A Robust Scene-Change Detection Method for Video Segmentation

Chung-Lin Huang and Bing-Yao Liao

***Abstract*—This paper proposes a new method that combines the intensity and motion information to detect scene changes such as abrupt scene changes and gradual scene changes. Two major features are chosen as the basic dissimilarity measures, and self- and cross-validation mechanisms are employed via a static scene test. We also develop a novel intensity statistics model for detecting gradual scene changes. Experimental results show that the proposed algorithms are effective and outperform the previous approaches.**

***Index Terms*—Abrupt scene changes, gradual scene changes, scene-change detection.**

I. Introduction

FOR video, a common first step is to segment the videos into temporal "shots," each representing an event or continuous sequence of actions. A shot represents a sequence of frames captured from a unique and continuous record from a camera. The main problem of segmenting a video sequence into shots is the ability to distinguish between scene breaks and normal changes that happen in the scene. These changes may be due to the motion of large objects or the motion of the camera. When special effects are involved, two shots are merged using gradual transition. The types of gradual transitions used mostly are dissolve, fade in, and fade out. A fade is a gradual transition between a scene and a constant image (fade out) or between a constant image and a scene (fade in). A dissolve is a gradual transition from one scene to another, in which the first scene fades out and the second scene fades in.

To segment a video sequence into shots, a dissimilarity measure between two frames must be defined. This measure must return a high value only when two continuous frames fall in different video shots. Dissimilarity measure is mainly based on a pixel-based methods [1] and histogram-based methods [2], [3]. The pixel-based methods are highly sensitive to motion of objects. Histogram-based methods provide a better tradeoff between accuracy and speed, and its performance is good for the case of abrupt scene changes such as cuts. The best performance is obtained by χ^2 test [3]. Unfortunately, in presence of dissolve, the difference between consecutive frames may be too low to be misinterpreted as a difference due to motion. Operations on fully decompressed or uncompressed video do not permit rapid processing because of the data size. Other algorithms [4], [5] are developed to operate directly on MPEG compressed data without having to first perform full frame decompression.

The two main problems in most existing algorithms are: 1) they are threshold-dependent algorithms and 2) they suffer false detection with scenes involving fast camera or object motion. The histogram-based methods ignore the spatial distribution of the luminance or colors. Consecutive frames may have different spatial distribution, but similar histograms, and they are declared to belong to the same shots. This paper proposes a scene-change detection algorithm with three contributions: 1) relaxing threshold selection problem; 2) higher detection rate (i.e., scene change should not be missed); and 3) lower false alarm rate.

Manuscript received January 23, 2001; revised September 28, 2001. This work was supported by the National Science Council, Taiwan, R.O.C., under Grant NSC 89-2213-E007-081. This paper was recommended by Associate Editor K. Aizawa.

The authors are with the Electrical Engineering Department, National Tsing-Hua University, Hsin-Chu, Taiwan, R.O.C. (e-mail: clhuang@ee.nthu.edu.tw).

Publisher Item Identifier S 1051-8215(01)11032-3.

II. Abrupt Scene-Change Detection

Here, we will explain how to choose the basic features for the dissimilarity measure between frames and how to apply static scene test to characterize the scene-change type and to facilitate the detection process.

A. Measurement of the Changes Between Frames

The effectiveness of detecting the scene changes depends on the suitable choice of similarity metric between two frames. Let X and Y be two frames, and their difference is denoted by $d(X,Y)$. Several different classes of metrics are discussed below.

1) Pixel-Based Difference: The first class relies on the measurement of corresponding pixel–pixel differences. It is not a good metric for change detection due to its sensitivity to object and camera motion. One alternative is to smooth the image in the beginning. Smoothing can be performed on the spatial images before the differences are taken. We choose the DC image difference [4] as our first basic dissimilarity measure for scene detection. The DC image difference is defined as

$$d_{\mathrm{DC}}(X,Y) = \sum_m \sum_n |C_X(m,n) - C_Y(m,n)| \qquad (1)$$

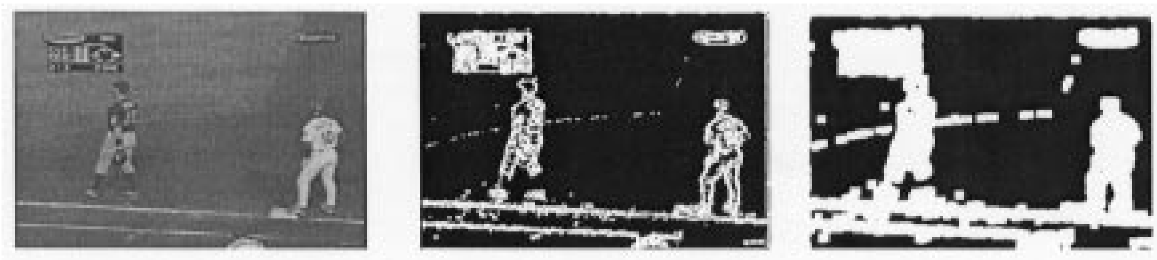
where $C_X(m,n)$ and $C_Y(m,n)$ are the DC image of frames X, and Y, which are denoted as $f_X(k,l)$ and $f_Y(k,l)$. They are divided into 8×8 blocks. The DC image of frame X is determined as

$$C_X(m,n) = \frac{1}{8} \sum_{i=0}^{7} \sum_{j=0}^{7} f_X(8m+i, 8n+j). \qquad (2)$$

2) Histogram-Based Difference: We survey several types of histogram based algorithms and find that χ^2 test has in general better performance with respect to other measures. So we adopt the χ^2 test [3] as our second feature

$$d_{\chi^2}(X,Y) = \sum_{j=1}^{K} \begin{cases} \frac{(H_X(j)-H_Y(j))^2}{\max(H_X(j),H_Y(j))}, & \text{if } (H_X(j) \neq 0) \cup (H_Y(j) \neq 0) \\ 0, & \text{otherwise} \end{cases} \qquad (3)$$

where $H_X(j)$ is the bin value of the histogram of frame X, and K is the overall number of bins.



(a) (b) (c)

Fig. 1. Example of edge detection: (a) the original image; (b) the edge image; and (c) dilated image with radius $r = 3$.

These two features are chosen for a number of reasons. First, they are easy to compute in the pixel domain. In addition, it is desirable to have both pixel- and histogram-based measures because they complement each other's weaknesses. Pixel-based techniques may generate false alarms whenever there are fast moving objects or fast camera movement in the video. Histogram-based techniques are fairly immune to these effects but they may miss scene changes if the luminance distribution of the frames does not change significantly. Here, we combine the two methods—as well as scene-transition analysis—and develop a robust scene-change method.

B. Static Scene Test

A static scene is characterized in the sense that all objects present in the scene exhibit rather small motion compared to the frame size, and global movement caused by the camera is slow and smooth. In a static scene, we find a minimal variation of both the spatial and temporal information of the consecutive frames. The temporal visual content of static scene does not have significant change. On the other hand, in dynamic scene, there exist fast object movements or camera zooming and panning motion. The dynamic activities are difficult to formularize by analyzing the spatial and temporal difference of the pixels or histogram.

1) Edge Detection: The static scene test is based on a simple observation: the intensity of edges of a static scene will be preserved around the spatial location near the original place for a period of time. Inter-frame edge variations are not obvious, thus it permits us to track the positions of intensity edges by edge detection. We use a simple gradient operator (i.e., Sobel masks) to compute the gradient image [see Fig. 1(b)].

2) Edge Dilation: To handle the small motions of multiple objects in the transition from frame F1 to the next frame F2, the edge pixels of succeeding image F2 are dilated by a radius r [6]. The edge pixels of F2 are referred as 'entering' edge pixel with respect to F1. All entering edge pixels that are less than a distance r from the closest entering edge pixel in F1 are able to cover the original edge pixels of F1 by dilation, if the transition is smooth and static. Dilation with radius r is achieved by replacing each edge pixel with a diamond whose height and width are $2r+1$ pixels. Fig. 1(c) gives an example with $r = 3$. A covering ratio α is then defined to indicate the static level, which is shown in the following:

$$\alpha = 1 - \frac{\sum_{x,y} E\{F_1(x,y)\} \overline{E\{F_2(x,y)\}}}{\sum_{x,y} E\{F_1(x,y)\}} \tag{4}$$

where $E\{\cdot\}$ denotes the edge-detection operation, and bar denotes dilation.

A lower covering ratio means that these two frames are more similar to each other from edge information point of view. Radius r controls the covering range and how smooth a static transition is. The transition of two consecutive frames with covering ratio larger than a predefined threshold is considered as a non-static—or dynamic—scene.

C. Scene Transition Classification

We can further extend the concept of the static scene test to classify the abrupt scene transition types. By analyzing the static property of a number of frames preceding the detected peak of potential scene change, and those of the succeeding frames, we generalize the scene transition types into three main categories:

1) static scene to static scene;
2) dynamic scene to static scene or vice versa;
3) dynamic scene to dynamic scene.

D. Detection Algorithm

Based on [4], our detection algorithm starts with the local peak selection using a sliding window, since scene change is a local activity in the temporal domain. We declare two kinds of abrupt scene change: genuine and ambiguous, in order to relax the threshold problem. Two gray zones are allowed in our two-phase system, and the final decision for ambiguous declaration will be made in the second phase. The overall system architecture is shown in Fig. 2.

In the first phase, we locate the highest and the second highest peaks of DC image difference in the midst of the sliding window, and then calculate the ratio n between the first and second peaks. Two thresholds are used, a larger n_{high} and a smaller n_{low}. The parameter n is imposed to avoid false alarm against fast panning or zooming scenes. For genuine scene change with a larger n, we are sure that only those dramatic changes are detected and thus false detections are avoided. For ambiguous scene change with a smaller n, we may have to go through the second phase. In the second phase, we take advantage of the second measure for cross-validating the ambiguous one that may be a scene change with blunt peak or a false alarm due to fast camera and object motion.

While the local peak in the midst of the difference sequence indicates an ambiguous scene change, we first employ the second feature (histogram) measured across the peak. In the second phase, we also use two thresholding values to classify the scene types in terms of five different video transition types (see Table I). If it is smaller than the lower threshold, then type E is declared, which indicates no scene change. If it is larger

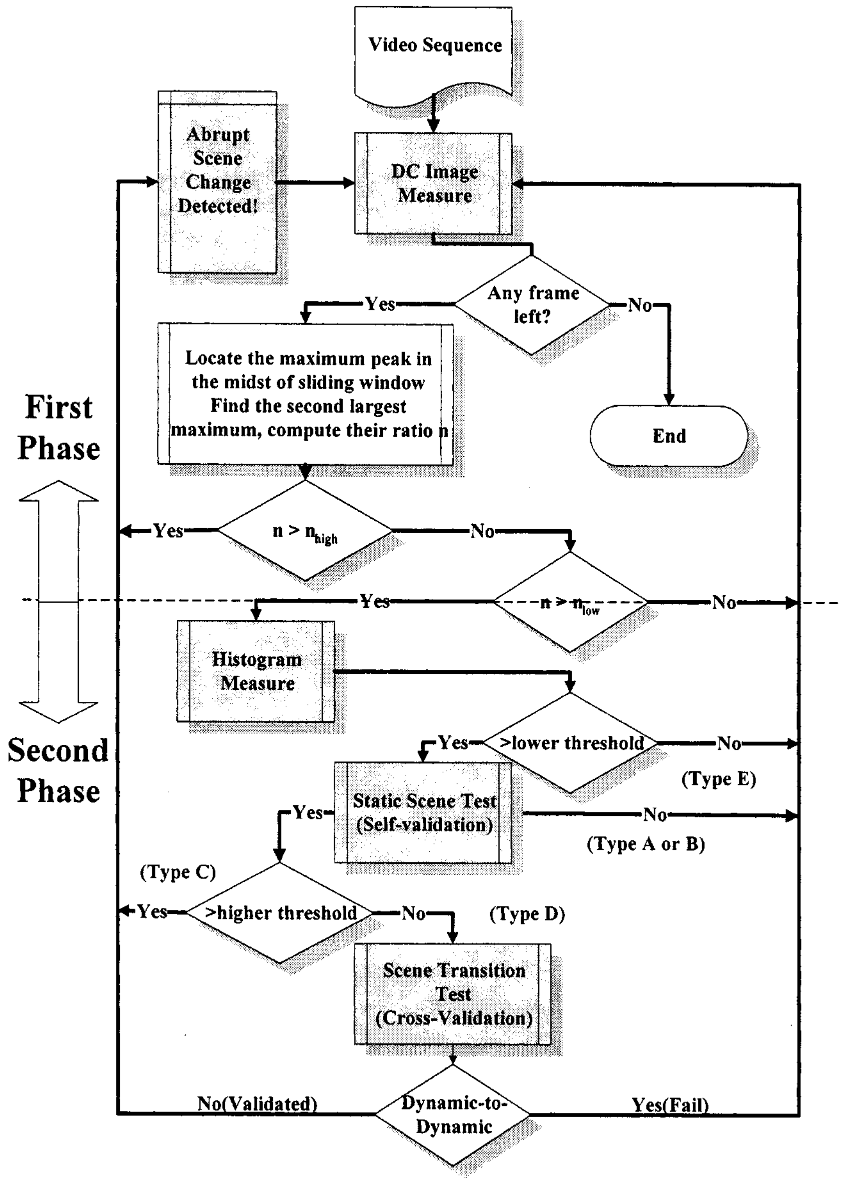


Fig. 2. Overall system of the abrupt change detection.

than the lower threshold, we employ the self-validation, which is basically a static scene test algorithm. It applies the static scene test algorithms (Section II-B) on the frames across the possible scene cut to differentiate whether the scene is static or dynamic. Second, if it is larger than the higher threshold and the scene is a dynamic scene (Type C), then a real scene cut is identified. Instead, if it is larger than the higher threshold and it is a static scene (Type A), then no scene cut is declared. Finally, if the histogram measure is between the smaller threshold and the larger threshold, then it may be types B or D. Still, the static scene test algorithm is required to differentiate the type B from type D. For type B, there is no scene cut declared, whereas for type D, we need to apply the cross-validation.

TABLE I
POSSIBLE VIDEO TRANSITION TYPES ACROSS THE AMBIGUOUS PEAK

Scene Test	Histogram measure thresholding	Scene Change	Transition Type
Static	≥ higher threshold	No	A
Static	≥ lower threshold < higher threshold	No	B
Dynamic	≥ higher threshold	Yes	C
Dynamic	≥ lower threshold < higher threshold	Ambiguous	D
Static or Dynamic	<lower threshold	No	E

For type D, we apply cross-validation to find out the possible scene transition type on both sides of the ambiguous peak [Section II-C]. From our observations of all the video sequences, human eyes cannot easily differentiate scene transitions such as dynamic-to-dynamic transitions. Normally, there are very few cases in video sequences that are found to be real scene changes with two dynamic scenes on the both sides of the peak. Dynamic-to-dynamic transition usually indicates continuous object or camera motion, rather than a real scene change. From our experiments, we find that we may declare a false alarm in this case. On the other hand, if the scene transition type is not a dynamic-to-dynamic, a real scene change will be declared. This second detector (histogram-based) uses the information of video transition to validate or invalidate the ambiguous scene cut declared by the first detector. Most of the false alarms declared by the histogram detector are due to sudden light changes, while the edge information is more or less invariant to these changes.

III. Gradual Scene-Change Detection

Gradual scene change consists of dissolve, fade-in, and fade-out. Here, we propose a novel detection algorithm based on an intensity statistics model.

A. Dissolve, Fade-In, and Fade-Out

The gradual transition in which the changes from frame n_1 to n_2 is much faster than those prior to frame n_1 and after frame n_2. Common form of gradual transition includes the special effect of dissolving, fading in, and fade out. A dissolve operation from scene X to Y in time duration T is a sequence of frames represented by

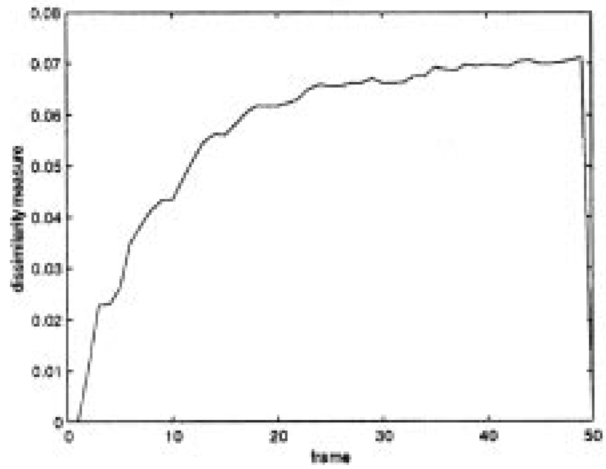
$$\frac{t}{T}Y + \left(1 - \frac{t}{T}\right)X, \quad 0 \leq t \leq T. \tag{5}$$

A fade-in is a special case with $X = 0$ and fade-out $Y = 0$. Comparison based on successive frames alone will not be useful for the detection of gradual scene transitions because changes are small in this case. Camera and object motions always introduce a larger variation than a gradual transition. Yeo's method [4] compares every frame with the following kth frame. Their method works for certain kinds of video sequences; however, its limitation is that the duration of detecting transition must be fixed. Besides, the "plateau" is not assuring flatness during the transition, and it will miss the gradual scene change. Next, we will develop a new approach to detect gradual scene change.

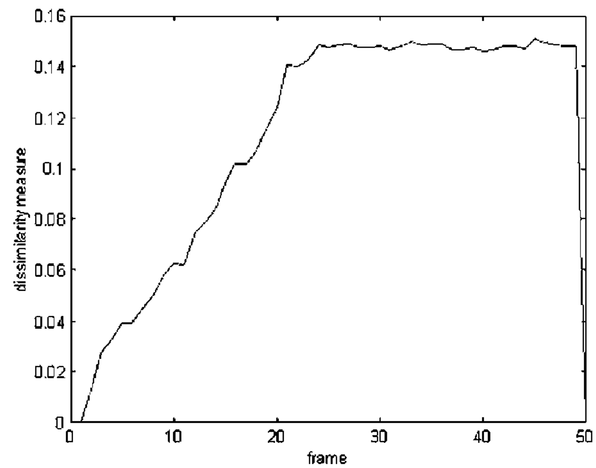
B. Intensity Statistics Model

The intensity similarity between two frames within a scene depends on their relative distance. The farther their distance is, the less correlation is observed. We develop an intensity statistics model based on this assumption. We assume that the correlation (similarity measure) between two frames within a shot is totally independent if their relative distance is long enough. We discover that, for any frames near the reference frame, their dissimilarity measure almost increases exponentially with their distance. For those who are "far" away from the reference frame, their difference measures are somehow limited. Since we assumed their independence with each other, the difference measures are randomly distributed but reaching a vibrated constant (with a small variation)—see Fig. 3.

We first define a "*seed*"' as the beginning frame of a gradual transition. Here, we find that the dissimilarity increases linearly with their distance during the transition. After the transition is over, the difference measures (over a temporal window distance N) are randomly distributed since the frames after transition are totally different from the *seed*. We also find that the difference measure during the transition is definitely nondecreasing, which is a very crucial property for our detection algorithm. Furthermore, we define the difference measure generated by comparing



(a)



(b)

Fig. 3. (a) Dissimilarity sequences using DC image difference for comparing a fixed frame with all its successive 49 frames. (b) Dissimilarity sequences for comparing a seed with all its successive 49 frames.

a frame with itself and its successive $(N-1)$ frames as the "N*-distance measure*"

$$\underline{M}_{N\text{-distance}} = \{d(X_i, X_j) | j = i, i+1, \ldots, i+N-1\} \tag{6}$$

where i denotes its frame number, $d(\cdot)$ represents the DC image difference measure.

The N*-distance measure* model shown in Fig. 4 is divided into four segments. In segment 1 (with a constant length C), frames within this segment are correlated to the reference frame X' to some degree and their differences increase with their distances. As the temporal distance increases, the monotonic increasing difference will finally become saturated. In segment 2, since the temporal distance from the frames to the reference frame is so long that there is almost no correlation between the frames in segment 2 and reference frame X'. However, the different measure in segment 3 is linear increasing. In segment 3 (a gradual transition with an unknown length G), the temporal distance from frame X and the reference frame X' is so long that $d(X, X')$ (difference between X and X') is random. The same scenario occurs for the rest of the frames Z in segment 3. Therefore, the distance measure of the frames in segment 3

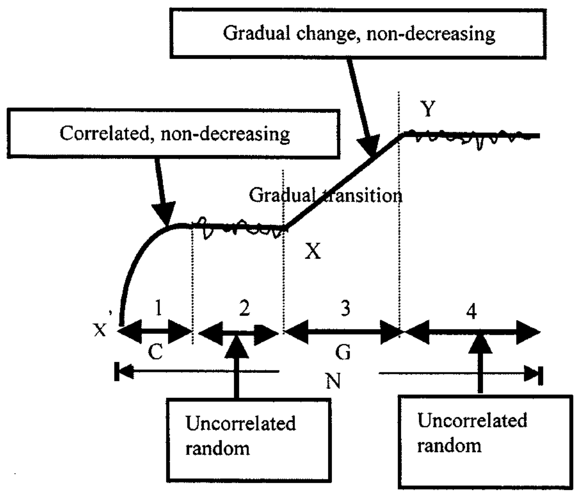


Fig. 4. Ideal model of the N-distance measure of a frame nears a seed.

has little relation with the reference frame X. The difference between Z and X depends on the dissimilarity measure between Z and X themselves, which increases with their relative distance. The statistics property of segment 4 is similar to segment 2.

C. Gradual Scene-Change-Detection Algorithm

The goal of our gradual scene-change detection is to locate the transitions in which the *N-distance measure* variations seem to be linear (or gradual). If we further differentiate the *N-distance measure*, it will become linear in segment 1 and a constant in segment 3, respectively. In segments 2 and 4, the difference distributions after differentiation are still random. To distinguish these segments, we may compute their "*zero crossing rate*". In segments 2 and 4 (with random characteristic), we may find high zero crossing rate, whereas in segments 1 and 3, we expect low zero crossing rate. Our method is illustrated as follows.

1) *N-distance measure*: For any frame, *seed* or *nonseed*, we may perform the N-distance measure using (6).
2) *Difference operation*: Form a difference operation on $\underline{M}$ to generate $\underline{M}'$ as

$$\begin{aligned}\underline{M}' &= \{d'(X_i, X_j, X_{j+1}) \,|\, j = i, i+1, \ldots, i+N-2\} \\ &= \{d(X_i, X_{j+1}) \\ &\quad - d(X_i, X_j) | j = i, i+1, \ldots, i+N-2\}. \quad (7)\end{aligned}$$

3) *Zero-crossing rate calculation*: Compute the zero crossing rate of $\underline{M}'$

If $d'(X_i, X_{k-1}, X_k) \times d'(X_i, X_k, X_{k+1}) < 0$ for $i \le k \le i+N-2$,
then a zero crossing at frame k is observed.

4) *Low-pass filtering*: Implement a simple low-pass filter to keep the low frequency segments and to remove the high-frequency segments.

For zero crossing at frame k and next zero crossing at frame l that $d'(X_i, X_{k-1}, X_k) \times d'(X_i, X_k, X_{k+1}) < 0$ and $d'(X_i, X_{l-1}, X_l) \times d'(X_i, X_l, X_{l+1}) < 0$, where $i \le k \le i+N-2$, $i \le l \le i+N-2$ and $l > k$. Now if the number of zero crossings between frame k and frame l is larger than a threshold, then we declare the fragment from frame k to frame l is high frequency fragment, else it is low frequency fragment. Once several continuous low frequency fragments are identified, the gradual scene-change segment (consists of fragments) is located.

Here, we use a quantitative measure to differentiate the low frequency fragments from the high-frequency fragments. We employ a local "score" record mechanism $\mathrm{Score}_i(q)$, $q = 0, \ldots N-1$ record mechanism for each frame j of the N-distance measure $\{d(X_i, X_j)\}$, and a global "track" record mechanism $Track(p)$, $p = 1, \ldots L$, for all frames. The length of the $Track$ is L that is the total number of frames in the video sequence ($L \gg N$). For low frequency fragments, they score "1", whereas score "0" is given for high frequency fragments. For instance, frames within the duration of nondecreasing $\underline{M}$ have been marked score "1". We initialize $Score_i(q) = 1$, for $q = 0, \ldots N-1$. For a high frequency fragment $[k, l]$, we reset $Score_i(q) = 0$, for $q = k, k+1, \ldots, l$. The seed-searching step S is the fragment length $S = l - k$.

From our model discussed in Section III-B, frames in segments 1 and 3 have score "1". Since gradual scene change does occur in segment 3 only, we need to ignore the scores in segment 1 due to correlation behavior of the reference frame and its neighboring frames. The correlated distance in segment 1 is "C," which is an important parameter for our proposed model. All local scores for frames are then stored and accumulated in the *track* record with proper shifts. The following procedure shows how to accumulate the $Score_i(q)$ record to the $Track(p)$ record

$$\begin{aligned}\text{for}\, q &= 0, 1, \ldots, N-1 \\ do\, p &= q + i \\ Track(p) &= Track(p) + Score_i(q).\end{aligned}$$

To develop a fast seed-searching process, we select one from every S consecutive frames for *N-distance measure*. The seed-searching algorithm finds a frame index p of which the $Track(p)$ is the maximum accumulated score as: $N/S - C/S$. Where N denotes the distance of *N-distance measure*, S indicates the seed-searching step and C is the assumed correlated distance, respectively. Note that the score is the highest score a frame can possibly achieve. All frames within the gradual transition should also have the same score. We declare a gradual transition with the beginning frame p and duration G, where frames $p, p+1, \ldots, p+G-1$ have the same highest scores observed in the track record.

To prevent the false alarm of the sub-sequence that illustrates the similar *N-distance measure* with the pre-defined model, we assume:

1) a significant histogram change will be observed between the two extremes of declared gradual transition—else, we bypass this seed and look for the next seed;
2) G is supposed to be larger than a constant (in our case, $G = 8$).

IV. Experimental Results

Here, we evaluate the performance of our proposed method. Abrupt and gradual scene changes are separately evaluated because of their different performance measures. The success of

TABLE II
VIDEO SEQUENCES TYPES USED IN THE EXPERIMENTS

Sequence	Sequence Type	Number of frames
French Kiss	Situation Comedy	14259
The Rock	Action Movie	9669
Movie Clips	Mixed	10562
News	News	7965
Music Videos	Mixed	12968
Advertisements	Mixed	2607
Sports Programs	Mixed	13627

TABLE III
SCENE-CHANGE STATISTICS I OF OUR SYSTEM

Sequence	Abrupt Change			Gradual Change		
	Total	Miss	False	Total	Miss	False
French Kiss	95	0	0	2	0	2
The Rock	151	3	10	0	0	0
Movie Clips	71	4	1	0	0	0
News	45	0	0	3	0	1
Music Videos	67	0	0	59	3	13
Advertisements	72	3	2	0	0	0
Sports Programs	31	2	4	57	5	15

TABLE IV
SCENE-CHANGE STATISTICS II OF OUR SYSTEM

Recall			Precision		
Abrupt	Gradual	Cover	Abrupt	Gradual	Cover
97.7%	93%	72%	96.8%	78%	54%

gradual scene-change detection involves the precision of the duration detected. Various types of video streams are tested in the experiments, which are summarized in Table II. About 500 abrupt scene changes and 120 gradual scene changes in total are tested in the experiments and the results are verified manually by human observation. We apply the performance measurement [7], and then we compute the statistics of the experimental results of the testing video streams. Finally, some individual cases are selected and discussed.

A. Performance Parameters

The performance of a scene-change-detection algorithm is usually expressed in terms of recall and precision. The recall parameter defines the percentage of true detection (performed by the detection algorithm) with respect to the overall events (scene changes) in the video streams. Similarly, the precision is the percentage of correct detection with respect to the overall declared event. The recall and precision are defined as

$$\mathrm{Recall} = \frac{N_c}{N_c + N_m} * 100\% \text{ and}$$
$$\mathrm{Precision} = \frac{N_c}{N_c + N_f} * 100\% \tag{8}$$

where

- N_c number of correct detection;
- N_m number of miss;
- N_f number of false detection;
- $N_c + N_m$ number of the existing events;
- $N_c + N_f$ number of overall declaration.

In case of dissolves, these two parameters do not indicate the precision of the detected duration. The detected dissolve does not always coincide with the real dissolve, sometimes it is included in the real dissolve. Sometimes it ends a few frames later. To consider such "partial" error, two new parameters have been defined: *cover recall* and *cover precision*. The cover recall is defined as the percentage of covered length of correct detected dissolve with respect to the length of the real dissolve; whereas, the cover precision can be defined similarly as

$$\mathrm{Recall_{cover}} = \frac{b}{a} * 100\% \text{ and } \mathrm{Precision_{cover}} = \frac{b}{c} * 100\% \tag{9}$$

where a is the length of the real dissolve, c is the length of the declared dissolve and b is the length of the real dissolve covered by the declared dissolve.

B. The Statistics of the Results

Various types of video streams are tested in the experiments, their contents are summarized in Table II. We give the statistics of our experimental results in Tables III and IV.

1) Abrupt Changes: For abrupt scene-change detection, the reported recall is about 97.7%. Most of the misses are due to the outcome of the second phase detection when video transition Type D is found, and cross-validation indicates dynamic-to-dynamic scene transition. Some misses occurred because the ratio between the first peak and the second peak are even smaller than the lower threshold that makes the first phase detector fail. In other words, they suffer great variations in scene contents before or after the scene change. False detections are dominantly decided by the first detector (pixel-based) only, and the results are not effectively cross-validated by the histogram measure (i.e., the ratio between the first peak and the second is larger than the higher threshold). In general, they manifest themselves as the local peaks, but are small from global point of view. If we apply the self and cross validation to both genuine scene change and ambiguous scene change, the precision increases dramatically to almost 98%.

2) Gradual Change: We show two examples of gradual scene change detected by our algorithm. We demonstrate the plot of the track of a long dissolve (about 2 s) and the plot of the track of four dissolves (obtained from an MTV video) in Figs. 5 and 6, respectively. In both cases, $N = 50$, $S = 5$, and $C = 20$. From Table III, we find that the proposed gradual scene-change detection method is prone to false alarm than miss. Since our statistical model is based on segmenting the linear behavior of frame dissimilarity, once the characteristics of frames are linear it will be declared gradual scene change. However, sometimes, false alarms may happen. We give two false alarm examples. In Fig. 7, duration from frame 338 to 368, there is a slow panning sequence. Since the camera panning motion generates a long sequence with $Score_i(p) = 1$ and linear increasing accumulated score is identified, a gradual scene change is declared. In Fig. 8, from frames 1316 to 1365, there are a linear increasing difference, however, the presence of two

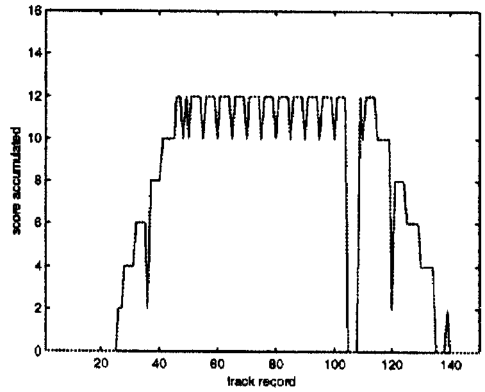


Fig. 5. Track plot of a long dissolve segment—frame 45 to 104.

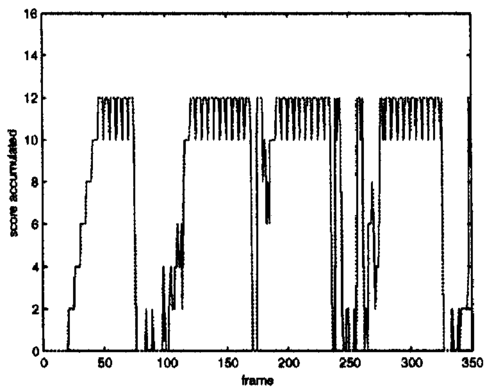
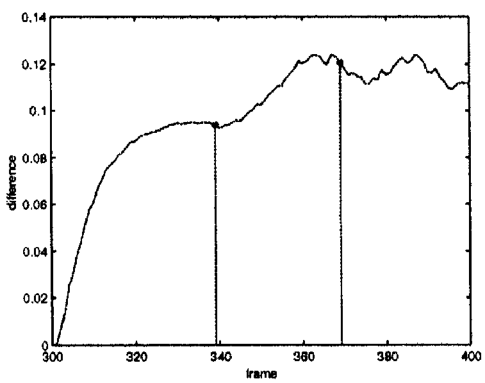
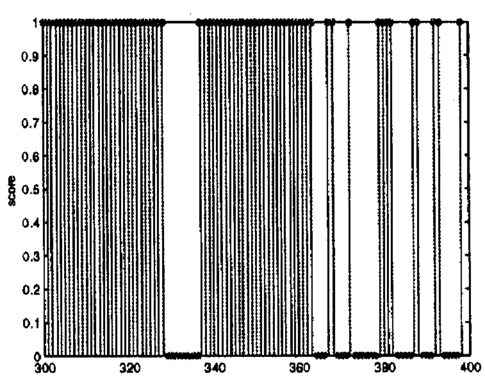


Fig. 6. Track plot of four dissolves detected in an MTV video.

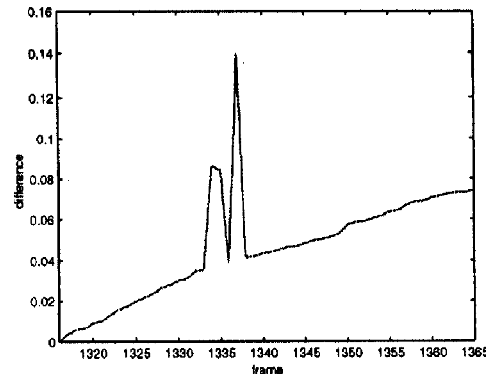


(a)

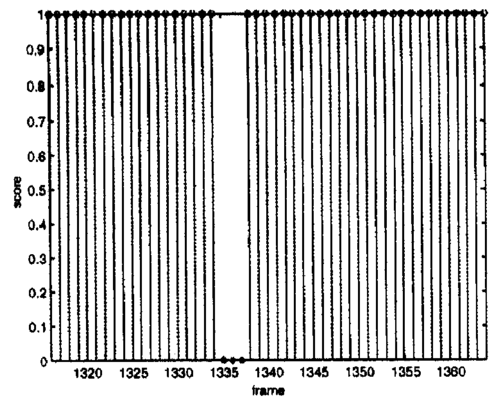


(b)

Fig. 7. False alarm due to the statistical similarity: (a) 100-distance measure of frame 300 (NBA basketball sequence) and (b) local score distribution.



(a)



(b)

Fig. 8. False alarm due to the statistical similarity: (a) 50-distance measure of frame 1316 and (b) local score distribution. Note two peaks in (a) are due to flashlight.

peaks of difference (due to sudden flashlight) are interpreted as an uncorrelated difference segment (segment 2). The original segment with monotonic increasing difference is divided into three segments, and the last segment is declared as the gradual scene-change segment.

Currently, we use a fixed C (i.e., correlation distance) in our system. For frames with correlation distance larger than this predefined constant, we cannot ignore the scores outside this fixed segment but to mistakenly identify the scores indicating the existence of a seed. This is the main reason why lots of false alarms are declared. To overcome this problem, we modify our algorithm as follows. If the Nth difference value (comparing to the first reference image frame) is less than a threshold, which means that the scene content does not change significantly in the duration N, then we ignore all scores reported for this reference frame by redefining $C = N$. Otherwise, $C =$ predefined constant. Therefore, the false alarms can be reduced, however, we can still improve our algorithm by determining C adaptively.

V. Conclusion

We have illustrated the advantages of our method over the conventional threshold problem in avoiding the false alarms by using the validation mechanism. Experimental results show that a very high detection rate is achieved while the false alarm rate is comparatively low. It also proves that the statistical model-based approach is reliable for gradual scene-change detection.

REFERENCES

[1] H. J. Zhang, A. Kankanhalli, and S. W. Smoliar, "Automatic partitioning of full-motion video," *Mutlimedia Syst.*, vol. 1, pp. 10–28, 1993.

[2] A. Nagasaka and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," *Visual Database Syst. II*, pp. 113–127, 1992.

[3] I. K. Sethi and N. Patel, "A statistical approach to scene change detection," *SPIE*, vol. 2420, pp. 329–338, 1995.

[4] B. L. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 533–544, 1995.

[5] Y. Nakajima, K. Ujihara, and A. Yoneyama, "Universal scene change detection on MPEG coded data domain," *Visual Commun. and Image Processing*, vol. 3024, pp. 992–1003, 1997.

[6] R. Zabhi, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," in *Proc. ACM Multimedia '95*, 1995, pp. 189–200.

[7] G. Lupatini, C. Saraceno, and R. Leonardi, "Scene break detection: A comparison," in *Proc. 8th Int. Workshop on Continuous-Media Databases and Applications*, 1998, pp. 34–41.

[8] C. Taskiran and E. J. Delp, "Video scene change detection using the generalized sequence trace," in *IEEE Proc. ICASSP*, 1998, pp. 2961–2964.