863, Barcelona, Spain, 2000.

[33] Z. Cernekova, C. Kotropoulos, and I. Pitas, “Video shot-boundary detection using singular-value decomposition and statistical tests,” *Journal of Electronic Imaging*, vol. 16, no. 4, Article ID 043012, pp. 51–59, 2007.

[34] W. Cheng, D. Xu, Y. Jiang, and C. Lang, “Information theoretic metrics in shot boundary detection,” in *Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES ’05)*, vol. 3683 of *Lecture Notes in Computer Science*, pp. 388–394, Melbourne, Australia, September 2005.

[35] A. Hanjalic and H. Zhang, “Optimal shot boundary detection based on robust statistical models,” in *Proceedings of the 6th International Conference on Multimedia Computing and Systems (ICMCS ’99)*, vol. 2, pp. 710–714, Florence, Italy, June 1999.

[36] A. Pardo, “Probabilistic shot boundary detection using interframe histogram differences,” in *Proceedibgs of the 11th Iberoamerican Congress in Pattern Recognition (CIARP ’06)*, vol. 4225 of *Lecture Notes in Computer Science*, pp. 726–732, Cancun, Mexico, November 2006.

[37] A. Miene, A. Dammeyer, T. Hermes, and O. Herzog, “Advanced and adapted shot boundary detection,” in *Proceedings of ECDL WS Generalized Documents*, pp. 39–43, 2001.

[38] “VCM: Video Content Management by Technologie-Zentrum Informatik (www.tzi.de),” http://atlas.ced.tuc.gr/.