



Dynamic Scene Change Detection in Video Coding

Y. A. Salih^{*a}, L. E. George^b

^a Department of Computer Science, University of Sulaimani, Sulaimani Polytechnic University, KRG, Iraq

^b Department of Remote Sensing and GIS, Baghdad University, Iraq

PAPER INFO

Paper history:

Received 11 February 2020

Received in revised form 11 March 2020

Accepted 28 March 2020

Keywords:

Scene Change Detection

Dynamic Threshold

Precision

Recall

F1- Score

ABSTRACT

Video compression has become a source of different research studies. It is necessary in order to address channel bandwidth limitations and growing video demand, including digital libraries and streaming media delivery via the Internet. A video is a number of frames captured by a camera while a scene is a series of consecutive frames captured from a specific narrative viewpoint. To compress a video, firstly, the intra frames are separated from inter frames using scene change detection methods. Then, block-based motion estimation algorithms are used to eliminate the temporal redundancy between successive frames. This paper describes some scene change detection methods for use on the uncompressed video to detect scene types such as cut, dissolve, wipe, etc. Absolute Frame Difference (AFD), Mean Absolute Frame Differences (MAFD), Mean Histogram Absolute Frame Difference (MHAFD), and Maximum Gradient Value (MGV) techniques are adaptively tested on different video types to identify accurate scene change in both low and high object motion scenes. Test results show that the proposed approach (MHAFD) obtains a better accuracy, F1-score measure of 100%, especially for cuts and gradual transitions (wipe) video types. Dissolve scene change is detected with a high precision of 100% (i.e., no false detection) with the (MAFD) detector. Besides, in terms of time complexity for analyzing all the video samples, the proposed method (MHAFD) provides the best result compared to the selected detectors.

doi: 10.5829/ije.2020.33.05b.30

1. INTRODUCTION

Scene change detection has great importance in video coding applications. It can be employed as a decisional pre-processing algorithm, used to force intra-frame encoding (I) rather than temporal inter-frame prediction (P) when scene change takes place, and to verify the coding for the remaining frames in terms of either (P) or (B) bi-directional coding. Video segmentation is required in any video compression process and is done through the utilization of scene change detection. A long video can be broken into video shots. A shot is a series of frames in which one continuous action in time and space is captured by a single camera. Figure 1 [1] presents a hierarchical description of a video sequence. Scene transitions can be classified into gradual transitions (fades, dissolves, and wipes) and abrupt transitions (cuts). A cut is a rapid transition between two contiguous frames; this process is employed to go from picture A to

B, as depicted in Figure 2. The dissolve/fading occurs when the proportion of two image signals are combined, making them appear blended when shown on screen, as depicted in Figure 3. Dissolve occurs only when the contribution of picture A on screen falls from 100% to zero and the opposite (zero to 100%) takes place in picture B; when picture A is a fade-in, this picture appears to be in a solid color. However, when picture B is a fade-out, this picture appears to be in a solid color [1]. Wiping involves having a virtual boundary across the screen, from which the old scene is replaced with a new one [2].

The literature provides many types of scene change detection systems. Computational schemes define the measure of similarity between two consecutive frames; scene change is detected if the measure reaches the given threshold. Furthermore, all video types are not well served by a fixed threshold value mainly because of their diverse features. For a fixed threshold, the major issue is

*Corresponding author Email: yusra.salih@univsul.edu.iq (Y. A. Salih)

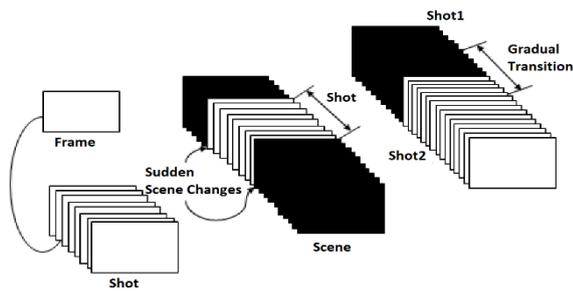


Figure 1. A hierarchical description of a video sequence



Picture (A) Picture (B)
Figure 2. Abrupt transition (cuts)

to obtain an optimal value. There is a high possibility that certain cuts will remain undetected if it is set too high; if it is set too low then false detection will arise. In this paper, we offer several techniques for overcoming this problem by extracting local statistical properties (mean value and standard deviations). We use them to further identify an automatic threshold that continuously updates. The remainder of this paper is organized as follows: Section 2 presents the literature review while Section 3 defines the research methods. Section 4 describes the experimental results; Section 5 is the discussion. Finally, Section 6 concludes the main points and suggests future research studies.

2. LITERATURE REVIEW

In this section, previously conducted research related to scene change detection in video files will be presented. Yi et al. [3] presented a new crude but effective way of identifying abrupt scene change by relying on pixel values only. They formulated a two-phase procedure. Initially, the mean absolute frame differences (MAFD) were tested using a relaxed threshold, where approximately 90% of the non-scene change frames were discarded. For the remaining 10% of frames, a normalized histogram equalization process was employed. To some extent, their approach is exempt from sharp illumination changes, moving objects and camera motion.

Gao et al. [4] proposed a PCA-based approach for video scene change detection in compressed video. Based on



(a)



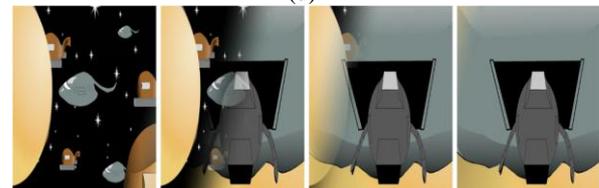
(b)



(c)



(d)



(e)

Figure 3. Gradual transitions types: (a) Dissolve, (b) Fad - in, (c) Fad-out, (d) Wipe 1 and (e) Wipe 2

the strong similarity of two adjacent video frames, they suggested that only the eigenvector with the largest eigenvalue is kept in the main component analysis (PCA) of video data. PCA displays a higher performance in comparison with the histogram and pixel feature methods. Based on this PCA feature, the detection algorithm is then made to detect those transitions that are both abrupt and gradual. The results demonstrate further improvement in recall and precision for scene change achieved through their detection algorithm.

Ding and Yang [5] offered adaptive group-of-pictures (GOP) and scene change detection approaches depending on existing H.264 advanced video coding information. The presented adaptive GOP detection (AGD) with SCD methods can normally raise the peak signal-to-noise ratio (PSNR) by 0.62 dB compared to the H.264/AVC that is run with a fixed GOP size. In addition, the scene changes

the detection rate of the suggested SCD reached a value of 98%.

Adhikari et al. employed one of the color histogram methods to underline the simulation of video shot boundary detection. In this method, an added feature was the scaling of the histogram metrics. They assessed the difference between the histograms of two successive frames. They conducted a simulation of the rapid change in video, resulting in 90% recall and precision values [2]. Mittal et al. formulated a method to distinguish between motion and image structures that are repetitive and permanent compared to those that are “new”. They suggested the use of a proper subspace made from image structures in order to utilize the appearance characteristics of those scenes. In addition, they made use of a prediction strategy in such a subspace to capture the dynamic characteristics. Because the model needs to adjust to long-term variations in the background, they recommended an incremental method, in which model parameters can quickly adapt themselves online. Such adaptive models lead to reliable and effective measures for scene detection, in which motion and structural changes are taken into account. What determines the efficient performance of the suggested approach, particularly in complex backgrounds, is the positive experimental result compared with the existing background modeling/subtraction methods [6].

Haberdar and Shah [7] suggested a new refinement method in a surveillance context in order to detect objects of interest in a dynamic open space. They used a mobile camera to capture two video sequences taken from distinct perspectives and times. They employed both texture and intensity gradient features to detect regions of interest. The results demonstrated the ability of this method to detect objects of different size and texture. This is in addition to the ability of this method to identify whether or not the scene changes, which leads to a decrease in false positives.

Radwan et al. [8] proposed an innovative and simple scene change detection algorithm that depends on the correlation between the video frames. They took as a reference to the video’s first frame. They computed the correlation between the histograms of the first frame and the remaining video frames. Their findings demonstrate that this approach is functional for motion as well as abrupt and gradual shot transition detection. It achieves an F-measure—a measure that combines both recall and precision—that exceeds a value of 0.89 for gradual shot transition in comparison with a value of 0.84 when using a PCA-based method.

Chauhan et al. [9] formulated an efficient scene change detection approach for uncompressed video. In this method, they used a canny edge detector in consecutive frames after dividing the frames into blocks. A change in the number of pixels (ones) in each block of successive frames indicates the occurrence of a scene

change. Here, a combined approach is presented wherein scene change detection is used alongside block-based motion estimation algorithms (BME) in the compression of video. The simulation result showed that this combined approach decreased the cost of computational complexity compared with NTSS, 4SS and DS algorithms. In addition, this approach gave around the same PSNR result.

Sonal and Bhide [10] discuss the various ways used to detect a shot boundary relying on the contents and the change in contents of the video. The significant information must not be missed because the key frames need to be processed for annotation purposes. This study considered the Histograms Differences as the most reliable method employed for image comparison. Shot boundaries can be detected by using this method to find those images whose histograms change greatly compared with the previous image histogram.

Sakurada and Okatani [11] developed a unique change detection approach. This approach employs features of a convolutional neural network (CNN) together with super-pixel segmentation. A low-resolution map of scene changes, which is reliable for detecting illumination changes and perspective differences, is obtained from a comparison of CNN features. A low-resolution map of scene changes is combined with super-pixel segmentation of the scene images to estimate the exact segmentation boundaries of the changes. The aim is to create an approach to detect city-scale changes that can be employed to visualize the damage resulting from natural catastrophes and to enrich the 3D model of any city. They have established a Panoramic Change Detection Dataset, which can be used by the public to assess the performance of change detection methods in such settings. The strength of this approach lies in the efficiency of the experimental results obtained from the dataset.

Singh et al. [12] developed a real-time scene change detection system with a very high frame rate by using a VLSI design to yield maximum performance. This was made possible by proposing, designing, and implementing area-efficient scene change detection VLSI architecture on an FPGS-based IDP Express platform. This system has the ability to process 2000 frames per second for 512×512 video resolution. The system can also render at high speed live video streams captured by cameras. It is also worth mentioning that this system is able to adapt to different video resolutions and frame rates.

Bulut and Osmani [13] created a method wherein they firstly calculated the color histogram values of consecutive frames. The scene change is detected when variation in the histogram values of a pair of frames in a row goes beyond a threshold value. In the experimental studies, the following palettes are applied to a group of video files and compared with each other: 3-Bit RGB

(Red Green Blue), 6-Bit RGB, 8- Bit RGB, 9-Bit RGB, 1-Bit Binary, 4-Bit Gray, and 8-Bit Gray. They employed the metrics that follow in comparing the palettes: accuracy, precision, recall, and F1- Score performance. Among all the palettes in the experimentation, the 6-Bit RGB color palette with a threshold level value of 35% was considered to be the best [13].

Thakur [14] developed non-static, dual threshold-based schemes in order to detect sudden scene change in a video. In this paper, the schemes proposed to employ dynamic and static thresholds introduced over a set of features, which contain Mean Squared Error (MSE), Entropy, and Count of displaced blocks. Wide-ranging experiments were carried out so as to indicate the values for the proposed thresholds to effectively detect with good accuracy the sudden scene alterations in videos .

Dayou and Jongweon [15] presented a new video scene change detection system focused on deep learning architecture and matching the SIFT feature key points. The SIFT algorithm is adopted for the extraction of image matching features and for the elimination of failure detection. The tests were performed on many video types. The proposed system could detect the changes in scene and classify frames into scene units at a low rate of errors.

3. MATERIAL AND METHOD

In this research, several detectors are used for scene change detection, which will be discussed in the following sections:

3. 1. Detector Type

3. 1. 1. Absolute Frame Difference The main idea behind the algorithm is that the intensity of the pixel changes significantly at the scene change. Firstly, the color image (frame) is separated into three color components: red, green and blue. Then, the absolute difference is computed over the intensity values of corresponding pixels of two consecutive frames as shown below:

$$D_{red}(i, j) = |f_k^r(i, j) - f_{k-1}^r(i, j)| \tag{1}$$

$$D_{green}(i, j) = |f_k^g(i, j) - f_{k-1}^g(i, j)| \tag{2}$$

$$D_{blue}(i, j) = |f_k^b(i, j) - f_{k-1}^b(i, j)| \tag{3}$$

Here, $f_k^r(i, j)$ is the red pixel intensity of the current frame at position (i, j) and $f_{k-1}^r(i, j)$ is the red pixel intensity of the previous frame at the same position. The threshold value is determined by the following two steps:

- A new feature is defined as C_{red} , C_{green} and C_{blue} ; they represent the number of red, green and blue pixels as shown in Equations (1), (2) and (3) which

are greater than some predefined value determined through trial and error.

- If C_{red} or C_{green} or $C_{blue} > T$, where T is some threshold expressed as 10 % of the frame size, then shot change is supposed.

3. 1. 2. Mean Absolute Frame Difference A test of dissimilarity on consecutive frames is often used to detect changes in the scene. The test should differentiate between true scene change and non-scene changes. Initially, each frame in the video sequence is subsampled to speed up the scene change detection process. Then, color space reduction is applied to convert color images into grayscale levels. A simple test for the k^{th} frames are mean absolute frame differences (MAFD), as illustrated by Equation (4).

$$MAFD_K = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N |f_k(i, j) - f_{k-1}(i, j)| \tag{4}$$

In the above equation, M and N are the width and the height of the frames $f_k(i, j)$ the pixel intensity at position (i, j) of k^{th} frames, and $f_{k-1}(i, j)$ is the pixel intensity at the same position of the frame $k - 1^{th}$. The first contribution of this work is the use of a dynamic threshold technique that corresponds with the sequence characteristics and does not need to be determined before detection and after the entire sequence. The method is based on computing the MAFD between the first frame and the next frame. The obtained values are used to construct the threshold criteria. After the first scene change is detected, the threshold value is automatically updated and involved in the next iteration of the detection process. The threshold Equation (5) is as follows:

$$T = C \times (MAFD) \tag{5}$$

Here, C is the constant value. Various values of C are tested for better performance, such as 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5. Experimental results show that the optimal value of C against largest F1- Score value is equal to 4 for all tested video samples (V1, V2, V3 and V4). As depicted in Figure 4.

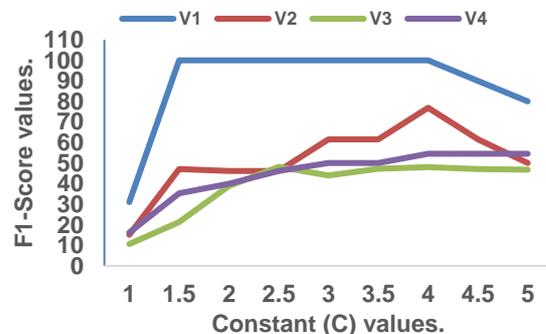


Figure 4. F1-Score values vs constant (C) values

3. 1. 3. Mean Histogram Absolute Frame Difference

To determine whether two shots are separated by an abrupt cut or gradual transition, a search strategy is used for identifying the difference between adjacent frames. MHAFD is the most common feature-based method used for this purpose. A histogram is used to determine the variation of pixels at different image intensity frequencies. For gradual transition and cut, the changes in the intensity histogram of a motion scene can occur gradually or sharply while they can be virtually constant in the case of no scene change. The basic idea of the proposed method is summarized as follows:

- Convert two consecutive frames $f_k(i, j)$ and $f_{k-1}(i, j)$ from RGB color space into gray scale levels.
- Calculate the absolute frame difference using the below Equation (6).
- Compute the histogram of the absolute frame difference, as in Equation (7).

$$D_{gray}(i, j) = |f_k(i, j) - f_{k-1}(i, j)| \tag{6}$$

$$H_d(l_k) = n_k \tag{7}$$

Here, l_k is the k^{th} gray scale level and n_k is the number of pixels in the difference frame whose intensity level is l_k . Then, the mean value of the histogram absolute frame difference is calculated using Equation (8).

$$M_{H_d} = \frac{1}{n} \sum_1^n H_d \tag{8}$$

Here, n is the number of grayscale levels l_k . In the threshold selection technique, the second contribution of the proposed method is provided. It is assumed that a sharp cut occurs when the difference between the two frames is relatively large. Thus, it is possible to identify sharp cuts and shot changes as a single peak in the time series of the plot of the proposed MHAFD. A new threshold criterion is developed based upon the detection of these peaks, as illustrated in Figure 5.

To develop a dynamic threshold technique, the method of utilizing some stastical values such as mean

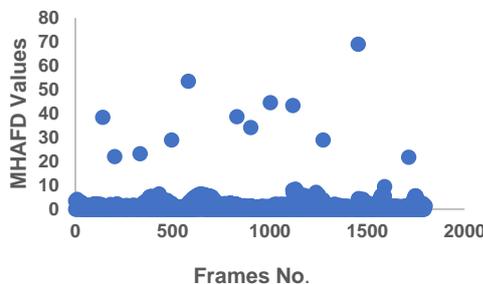


Figure 5. Relationship between MHAFD and frame numbers

and standard deviations is used as described in Equation (9), and this effectively detects sharp cuts without any false or missed detection.

$$\left| \frac{X_i - M}{SD} \right| > |\pm 3| \tag{9}$$

Here, X_i represents the y-axis data(Figure 5), M is the mean and SD is the standard deviation.

3. 2. Maximum Gradient Value This algorithm depends on computing the image gradient. By determining the derivative of the intensity value across the image pixels and finding those points with maximum derivative, the scene change between successive frames can be detected. The basic steps of the suggested method are summarized as follows:

- Split the color image (frame) into three corresponding channels: red, green and blue.
- For each color channel, find the components of the gradient using Equations (10) and (11).

$$\Delta x = f(i, j) - f(i, j + 1) \tag{10}$$

$$\Delta y = f(i, j) - f(i + 1, j) \tag{11}$$

- Determine the maximum intensity change across the Δx and Δy for each color channel individually. Then, count the number of pixels within some predefined value.
- Find the difference between the obtained counted pixels for each two successive frames f_k and f_{k-1} . Then, compare it with a selected threshold, which is determined by using Equation (9).

3. 3. Proposed Approach Framework In this work, many detectors are used to detect scene changes in video sequences. The general flowchart of the proposed approach is presented in Figure 6.

4. RESULTS AND ANALYSIS

In this section, the test video samples and the involved parameters which affect the detection accuracy, Precision, Recall, and F1-Score measure are described in the following sub-sections.

4. 1. Test Samples A set of uncompressed test video samples with various resolutions, video length, and frame rate that used to analyze the performance of the proposed algorithm². Also, the test video sequences are classified into two categories: motion types and scene change types. Fast Object Motion (FOM) and Slow Object Motion (SOM) describe the forms of motion. Furthermore, scene change types are considered by using Gradual Scene Change (GSC) and Abrupt Scene Change (ASC). The sample test sequences are described in Tables 1 and 2, and Figure 7.

² ftp://vqeg.its.bldrdoc.gov/HDTV/NTIA_source.

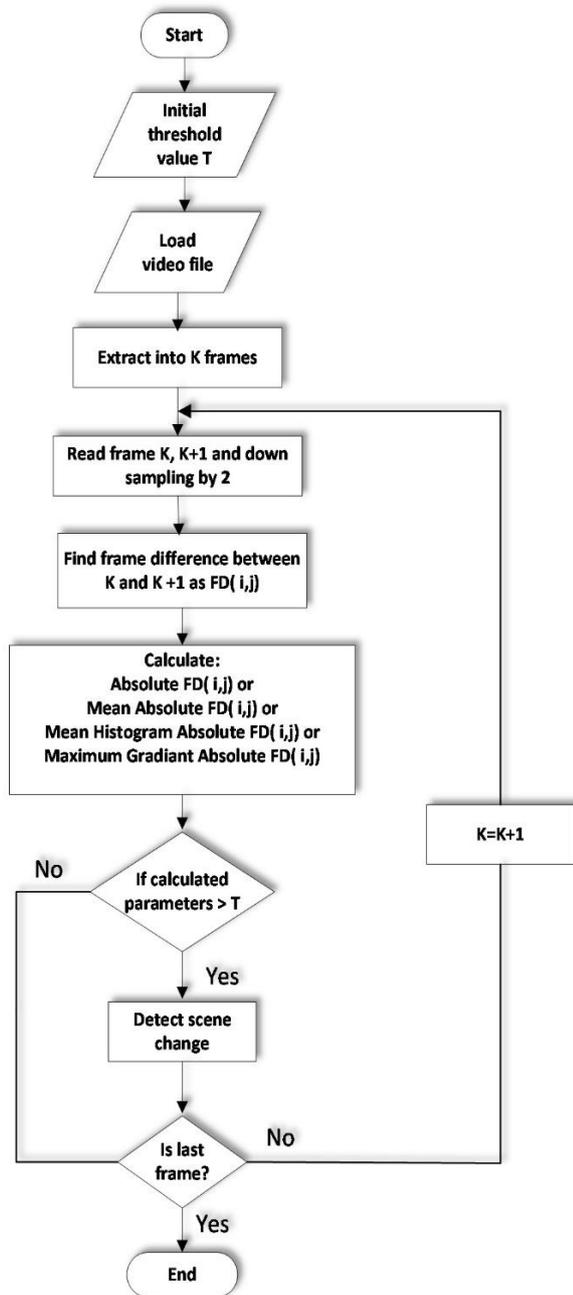


Figure 6. Proposed approach flowchart

TABLE 1. Test video sample description

Test Sequences	Name	No. of Frames	Frame Rate(f/s)	Length (Seconds)
Aspen.avi	V1	570	30	00:00:19
Controlled burn.avi	V2	570	30	00:00:19
Spring.avi	V3	1770	30	00:00:59
BigBunny.avi	V4	325	25	00:00:13

TABLE 2. Test video samples categories

Test Sequences	Resolution	Motion & Scene Type
Aspen.avi	1080×1920	ASC & FOM
Controlled burn.avi	1080×1920	ASC & GSC
Spring.avi	854×480	ASC & FOM
BigBunny.avi	1280×720	ASC & SOM& GSC



Figure 7. Test video samples

4. 2. Performance Parameters

Recall and Precision are the two commonly used measurements in most scene change detection algorithms. These measures are determined as described in Equations (12) and (13).

$$Precision = \frac{N_c}{N_c + N_f} \tag{12}$$

$$Recall = \frac{N_c}{N_c + N_m} \tag{13}$$

Here, N_c represents the number of correctly detected scenes, N_f is the number of false alarms and N_m denotes the number of missed detections. F1-Score is a common measure combining both Recall and Precision, given in Equation (14):

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{14}$$

4. 3. Test Results

The test results of Precision, Recall, F1-Score measure and computational time for four test video samples are presented in the next sub sections. The exact number of scene changes for each test video sample is manually determined, which can be considered as a ground truth. Based on the scene numbers, one can determine the number of missed detections and false alarms.

4. 3. 1. AFD Test Results

In this section, results of Precision, Recall, F1- Score measures and computational time are presented. According to Table 3, the AFD method is optimal in terms of Precision for videos (V1, V3, and V4). However, fixed threshold conditions are applied to all test video samples, except

for V2, which can be observed from the number of false alarms and missed detections; the detection measurements are not optimal. The reason underlying this is that the video content type has the dissolve type scene changes.

4.3.2. MAFD Test Results In Table 4, the MAFD detector test results are presented. Compared with the previous AFD detector, an improvement in the computational time of every test video sample is evident. As shown in Table 4, another enhancement is the reduction of false alarms compared with that presented by V2 in Table 3.

4.3.3. MHAFD Test Results False alarms have a direct impact on the Precision value, which is, in turn, affects the F1-Score value. Observing these values in Table 5, it is evident that the false alarms corresponding to V2 are too high and therefore the Precision and F1-Score values are low for that test video sample. This occurred because each test video sample is subjected to a fixed value of the threshold, which is acceptable for that video type.

To overcome the problem of the static threshold rule, an adaptive threshold approach was developed as mentioned in section 3.1.3. Table 6 presents the

TABLE 3. Test results of AFD detector

Test Sequencess	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	10	5	13	3
N_f	-	10	-	-
N_m	-	3	2	-
Precision %	100	33.333	100	100
Recall %	100	62.5	86.666	100
F1-Score %	100	43.478	92.857	100
Avg.Time (sec.)	0.0153	0.0146	0.0023	0.0052

TABLE 4. Test results of the MAFD detector

Test Sequencess	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	10	5	12	3
N_f	-	-	24	5
N_m	-	3	3	-
Precision %	100	100	33.333	37.5
Recall %	100	62.5	80	100
F1-Score %	100	76.923	47.0587	54.545
Avg.Time (sec.)	0.0080	0.0072	0.0012	0.0026

TABLE 5. Test results of MHAFD with a fixed threshold

Test Sequencess	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	10	7	14	3
N_f	-	30	-	-
N_m	-	1	1	-
Precision %	100	18.918	100	100
Recall %	100	87.5	93.333	100
F1-Score %	100	31.109	96.551	100
Avg.Time (sec.)	0.0075	0.0074	0.00123	0.00237

simulation results of using MHAFD with an adaptive threshold value. As shown in Table 6, MHAFD is optimal in terms of Precision, Recall and F1-Score measure for videos (V1, V3, and V4). In addition, the F1-Score value for V2 test video sample is increased compared with that in Table 5. Also, there is a significant reduction in the computational time of all test video samples.

4.3.4. MGVS Test Results The simulation results of implementing MGVS with a static threshold method are presented in Table 7. As shown in Table 8, when

TABLE 6. Test results of MHAFD with a dynamic threshold

Test Sequencess	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	10	5	15	3
N_f	-	10	-	-
N_m	-	3	-	-
Precision %	100	33.333	100	100
Recall %	100	62.5	100	100
F1-Score %	100	43.475	100	100
Avg.Time (sec.)	0.0071	0.0069	0.0014	0.0022

TABLE 7. Test results of MGVS with a fixed threshold

Test Sequencess	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	10	7	13	2
N_f	-	12	-	-
N_m	-	1	2	1
Precision %	100	36.842	100	100
Recall %	100	87.5	86.667	66.667
F1-Score %	100	51.852	92.857	79.999
Avg.Time (sec.)	0.229	0.229	0.039	0.095

TABLE 8. Test results of MGV with an adaptive threshold

Test Sequences	V1	V2	V3	V4
No. of Actual Scenes	10	8	15	3
N_c	9	5	13	2
N_f	-	4	-	-
N_m	1	3	2	1
Precision %	100	55.55	100	100
Recall %	90	62.5	86.667	66.667
F1-Score %	94.737	58.82	92.857	79.999
Avg.Time (sec.)	0.245	0.236	0.0549	0.103

applying the adaptive threshold method on the test video samples, there is a reduction in the number of false alarms obtained in V2.

In contrast, there is an increase in the number of missed detections for V1 and V2. Also, the average time of this technique is higher than all other techniques, as demonstrated in Tables 7 and 8.

5. DISCUSSION

Proposed scene change detectors were tested on a set of video samples. The test results indicate that the drawback of using a fixed threshold value is the detection of a high number of false alarms. Also, the proposed approach has a low detection performance because all the test video samples are subjected to standard judgments. Moreover, the manual selection of threshold value by a trial-and-error method is considered time-consuming. To overcome the above-mentioned issues, a dynamic threshold technique was developed to improve the F1-Score measure of some video samples due to the reduction of the number of false alarms. The Precision, Recall and F1-Score values of video samples tested with four proposed techniques are shown in Tables 3, 4, 6 and 8. The test results demonstrate that the Precision, Recall and F1-Score measure values have an optimal value, approximately 100% for all videos with hard cuts, high movement scenes and sudden object appearance with slow object movement (V1, V3, and V4). The Recall value is in the range of 62.5 to 100 for all test video samples due to the number of missed detections, which is considered acceptable.

Another improvement in Precision, Recall and F1-Score values occurs through the use of an adaptive threshold technique, especially for a video sample (V2) with a graduation transition (Dissolve) scene change. Results in Tables 6 and 8 indicate that the number of false alarms is reduced from 30 to 10, and the F1-Score value increases from 31.109 to 43.475 in Table 6. Whereas the

number of false alarms is reduced from 12 to 4, and the F1-Score value increases from 51.852 to 58.82 in Table 8. Finally, the optimal average time among all detectors is obtained by the proposed method MHAFD as presented in Table 6. Figures 8, 9 show the Precision and Recall values for all detectors applied the test video samples (V1, V2, V3, and V4) respectively.

As shown in Table 9, compared to these benchmark methods [2, 16-18], the proposed approach further enhanced the accuracy of the scene change detection. In particular, the F1 score of the proposed method was as much as 10.592 (an improvement of 11.847 percent), 7.994 (an improvement of 8.689 percent), 5.708 (an improvement of 6.054 percent), 1.126 (an improvement of 1.139 percent) higher than that of methods 2, 17, 18, 19, respectively.

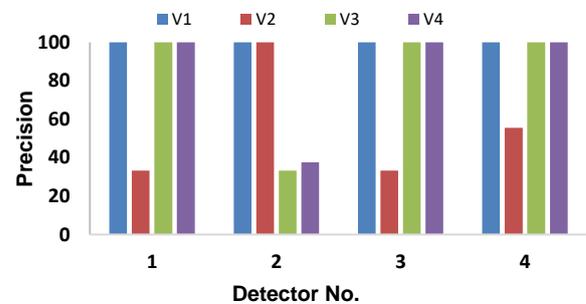


Figure 8. Precision values vs detector types

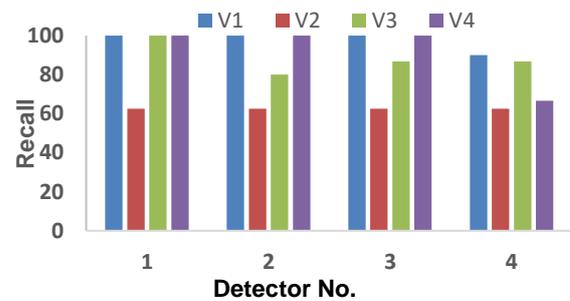


Figure 9. Recall values vs detector types

TABLE 9. F1-Score comparison between the proposed (MHAFD) and previous schemes

Schemes	Precision %	Recall %	F1-Score %
[2]	90.47	88.37	89.408
[17]	95.8	88.5	92.006
[18]	97.18	91.57	94.292
[19]	99.15	98.60	98.874
Proposed MHAFD	100	100	100

6. CONCLUSION

In this research work, an adaptive threshold approach based on scene change detection was developed to address the problem of manual threshold selection (fixed value). To reduce the number of false alarms obtained in video with dissolve effects, an adaptive threshold method was applied. The results show that the Precision, Recall and F1-score measure values are 100% when MHAJD is applied to sample videos with hard cuts (V1 and V3) and gradual transition (wipe1-V4). For the dissolving type, a Precision value of 100% is achieved using the MAFD detector. Test results indicate that the minimum average time is acquired when MHAJD is applied to all the video samples, as compared with the rest of the detectors. Future work will use other techniques to detect in advance the change of scenes by adding some conditions which are capable of recognizing the scene type either hard (cut) or graduation transition. Depending on the scene types, one can select the appropriate detectors.

7. REFERENCES

1. Fernando, W.A.C., Canagarajah, C.N. and Bull, D.R., "A unified approach to scene change detection in uncompressed and compressed video", *IEEE Transactions on Consumer Electronics*, Vol. 46, No. 3, (2000), 769-779.
2. Adhikari, P., Neeta, G., Jyothi, D. and Hogade, B., "Abrupt scene change detection", *Intelligent Links*, Vol. 35, (2008).
3. Yi, X. and Ling, N., "Fast pixel-based video scene change detection", in 2005 IEEE International Symposium on Circuits and Systems, IEEE., (2005), 3443-3446.
4. Gao, L., Jiang, J., Liang, J., Wang, S., Yang, S. and Qin, Y., "Pca-based approach for video scene change detection on compressed video", *Electronics letters*, Vol. 42, No. 24, (2006), 1389-1390.
5. Ding, J.-R. and Yang, J.-F., "Adaptive group-of-pictures and scene change detection methods based on existing h. 264 advanced video coding information", *IET Image Processing*, Vol. 2, No. 2, (2008), 85-94.
6. Mittal, A., Monnet, A. and Paragios, N., "Scene modeling and change detection in dynamic scenes: A subspace approach", *Computer Vision and Image Understanding*, Vol. 113, No. 1, (2009), 63-79.
7. Haberdar, H. and Shah, S.K., "Disparity map refinement for video based scene change detection using a mobile stereo camera platform", in 2010 20th International Conference on Pattern Recognition, IEEE. (2010), 3890-3893.
8. Radwan, N.I., Salem, N.M. and El Adawy, M.I., Histogram correlation for video scene change detection, in *Advances in computer science, engineering & applications*. 2012, Springer.765-773.
9. Chauhan, A.P., Parmar, R.R., Parmar, S.K. and Chauhan, S.G., "Hybrid approach for video compression based on scene change detection", in 2013 IEEE International Conference on Signal Processing, Computing and Control (ISPPCC), IEEE. (2013), 1-5.
10. Waghmare, M.S.P. and Bhide, A., "Shot boundary detection using histogram differences", *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, Vol. 3, (2014), 1460-1464.
11. Sakurada, K. and Okatani, T., "Change detection from a street image pair using cnn features and superpixel segmentation", in *BMVC*. (2015), 61.61-61.12.
12. Singh, S., Saini, R., Saurav, S., Tanwar, P., Raju, K.S., Saini, A.K., Chaudhury, S. and Ishii, I., "High frame rate real-time scene change detection system", in *International Conference on Computer Vision, Graphics, and Image processing*, Springer. (2016), 157-167.
13. Bulut, F. and Osmani, S., "Scene change detection using different color pallets and performance comparison", Vol., No., (2017).
14. Thakur, M.K., "Dynamic dual threshold based schemes for abrupt scene change detection in videos", *Journal of Theoretical and Applied Information Technology*, Vol. 96, No. 14, (2018).
15. Jiang, D. and Kim, J., A scene change detection framework based on deep learning and image matching, in *Advanced multimedia and ubiquitous engineering*. 2018, Springer.623-629.
16. Kang, S.-J., "Adaptive luminance coding-based scene-change detection for frame rate up-conversion", *IEEE Transactions on Consumer Electronics*, Vol. 59, No. 2, (2013), 370-375.
17. Feng, H., Fang, W., Liu, S. and Fang, Y., "A new general framework for shot boundary detection and key-frame extraction", in *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*. (2005), 121-126.
18. Chasanis, V., Likas, A. and Galatsanos, N., "Simultaneous detection of abrupt cuts and dissolves in videos using support vector machines", *Pattern Recognition Letters*, Vol. 30, No. 1, (2009), 55-65.

Persian Abstract

چکیده

فشرده سازی فیلم به مرجعی برای تحقیقات مختلف تبدیل شده است. رفع محدودیت های پهنای باند کانال و پاسخ به افزایش تقاضای فیلم، از جمله کتابخانه های دیجیتال و پخش رسانه ها از طریق اینترنت ضروری است. یک ویدئو تعدادی فریم است که توسط یک دوربین ضبط می شود در حالی که یک صحنه مجموعه ای از فریم های متوالی است که از یک دیدگاه روایی خاص گرفته شده است. برای فشرده سازی یک فیلم، ابتدا فریم های داخلی با استفاده از روش های تشخیص تغییر صحنه از فریم های داخلی جدا می شوند. سپس از الگوریتم های برآورد حرکت مبتنی بر بلوک استفاده می شود تا افزونگی زمانی بین فریم های پی در پی را از بین رود. در این مقاله برخی از روشهای تشخیص تغییر صحنه برای استفاده از فیلم غیر فشرده برای شناسایی انواع صحنه مانند برش، حل شدن، پاک کردن و غیره توضیح داده شده است. برای تشخیص تغییرات دقیق صحنه در هر دو صحنه با حرکات کم و زیاد، روشهای تفاضل فریم مطلق، تفاضل فریم مطلق میانگین، تفاضل فریم مطلق هیستوگرام میانگین و مقدار گرادیان بیشینه بر روی انواع مختلف ویدئو آزمایش می شوند. نتایج آزمون نشان می دهد که رویکرد پیشنهادی در صورت استفاده از تفاضل فریم مطلق هیستوگرام میانگین، به خصوص برای برش ها و انتقال تدریجی (پاک کردن) انواع ویدئو، دقت بهتر و مقدار $F1\text{-Score} = 100\%$ ارایه می دهد، به خصوص برای برش ها و انتقال تدریجی (پاک کردن) انواع ویدئو. تغییر صحنه با دقت بالای 100٪ (یعنی عدم تشخیص کاذب) با روش ارایه شده تشخیص داده می شود. علاوه بر این، از لحاظ پیچیدگی زمانی برای تجزیه و تحلیل تمام نمونه های ویدئویی، روش پیشنهادی در مقایسه با آشکارسازهای انتخاب شده بهترین نتیجه را فراهم می نماید.