2010 International Conference on Multimedia Information Networking and Security

# 2PASCD: An efficient 2-Pass Abrupt Scene Change Detection algorithm

Aditya Vashistha, Rajarathnam Nallusamy, and Sanjoy Paul

SETLabs, Infosys Technologies Limited, Bangalore, India 560100
Email: {aditya_vashistha, rajarathnam_n, sanjoy_paul}@infosys.com

## Abstract

*Scene change detection in videos is a primary requirement of video processing applications used for the purpose of generating data needed by video data management systems and digital rights management (DRM) systems. Scene change data can be used in DRM systems for effective IPR protection by means of watermarking and fingerprinting selected scenes. In this paper, we present a 2-pass algorithm for abrupt scene change detection in videos based on* $\chi^2$ *color histogram comparison of sub-windows of successive frames and sliding window detector method. In the proposed method, we improve the efficiency of scene change detection by a second level inspection of the detected scene changes using a fixed average and a moving average to ascertain whether the detected scene boundary is correct or the scenes need to be combined. This method gives precision and recall values better than any existing method.*

***Keywords*-abrupt scene change detection; videos; multimedia;**

## I. Introduction

The convergence of networks, devices, and services combined with the technological advancements in digital storage, multimedia compression, and miniaturization of digital cameras has lead to an explosive growth of online video content. In addition to the professionally produced video content, user-generated content and content produced by hardcore amateurs are also on the rise. Videos can easily be shared over the Internet using popular video sharing sites such as YouTube and Yahoo! Video. Increasing volumes of online digital video content and large amount of information contained within each video make it a challenge to search and retrieve relevant video files from a large collection. Video data management systems aim at reducing this complexity by indexing the video files.

The ease with which online digital content can be copied and redistributed poses a serious threat to copyrighted content. Also, easy availability of freely downloadable software tools for editing and transcoding multimedia content makes it easier to modify and distribute pirated content as genuine content for economic benefits. This underlines the need for an effective copyright protection scheme for online content. Digital watermarking continues to remain the main technique used for this purpose.

Indexing of video content as well as many digital watermarking algorithms require the video to be split into scenes. Scene change detection (SCD) is the initial step in the segmentation of videos into contiguous scenes. Segmentation of digital videos finds applications in video data management systems [1]–[4] and intellectual property rights (IPR) protection systems [5], [6]. Scene changes in videos can either be gradual or abrupt. Abrupt scene changes can result from editing *cuts*. Gradual scene changes result from spatial effects such as zoom, camera pan and tilt, dissolve, fade in, fade out, etc. SCD usually involves measurement of some differences between successive frame images. Our study is focused on detecting abrupt scene changes in videos for the purpose of watermarking all or selected scenes.

A video is a sequence of scenes and a scene is a sequence of images called *frames*. Detection of scene changes effectively depends on finding the similarity or the difference between adjacent frames. There are several metrics used to compute the difference between two frames [7], [8]. Template matching, histogram comparison, and $\chi^2$ color histogram comparison are some of the techniques used to measure the inter-frame difference. We use local $\chi^2$ color histogram comparison of adjacent frames after dividing them into sub-windows. We propose a 2-pass algorithm for detecting abrupt scene changes that gives high precision and recall values.

This paper is organized as follows. Section II presents related work in SCD using frame differences. The proposed 2-pass abrupt scene change detection (2PASCD)

978-0-7695-4258-4/10 $26.00 © 2010 IEEE
DOI 10.1109/MINES.2010.17

IEEE computer society

algorithm is presented in Section III. The experiments conducted along with the results are given in Section IV. Finally, the conclusions are presented in Section V.

## II. Related work

In this section, we present the sliding window detector based scene change detection method which is built on color histogram based frame difference [2], [9].

### A. Local $\chi^2$ color histogram comparison

Color histogram of a frame (image) is a representation of the distribution of colors in that frame. To compute the color histogram of an image, its color space, for example, RGB, is divided into *n bins* containing similar colors, and then number of pixels falling in each *bin* is counted. The color histogram $H_F$ of a frame *F* is a vector $< h_1, h_2, .., h_n >$ of n bins, wherein each bin $h_j$ contains the number of pixels of color $j$ in that frame. In our implementation, $n = 256$ bins are used for computation of histogram.

If $F_i$ and $F_j$ are two consecutive frames of the video, then the color histogram difference metric is given by

$$d_{r,g,b}(F_i, F_j) = \sum_{n=1}^{256}(|H_i^r(n) - H_j^r(n)| + |H_i^g(n) - H_j^g(n)| + |H_i^b(n) - H_j^b(n)|) \tag{1}$$

where $H_i^r(n)$, $H_i^g(n)$, and $H_i^b(n)$ represent the number of pixels in $n^{th}$ bin of each color space (r, g, b) of $i^{th}$ frame $F_i$.

Using weight for brightness grade change of each color space, we can redefine equation 1 as

$$\begin{aligned} d_{wr,wg,wb}(F_i, F_j) = \sum_{n=1}^{256}(&|H_i^r(n) - H_j^r(n)| \times \alpha + \\ &|H_i^g(n) - H_j^g(n)| \times \beta + \\ &|H_i^b(n) - H_j^b(n)| \times \gamma) \end{aligned} \tag{2}$$

where $\alpha$, $\beta$, $and$ $\gamma$ are constants representing the brightness grade according to NTSC standard and has values $\alpha = 0.299$, $\beta = 0.587$, and $\gamma = 0.114$.

Since $\chi^2$ color histogram comparison gives better results [2], [10], [11], several other researchers have used it to detect abrupt scene changes in videos [9], [12]. For $\chi^2$ color histogram comparison, equation is modified as

$$d_{w\chi^2}(F_i, F_j) = \begin{cases} \sum_{n=1}^{256} \frac{((H_i(n) - H_j(n))^2}{max((H_i(n), H_j(n))}, & \text{if}(H_{i,j} \neq 0) \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

It has also been shown that partitioning of successive video frames into sub-windows of equal size and $\chi^2$ color histogram comparison of corresponding sub-windows yield better results in detecting scene changes in videos [8]. For *m* $(m1 \times m2)$ sub-windows or blocks, equation can be modified as

$$d(F_i, F_j) = \sum_{bl=1}^{m} d_{\chi^2}(F_i, F_j, bl) \tag{4}$$

The $\chi^2$ color histogram comparison of sub-windows or blocks then becomes

$$\begin{aligned} d_{\chi^2}(F_i, F_j, bl) = \sum_{k=1}^{N} &\Big(\frac{(H_i^r(k) - H_j^r(k))^2}{\max(H_i^r(k), H_j^r(k))}\Big) \times \alpha + \\ &\Big(\frac{(H_i^g(k) - H_j^g(k))^2}{\max(H_i^g(k), H_j^g(k))}\Big) \times \beta + \\ &\Big(\frac{(H_i^b(k) - H_j^b(k))^2}{\max(H_i^b(k), H_j^b(k))}\Big) \times \gamma \end{aligned} \tag{5}$$

### B. Sliding window detector

As can be seen in Figure 1, there is a large variation of the frame difference values obtained using equation 5 and it is difficult to obtain information about connected frames of a scene. Hence, we used a sliding window detector to extract robust scene changes from frame differences by comparing the frame difference value of the frame corresponding to the point of local maxima. There is a scene change at the point of local maxima, if the average frame difference of previous five frames is less than an appropriate threshold value.

## III. Proposed 2PASCD algorithm

The first pass of the algorithm presented below is based on methods presented by many researchers [2], [9].

### A. First pass of the algorithm

(i). Identify the frame difference between the consecutive frames in a given video. If there are, say, $N$ frames in the video, then total consecutive frame difference values will be $N - 1$. For two frames, $F_i$ and its previous frame $F_{(i-1)}$, let the frame difference be denoted by $D_{(i-1),i}$. It is calculated by computing local $\chi^2$ color histogram comparison using k bins (where k is power of 2) and breaking down the frames into $(m \times m)$ sub-windows.

(ii). Identify peak values amongst all $N - 1$ computed values of $D_{(i-1),i}$. Peak $D_{(i-1),i}$ values are the frame difference values which are greater than previous as well as next difference value. i.e. $D_{(i-1),i}$ is a peak if it is greater than both $D_{(i-2),(i-1)}$ and $D_{i,(i+1)}$. Let there be *R* peak values. Peak values are the points of local maxima.

(iii). Find *D*, the average of previous 5 frame difference values corresponding to each peak value of $D_{(i-1),i}$. Identify appropriate threshold factor *X*. If $D < D_{(i-1),i}/X$, then $F_i$ is the start frame of a new scene.

(iv). Identify the first set of scenes detected from the first pass using steps (i) to (iii) above. Let the total number of scenes obtained from the first pass be P where $P \subseteq R$.

## B. Second pass of the algorithm

In this section, We introduce a second pass to improve the efficiency of abrupt scene change detection by inspecting and refining the scene changes detected by first pass.

(i). Identify completely black or white scenes among the scenes detected in the first pass. If the *median* of all the frame difference values, $D_{(i-1),i}$, in the scene is zero, then the scene can be termed as completely black or white.

(ii). Compute the frame difference at the scene boundary by local $\chi^2$ color histogram comparison. Let the frame difference at the boundary between $(j-1)^{th}$ and $j^{th}$ scenes be $DS_{(j-1),j}$ where *j* varies from *1 to P-1*. This difference is called *scene boundary difference* and is nothing but the frame difference between the last frame of $(j-1)^{th}$ scene and the first frame of $j^{th}$ scene. The scene boundary difference is a measure of abruptness at the scene boundary. Larger the $DS_{(j-1),j}$ value, higher the chance that scenes $j-1$ and *j* are distinct and can not be clubbed together.

(iii). Find the *fixed average value* $D_{favg}$ of all the scene boundary differences, $DS_{(j-1),j}$ where *j* varies from *1 to P-1*.

$$D_{favg} = \frac{\sum_{j=1}^{P-1} DS_{(j-1),j}}{P-1} \quad (6)$$

Here, the value of $DS_{(j-1),j}$ is considered as *0* if either scene $(j-1)$ or scene *j* is black or white.

(iv). Next step is to compute the scene based varying average, $D_{vavg}$. Scene based varying average is used to detect the relative difference between the current scene and the last detected abrupt scene. The idea is to club the scenes that are incorrectly fragmented as different scenes in the first pass of the algorithm, even though they belong to the same scene. Initially, set $D_{vavg} = D_{favg}$. If, $DS_{(j-1),j} < D_{favg}$ and $DS_{(j-1),j} < D_{vavg}$, then combine scenes $(j-1)$ and *j*. Otherwise, scene *(j-1)* is an abrupt scene. Then, set the scene based varying average, $D_{vavg}$ value to half the scene boundary difference value between scenes $(j-1)$ and *j*. The plot of $D_{favg}$ and $D_{vavg}$ are given in Figure 2.

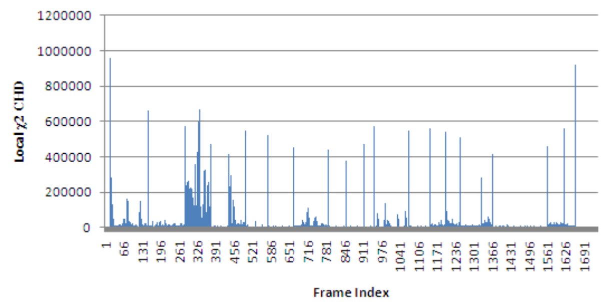


**Figure 1. Color histogram difference between consecutive frames.**

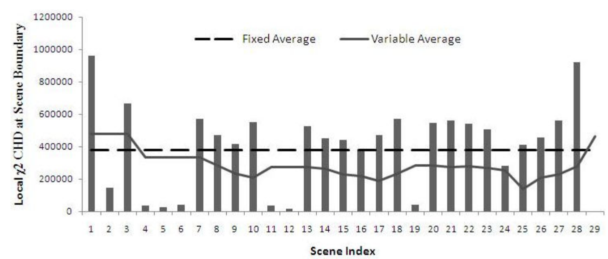


**Figure 2. Color histogram difference at scene boundary.**

# IV. Experimental results

In order to study the efficiency of the proposed 2PASCD algorithm, it is implemented using Java Media Framework API (JMF). Six different video files downloaded from *www.washingtonpost.com* were used as test samples after converting to *avi* format with a resolution of $800 \times 600$. To evaluate the performance of the proposed algorithm, we have used *precision* and *recall* metrics which are widely used in the area of information retrieval. They are defined in Section IV-A. Two different threshold values *(1/X = 25% and 33%)* and $(2 \times 2)$ and $(4 \times 4)$ sub-windows were used in the experiments. Number of scenes detected after the first pass and second pass of the algorithm are obtained. Actual number of scenes present in the video along with the number of missed scenes and falsely detected scenes were obtained after visually inspecting the frames in each scene and at the scene boundaries. The results and discussions are presented in Section IV-B.

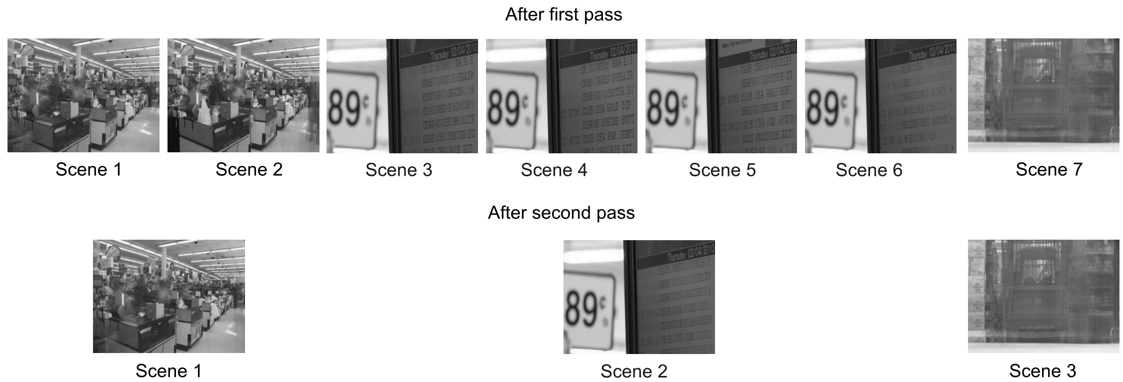


**Figure 3. Averaged scene images after first and second pass of the algorithm.**

### A. Performance metrics used

*Precision* and *recall* are the two metrics widely used in establishing the effectiveness of an information retrieval method. In SCD, *precision* is defined as the ratio of number of scene changes correctly detected to the total number of scene changes detected. *Recall* is the ratio of number of scene changes correctly detected to the total number of scene changes that exist. Thus, we have

$$Precision = \frac{N_d}{N_d + N_f} \tag{7}$$

and

$$Recall = \frac{N_d}{N_d + N_m} \tag{8}$$

where

$$\begin{aligned} N_d &= number\ of\ correctly\ detected\ scene\ changes \\ N_f &= number\ of\ falsely\ detected\ scene\ changes \\ N_m &= number\ of\ scene\ changes\ missed \end{aligned}$$

### B. Results and discussions

The results of the 2PASCD algorithm based on the number of scenes detected after first pass and second pass of the algorithm and the number of scenes missed and falsely detected by each pass are presented in Tables I and II.

i). From the data presented in Table I, it is clear that the 2PASCD is very efficient providing 100% precision and recall for half of the test videos with 25% threshold value. No method available in literature provides such a high precision and recall even in selected set of videos.

ii). From Table I, for the other videos, while the precision varies from 95.6% to 98%, the recall varies from 87.8% to 96.7%, which shows that the 2PASCD gives better results for a wide range of videos.

iii). Table II shows that dividing the frames into more number of sub-windows $(4 \times 4)$ improves the precision and recall values for videos which were giving lower precision and recall values when less number of sub-windows $(2 \times 2)$ were used. Precision is more important in detecting abrupt scene changes for the purpose of watermarking.

iv). A higher threshold value gives better recall values. In Table II, recall which varied from 87.8% to 100% for a threshold value of 25%, increases to 94.4% to 100% for a threshold value of 33%.

v). Figure 3 shows how a scene fragmented by first pass (scenes 3, 4, 5, and 6) because of the changing text in the display board are clubbed into a singe scene after the second pass of the algorithm. Scenes 1 and 2 fragmented by first pass because of movement of persons, is clubbed together in second pass. Scene 7 remains as it is and becomes scene 3 after second pass.

vi). It is observed that the 2PASCD also fragments a scene in which there are sudden illuminations in few frames like any other algorithm.

## V. Conclusions

In this paper, we have presented an efficient 2PASCD algorithm by introducing a second pass to existing abrupt scene change detection algorithms. By inspecting the frame differences at the scene boundaries, decision is made whether the scenes detected after the first pass actually are different scenes or fragmentation of same scene. By

**Table I. Improvement on the efficiency of abrupt scene change detection due to 2PASCD algorithm using 25% threshold and $(4 \times 4)$ windows.**

| Name of the video | Number of correct scenes | $1^{st}$ pass | | $2^{nd}$ pass | |
|---|---|---|---|---|---|
| | | Precision | Recall | Precision | Recall |
| Stocking up for the 'storm of the decade' | 21 | 75 | 100 | 100 | 100 |
| As the city sleeps | 14 | 34.1 | 100 | 100 | 100 |
| Journey into the 'Emerald Triangle' | 61 | 69.3 | 100 | 100 | 96.7 |
| A Portrait of Coney Island | 57 | 86.3 | 95 | 98 | 87.8 |
| Bless Your Pet | 21 | 84 | 100 | 100 | 100 |
| Moving thro-ugh Metro | 71 | 84.5 | 95.9 | 95.6 | 91.6 |

**Table II. Effect of threshold and sub-window size on precision and recall.**

| Name of the video | 25% threshold (2x2) windows | | 25% threshold (4x4) windows | | 33% threshold (4x4) windows | |
|---|---|---|---|---|---|---|
| | Preci-sion | Recall | Preci-sion | Recall | Preci-sion | Recall |
| Stocking up for the 'storm of the decade' | 95.2 | 95.2 | 100 | 100 | 87.5 | 100 |
| As the city sleeps | 100 | 100 | 100 | 100 | 100 | 100 |
| Journey into the 'Emerald Triangle' | 100 | 96.7 | 100 | 96.7 | 100 | 96.7 |
| A Portrait of Coney Island | 97.8 | 84.2 | 98 | 87.8 | 93.2 | 96.5 |
| Bless Your Pet | 100 | 100 | 100 | 100 | 100 | 100 |
| Moving thro-ugh Metro | 97 | 91.6 | 95.6 | 91.6 | 95.7 | 94.4 |

analysing six different videos, we have shown that the precision values are considerably improved by the 2-pass algorithm after the second pass. We have also shown that choosing a higher threshold value within the effective threshold range improves the recall values and higher number of sub-windows for local $\chi^2$ color histogram comparison improves both the precision and recall values for abrupt scene change detection. The 2PASCD algorithm presented here gives better precision and recall values as compared to existing algorithms.

## References


[1] A. Hampapur, T. Weymouth, and R. Jain, "Digital video segmentation," in *MULTIMEDIA '94: Proc. of the second ACM international conference on Multimedia*. New York, NY, USA: ACM, 1994, pp. 357–364.

[2] C. F. Lam and M.-C. Lee, "Video segmentation using color difference histogram," in *MINAR '98: Proc. of the IAPR International Workshop on Multimedia Information Analysis and Retrieval*. London, UK: Springer-Verlag, 1998, pp. 159–174.

[3] "Video content representation, indexing, and matching in video information systems," *Journal of Visual Communication and Image Representation*, vol. 8, no. 2, pp. 107 – 120, 1997.

[4] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," in *MULTIMEDIA '95: Proc. of the third ACM international conference on Multimedia*. New York, NY, USA: ACM, 1995, pp. 189–200.

[5] K. M. Singh, T. R. Singh, O. I. Singh, and T. R. Singh, "A blind video watermarking scheme based on scene change detection," in *Proc. of the Indo-US Conference & Workshop on Cyber Security, Cyber Crime and Cyber Forensics*, Kochi, India, Aug. 2009.

[6] A. Vashistha, R. Nallusamy, and S. Paul, "Watermarking video content using visual cryptography and scene averaged image," in *International Workshop on Content Protection and Forensics (CPAF 2010), In Conjunction with IEEE International Conference on Multimedia and Expo (ICME 2010), July 19 - 23, 2010, Singapore*, 2010.

[7] 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, pp. 28–43, 1996.

[8] H. Jiang, A. S. Helal, A. K. Elmagarmid, and A. Joshi, "Scene change detection for video database management systems - a survey," Department of Computer Sciences, Purdue University, West Lafayette, IN 47907, Tech. Rep. CSD-TR 96-058, Oct. 1996.

[9] P. Adhikari, N. Gargote, J. Digge, and B. Hogade, "Abrupt scene change detection," *World Academy of Science, Engineering and Technology*, vol. 42, pp. 711–716, 2008.

[10] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," in *Proc. of the IFIP TC2/WG 2.6 Second Working Conference on Visual Database Systems II*. Amsterdam, The Netherlands, The Netherlands: North-Holland Publishing Co., 1992, pp. 113–127.

[11] B.-L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 533–544, Dec. 1995.

[12] B. Sabata and M. Goldszmidt, "Fusion of multiple cues for video segmentation," in *In International Conference on Information Fusion*, 1999.