Multimed Tools Appl (2018) 77:8139–8161
DOI 10.1007/s11042-017-4707-9

# Video shot boundary detection using multiscale geometric analysis of nsct and least squares support vector machine

**Jaydeb Mondal[1] · Malay Kumar Kundu[1] · Sudeb Das[2] · Manish Chowdhury[3]**

Received: 14 June 2016 / Revised: 18 February 2017 / Accepted: 12 April 2017 /
Published online: 25 April 2017


**Abstract** The fundamental step in video content analysis is the temporal segmentation of video stream into shots, which is known as Shot Boundary Detection (SBD). The sudden transition from one shot to another is known as Abrupt Transition (AT), whereas if the transition occurs over several frames, it is called Gradual Transition (GT). A unified framework for the simultaneous detection of both AT and GT have been proposed in this article. The proposed method uses the multiscale geometric analysis of Non-Subsampled Contourlet Transform (NSCT) for feature extraction from the video frames. The dimension of the feature vectors generated using NSCT is reduced through principal component analysis to simultaneously achieve computational efficiency and performance improvement. Finally, cost efficient Least Squares Support Vector Machine (LS-SVM) classifier is used to classify the frames of a given video sequence based on the feature vectors into No-Transition (NT), AT and GT classes. A novel efficient method of training set generation is also proposed which not only reduces the training time but also improves the performance. The performance of the proposed technique is compared with several state-of-the-art SBD methods on TRECVID 2007 and TRECVID 2001 test data. The empirical results show the effectiveness of the proposed algorithm.

✉ Jaydeb Mondal
maharaj305@gmail.com

Malay Kumar Kundu
malay@isical.ac.in

Sudeb Das
sudeb.das@videonetics.com

Manish Chowdhury
manchowd@kth.se

[1] Machine Intelligence Unit, Indian Statistical Institute, 203 B.T.Road, Kolkata, 700108, India

[2] Videonetics Technologies Pvt. Ltd., Salt Lake City, Kolkata, 700091, India

[3] KTH School of Technology and Health, Huddinge, SE-14152, Stockholm, Sweden

**Keywords** Shot boundary detection · Abrupt transition · Gradual transition · Principal component analysis · Non-subsampled contourlet transform · Least squares support vector machine

## 1 Introduction

Recent advances in computer and multimedia technologies have made digital video, a common and important medium for various applications such as education, remote sensing, broadcasting, video conference and surveillance etc [36, 37]. Due to the enormous increasing rate of video production, development of effective tools for automatic analysis of video content becomes an important research issue. Partitioning of video into shots, known as Shot Boundary Detection (SBD) is the first and essential step towards analysis of video content; it provides a basis for nearly all types of video abstraction. A shot is defined as a sequence of consecutive frames taken from a single non-stop camera operation [14]. It basically represents a unit semantic information with respect to the whole video. If there is sudden transition from one shot to the other shot, i.e. if there is no intermediate frames between two shots, it is called Abrupt Transition (AT). On the other hand, if the transition occurs over several frames, it is called Gradual Transition (GT). The GT is further classified mainly into three sub-classes e.g. fade-in, fade-out, dissolve and wipe according to their effects [27]. Among the different types of shot boundaries, GTs are difficult to detect than ATs. This is due to the slow and low frame content changing nature of GTs over ATs. An effective SBD technique (detecting both AT and GT) has various applications in automatic video content analysis such that content based video retrieval, video summarization, video-restoration, video-quality analysis, video aesthetic analysis etc.

It is necessary to develop algorithms for effective extraction of features and classification with suitable dissimilarity measure to design an accurate tool for SBD. Literature on SBD is quite rich. A large number of methods for SBD have been proposed in the past [4, 14, 18, 27, 28, 36, 37]. Recent approaches can be divided broadly into two categories, namely pixel domain based approach and transform domain based approach. In the pixel domain approach, the simplest methods are based on pixel intensity difference between consecutive frames [14, 37]. These methods are easy to implement and computationally fast, but very sensitive to camera motions as well as abrupt changes in object motion. Usually people have divided each frame into a number of equal-sized non-overlapping blocks and compute features from a subset of these blocks to remove the influence of object motion [18, 37]. Color histograms in different color spaces, like RGB, HSV, YCbCr, L*a*b*, etc. have also been used as features to reduce the influence of object/camera motions [36]. Structural properties of the video frames such as edge characteristics is used as feature to reduce the influence of flashlight effects [16]. There are several other pixel domain approaches, that use joint entropy, correlation coefficient, Mutual Information, Scale Invariant Feature Transform (SIFT) and local key-point matching as features for SBD [19, 27].

On the other hand, various researchers have used unitary transforms like discrete cosine transform [1], fast fourier transform [25], Walsh Hadamard Transform (WHT) [19], to extract features for SBD. Although these methods are good in detecting AT but they fail to detect GT effectively. This is because of the fact that the duration and characteristics of GT (e.g. fade-in, fade-out, dissolve and wipe) varies widely from video to video. Moreover, in most of the real-life scenarios these existing schemes often produce false results which eventually hampers the task of automatic analysis of the video content. It can only be possible

to detect GT accurately, if variable or multi resolution approach is taken, as GT is detected better at lower resolution [7]. Since wavelet is well known for its capability to represent a signal in different scales, many authors have used Discrete Wavelet Transform (DWT) based features for SBD, especially for GT detection [7, 28]. However, the main problem of DWT based features is the inherent lack of support to directionality and anisotropy. Thus efficiency of the DWT based approaches suffer in the presence of large object/camera motions. A recent theory called Multi-scale Geometric Analysis (MGA) for high-dimensional signals has been introduced to overcome these limitations and several MGA tools have been developed like Curvelet (CVT), Contourlet (CNT) with application to different problem domains [6, 10].

Moreover, proper and automatic selection of no-transition (negative) frames for efficient training set generation is another important requirement for developing an effective SBD scheme. Improper selection of non-transition frames for training purpose often leads to imbalanced training procedure. This imbalanced training set results in improper trained model and also requires much higher training time [4, 14, 27, 36]. The conventionally used training set generation procedures (random selection, single threshold based selection) can not produce high quality training set and often fails to achieve the desired results [4, 27]. Therefore, we need a novel and effective way of proper training set generation procedure.

In this article a new SBD framework has been proposed which is capable of detecting both AT and GT with equal degree of accuracy. Our main motivation is to develop an effective SBD technique which is capable of correctly discriminating the changes caused by both types of transitions from one shot to the other in the presence of different types of disturbances like large movements of objects or camera and flashlight effects. The key contributions of the proposed technique are as follows:

- Rather than the conventionally used unitary transforms like discrete cosine transform, fast Fourier transform, Walsh Hadamard transform etc., to extract features for SBD, in the proposed scheme, a novel robust frame-representation scheme is proposed based on the multiscale, multi-directional, and shift-invariant properties of Non-Sub sampled Contourlet Transform (NSCT), which reduces the problems of object/camera motion and flashlight effects. Compared to the existing NSCT-based SBD scheme [30] capable only of AT detection, our proposed scheme can detect both AT and GT present in a video.
- A novel low-dimensional feature vector based on non-overlapping block subdivision of NSCT's high-frequency subbands is developed using the well-known Principal Component Analysis (PCA) technique from the dissimilarity values using the contextual information around neighboring frames to capture the distinct characteristics of both abrupt and gradual transitions.
- A new technique for training set generation is also proposed to reduce the effect of imbalanced training set problem, making the trained model unbiased to any particular type of transition.

In the proposed method, we have used Multiscale Geometric Analysis (MGA) of NSCT for generation of feature vectors. NSCT is an improvement of Contourlet Transform (CNT) to mitigate the shift-sensitivity and the aliasing problems of CNT in both space and frequency domains [9]. NSCT is a flexible multiscale, multi-directional, and shift-invariant image transform capable of producing robust feature vector which reduces the problems of object/camera motions and flashlight effects. The dimensionality of the feature vectors is reduced through PCA to achieve high computational efficiency as well as to boost the

performance. The Least Square version of Support Vector Machine (LS-SVM) is adopted to classify the frames into different transition events like AT, GT (fade-in, fade-out, dissolve etc.) and No-Transition (NT). Moreover, we have also proposed a novel method of training set generation using two automatically selected different threshold values. The performance of the proposed algorithm has been tested using several benchmarking videos containing large number of ATs and various types of GTs and compared with several other state-of-the-art SBD methods. The experimental results show the superiority of the proposed technique for most of the benchmarking videos.

The rest of the paper is organized as follows. In Section 2, theoretical preliminaries of NSCT is briefly described. Our proposed method is presented in Section 3. Section 4 contains experimental results and discussion. Finally, conclusions and future Work are given in Section 5.

## 2 Theoretical preliminaries

NSCT is a fully shift-invariant,multiscale, and multi-direction expansion with fast implementability [9]. NSCT achieves the shift-invariance property (not present in CNT) by using the non-subsampled pyramid filter bank (NSPFB) and the non-subsampled directional filter bank (NSDFB).

### 2.1 Non-subsampled pyramidal filter bank (NSPFB)

NSPFB ensures the multiscale property of the NSCT, and has no down-sampling or up-sampling, hence shift-invariant. It is constructed by iterated two channel Non-Subsampled Filter Bank (NSFB), and one low-frequency and one high-frequency image is generated at each NSPFB decomposition level. The subsequent NSPFB decomposition stages are carried out to decompose the low-frequency component available iteratively to capture the singularities in the image. NSPFB results in $k+1$ sub-images, which consist of one Low-Frequency Subband (LFS) and $k$ High-Frequency Subbands (HFSs), all of whose sizes are the same as the source image, where $k$ denotes the number of decomposition levels.

### 2.2 Non-subsampled directional filter bank (NSDFB)

The NSDFB is constructed by eliminating the downsamplers and upsamplers of the DFB [9]. This results in a tree composed of two-channel NSFBs. The NSDFB allows the direction decomposition with $l$ stages in high-frequency images from NSPFB at each scale and produces $2^l$ directional sub-images with the same size as that of the source image. Thus the NSDFB offers NSCT with the multi-direction property and provides more precise directional information. The outputs of the first and second level filters are combined to get the directional frequency decompositions. The synthesis filter bank is obtained similarly. The NSDFBs are iterated to obtain multidirectional decomposition and to get the next level decomposition all filters are up sampled by a Quincunx Matrix (QM) given by $QM = [1\ \ 1; 1\ \ -1]$.

The NSCT is obtained by combining the NSPFB and the NSDFB as shown in Fig. 1a. The NSCT's resulting filtering structure approximates the ideal partitioning of the frequency plane displayed in Fig. 1b. Detailed description of NSCT is presented in the article by Cunha et al. [9].

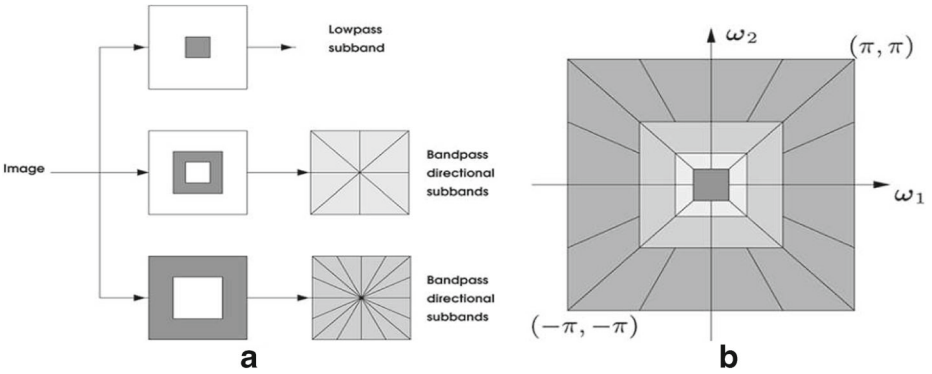


**Fig. 1** Non-subsampled contourlet transform (**a**) NSFB structure that implements the NSCT. (**b**) Idealized frequency partitioning obtained with the NSFB structure [9]

## 3 Proposed method

The proposed scheme consists of two main parts: feature representation and classification. The overall method is described by the block-diagram shown in Fig. 2. The left dotted portion indicates training phase and the right one indicates testing phase. In the training phase, for a labeled training video, frames are extracted and converted into CIE L*a*b* color space for better and uniform perceptual distribution [23]. After that features are extracted for each frame using NSCT and the dimension of the feature vectors is reduced using PCA. The

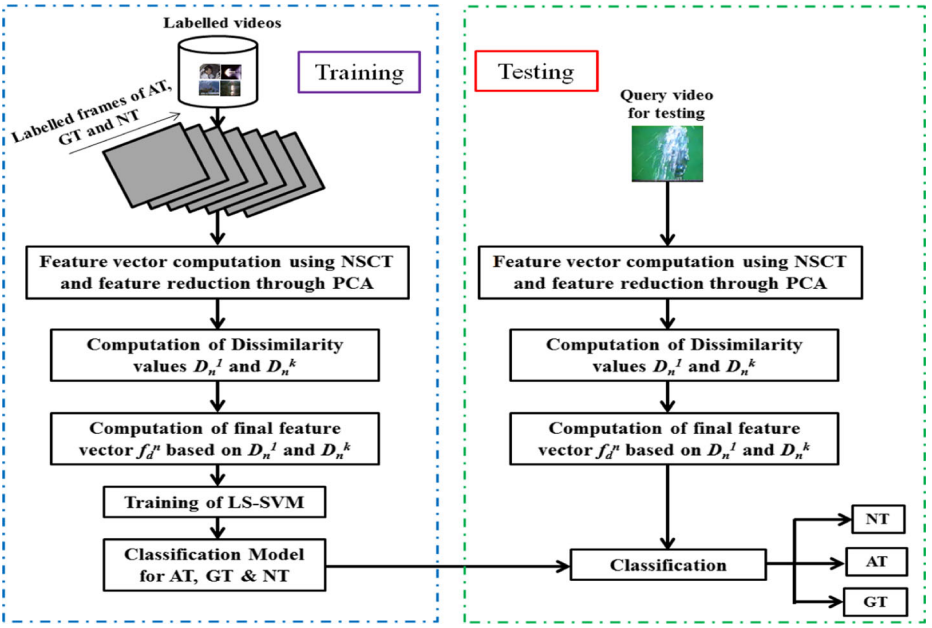


**Fig. 2** Block-diagram of the proposed method

dissimilarity values $D_n^1$ and $D_n^k$ are computed for each frame and the final feature vector $f_d^n$ is formed for each frame using these dissimilarity values in the next phase. These feature vectors are used to train the LS-SVM classifier. The above procedures are repeated on a given unknown video sequence in the testing phase for feature extraction and final feature vector generation. Finally, the frames of the unknown video sequence are classified using the trained classifier into three classes: AT, GT and NT. The proposed technique is described in detail in the following subsections.

### 3.1 Feature representation

The accuracy of a SBD system mostly depends on how effectively the visual content of video frames are extracted and represented in terms of feature vectors. The feature vector should be such that, it not only captures visual characteristics within a frame but also is able to integrate motion across video frames. One of the most important and desirable characteristics of the features used in SBD is the discriminating power between intra (within) shot and inter (between) shot frames. In view of the above facts, it is necessary to use a multiscale tool for feature extraction. Among the various multi-scale tools, NSCT has been selected due to some of its unique properties like multi-directional and shift invariancy. These properties are very important for SBD system and are not present in other multi-scale transforms. The detailed procedure of feature vector computation is shown by the block-diagram in Fig. 3.

Assuming a video sequence $V$ consists of $N$ number of frames and $f_n$ represents the $n^{th}$ frame. At first, each frame is converted into CIE L*a*b* color space. This is followed by applying NSCT on each color plane for feature extraction (with 3 level decomposition) which results in $L+1$ number of subband images denoted by $S_l^n$ having the same size as that of the original frame $f_n$, where $l = \{1, 2, ...., L+1\}$. Each subband $S_l^n$ (except the low frequency subband as it is crude low pass version of the original frame) is divided

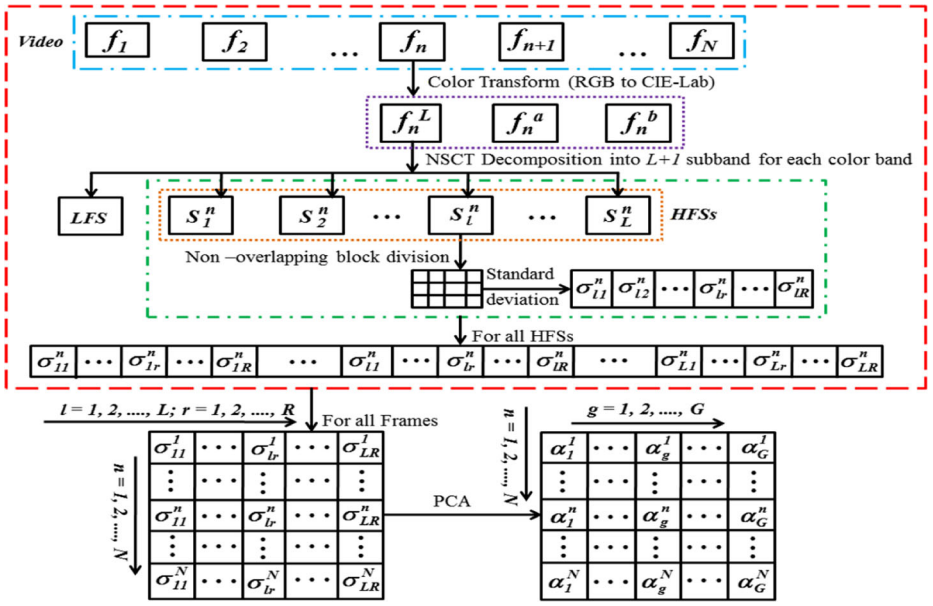


**Fig. 3** Block diagram of feature vector computation using NSCT and PCA

into $R$ number of non-overlapping blocks of size $p \times p$. Let $\{b_{lr}^n\}; l = \{1, 2..., L\}; r = \{1, 2, ..., R\}; n = \{1, 2, ..., N\}$ denotes the $r^{th}$ block of the $l^{th}$ subband. Each of these blocks is represented by the standard deviation ($\sigma$) computed from the NSCT coefficients values within these blocks. Therefore, the feature vector for each color plane of frame $f_n$ is defined as,

$$V_{f_n} = [\sigma_{11}^n, \sigma_{12}^n, ..., \sigma_{lr}^n, ..., \sigma_{LR}^n]; \tag{1}$$

where,

$$\sigma_{lr}^n = \sqrt{\frac{\sum_{x=1}^{p}\sum_{y=1}^{p}(C_{lr}^n(x, y) - \mu_{lr}^n)^2}{p \times p}}, \tag{2}$$

$C_{lr}^n(x, y)$ represents the NSCT coefficient value at $(x, y)$ of the $r^{th}$ block of the $l^{th}$ subband and $\mu_{lr}^n$ represents the mean of these coefficient values. As a result, each color plane generates a feature vector of size $L \times R$. Similar procedure is followed for other color planes of the frame $f_n$. Therefore each frame having three color planes generates a feature vector of dimension $(3 \times L \times R)$. The reason behind considering non-overlapping block subdivision of HFSs (unlike the usage of global subband's statistics used in [30]) is to reduce the disturbing impact of large object/camera motion as well as flash lighting effects. The reason is that object motion/flash light effects usually affect a part of the frame and not the entire frame. Therefore, it is expected that in these cases, local statistics of non-overlapping blocks will provide more discriminating capability.

The dimension of the feature vectors $(3 \times L \times R)$ is quite a large number. Moreover, the features generated using NSCT is highly correlated and contain some noisy features. In order to deal with these situations, proposed method uses PCA to reduce the dimension of the feature vectors. PCA is a dimensionality reduction technique which transforms the correlated features to uncorrelated ones and the features can be selected according to their variances [11]. The features with smaller variance can be omitted as they are least significant and consist of noises. The features corresponding to higher variations preserve the information of the data. Moreover, the features selected by PCA are uncorrelated. Therefore, PCA is a suitable option to deal with correlated and noisy features. Let the reduced feature matrix be denoted as $M_V = [\alpha_g^n]; g = \{1, 2, ..., G\}; n = \{1, 2, ..., N\}$ of size $(N \times G)$; where $G$ is the dimension of the reduced feature vector and $\alpha$ is a component of the reduced feature vector.

The dissimilarity value $D_n^1$ between two consecutive frames $f_n$ and $f_{n+1}$ is computed as follows:

$$D_n^1 = D(n, n+1) = \sqrt{\sum_{g=1}^{G}(\alpha_g^n - \alpha_g^{n+1})^2}; \ \forall n = 1, 2..., N-1 \tag{3}$$

Similarly, the dissimilarity value $D_n^k$ between frame $f_n$ and its following $k^{th}$ frame $f_{n+k}$ is computed as follows:

$$D_n^k = D(n, n+k) = \sqrt{\sum_{g=1}^{G}(\alpha_g^n - \alpha_g^{n+k})^2}; \ \forall n = 1, 2, ..., N-k \tag{4}$$

where $k > 1$, is the frame step [2]. The value of $k$ is chosen as $k = (t + 1)$, where $t$ is the average duration of GT computed from a large number of training videos.

A new feature vector $f_{d_n}^{AT}$ is computed for AT detection using dissimilarity values $D_n^1$ obtained from Eq. 3, over a window of size $(2w_1 + 1)$ at around each frame position and expressed as:

$$f_{d_n}^{AT} = \left[ D_{n-w_1}^1, ....., D_n^1, ....., D_{n+w_1}^1 \right] \tag{5}$$

Similarly, a new feature vector $f_{d_n}^{GT}$ is computed for GT detection using $D_n^k$ obtained from Eq. 4, over a window of size $(2w_2 + 1)$ at around each frame position and expressed as:

$$f_{d_n}^{GT} = \left[ D_{n-w_2}^k, ....., D_n^k, ....., D_{n+w_2}^k \right] \tag{6}$$

where, $w_1$ and $w_2$ are the window parameters.

The basic reason for considering a number of dissimilarity values over a group of frames for computation of $f_{d_n}^{AT}$ is to make the detection process reliable and robust [36]. For any true AT, the dissimilarity values around the transition follows a particular pattern as shown in Fig. 4a. If only one $D_n^1$ value between frame $f_n$ and $f_{n+1}$ is considered for the detection

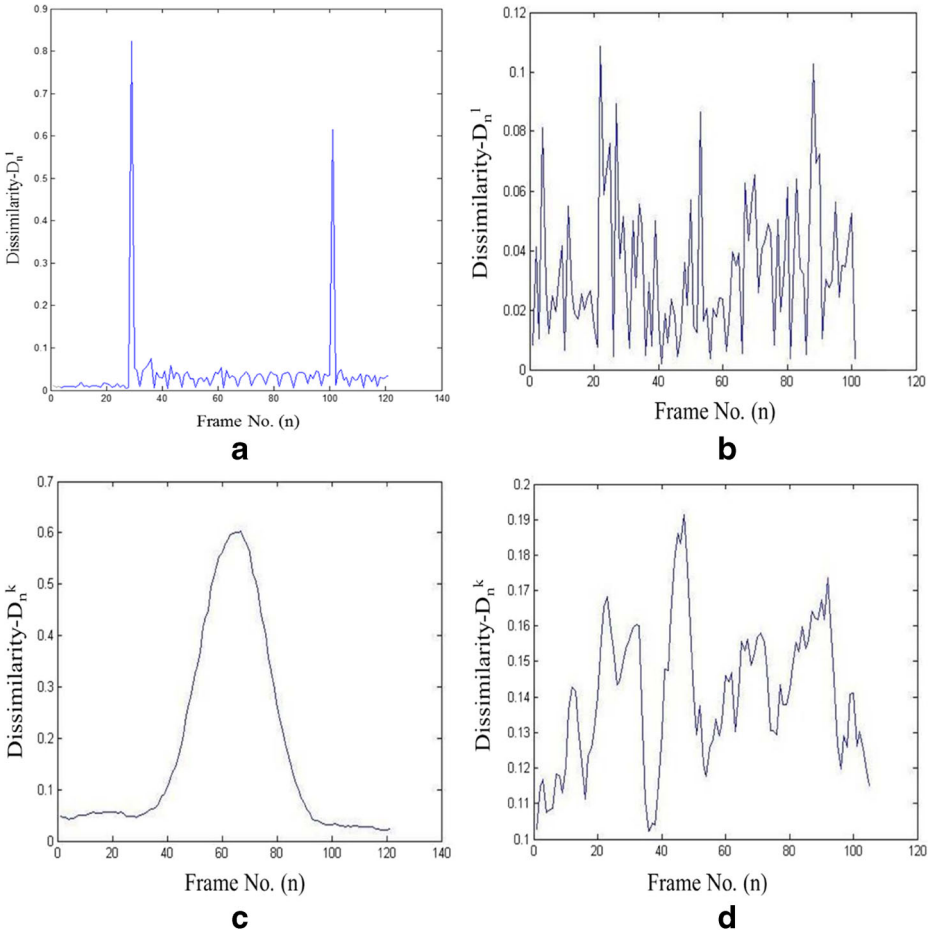


**Fig. 4** Plot of dissimilarity values ($D_n^1$ and $D_n^k$) vs Frame no. ($n$): (**a**) $D_n^1$ curve for True AT. (**b**) $D_n^1$ curve due to flashlight effects. (**c**) $D_n^k$ curve for True GT. (d) $D_n^k$ curve due to camera motion

of AT, it may lead to false detection when some spurious events like flashlights, sudden drop in intensity values are present because the magnitudes of $D_n^1$ values are often same to that generated by true AT. However, if a group of $D_n^1$ values is considered, then the pattern generated by these spurious events is not follow the pattern as shown in Fig.4a rather follows a pattern like Fig.4b. From Fig. 4a and b, it is seen that the peaks in $D_n^1$ values for genuine ATs are separated by a large number of frames, whereas for flashlight effects multiple peaks are found within a very short interval in the $D_n^1$ values. Similarly for GT, to capture the conventional pattern of $D_n^k$ values due to true GT effects as shown in Fig.4c, a collection of dissimilarity values of several frames (preceding and succeeding) are considered. This is done in order to remove false GT e.g. effects due to large camera movements and abrupt changes of object motion. Typical characteristics pattern due to camera movement is shown in Fig. 4d. Hence, from Fig. 4c and d one can easily discriminate true and pseudo GT. The window parameters $w_1$ for AT and $w_2$ for GT should be selected such that these can able to capture the above mentioned characteristics patterns for AT as well as GT and at the same time it does not merge two consecutive AT or GT.

A combined feature vector $f_d^n$ is computed for simultaneous detection of AT and GT as described by Chasanis et al. [4] and expressed as:

$$f_d^n = \left[ f_{d_n}^{AT},\ f_{d_n}^{GT} \right] \tag{7}$$

This $f_d^n$ is used as input feature vector to the LS-SVM classifier for training as well as testing.

### 3.2 Proper training set generation and classification

Even if we use an effective feature representation scheme to represent the video frames, it is not possible to achieve desired performance without a proper training feature set and an efficient classifier [4, 27]. A typical video contains very few transition (AT and GT known as positive samples) frames and a very large number of no-transition (known as negative samples) frames. Using all these frames of the video as a training set not only increases the training time but also makes the trained model biased towards the negative transitions. This is known as imbalanced training procedure and the training set is called imbalanced training set [4, 36]. Therefore, the goal is to use fewer selective no-transition frames as the negative samples. This is due to the fact that the selected negative samples have great impact on the performance of the trained model produced by the classifier. The important negative samples are actually those which belong nearest to the class separating hyperplane i.e. those frames which lie nearest to the decision boundary regions. The conventionally used random selection of negative samples often leads to improper trained model and results in poor performance. Because, if we randomly select negative samples for training - then there is no certainty that the chosen negative samples lie nearest to the decision boundary. To overcome from this problem, many researchers have used a thresholding technique to select the negative sample frames. However, selecting a proper threshold is challenging. A high threshold might select some of the positive samples (specifically the gradual transition's frames) as the no-transition frames, whereas a low threshold increases the number of negative samples which lie far apart from the decision boundary regions. In the proposed SBD framework, apart from selecting the positive samples (obtained from the available ground truth data) we have also selected those negative samples whose characteristics patterns are similar to that of the actual transitions. To do this automatically, we have used two different threshold values $T_1$ and $T_2$. Considering the two different window parameters $w_1$ and $w_2$ mentioned

in the Eqs. 5 and 7 for two different kinds of frame transitions AT and GT, $T_1$ is used for finding the highest peak of the difference pattern and $T_2$ is used for finding the second highest peak. The magnitude of these two peaks for actual transitions are quite different and in the proposed method we have set $T_1 = 0.5$ and $T_2 = \frac{magnitudeHighestPeak}{magnitude2ndHighestPeak} = 1.67$. These values are set empirically after extensive experiments. This makes the highest peak ($T_1$) as the middle value (approximately) of the actual transition (AT or GT) characteristics pattern and $T_2$ belonging to nearby no-transitions frame's pattern. We have exploited these two characteristics to select the negative samples. Thus the volume of the training set is largely reduced which solves the imbalance problem as well as reduces training time.

The other critical aspect of SBD is the evaluation of computed dissimilarity values since the final output of the SBD algorithms largely depends on this evaluation method. Earlier works for SBD mainly depend upon hard thresholds which are selected experimentally. The first effort to make it automatic was done by Zhang et al. in [37], where they proposed two different thresholds for the detection of AT and GT, respectively. These thresholds have a major drawback that these cannot incorporate contextual information (the dissimilarity values around the current frame under consideration) and therefore lead to many false detections as well as missed detections. The reason is that the transition is a local phenomenon and there exists a clear relationship among the frames corresponding to a transition event and the frames closely surrounded to it [2], which are shown in Fig. 4a and c, respectively. A better alternative is to use adaptive thresholding that considers the local information. However, it still suffers from some parameters chosen experimentally which basically controls the false detection rate and these parameters usually vary from video to video [14]. Recently the development of Machine Learning (ML) algorithms have shown vast improvement in the SBD system. The reason for the success of ML algorithms is that they make decisions via the recognition of the patterns that different types of shot boundary generates, instead of the evaluation of the magnitudes of content variations [36]. They can perform reliably on any unknown video sequence once the parameters are set by proper training mechanism. Various types of ML tools are successfully employed by various researchers such as K-Nearest Neighbor, Multilayer Perceptron, Support Vector Machine etc. for SBD [27, 31, 36]. SVM is considered as one of the most successful ML algorithm for classification of different unknown data patterns. It has solid theoretical foundations as well as different successful applications [36]. The recent study of Smeaton et al. has shown that most of the top performing SBD methods have used SVM as the ML tool, indicating it is well suited for SBD [31].

However, the major drawback of SVM is its high computational complexity for data sets of large dimension. To reduce the computational cost, a modified version called Least Square SVM (LS-SVM) is adopted as a classifier in this paper. The LS-SVM does not require solving quadratic programming problems and simplifies the training procedure by solving a set of linear equations [32].

LS-SVM is originally developed for binary classification problems. A number of methods have been proposed by various researchers for extension of binary classification problem to multi-class classification problem. It's essentially separate $M$ mutually exclusive classes by solving many two-class problems and combining their predictions in various ways. One such technique which is commonly used is Pair-Wise Coupling (PWC) or One-vs.-One is to construct binary SVMs between all possible pairs of classes. PWC uses $M * (M - 1)/2$ binary classifiers for M number of classes, each of these classifiers provide a partial decision for classifying a data point. During the testing of a feature, each of the $M * (M - 1)/2$ classifiers vote for one class. The winning class is the one with the largest

number of accumulated votes. Hsu et al. [15], has shown that the PWC method is more suitable for practical use than the other methods discussed.

## 4 Experimental results and discussion

The effectiveness of the proposed algorithm is evaluated over several benchmarking videos on the standard performance measures and compared against several state-of-the-art techniques. The details of the dataset, experimental setup and results are described in the following subsections.

### 4.1 Description of datasets and evaluation criteria

Separate training and test set videos are used to test the effectiveness of our proposed SBD system. Some of the videos (totaling 47,186 frames) from http://www.open-video.org/ are taken as the initial training data which do not belong to the TRECVID-2001 test set. After applying the proposed active learning strategy approximately 12.32% of those initial frames are selected as the final training set. The description of the training data is given in the Table 1. We have tested the performance of our proposed method on TRECVID 2007 and TRECVID 2001 benchmark video datasets obtained from http://www.open-video.org/ [31, 33]. Both the dataset include large number of ATs and GTs along with different types of disturbances (events) like abrupt movement of objects, sudden appearance of objects in front of camera, large camera motions as well as flashlight effects. The descriptions of the videos in these dataset are given in Table 2 and in 3. These videos are chosen for comparison because it has been widely used by various researchers due to the presence of different types of true and pseudo transitions (disturbances) in it. Each frame of the videos 'anni005', 'anni009' is of size 240 × 320 and that for rest of the videos is 240 × 352 for TRECVID 2001 dataset, whereas the resolution of frames of all the videos in TRECVID 2007 dataset is of 288 × 352. For uniformity of testing, all the video frames are resized into 128 × 128 in the proposed method. We have implemented the proposed technique in MATLAB, and experiments have been carried out on a PC with 3.40 GHz CPU and 8 GB RAM.

The performance of the proposed method is compared with the state-of-the-art techniques using the following standard quantitative measures:

$$Precision\ (P) = \frac{N_c}{N_c + N_f} \tag{8}$$

$$Recall\ (R) = \frac{N_c}{N_c + N_m} \tag{9}$$

where $N_c$ denotes the number of correct detections, $N_m$ denotes the number of missed detections and $N_f$ denotes the number of false detections ~~[6]~~. F-Measure can be defined in the following way:

$$F - Measure\ (F1) = \frac{2 \times P \times R}{P + R} \tag{10}$$

**Table 1** Details of the training dataset

| Description | # NT (no-transitions) | # AT | # GT |
|---|---|---|---|
| Total | 45,300 | 352 | 115 |
| Used for training | 4,072 | 308 | 97 |

**Table 2** Details of the TRECVID 2001 SBD dataset

| File | Duration (mm:ss) | # Frames | # AT | # GT |
|---|---|---|---|---|
| anni005 | 6:19 | 11,363 | 38 | 27 |
| anni009 | 6:50 | 12,306 | 38 | 65 |
| nad31 | 29:08 | 52,395 | 187 | 55 |
| nad33 | 27:39 | 49,734 | 189 | 26 |
| nad53 | 14:31 | 26,115 | 83 | 75 |
| nad57 | 6:57 | 12,510 | 45 | 31 |
| bor03 | 26:56 | 48,450 | 231 | 11 |
| bor08 | 28:07 | 50,568 | 380 | 151 |
| **Total** | **144:67** | **2,63,441** | **1191** | **441** |

The value of recall reflects the rate of miss-classification i.e., higher the value of recall, lower the rate of misclassification. On the other hand, precision reflects the rate of false positives; lower the rate of false positives, higher the precision. F-measure is the harmonic mean of recall and precision. The value of F-measure is high only when both the recall and precision are high i.e. only when the miss-classification rates as well as rate of false positives both are low.

## 4.2 Parameters selection

In the proposed technique, 3 level $[0, 1, 2]$ decomposition of NSCT is used, whereas 'pyrexc' and 'pkva' are selected as the pyramidal filter and directional filter respectively.

**Table 3** Details of the TRECVID 2007 SBD dataset

| Video Name | Duration (mm:ss) | # Frames | # AT | # GT |
|---|---|---|---|---|
| BG2408 | 23:55 | 35,892 | 101 | 20 |
| BG9401 | 33:22 | 50,049 | 89 | 3 |
| BG11362 | 10:56 | 16,416 | 104 | 4 |
| BG14213 | 55:24 | 83,115 | 106 | 61 |
| BG34901 | 22:55 | 34,389 | 224 | 16 |
| BG35050 | 24:39 | 36,999 | 98 | 4 |
| BG35187 | 19:20 | 29,025 | 135 | 23 |
| BG36028 | 29:59 | 44,991 | 87 | – |
| BG36182 | 19:44 | 29,610 | 96 | 13 |
| BG36506 | 10:08 | 15,210 | 77 | 6 |
| BG36537 | 33:20 | 50,004 | 259 | 30 |
| BG36628 | 37:42 | 56,564 | 192 | 10 |
| BG37359 | 19:16 | 28,908 | 164 | 6 |
| BG37417 | 15:20 | 23,004 | 76 | 12 |
| BG37822 | 14:38 | 21,960 | 119 | 10 |
| BG37879 | 19:20 | 29,019 | 95 | 4 |
| BG38150 | 35:05 | 52,650 | 215 | 4 |
| **Total** | **421:43** | **6,37,805** | **2,237** | **226** |

This NSCT decomposition leads to one LFS and seven HFSs, for each color plane of a frame. Each such HFS is decomposed into blocks of size 16 × 16, which results in 64 ($\frac{128\times128}{16\times16}$) number of blocks. The choice of block size is not a well defined problem. Larger block size will lead to poor representation and smaller block size increases computational complexity. Generally a trade-off is made and block-size of 16 × 16 is selected in the proposed method as a reasonable choice. The standard deviation ($\sigma$) is computed from the coefficient values within each block, which is used as a component of the feature vector. Therefore, each frame generates a feature vector of dimension 1344 (64 × 7 × 3). Even a feature vector of size 1344 is quite large. The computation of distance between two frames based on the feature vectors of this size is quite expensive. Moreover, all components of feature vectors may not carry significant information about the contents of the frame. Therefore, to achieve better efficiency of the system, reduction of the dimension of the feature vectors is desired which is achieved using PCA. We did several experiments to evaluate the effectiveness of the usage of PCA over non-usage of PCA in our proposed scheme considering the videos of TRECVID 2001 dataset. The visual results for AT as well as GT correctly detected by the proposed technique on TRECVID 2001 data set is shown in Fig. 5. Conventionally, the importance of the used features is ranked according to the number of principal components (NPC). In these experiments, considering only abrupt transitions, we used different NPC which were selected by using variance information on accuracy (F-measure) and the result is shown in the Fig. 6. It can be clearly seen from the graph of the Fig. 6 that

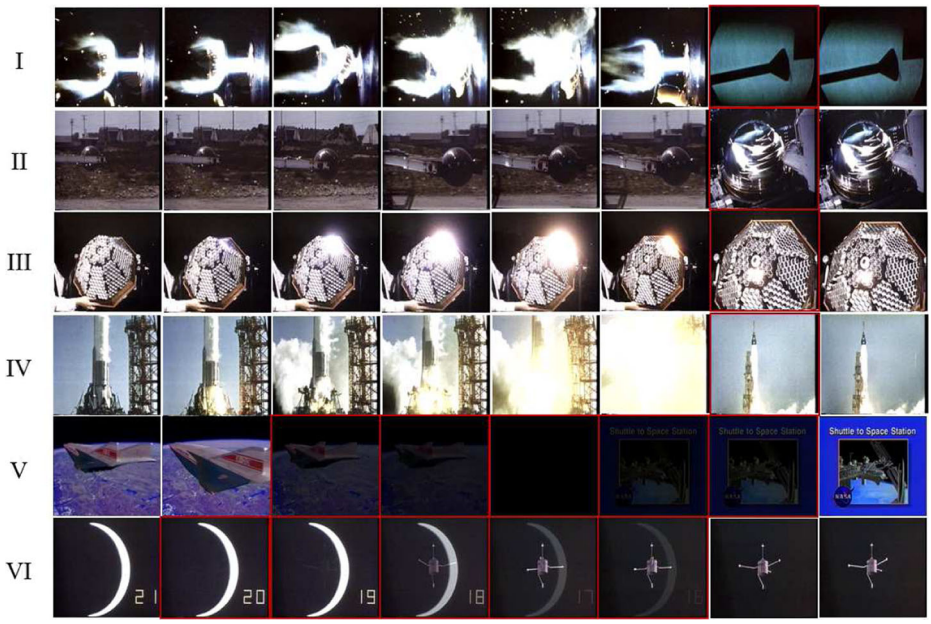


**Fig. 5** Results of correctly detected transition frames by our algorithm: (All the transition frames are marked by red boxes.)(**a**) Row I-IV are the results for cut detection Row-I: The detection of AT in presence of large object motion. Row-II: the detection of AT in presence of camera motion (camera pan) Row-III: First row: in the presence of large object motion (ii) second row: in the presence of large camera pan (iii) Third and fourth row: in the presence of flashlight and other effects. (**b**) Fifth row: a fade in and fade out detection (d) and the last row show a dissolve detection. The detected frames are marked by red boxes. The visual results given are on TRECVID 2001 datasets

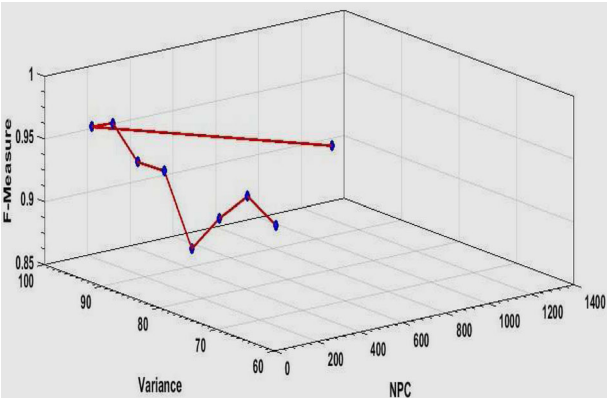


**Fig. 6** Performance comparison PCA vs. non-PCA based feature representation

considering a reduced feature vector (using PCA) over the full-dimensional feature vector (non-PCA), the proposed system is providing better result in terms of F-measure. Moreover, a lower-dimensional feature vector (approximately 96% dimension reduction over the full-dimension 1344) also helps to achieve computational efficiency. A total of first 50 components from the ranked principal components are selected which preserve 90% of the total variance information of NSCT decomposed features. The other tunable parameters are the frame step $k$, average durations of GT $t$, and window parameters $w_1$ for AT and $w_2$ for GT. The frame step $k$ is selected as $k = t + 1$, where t is the average duration of GT. From a large collection of ~~training~~ videos taken from various sources, including many videos from TRECVID 2001 dataset which are not used as a test video, it is found that the duration of GT varies from 15 to 30 frames. Hence the value of $k$ is selected as 24 in the proposed technique. The window parameters $w_1$ for AT is set as 10 and $w_2$ for GT is set as 30. The very lower values of $w_1$ and $w_2$ cannot capture the characteristic patterns for AT and GT as shown in Fig. 4a and c, whereas large values of $w_1$ and $w_2$ will merge two successive ATs and GTs respectively.

To make the LS-SVM classifier more reliable and generalized, $5 \times 5$ fold stratified Cross Validations (CV) are employed. We have used the Radial Basis Function (RBF): $K(x_i, x_j) = exp(-\gamma||x_i - xj||^2), \gamma > 0$, as the kernel. There are two tunable parameters while using the RBF kernel: $C$ and $\gamma$. The kernel parameter $\gamma$ controls the shape of the kernel and regularization parameter $C$ controls the trade-off between margin maximization and error minimization. It is not known beforehand which values of $C$ and $\gamma$ are the best for the classification problem at hand. Hence various pairs of $(C, \gamma)$ were tried with, over the course of CV procedure, and the one with the lowest CV error rate was picked. After obtaining best values of the parameters $C$ and $\gamma$, these values were used to train the LS-SVM model, and the test set was used to measure the error rate of the classification. The LS-SVM toolbox 'LS-SVMlab v1.8' is used for implementation of the classifier in the proposed method [3]. The training set is manually constructed using videos of different genres such as documentary, sports, news and movies etc. The transitions are manually annotated as positive examples while negative examples are selected as the frames corresponding to disturbing effects such as large object movements, camera motions as well as flashlight effects. The same model is used to test both the datasets, i.e. TRECVID 2001 dataset as well as TRECVID 2007 dataset.

### 4.3 Comparison with related works

We have performed several experiments to validate the effectiveness of the proposed technique. Tables 4 and 5 contain the results of the performance comparison with several other techniques on TRECVID 2001 data set. Evaluation of the proposed technique on TRECVID 2007 data set is given in Table 6. Table 7 contains the comparison results with other existing techniques on TRECVID 2007 data set. To support our choice of NSCT over other existing MGA tools like CVT and CNT, we have conducted several other experiments. The findings of the experiments are tabulated in the Table 8. Even though, the Table 8 contains results of only 4 videos (2 from TRECVID 2001 and the other 2 from TRECVID 2007 dataset), we have got similar results for the other videos of the datasets. To make the comparison fair, the setups of the experiments (frame's size, filters, decomposition levels etc.) are remained similar for all the above-mentioned transform based features. The comparison results are given only on F-measure value. It is evident from the Table 8 that NSCT based feature representation performs significantly better than that of CVT and CNT based feature representations considering both normal and complex video scenes.

The proposed method is also compared with some state-of-the-art SBD techniques to determine its effectiveness and the results are reported in Tables 4, 5 and 7. Tables 4 and 5 compare the results on $TRECVID2001$ dataset whereas Table 7 represents the comparison on $TRECVID2007$ dataset with the top performer of the 2007 SBD task as well as with one state-of-the art technique Lakshmipriya et al. [19]. Table 4 shows the results for AT and Table 5 demonstrates the performance for GT. The best results for each category are marked as bold-faces. From Table 4, it is seen that the proposed method performs much superior on an average for AT detection than the other methods in terms of both recall and precision, resulted in overall average F-measure value of 0.969 where the next best average F-measure value is 0.928. These high performances for AT detection follows our expectation where the system able to discriminate correctly the pattern for AT as shown in Fig. 4a and that of non-AT as shown in Fig. 4b. Similarly for GT, the system able to correctly discriminate the patterns for GT as shown in Fig. 4c and non-GT as shown in Fig. 4d. From Table 5, it is observed that the F-measure value of the proposed method for most of the videos are higher than that of the other methods for GT detection, although the proposed technique does not result best performances in terms of precession and recall. The only case where the proposed method performs superior in respect to all the three measures is for the video sequence 'bor08'. The other methods perform superior than the proposed technique only in terms of either precision or recall but not for both. In fact, it is seen from Table 4 that those methods whose performance in terms of recall is superior, their performance in terms of precision is poor and vice versa. In other words, the existing techniques either give good accuracy to the cost of much higher false detection rate or very low false detection rate to the cost of very low accuracy. Only, the method proposed by Choudhury et al. [5] performs well in terms of both recall and precision. The recall of this method is superior for the video sequences 'anni005', 'nad31', 'nad53', 'nad57', 'bor03' and the average respectively. However they do not perform superior in terms of precision or F-measure for any one of the video sequences. Whereas the proposed method performs equally well in terms of both recall and precision which results in higher F-measure for most of the video sequences. Only case where the proposed technique lags in F-measure is for the video sequence 'nad33'. Visual results shown in Fig. 5 shows that the proposed technique is really robust because in presence of various kinds of disturbances the system could correctly identify the positions of transitions i.e., the transition frame for AT and a collection of

**Table 4** Performance comparison of Various Shot Boundary Detection System for AT on TRECVID 2001 dataset

| File | Proposed Method | | | Choudhury's Method [5] | | | Gianluigi's Method [13] | | | Ma's Method [24] | | | Li's Method [21] | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R | P | F1 | R | P | F1 | R | P | F1 | R | P | F1 | R | P | F1 |
| anni005 | **1.000** | **0.975** | **0.987** | **1.000** | 0.826 | 0.905 | 0.868 | 0.943 | 0.904 | 0.973 | 0.948 | 0.960 | **1.000** | 0.736 | 0.848 |
| anni009 | 0.974 | **0.974** | **0.974** | **1.000** | 0.760 | 0.864 | 0.947 | 0.857 | 0.899 | 0.842 | 0.888 | 0.864 | **1.000** | 0.655 | 0.791 |
| nad31 | 0.935 | 0.972 | **0.956** | **0.963** | 0.818 | 0.884 | 0.043 | **1.000** | 0.082 | 0.912 | 0.938 | 0.925 | 0.901 | 0.945 | 0.922 |
| nad33 | **0.994** | 0.954 | 0.973 | 0.947 | 0.861 | 0.902 | 0.910 | 0.950 | 0.930 | 0.989 | 0.963 | **0.976** | 0.984 | **0.964** | 0.974 |
| nad53 | **1.000** | **1.000** | **1.000** | 0.988 | 0.802 | 0.885 | 0.842 | 0.920 | 0.879 | 0.975 | 0.975 | 0.975 | 0.952 | 0.941 | 0.946 |
| nad57 | **1.000** | 0.883 | 0.938 | 0.977 | 0.915 | 0.945 | 0.909 | 0.909 | 0.909 | 0.772 | **0.971** | 0.860 | 0.958 | 0.939 | **0.948** |
| bor03 | **0.991** | **0.962** | **0.976** | 0.978 | 0.904 | 0.939 | 0.627 | 0.923 | 0.747 | 0.938 | 0.954 | 0.946 | 0.961 | 0.949 | 0.955 |
| bor08 | 0.931 | 0.972 | **0.951** | 0.934 | 0.868 | 0.900 | 0.497 | **0.990** | 0.662 | 0.965 | 0.882 | 0.922 | **0.973** | 0.901 | 0.949 |
| average | **0.978** | **0.961** | **0.969** | 0.973 | 0.844 | 0.903 | 0.705 | 0.940 | 0.752 | 0.921 | 0.940 | 0.928 | 0.966 | 0.879 | 0.917 |

**Table 5** Performance comparison of Various Shot Boundary Detection System for GT on TRECVID 2001 dataset

| File | Proposed Method | | | Choudhury's Method [5] | | | Gianluigi's Method [13] | | | Ma's Method [24] | | | Li's Method [21] | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R | P | F1 | R | P | F1 | R | P | F1 | R | P | F1 | R | P | F1 |
| anni005 | 0.852 | **0.951** | **0.899** | **0.888** | 0.828 | 0.857 | 0.000 | 0.000 | 0.000 | 0.666 | 0.782 | 0.719 | 0.786 | 0.880 | 0.839 |
| anni009 | **0.963** | 0.893 | **0.927** | 0.907 | 0.881 | 0.894 | 0.046 | 0.750 | 0.087 | 0.507 | 0.733 | 0.599 | 0.848 | **0.926** | 0.885 |
| nad31 | 0.741 | **0.884** | **0.806** | **0.818** | 0.789 | 0.803 | 0.000 | 0.000 | 0.000 | 0.535 | 0.428 | 0.476 | 0.708 | 0.687 | 0.697 |
| nad33 | 0.794 | 0.822 | 0.778 | 0.923 | 0.800 | 0.857 | 0.231 | **0.857** | 0.364 | 0.692 | 0.382 | 0.492 | **0.943** | 0.805 | **0.868** |
| nad53 | 0.892 | **1.000** | **0.943** | **0.970** | 0.913 | 0.942 | 0.040 | 0.750 | 0.076 | 0.805 | 0.696 | 0.746 | 0.826 | 0.947 | 0.882 |
| nad57 | 0.925 | 0.951 | **0.938** | **0.956** | 0.846 | 0.898 | 0.087 | **1.000** | 0.160 | 0.826 | 0.593 | 0.690 | 0.852 | 0.885 | 0.868 |
| bor03 | **1.000** | 0.855 | **0.922** | **1.000** | 0.688 | 0.815 | 0.182 | **1.000** | 0.308 | 0.818 | 0.281 | 0.418 | 0.929 | 0.382 | 0.541 |
| bor08 | **0.973** | **0.943** | **0.958** | 0.960 | 0.913 | 0.936 | 0.007 | 0.500 | 0.014 | 0.758 | 0.816 | 0.786 | 0.716 | 0.930 | 0.809 |
| average | 0.892 | **0.920** | **0.896** | **0.923** | 0.832 | 0.875 | 0.074 | 0.482 | 0.126 | 0.701 | 0.589 | 0.616 | 0.826 | 0.805 | 0.815 |

**Table 6** Performance evaluation of the proposed method on TRECVID 2007 SBD task dataset

| Dataset | Abrupt Transition | | | Gradual Transition | | | Overall Transition | | |
|---|---|---|---|---|---|---|---|---|---|
| | R | P | F1 | R | P | F1 | R | P | F1 |
| BG2408 | 1.000 | 0.971 | 0.985 | 0.712 | 1.000 | 0.832 | 0.950 | 0.975 | 0.962 |
| BG9401 | 0.988 | 0.978 | 0.983 | 1.000 | 1.000 | 1.000 | 0.989 | 0.978 | 0.984 |
| BG11362 | 0.939 | 0.970 | 0.954 | 0.800 | 0.800 | 0.800 | 0.935 | 0.962 | 0.948 |
| BG14213 | 0.991 | 0.850 | 0.915 | 0.879 | 0.908 | 0.893 | 0.952 | 0.869 | 0.909 |
| BG34901 | 0.991 | 0.982 | 0.987 | 0.823 | 0.877 | 0.849 | 0.979 | 0.975 | 0.977 |
| BG35050 | 1.000 | 0.942 | 0.971 | 0.700 | 1.000 | 0.824 | 0.990 | 0.944 | 0.967 |
| BG35187 | 0.940 | 0.928 | 0.934 | 0.750 | 0.885 | 0.812 | 0.911 | 0.923 | 0.917 |
| BG36028 | 0.989 | 0.956 | 0.972 | – | – | – | 0.989 | 0.956 | 0.972 |
| BG36182 | 0.926 | 0.978 | 0.951 | 0.867 | 0.933 | 0.899 | 0.917 | 0.971 | 0.943 |
| BG36506 | 0.987 | 0.987 | 0.987 | 0.600 | 0.800 | 0.700 | 0.964 | 0.976 | 0.970 |
| BG36537 | 0.951 | 0.975 | 0.963 | 0.850 | 0.920 | 0.884 | 0.941 | 0.971 | 0.956 |
| BG36628 | 0.984 | 0.970 | 0.977 | 0.850 | 0.810 | 0.830 | 0.980 | 0.961 | 0.971 |
| BG37359 | 0.976 | 1.000 | 0.988 | 0.600 | 0.800 | 0.700 | 0.965 | 0.994 | 0.979 |
| BG37417 | 0.987 | 0.935 | 0.960 | 0.750 | 0.830 | 0.788 | 0.955 | 0.923 | 0.939 |
| BG37822 | 0.983 | 0.936 | 0.959 | 0.800 | 0.900 | 0.850 | 0.969 | 0.933 | 0.951 |
| BG37879 | 1.000 | 1.000 | 1.000 | 0.770 | 0.900 | 0.830 | 0.990 | 1.000 | 0.995 |
| BG38150 | 0.976 | 0.981 | 0.979 | 0.600 | 0.700 | 0.650 | 0.968 | 0.977 | 0.973 |
| Average | 0.982 | 0.984 | 0.983 | 0.772 | 0.870 | 0.818 | 0.970 | 0.980 | 0.975 |

transition frames for GT. In Table 6, the performances of the the proposed technique for AT, GT and overall transitions are presented. From this table, it is seen that the average F1 value for AT is 0.983 and for GT is 0.818. The low F value for GT is due to the fact of very

**Table 7** Performance comparison of the proposed method with the top results of TRECVID 2007 SBD task

| Dataset | Abrupt Transition | | | Gradual Transition | | | Overall Transition | | |
|---|---|---|---|---|---|---|---|---|---|
| | R | P | F1 | R | P | F1 | R | P | F1 |
| Proposed | 0.982 | 0.984 | 0.983 | 0.772 | 0.870 | 0.818 | 0.970 | 0.980 | 0.975 |
| Lakshmipriya et. al.[19] | 0.972 | 0.976 | 0.974 | 0.869 | 0.719 | 0.780 | 0.965 | 0.957 | 0.961 |
| AT T run5[22] | 0.979 | 0.966 | 0.972 | 0.709 | 0.802 | 0.753 | 0.956 | 0.954 | 0.955 |
| AT T run3[22] | 0.977 | 0.968 | 0.972 | 0.704 | 0.780 | 0.740 | 0.955 | 0.953 | 0.954 |
| THU11[35] | 0.968 | 0.982 | 0.975 | 0.718 | 0.733 | 0.725 | 0.947 | 0.962 | 0.954 |
| THU05[35] | 0.968 | 0.982 | 0.975 | 0.743 | 0.695 | 0.718 | 0.949 | 0.956 | 0.952 |
| BRAD[29] | 0.973 | 0.982 | 0.977 | 0.587 | 0.425 | 0.493 | 0.941 | 0.919 | 0.929 |
| NHK2[17] | 0.933 | 0.965 | 0.965 | 0.607 | 0.691 | 0.646 | 0.905 | 0.944 | 0.924 |
| NHK3[17] | 0.916 | 0.975 | 0.945 | 0.578 | 0.768 | 0.660 | 0.923 | 0.960 | 0.916 |
| Marburg1[26] | 0.945 | 0.942 | 0.944 | 0.766 | 0.595 | 0.670 | 0.931 | 0.907 | 0.919 |
| Marburg2[26] | 0.957 | 0.930 | 0.943 | 0.777 | 0.570 | 0.658 | 0.942 | 0.893 | 0.917 |

**Table 8** Performance comparison of various MGA tools

| Dataset | File | AT (F-measure) | | | GT (F-measure) | | |
|---|---|---|---|---|---|---|---|
| | | CVT | CNT | NSCT | CVT | CNT | NSCT |
| **TRECVID 2001** | anni005 | 0.923 | 0.885 | **0.987** | 0.863 | 0.842 | **0.899** |
| | anni009 | 0.945 | 0.861 | **0.974** | 0.837 | 0.781 | **0.927** |
| **TRECVID 2007** | BG11362 | 0.912 | 0.892 | **0.954** | 0.763 | 0.652 | **0.800** |
| | BG36506 | 0.937 | 0.924 | **0.987** | 0.615 | 0.623 | **0.700** |

few number of GTs presents in these videos and their duration is of varying nature which results in inconsistent patterns in the $f_{d_n}^{GT}$ values as well as $f_d^n$ values. However, the F value for overall transition is 0.975 which shows the excellent performance of the proposed technique.

We have also compared the performance of the proposed technique with the best techniques reported on TRECVID 2007 dataset and the results are reported in Table 7. From Table 7, it is seen that the proposed technique performs superior for AT detection in terms of all three measures R, P and F values than the other methods. The best R value for GT detection is of Lakshmipriya et al.'s method [19] which is 0.869 whereas the R value of proposed technique is 0.772. However, the P value of the proposed method is 0.870 which is much superior than the other methods. The other best P value is 0.802 of AT T run5. Therefore our F value for GT is also much higher than the other methods. Thus, the proposed technique performs superior for overall transitions.

We have also compared the computational efficiency of the proposed technique with some of the other state-of-the-art methods. It is to be noted that the time requirement of a SBD scheme mainly depends on the complexity of the frame-content representation (feature extraction) step. Therefore, only the time requirement of the feature extraction step is considered for this performance comparison. To make a fair comparison we have run all the compared SBD schemes on the same machine configuration. The detail of the comparison is tabulated in the Table 9.

It is clear from the results given in the Table 9 that the proposed scheme is computationally more expensive than the state-of-the-art SBD scheme described in [19]. But, from the results given in the Table 7, it is evident that the proposed scheme is more accurate than the method of [19]. At the same time the proposed method is not only computationally much more efficient than the NSCT-based SBD technique proposed in [30], but also performs significantly superiorly. It is also to be noted that after the submission of a video frame to the

**Table 9** Comparison of computation time on TRECVID 2007 dataset

| Scheme | Computation Time (sec) | AT | | | GT | | |
|---|---|---|---|---|---|---|---|
| | | P | R | F | P | R | F |
| **Proposed** | 0.1240 | **0.984** | **0.982** | **0.983** | **0.870** | 0.772 | **0.818** |
| **Lakshmi Priya et al.** [19] | **0.0781** | 0.976 | 0.972 | 0.974 | 0.719 | **0.869** | 0.780 |
| **Sasithradevi et al.** [30] | 34.5469 | 0.865 | 0.785 | 0.823 | – | – | – |

proposed SBD system, it provides response in approximately 0.2 sec. (feature extraction + classification), which is acceptable for offline automatic video-processing and analysis.

## 5 Conclusions and future work

In this paper a new SBD technique is presented for the detection of both AT and GT following the general SBD framework. The method is able to detect accurately both types of transitions AT as well as GT, even in the presence of different disturbing effects like flashlights, abrupt movement of objects and camera motions. The features from the video frames are extracted using NSCT which has unique properties like multi-scale, multi-directional, and shift invariancy. Thus the extracted features are invariant to the disturbing effects like flashlights, abrupt movement of objects and motion of camera. Furthermore, the dimensionality of the feature vectors is reduced through PCA to achieve computational efficiency as well as to reduce the noisy features. The contextual information of the video frames, i.e., the dissimilarity values around the current frame is taken under consideration to improve the accuracy as well as to reduce the rate of false detection. Finally, cost efficient LS-SVM classifier is adopted to classify the frames of a given video sequence into AT, GT and NT classes. A novel efficient method of training set generation is also proposed which not only reduces the training time but also improves the performance. Experimental results on TRECVID 2001 and 2007 dataset and comparison with state-of-the-art SBD methods show the effectiveness of the proposed technique.

In future, we will use the NSCT based feature vectors for the detection of different types of camera motions such as panning, zooming, tilting etc. and the flashlight effects. The detection of these effects is important for the further improvement of SBD techniques as these are the most disturbing effects for SBD as well as for further analysis of a video sequence as these effects bear important information about a video.

**Acknowledgements** The first author acknowledges Tata Consultancy Services (TCS) for providing fellowship to carry out the research work. Malay K. Kundu acknowledges the Indian National Academy of Engineering (INAE) for their support through INAE Distinguished Professor fellowship. The authors would like to thank the National Institute of Standards & Technology (NIST) for providing TRECVID data set.

## References

1. Arman F, Hsu A, Chiu MY (1994) Image Processing on encoded video sequences. Multimedia Syst 1(5):211–219
2. Bescos J, Cisneros G, Martinez JM, Menendez JM, Cabrera J (2005) A unified model for techniques on video-shot transition detection. IEEE Trans Multimedia 7(2):293–307
3. Brabanter KD, Karsmakers P, Ojeda F, Alzate C, Brabanter JD, Pelckmans K, Moor BD, Vandewalle J, Suykens JAK (2011) LS-SVMlab Toolbox Users Guide version 1.8, ESAT-SISTA Technical Report 10-146 pp 1–115
4. Chasanis V, Likas A, Galatsanos N (2009) Simultaneous detection of abrupt cuts and dissolves in videos using support vector machines. Pattern Recogn Lett 30(2009):55–65
5. Choudhury A, Medioni G (2012) A framework for robust online video contrast enhancement using modularity optimization. IEEE Trans Circuits Syst Video Technol 22(9):1266–1279
6. Chowdhury M, Kundu MK (2014) Comparative assessment of efficiency for content based image retrieval systems using different wavelet features and pre-classifier. Multimedia Tools and Applications 72(3):1–36

7. Chua T-S, Feng H, Chandrashekhara A (2003) An unified framework for shot boundary detection via active learning. In: Proceedings Int. Conf. Acoust. Speech Signal Proces, pp 845–848
8. Cooper M, Liu T, Rieffel E (2007) Video segmentation via temporal pattern classification. IEEE Trans Multimedia 9(3):610–618
9. da Cunha AL, Zhou J, Do MN (2006) The nonsubsampled contourlet transform: theory, Design, and Applications. IEEE Trans Image Process 15:3089–3101
10. Do MN, Vetterli M (2005) The Contourlet Transform: an efficient directional multiresolution image representation. IEEE Trans Image Process 14(12):2091–2106
11. Duda RO, Hart PE, David G (2012) Pattern classification, John Wiley & Sons
12. Garcia-Perez AM (1992) The perceived image: Efficient modelling of visual inhomogeneity. Spat Vis 6(2):89–99
13. Gianluigi C, Raimondo S (2006) An innovative algorithm for key frame extraction in video summarization. J Real-Time Image Proc 1(1):69–88
14. Hanjalic A (2002) Shot-boundary detection: unraveled and resolved? IEEE Trans Circuits Syst Video Technol 12(2):90–105
15. Hsu CW, Lin CJ (2002) A comparison of methods for multi-class support vector machines. IEEE Trans Neural Netw 13(2):415–425
16. Jang H (2006) Gradual shot boundary detection using localized edge blocks, vol 28
17. Kawai Y, Sumiyoshi H, Yagi N (2007) Shot boundary detection at TRECVID 2007. In: Proceedings TREC Video Retr. Eval Online
18. Kundu MK, Mondal J (2012) A novel technique for automatic abrupt shot transition detection. In: Proceedings Int. Conf. Communications, Devices and Intelligent Systems, pp 628–631
19. Lakshmi Priya GG, Domnic S (2014) Walsh-Hadamard Transform kernel-based feature vector for shot boundary detection. IEEE Trans Image Process 12:23
20. Li S, Yang B, Hu J (2011) Performance comparison of different multi-resolution transforms for image fusion. Information Fusion 12(2):74–84
21. Li W-K, Lai S-H (2002) A motion-aided video shot segmentation algorithm. In: Pacific rim Conference Multimedia, pp 336–343
22. Liu Z, Zavesky E, Gibbon D, Shahraray B, Haffner P (2007) AT&T research at TRECVID 2007. In: Proceedings TRECVID Workshop
23. Lopez F, Valiente JM, Baldrich R, Vanrell M (2005) Fast surface grading using color statistics in the CIE lab space. In: Proceedings Pattern Recognition and Image Analysis, pp 666–673
24. Ma YF, Sheng J, Chen Y, Zhang HJ (2001) Msr-asia at trec-10 video track: Shot boundary detection. In: Proceedings TREC
25. Miene A, Dammeyer A, Hermes T, Herzog O (2001) Advanced and adaptive shot boundary detection. In: Proceedings ECDL WS Generalized Documents, pp 39–43
26. Mithling M, Ewerth R, Stadelmann T, Zofel C, Shi B, Freislchen B (2007) University of Marburg at TRECVID 2007: Shot boundary detection and high level feature extraction. In: Proceedings REC Video Retr. Eval Online
27. Mohanta PP, Saha SK, Chanda B (2012) A model-based shot boundary detection technique using frame transition parameters. IEEE Trans Multimedia 14(1):223–233
28. Omidyeganeh M, Ghaemmaghami S, Shirmohammadi S (2011) Video keyframe analysis using a segment-based statistical metric in a visually sensitive parametric space. IEEE Trans Image Process 20(10):2730–2737
29. Ren J, Jiang J, Chen J (2007) Determination of Shot boundary in MPEG videos for TRECVID 2007. In: Proceedings TREC Video Retr. Eval Online
30. Sasithradevi A, Roomi SMdM, Raja R (2016) Non-subsampled Contourlet Transform based Shot Boundary Detection. IJCTA 9(7):3231–3228
31. Smeaton AF, Over P, Doherty AR (2010) Video shot boundary detection: Seven years of trecvid activity. Comput Vis Image Underst 114(4):411–418
32. Suykens JAK, Vandewalle J (1999) Least squares support vector machine classifiers. Neural Process Lett 9(3):293–300
33. TRECVID Dataset. Available: http://trecvid.nist.gov/
34. Youseff SM (2012) IC,TEDCT-CBIR: Integrating curvelet transform with enhanced dominant colors extraction and texture analysis for efficient content-based image retrieval. Comput Electr Eng 38(5):1358–1376
35. Yuan et al (2007) THU And ICRC at TRECVID 2007. In: Proceedings TREC video retr. Eval. Online
36. Yuan J, Wang H, Xiao L, Zheng W, Li J, Lin F, Zhang B (2007) A formal study of shot boundary detection. IEEE Trans Circuits Syst Video Technol 17(2):168–186
37. Zhang HJ, Kankanhalli A, Smolier SW (1993) Automatic partitioning of full-motion video. Multimedia Systems 1(1):10–28

**Jaydeb Mondal** received his B. Tech. degree in Computer Science and Engineering from WBUT, India and M. Tech degree in Computer Science and Engineering from University of Kalyani, India. Currently he is a Research Scholar at Machine Intelligence Unit, Indian Statistical Institute Kolkata. His research interests include Video Content Analysis, Applications of Machine Learning algorithms and Computer Vision.

**Malay Kumar Kundu** received his B. Tech., M. Tech. and PhD (Tech.) degrees in Radio physics and Electronics all are from the University of Calcutta. In 1982, he joined the Indian Statistical Institute, Calcutta, as a faculty member. He superannuated from the service of the institute as Professor (HAG) in December 2013. Currently, he is the INAE distinguished professor in the Machine Intelligence Unit of the ISI. He is a Fellow of the International Association for Pattern Recognition, USA (FIAPR), Indian National Academy of Engineering (FNAE), National Academy of Sciences (FNASc.), India and the Institute of Electronics and Telecommunication Engineers (FIETE), India. A senior member of the IEEE, USA and the founding life member & was Vice President of the Indian Unit for Pattern Recognition and Artificial Intelligence (IUPRAI) for 10 years till 2014. He received the prestigious VASVIK award for industrial research in the field of Electronic Sciences & Technology for the year 1999. In the year 1986, he received the Sir. J. C. Bose memorial award from the Institute of Electronics and Telecommunication Engineers (IETE), India. His current research interest includes Medical image analysis, Machine learning, Content based Image & video retrieval, Digital watermarking, video processing & analysis, soft computing and Computer vision. He has contributed 5 edited book volumes, about 160 research papers in well known and prestigious archival journals, international refereed conferences and in the edited monograph volumes. He is the holder of ten U.S patents, two International and two E.U patents.

**Sudeb Das** received the M.C.A. degree from WBUT, India, in 2008. He worked as a Project Linked Personnel at Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India from December 2008 to December 2013. He got his Ph.D. degree from Calcutta University in Computer Sci. (Tech.) in December 2015. He has contributed more than 15 research papers to well-known prestigious archival journals and international refereed conferences. Dr. Sudeb Das is currently working as a Senior Research Engineer in Videonetics Technologies Pvt. Ltd., Salt Lake City, Kolkata, West Bengal, Pin: 700 091, India. Prior to joining Videonetics, Dr. Das had worked as a Consultant at Innovation Lab, Tata Consultancy Services, Kolkata, West Bengal, India. His research is related to processing, analyzing and managing of digital image/video data for solving various real-life problems. He is also interested in the applications of various machine learning and soft computing schemes to process digital image/video data in real-time.

**Manish Chowdhury** received his B. Tech., degree in Electronics from the University of Pune, India. Currently he is a postdoctoral fellow in the School of Technology and Health, KTH Royal Institute of Technology, Sweden. He had been a Project Linked Personnel with the Machine Intelligence Unit, Indian Statistical Institute, Kolkata, where he has completed his PhD work. He presently has around 13 publications in international journals/conferences. He has also acted as a reviewer for few international journals/conferences. His current research interests include Pattern Recognition, Image Processing, 2D/3D medical image processing, Image Retrieval and Graph Theory.