

Real-time shot cut detection in compressed domain

J. Jiang^{1,2}, Z. Li³, G. Xiao¹ and J. Chen²

¹Faculty of Computing & Information Sciences, Southwest University, China

²School of Informatics, University of Bradford, United Kingdom

³Guangdong Industry Training Centre, Guangdong Polytechnic Normal University, China

ABSTRACT: In this short paper, we propose a fast and simple shot cut detection algorithm, which directly operates in compressed domain and suitable for real-time implementation. The proposed algorithm exploits the existing MPEG techniques by examining the prediction status for each macro-block inside B frames and P frames. As a result, locating both abrupt and dissolved shot cuts is operated by a sequence of comparison tests, and thus no feature extraction or histogram differentiation is needed. Although the description of the algorithm is primarily based on MPEG-1 and MPEG-2 streams, the scheme can be readily extended to other video compression standards such as MPEG-4 and H.264 by following the principle on monitoring: (i) balance between forward prediction and backward prediction; and (ii) boundaries among P, B and I frames. Extensive experiments illustrate that the proposed algorithm outperforms similar existing algorithm, providing a useful technique for fast and on-line video content processing.

Indexing Terms: shot cut detection, compressed domain video processing, video content analysis, and MPEGs

1. Introduction

In order to achieve effective and efficient motion estimation and compensation inside digital videos, MPEG arranged video sequences into group of pictures (GoP), for which the structure of such arrangement can be illustrated in Figure-1. Inside each GoP, the MPEG video structure has the feature that there exist two B-frames between every pair of I-P frames or P-P frames. For the convenience of discussion, we refer the first B-frame as B_f (front-B) and the second B-frame as B_r (rear-B). As a result, the proposed algorithm can be designed in terms of these front-B and rear-B frames.

In principle, shot cut detection is mainly conducted in pixel domain by detecting the major differences between adjacent frames. Representative techniques include histogram comparisons and edge difference examinations etc. [2,3,6]. The proposed algorithm, however, follows the principle that the MPEG properties and its embedded motion estimation and compensation scheme should be further exploited in the process of algorithm development for shot cut detection[1~5]. Fernando et al. [1] proposed an algorithm to detect shot cuts by exploiting motion vectors available inside MPEG compressed videos, Lelescu [2] models video sequences as stochastic processes, in which changes of characteristics or parameters are exploited to detect scene changes. Kobla et al. [3] provided a detailed analysis of MPEG compressed domain, based on which shot cut detection is proposed via exploiting the MPEG motion estimation and compensation scheme. Earlier attempt for shot cut detection in compressed domain was reported in [4], where Meng et al. used motion vectors and I,P-frame DC images to detect shot cuts. In [5], Pei and Chou proposed a simpler MB-type based scene change detection algorithm for both abrupt and gradual scene changes. This algorithm differs from all others as such that only the type of MBs is monitored to exploit the motion estimation and compensation scheme in MPEG, in which no motion vector or prediction status is analyzed. Therefore, such scheme has the advantage of extremely low computing cost. Specifically, the scheme firstly counts the number of MBs in intra-coding mode inside P-frames. Whenever the number of intra-coded MBs is above a pre-defined threshold, two separate operations for scene change detection are activated for detecting abrupt changes and gradual changes respectively. For abrupt changes, their detection is based on one of the three motion estimation and compensation forms in MPEG videos as illustrated in Figure-2. For gradual changes, their detection is based on two conditions. One is that a significant number of MBs inside P-frames are intracoded, indicating significant change of content; and the other is that a dominant number of MBs inside B-frames are interpolative motion compensated. While the scheme is simple with low computing cost, detailed analysis reveals that there exist a range of weaknesses, which include: (i) the scene change detection is dependent on four fixed and pre-defined thresholds; (ii) abrupt change detection and gradual scene change detection are two separate operations; and (iii) the performance is low in terms of precision in detecting scene changes. To improve the scheme, we propose to: (i) introduce an adaptive mechanism as such that comparison tests are adaptive to the outcome of motion estimation and compensation for each B frame; (ii) control all comparison tests by using only three parameters; (iii) combine abrupt change detection and gradual change detection into an integrated shot cut detection algorithm.

The rest of this article is structured into three sections. While section 2 provides detailed description of the proposed algorithm design, section 3 reports the experimental results and section 4 draws concluding remarks to provide overall analysis towards fast dissemination of the proposed technique.

2. The Proposed Algorithm Design

Following the spirit of the work reported in [5], we exploit the MPEG motion estimation and compensation technique to detect shot cuts by monitoring the number of predicted macroblocks inside each P or B frame. For the convenience of description and presentation, we define a range of essential variables as follows,

N_i : the number of intra-coded blocks inside P-frames

N : the total number of blocks inside each video frame;

N_{fb} : the number of both forward and backward predicted macroblocks inside each B-frame;

N_b : the number of backward predicted macroblocks inside each B-frame;

N_f : the number of forward predicted macroblocks inside each B-frame.

and a global view of the overall proposed algorithm is illustrated in Figure-3.

Given the input video sequence, we firstly monitor the number of intra-coded blocks inside each P-frame via the following comparison test:

$$\frac{N_i}{N} > T \quad (1)$$

Where T stands for a threshold, which is to be determined empirically.

The essence of the ratio between N_i and N calculated in (1) determines whether this tested P-frame is content correlated to its reference frame or not. When the ratio is smaller than the threshold, it indicates that most of the macroblocks can be motion compensated by its reference frame, and hence a significant extent of correlation between this P-frame and its reference frame can be established. Therefore, it is not likely that there exists any shot cut around this P-frame. As a result, we should carry on examining the next P-frame. Otherwise, if the condition represented by (1) is satisfied, it should indicate that most of the macroblocks are not well compensated by the reference frame, and hence it is likely that there exist a shot cut in

the neighborhood of this P-frame. As a result, further confirmation of such shot cut needs to be tested by examining its subsequent B-frames to find out: (i) whether such shot cut is abrupt or gradual; (ii) its exact location of the shot cut.

According to MPEG, all B-frames have three possibilities for their specific process of motion estimation and compensation, which include: (i) forward prediction, (ii) backward prediction, and (iii) bi-directional prediction. To confirm a shot cut and its location, we need to determine for each B frame whether its content is more correlated to its preceding reference frame (P or I) or subsequent reference frame (P or I) by monitoring its prediction status, i.e. forward, backward, or bidirectional prediction. A B-frame is more content correlated to its preceding reference frame if there exist overwhelming number of forward predicted macroblocks. In other words, MPEG motion estimation and compensation is more balanced towards forward prediction. This corresponds to the illustration in part (a) of Figure-2. Equally, a B-frame will be more content correlated to its subsequent reference frame if more macroblocks are backward predicted, which corresponds to the illustration given in Part (c) of Figure-2. Therefore, to test whether the B-frame is more content-correlated to its preceding reference frame, we propose the following test:

$$N_f > \lambda N_b \quad (2)$$

Where λ is a parameter controlling the balance between the backward prediction and forward prediction.

Similarly, the balance towards backward reference frame can be detected via:

$$N_b > \lambda N_f \quad (3)$$

Correspondingly, the abrupt shot cut detection can be designed into the following two steps:

Step-1: By using equation (1), examine every P-frame inside the video sequence to see if the shot cut detection process should be activated or not. For positive test of (1), go to step-2. Otherwise, carry on with step-1 to examine the next P-frame;

Step-2: Following satisfaction of (1), examine B_f and B_r to detect an abrupt shot cut according to one of the three situations: (i) if both B_f and B_r have more forward predicted blocks (satisfaction of equation (2)), an abrupt shot cut is detected between B_r and its subsequent P or I frame (see part (a) of Figure-2); (ii) if both B_f and B_r have more backward predicted blocks (satisfaction of equation (3)), an abrupt shot cut is detected between B_f and its preceding P or I frame (see part (c) of Figure-2); (iii) if B_f has more forward predicted blocks yet B_r has more backward predicted blocks, an abrupt shot cut is detected between B_f and B_r (see part (b) of Figure-2).

If none of the conditions given in Step-2 is detected, we need to check if there exists a possible gradual shot cut by examining the B-frames with the following test:

$$N_{bf} > \alpha(N_f + N_b) \quad (4)$$

Where α is another parameter indicating the dominance of bi-directional prediction of MBs inside B-frames.

The above test exploits the fact that gradual shot cuts incur gradual content change and such gradual change support bidirectional prediction inside B-frames such as fade-ins and fade-outs. To detect the possible gradual shot cuts out of those candidates satisfying (4), we follow our earlier work and analysis [6] that gradual cuts can only be detected by examining a consecutive number of such candidate frames. In general, the duration of most gradual shot boundaries is more than 1 second, which means that the duration of such changes is around 30 frames. To this end, a simple counting process is adopted to monitor the number of consecutive B frames that satisfy the condition given in (4). As illustrated in Figure-3, whenever the consecutive number of candidate B-frames is greater than 10, a gradual cut is detected. This value of 10 is determined on both theoretical and empirical basis, where the duration of 30 frames corresponds to around 20 B frames (see Figure-1), leading to around 10 P frames. Our empirical investigation also verifies that consecutive satisfaction of (4) for 10 times is an appropriate indicator for a gradual shot cut in consideration of general trend on most duration of gradual changes. For other compression schemes, this value needs to be adjusted according to specific ratio between P frames and B frames.

3. Experimental Results and Conclusions

To evaluate the performance of the proposed shot cut detection algorithm, we prepared a test set with three groups of video sequences, including news (12 video clips), movies (8 video clips) and cartoons (7 video clips). The test set contains a total of 403 abrupt shot cuts and 87 dissolved shot cuts lasting around 90 minutes. The selected test sequences are complex with extensive graphical effects. Videos were captured at a rate of 30 frames/s and resolution of 640×480 pixels.

For benchmarking purposes, we selected the algorithm reported in [5] as a representation of the existing techniques to provide a comparison for the evaluation of the proposed algorithm. We understand that there exists extensive research work on shot cut detections, with numerous algorithms reported in both pixel domain and compressed domain. However, we justify our selection by applying the principle that any benchmark selected should maintain fair comparisons with comparable computing cost and algorithm complexity. To provide some further information about the proposed algorithm, we also used our earlier work as the second benchmark [6], which is an unfair comparison since this algorithm has much higher complexity and computing cost.

To measure the performances on shot cut detection for the evaluated algorithms, we follow the common practice to use the recall and precision rates [6~8]. For the three parameters, (T, λ, α) , we used (5%, 3.6, 2.5) in our experiments, which is determined empirically. Such empirical approach has the same nature as all those threshold-based techniques reported in pixel domain [6~8], where certain range of flexibility exists. As indicated by equation (1), higher value of T increases the reliability of shot cut detection but misses more shot cuts. Similarly, equations (2) and (4) also indicate that, while higher value of λ or α reduces the false positive detection rate, it could increase the number of shot cuts being missed.

All the experimental results are summarized in Table-I, where the number in the first column specifies the total number of shot cuts inside the video sequences, and the pair of values inside brackets is (recall, precision). For the convenience of visual inspection, samples of detected shot boundaries are illustrated in Figure 4, where each row of frames correspond to one category of the video clips.

From Table-I, it can be seen that the proposed algorithm outperforms the benchmark-1 in terms of both recall and precision rates for abrupt and dissolved shot cut detections. Specific comparisons between the proposed and benchmark-1 can be summarized as: (i) For abrupt shot cut detection, the proposed algorithm achieves on average a 86% recall rate and a 85% precision rate, yet the benchmark achieves 34%

recall rate and 68% precision rate; (ii) For dissolved shot cut detection, the proposed algorithm delivers 63% recall and 64% precision rate, yet the benchmark delivers 33% recall and 51% precision; (iii) both the techniques work better for abrupt shot cut detection than dissolved shot cut detection.

Compared with benchmark-2, the proposed algorithm is about 12% inferior on all measurements. Considering the fact that benchmark-2 has much more complicated operations, such as multiple feature extraction (color histograms, motion compensation, and textures) and fuzzy inferences [6], the proposed algorithm has its unique advantages in terms of real-time applications and compressed domain operation. Further comparisons in practical terms are given below.

As the work is prompted by an EU funded FP-6 integrated project, the figures illustrated in Table-I can be further interpreted in practical applications to reveal its general performances. In this integrated project, a video processing tool is to be developed to extract series of semantic features, including input video types (documentary, sporting, news, films etc.), 60s/70s styles or settings, close-up of human objects etc. The first step of such a tool is to cut the input video sequence into sections, where consistent spatial content can be identified and thus semantics can be extracted within each section. In this circumstance, we applied both benchmark-2 and the proposed algorithm to the video processing system. It is discovered that both algorithms meet our need in semantics feature extraction and no noticeable difference is made during the comparative tests. Our further observations explain the reason that: (i) although some cuts were missed by the proposed algorithm, the visual content still maintains certain level of consistency, such as both shots follow the same human objects but with different background (different corners of the same room etc.). To this end, the missed cuts do not produce any noticeable negative impact upon semantics feature extraction; (ii) for those false positive cuts detected, it is noticed that the boundary visual content does present significant differences, and thus such false alarms do not produce any noticeable negative impact either.

While the proposed algorithm is primarily based on MPEG-1 or MPEG-2 schemes, it is readily extendable to other video compression standards such as MPEG-4 and H.264. The major implication is two folds. One is the variable block sized motion estimation and compensation, and the other is variable allocation of B-frames and P-frames. For the issue of variable block size, the proposed algorithm will still work without any significant change, since the balance between forward-prediction and backward-prediction can still be monitored by testing the number of predicted blocks. Further consideration may be required by looking into the size of each predicted block, which can be implemented via introducing a table of fixed weighting factors. Regarding the issue of variable allocation of B-frames and P-frames, detailed analysis of their corre-

sponding modes of motion estimation and compensation is required. However, the same principle can still be applied to such analysis in the sense that only three boundaries need to be examined. That is: (i) the boundary between the front P-frame and all other B-frames; (ii) the boundary between B-frames, and (iii) the boundary between the last P-frame or I-frame and all other B-frames. This is illustrated in Figure-5.

Finally, to provide a wider evaluation of the proposed algorithm, we also compared with the motion-vector based scene change detection algorithm [4]. This algorithm is very different from the proposed. The major difference can be highlighted by the fact that: (i) Meng's algorithm monitors motion vectors, yet the proposed algorithm monitors the MB types; (ii) while Meng's work could be interpreted as monitoring MB type through motion vectors, no consideration is given for bi-directional prediction; (iii) the proposed algorithm goes one step further by monitoring the MB types to detect the correlation of B frames, i.e., more correlated to front reference frame or more correlated to rear reference frame (see Figure 2). (iv) Meng's algorithm requires the variances of DC image for I and P frames; (v) Meng's work relies on distances among suspected scene changed frames to detect the scene changes, such as T_{reject} , yet the proposed uses the content adaptive thresholds to determine how close the B frames are correlated to their front or rear reference frames.

To highlight the comparison in terms of exploiting MPEG motion estimation and compensation scheme, we implemented this algorithm based on the three ratios of R_p , R_b , R_f to detect their peaks and then decide the scene changes by using the distance criteria $T_{\text{rejection}} = \text{default}$ [4]. The experimental results are shown in Table-II. From the results, it can be seen that the proposed algorithm outperforms Meng's work in terms of abrupt shot cut detection. While Meng's recall rates are close to the proposed algorithm, its precision rates are significantly lower due to large number of false alarms. As suggested by the reviewer, we did not implement Meng's algorithm on dissolved cut detection since it requires variance of DC images, which can not be implemented within the designated deadline. In addition, such variance-based parabolic detection remains the same as conventional techniques in pixel domain, which is much more computing intensive than the proposed.

4. Conclusions

In this paper, we described a simple and fast algorithm for detection of both abrupt shot cuts and dissolved shot cuts. The proposed algorithm works in compressed domain via exploiting existing MPEG motion esti-

mation and compensation mechanisms. While the proposed algorithm can save a lot of computation cost, extensive experiments support that the proposed algorithm also achieves superior performances over the existing counterpart. In summary, the feature and the advantage of the proposed algorithm can be highlighted as: (i) an integrated technique for both abrupt shot cut and dissolved shot cut detection; (ii) directly operates in compressed domain and thus suitable for real-time implementation; and (iii) only two thresholds and two parameters are required for shot cut detection and yet the detection mechanism is made adaptive to the input video content, or the performance of motion estimation and compensation embedded inside the MPEG-2 techniques; (iv) such technique will provide a range of useful applications for on-line video content analysis, processing, and management. Specific examples include real-time scene change analysis for MPEG compressed video streams, on-line annotation of video sequences, and shot-based video content retrievals.

Finally, the authors wish to acknowledge the financial support under EU IST FP-6 Research Programme as funded for the integrated project: LIVE (Contract No. IST-4-027312).

References

1. Fernando W. A. C., Canagarajah C. N. and Bull D. R., "Video Segmentation and Classification for Content Based Storage and Retrieval Using Motion Vectors", SPIE Conference on Storage and Retrieval for Image and Video Databases, Vol. 3656, pp. 687-698, 1999.
2. Dan Lelescu and Dan Schonfeld, "Statistical sequential analysis for real-time video scene change detection on compressed multimedia bitstream", IEEE Transactions on Multimedia, Vol.5, No.1, March 2003, pp.106-117;
3. Kobla V., Doermann D. and Lin K., "Archiving, Indexing, and Retrieval of Video in the Compressed Domain", SPIE Conference on Multimedia Storage and Archiving Systems, pp. 78-89, 1996.
4. Meng J., Juan Y. and Chang S., "Scene Change Detection in a MPEG Compressed Video Sequence", SPIE Conference on Digital Video Compression: Algorithms and Technologies, 1995, pp.14-25, 1995.
5. Soo-Chang Pei and Yu-Zuon Chou, "Novel error concealment method with adaptive prediction to the abrupt and gradual scene changes", IEEE transactions on multimedia, Vol. 6, No.1, February 2004, pp.158-173;

6. H. Fang and J. Jiang 'A fuzzy logic approach for detection of video shot boundaries', *Pattern Recognitions*, Vol 39, pp2092-2100, 2006;
7. J. Bescos, "Real-time Shot Change Detection Over Online MPEG-2 Video", *IEEE Transaction on Circuits and Systems for Video Technology*, Vol.14, No.4, April 2004;
8. H. Koumaras and G. Gardikis and G. Xilouris and E. Pallis and A. Kourtis 'Shot boundary detection without threshold parameters' *Journal of Electronic Imaging*, Vol 15, No 2, id: 020503;

Table-I: Summary of experimental results for Abrupt Shot Cut Detection

Video Sequences	The proposed		The benchmark-1		The benchmark-2	
	Abrupt	Dissolved	Abrupt	Dissolved	Abrupt	Dissolved
News (93)	(87%,82%)	(72%,52%)	(38%,64%)	(35%,46%)	(96%,94%)	(87%,73%)
Movies (138)	(88%,85%)	(53%,68%)	(31%,78%)	(28%,50%)	(100%,100%)	(81%,77%)
Cartoons (172)	(84%,90%)	(63%,73%)	(32%,39%)	(36%,58%)	(98%,100%)	(79%,80%)
Total (403)	(86%,85%)	(63%,64%)	(34%,68%)	(33%,51%)	(98%,98%)	(82%,77%)

Table-II: Comparative experiments for Abrupt Cut Detection

Video Sequences	The proposed	Meng's algorithm
News (93)	(87%,82%)	(85%,63%)
Movies (138)	(88%,85%)	(74%,38%)
Cartoons (172)	(84%,90%)	(81%,47%)
Total (403)	(86%,85%)	(83%,78%)

List of Figure Captions:

Figure 1: Illustration of MPEG video structure for motion estimation & compensation

Figure 2: Illustration of abrupt shot cut detection

Figure 3: Overview of the proposed shot cut detection algorithm

Figure 4: Frame sample illustration of the test video clips.

Figure 5: Analysis of shot cut modes in variable video streams.

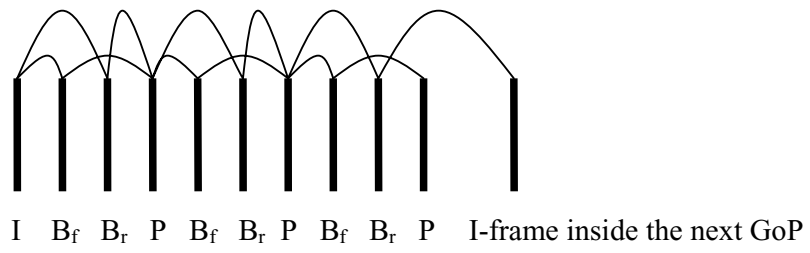


Figure-1: *Illustration of MPEG video structure for motion estimation & compensation*

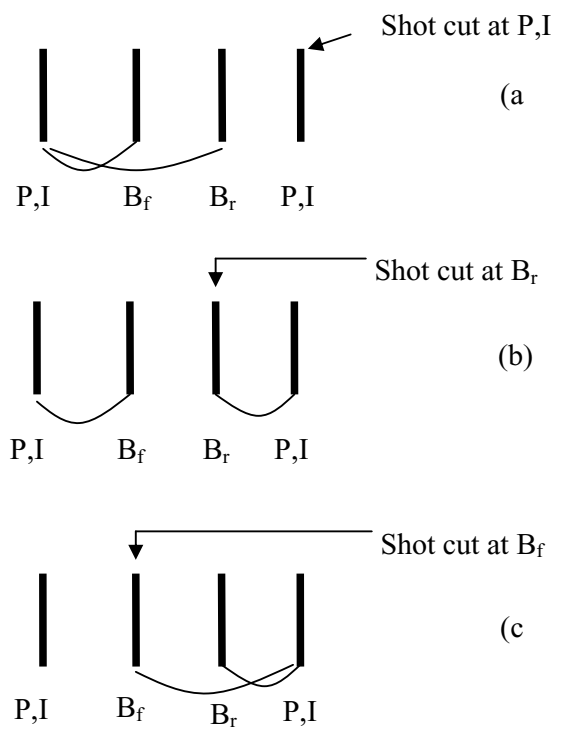


Figure-2: *Illustration of abrupt shot cut detection*

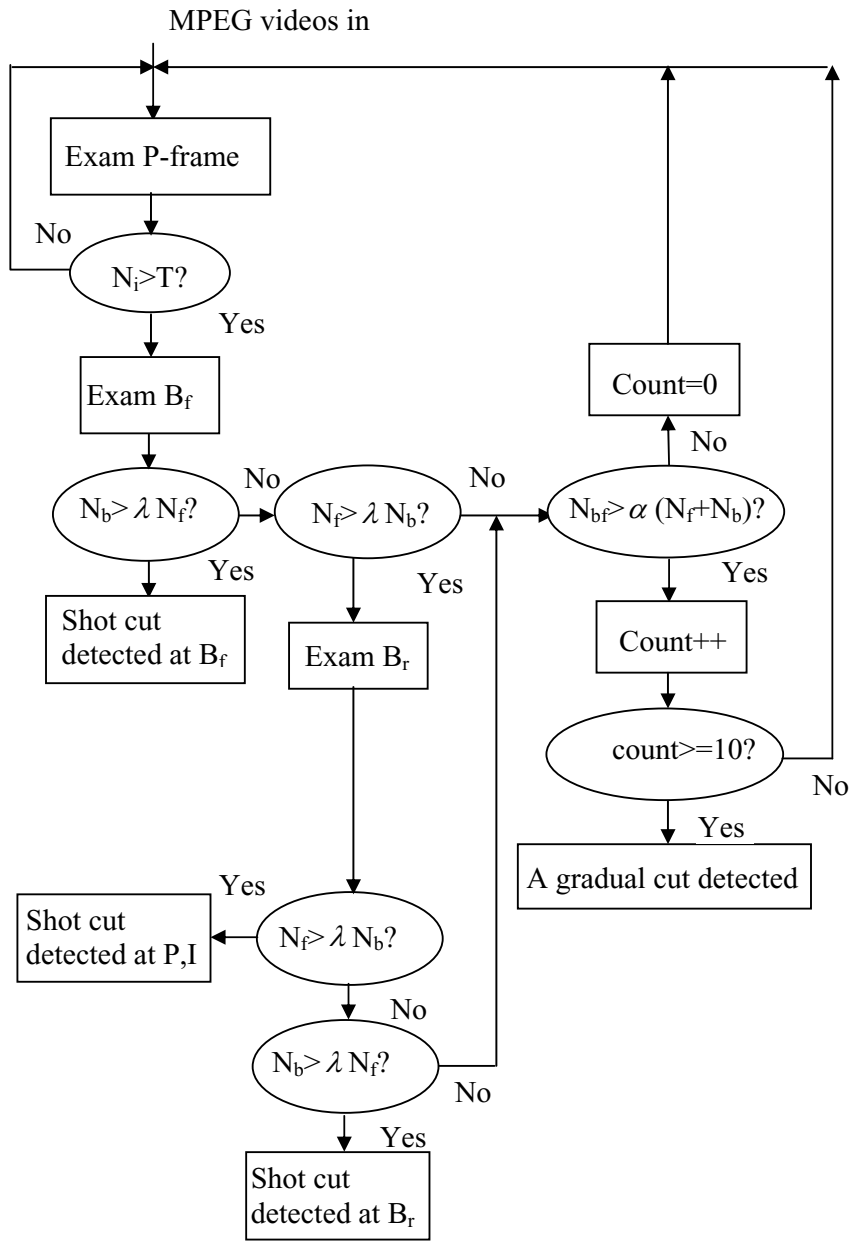


Figure 3: Overview of the proposed shot cut detection algorithm



(a): Shot boundary samples in news.



(b): Shot boundary samples in movies.



(c): Shot boundary samples in cartoons.

Figure-4: Frame sample illustration of the test video clips.

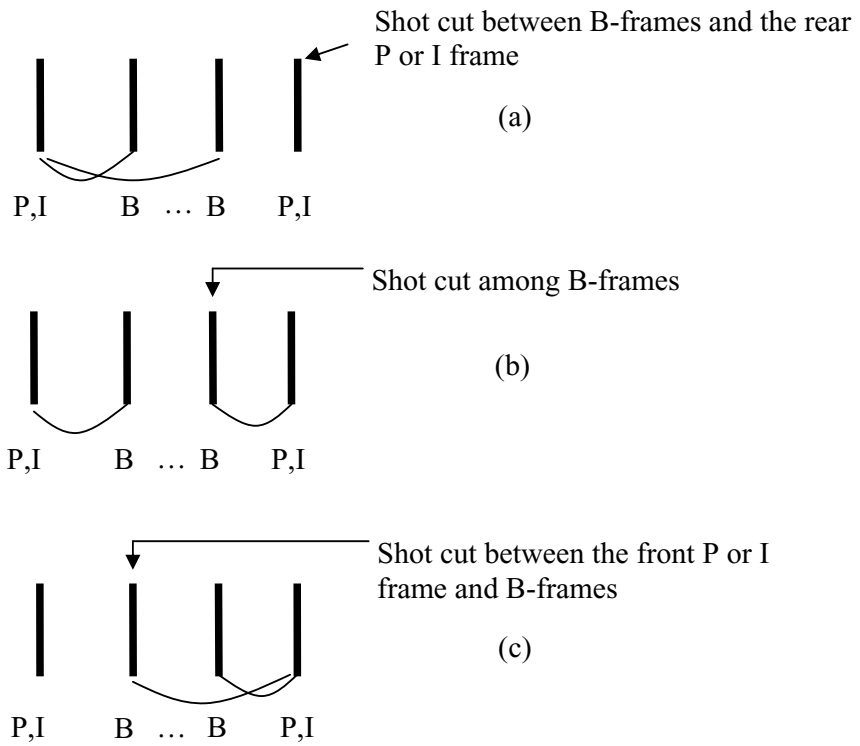


Figure-5: Analysis of shot cut modes in variable video streams.