Sensors & Transducers, Vol. 167, Issue 3, March 2014, pp. 104-109

www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

Robust Shot Boundary Detection from Video Using Dynamic Texture

[1, 3] Peng Taile, [2] Zhang Wenjun

[1] School of Communication & Information Engineering, Shanghai University, Shanghai, 200072, China

[2] School of Film and TV Arts & Technology, Shanghai University, Shanghai 200072, China

[3] School of Computer Science and Technology, Huaibei Normal University, Huaibei, Anhui, 235000, China.

[1] E-mail: tailep@163.com

Received: 4 March 2014 /Accepted: 28 March 2014 /Published: 31 March 2014

Abstract: Video boundary detection belongs to a basis subject in computer vision. It is more important to video analysis and video understanding. The existing video boundary detection methods always are effective to certain types of video data. These methods have relatively low generalization ability. We present a novel shot boundary detection algorithm based on video dynamic texture. Firstly, the two adjacent frames are read from a given video. We normalize the two frames to get the same size frame. Secondly, we divide these frames into some sub-domain on the same standard. The following thing is to calculate the average gradient direction of sub-domain and form dynamic texture. Finally, the dynamic texture of adjacent frames is compared. We have done some experiments in different types of video data. These experimental results show that our method has high generalization ability. To different type of videos, our algorithm can achieve higher average precision and average recall relative to some algorithms. *Copyright © 2014. IFSA Publishing, S. L.*

Keywords: Shot boundary detection, Average gradient direction, Dynamic texture, Texture measure.

1. Introduction

Video is an important form of multimedia data. With Internet and electronics technology developing, video data are rapidly increasing in video websites, libraries and mobile devices, etc. In the vast amounts of data, we always hope to get available video data. Because of the complexity of the video structure, it is more difficult to retrieve video from that large database. Therefore there is a need for efficient and accurate video retrieving algorithms. Among the algorithms, an important approach is to break the video into shots and get key frame for each shot. So video retrieval can be converted to retrieval based on image. As we know that a shot of video is a continue sequence of frames to depict some scene or event, video structure is shown in Fig. 1.

2. Review of Existing Techniques

Nowadays, shot boundary detection, a fundamental problem of computer vision, has been intensively studied in the past several years. There have been some methods in the published literature to solve the shot boundary detection [1-8]. These shot boundary

Article number P_RP_0084

detection techniques can be classified into four approaches, namely, 1) Pixel comparison methods; 2) Histogram comparison method; 3) Edge comparison method, and 4) Methods based on machine learning.

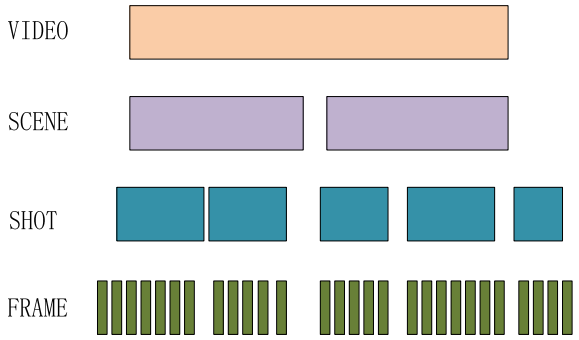

Fig. 1. Video structure.

1) Pixel comparison methods are based on the assumption that there is difference among the pixels. Thus, we may define a threshold and draw difference of the adjacent frame through calculating their value (grey level, color value, etc). Further, according to the difference between the adjacent frames, it can be judged whether shot boundary changes.

Pixel comparison methods have low complexity of calculation, but some factors (pixel brightness change caused by movement of digital cameras and video object, illumination variation, etc) have large influence to results, these factors can be easy to cause a fault shot boundary.

2) Histogram comparison methods use gray histogram to compare the adjacent frames. The gray histogram of frame usually classify several ratings to brightness of each pixel, gray of each pixel or color of each pixel in a frame, and sums the total number of per level pixels, so as to compare the frame boundary. On the basis of the statistics, the methods can get better result of boundary detection to the video where there are slow motion objects and slow camera moving. On the one hand, these methods have lower computation complexity, on the other hand, when the light intensity change or shots have quick movement, the histogram can produce distortion, thus we receive error detection.

3) Edge comparison methods rely on the assumption when the adjacent shots happen to change, the new shot should be far from the previous one. The methods are easy to detect the shot with simple structure.

4) Machine learning methods usually use machine learning to train of frame features, and achieve the shot boundary. These machine learning methods include the SVM method, K-means method, AdaBoot algorithm, etc.

Texture feature is an important the image feature and annotation methods for object within image and the whole image. In this paper we extend image texture concept to video, then form a dynamic texture. Dynamic texture [9] can be used to describe the dynamic and timing sequence image object, such as video shot. In general, for the adjacent frames of the same shot, if the objects within a frame move in the local area of frame, frame texture feature locally changes too. Based on the factors, this paper proposes a new “video dynamic texture” concept. In this paper, the frame of video will be segmented into the same size sub-domain that is image blocks. The following thing is to calculate the average gradient direction of each area, and form "video dynamic texture" based on these average gradient direction. Then, according to the change of "video dynamic texture" between the adjoining frames, we can determine if the boundary of the adjacent frames have a large change.

3. Background and Detection

As we know that texture feature is a basic feature of the image. From the view of locality, image texture is in out of order, which is in order in global view, and which is easy to be cognitive and difficult to be defined. Video texture is abstract, whose definitions are formed due to the different understanding on the texture, and depends on the specific application. In this paper we treat a group of average gradient direction as “video dynamic texture”. Video dynamic texture of color video is different from gray video. Calculation for video dynamic texture of color video is more difficult than gray video. Firstly, let's discuss video dynamic texture of gray video. For an assumed grayscale image block (sub-domain), as we know that the average gradient direction of the sub-domain can always reflect the regional gray intensity. If we segment
an image into several sub-domains, and calculate the average gradient direction of each sub-domain, we will quantitatively obtain gray distribution.

In this work, we bring up a video clip, read the adjacent frame f_i and f_{i+1}, make normalization to frame size, and calculate the average gradient direction of each sub-domain $Gradient\_\alpha_{i,j}$. Thus $Matrix_i$, $Matrix_{i+1}$ (average gradient direction matrix) are formed by $Gradient\_\alpha_{i,j}$ .The following thing is to compare $Matrix_i$ with $Matrix_{i+1}$. Flowchart of algorithm is shown in Fig. 2.

3.1. Initialization Operation

In video stream, the video clip can be from different cameras, different video post-production, so there may be a not consistent size among the shots, For facilitating the comparison we request all the frames must be in the same size. In this paper all the frames are normalized, into a given size.

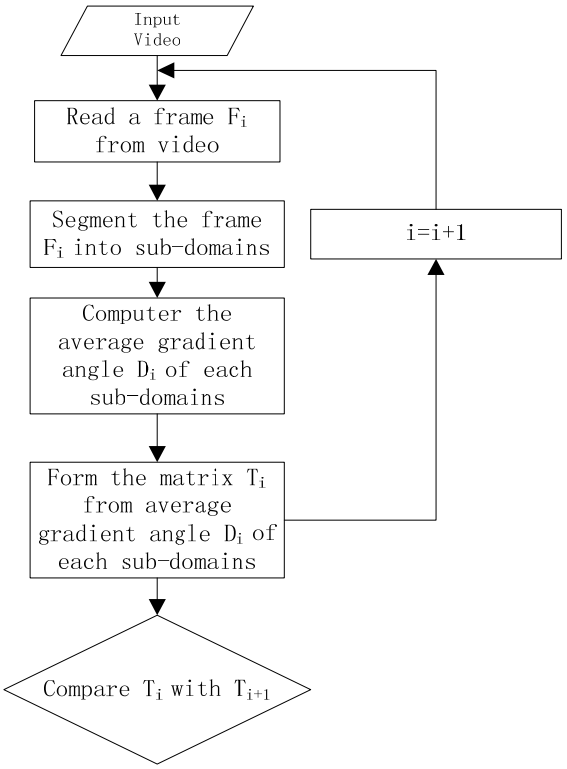

Fig. 2. Formation of video dynamic texture.

3.2. Correlative Concepts Definition

For a normalized frame f^*, we can calculate gradient of sub-domain by several methods. We know that difficulty that color frame obtain the gradient of sub-domain is greater than the gray image. Due to the color images having more application, so this paper uses color video data as experiment data. For gray image, the gradient and the gradient direction can be calculated by the Formula 1- Formula 4. We assume that gradient based on gray （bright）function *H(x, y)* in the point (x, y) is a vector with value and direction. G_x and G_y represent gradient along the x direction and y direction. *H(x, y)* represent the grey value of pixel (x, y). Then the gradient vector can be represented as follows:

$$G_x = H(x+1, y) - H(x-1, y), \quad (1)$$

$$G_y = H(x, y+1) - H(x, y-1), \quad (2)$$

The gradient value G (x, y) is:

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2}, \quad (3)$$

The gradient direction is:

$$\alpha(x, y) = \arctan(G_y(x, y) / G_x(x, y)), \quad (4)$$

In this paper, sub-domain gradient is calculated in the three fields (R, G, B) in RGB color space, then eventual gradient can obtained by averaging gradient from R, G, B field. Average gradient of frame sub-domain (FSAGT), average gradient direction of frame sub-domain (FSAGTD), dynamic texture are defined as follows:

Definition 1. Frame sub-domain average gradient (FSAGT).

In RGB color space, for a sub-domain $D_{i,j}$ of a given fame, we assume $D_{i,j} \subseteq R^{L\times L}$ (L belongs to odd number), FSAGT is average gradient value of sub-domain pixels except for boundary pixels.

$$FSAGT(I,J)_x = \\ = \sum_{K=1}^{L-2} \sum_{z=R,G,B} (H(x+k+L*i+1, y) - , \quad (5) \\ - H(x+K+L*i-1, y))_z / 3$$

$$\sum_{k=1}^{L-2} \sum_{Z=R,G,B} (H(x, y+K+L*j+1) - \\ - H(x, y+k+L*j-1))_z / 3, \quad (6)$$

where i, j=0, 1, 2.

Definition 2. Frame average gradient direction (FSAGT D).

In RGB color space, for a sub-domain $D_{i,j}$ of a given fame, we assume $D_{i,j} \subseteq R^{L\times L}$ (L belongs to odd number), FSAGT D ($Gradient\_\alpha_{I,J}$) can be describe as:

$$Gradient\_\alpha_{I,J} = \\ = \arctan(FSAGT(I,J)_y / FSAGT(I,J)_x), \quad (7)$$

Definition 3. Video dynamic texture.

For a given frame f_i, we normalize it as the frame f_i^* at first. The frame f_i^* is segmented into M * N areas. Then we calculate the gradient direction of each sub-domain. At last, a matrix is made up with M * N gradient directions $Gradient\_\alpha$.

$$W = \begin{bmatrix} Gradient\_\alpha_{0,0}, Gradient\_\alpha_{0,1}, \ldots Gradient\_\alpha_{0,N} \\ \vdots \\ Gradient\_\alpha_{M,0}, Gradient\_\alpha_{M,1}, \ldots Gradient\_\alpha_{M,N} \end{bmatrix}$$

The matrix W is named as dynamic texture of the frame.

4. Dynamic Texture Measure

Image texture features can be measured by Coarseness, Contrast, Line Likeness, regularity, Roughness and directionality [10]. Similarity among image texture may use statistical moment of histogram, gray level co-occurrence matrix, spectral measure and the fractal dimension video dynamic texture in this paper is different from the traditional image texture. Video dynamic texture is defined by average gradient direction of frame sub-domain. We define texture which can reflect the dynamic global characteristics of video frames, can also reflect the local characteristics of image. For two adjacent frames, the change value of gray gradient direction of the same sub-domain reflects similarity of the two adjacent frames at the sub-domain. The change value of dynamic texture of two adjacent frames reflects similarity of the two adjacent frames. For the adjacent frame class dynamic texture can be compared by Formula 8 and Formula 9.

$$\frac{\left|Gradient_{j,k}^{i+1} - Gradient\_\alpha_{j,k}^{i}\right|}{\sqrt{(Gradient_{j,k}^{i+1})^2 + (Gradient\_\alpha_{j,k}^{i})^2}} < \delta, \quad (8)$$

$$\Delta W = \left|W_{i+1} - W_i\right|, \quad (9)$$

In the Formula 8, δ is a given experience decimal. Due to many factors, frames always have noise. If Formula 8 is established, we will think that the gradient direction of sub-domain (j, k) does not change. We assume that N_s is number of matrix elements which meet Formula 8 in the adjacent frames. For a given shot, if matrix ΔW meets Formula 9, we think matrix ΔW is in sparse, frame f_i and f_{i+1} belong to the same shot. Otherwise the ΔW is dense matrix, frame f_i and f_{i+1} does not belong to the same shot.

$$N_s / N_{all} < \varepsilon, \quad (10)$$

where N_{all} is the total number of matrix elements; ε is a given decimals.

5. Shot Boundary Detection Algorithm Based on Dynamic Texture

The basic steps of our method are represented as:

Step 1. For a given video clips, the algorithm need to read for two adjacent frames f_i and f_{i+1}. In RGB space, the two frames will be normalized, divided into several sub-domain. Then we calculate its dynamic texture w_i and w_{i+1}.

Step 2. According to Formula 9 and Formula 10, the algorithm calculates Δw and estimates sparsity of Δw.

Step 3. If Δw is sparse, we think that f_{i+1} is first frame of new shot. Otherwise, $f_i = f_{i+1}$, the algorithm turns to step 1.

Step 4. Detection judgment.

For some video, such as video which has single fluid frame object, although there will be an obvious difference for the dynamic texture of two adjacent frames, but we still can't confirm that the two frames belong to two shots. On the basis of algorithm of literature [11], we will compare histogram of the two frames. If they still have obvious difference for histogram, the two frames belong to two different shots, otherwise, they belong to the same shot.

6. Results and Analysis of Experiment

In order to prove the performance of the algorithm in this paper, we choose 40 pieces of video clips including several kinds of video data (news, sports video, advertising, movie, etc.) from Internet to compare several methods. Typical video data is shown in Fig. 3.

Fig. 3. Typical video data.

In this paper, the experimental platform is Windows 7 operating system. The computer configuration for this experiment is as follows: Inter core2 Quad CPU Q8300, 2 GB memory and Matlab2010a.

In this paper we compared our methods with two very commonly used methods. One of these algorithms is based on color histogram method, the others is pixel comparison method.

We implement and run all algorithms under the same development environment to have a fair timing comparison. The comparisons amomg the three algorithms are based on shot average precision ($AVS_{precision}$), shot average recall(AVS_{recall}), to test effect of the three methods.

$AVS_{precision}$ and AVS_{recall} can be defined as follows:

$$\mathrm{AVS}_{precision} = \sum_{i=1}^{n} \frac{Nts_i}{Nas_i} \times 100\%, \quad (11)$$

$$\mathrm{AVS}_{recall} = \sum_{i=1}^{n} \frac{Nts_i}{Nt_i + Nf_i} \times 100\%, \quad (12)$$

where Nts_i is the right number of shot detected of video i; Na_i is the number of shots which have been detected; Nf_i is the number of lost shots.

In the experiments, we adopt that sub-domain size of frame is 13×13. Why we adopt the size 13×13? If the size is smaller, it will increase the running time of algorithm. If the size is larger, it is impossible to accurately describe the change of gradient of frame.

6.1. Experimental Data Analysis

The Formula 11 and Formula 12 we defined can more accurately embody shot boundary detection results from some video. The experimental results are shown in Table 1 – Table 4. For sports video, there are 723 shots. Our method detected 632 boundaries. Among them shot number being detected is 583. We can see from table 1, the average recall ratio and precision ratio reached 85.37 % and 85.6 %.

As can be seen from Table 1 to Table 4, our method can obtain high boundary detection efficiency for cut shots or gradient shots, and also has high generalization ability.

Table 1. Experiment result for sports video.

Algorithm	Number of video	AVS_{recall} (%)	$AVS_{precision}$ (%)
Pixel comparison method	10	83.2	85.09
Color histogram	10	77.36	84.97
Our methods	10	85.37	85.6

Table 2. Experiment result for Movie.

Algorithm	Number of video	AVS_{recall} (%)	$AVS_{precision}$ (%)
Pixel comparison method	10	86.09	78.36
Color histogram	10	80.6	82.87
Our methods	10	93.18	88.53

Table 3. Experiment result for news video.

Algorithm	Number of video	AVS_{recall} (%)	$AVS_{precision}$ (%)
Pixel comparison method	10	89.59	85.46
Color histogram	10	86.4	86.08
Our methods	10	93	91.72

Table 4. Experiment result for advertisement.

Algorithm	Number of video	AVS_{recall} (%)	$AVS_{precision}$ (%)
Pixel comparison method	10	93	83.4
Color histogram	10	91.74	83.4
Our methods	10	92.8	84.88

7. Conclusion

For several kinds of video data, the paper presented a novel video boundary detection method based on dynamic texture. We define a novel video dynamic texture based on gradient direction of sub-domain of frame, which can reflect frame local changes and global changes.

The method of shot boundary detection isn't only effective to abrupt shots, but also to gradual shots. We have experimented with sports video, movie, news video, and advertisement video, with good results.

The approach also has some short need to improve, such as how to select δ and ε. In future work, we will be combined with other information to improve the method in order to obtain better generalization ability and better detection results.

Acknowledgements

This work was financially supported by the University Science Research Key Project of Anhui China (Grant No. KJ2010A304).

References

[1]. E. Sahouria, A. Zakhor, Content Analysis of Video Using Principal Components, *IEEE Transactions on Circuits and Systems for Video Technology,* Vol. 9, No. 8, December 1999, pp. 1290-1298.

[2]. E. Stringa, C. S. Regazzoni, Real-time Video-shot Detection for Scene Surveillance applications, *IEEE Trans. on Image Processing,* Vol. 9, No. 1, January 2000, pp. 69-79.

[3]. I. Koprinska, S. Carrato, Video segmentation of MPEG compressed data, in *Proceedings of the IEEE Int. Conf. on Electronics, Circuits and Systems,* Vol. 2, 1998, pp. 243-246.

[4]. Zhang, H. J., Kankanhalli, A., Smoliar, S., Automatic partitioning of full-motion video, *Multimedia Systems,* 1, 1, 1993, pp. 10-28.

[5]. Ford Ralph M, Robson Craig, Temple Daniel, et al., Metrics for shot boundary detection in digital video sequences, *Multimedia Systems,* 8, 1, 2000, pp. 37-461.

[6]. Zhu Xi, Lin Xing-gang, Survey on video temporal segmentation, *Chinese Journal of Computer,* 27, 8, 2004, pp. 1027-10351.

[7]. Nagasaka A, Tanaka Y., Automatic video indexing and full-video search for object appearances, in *Proceedings of the IFIP 2nd Workshop Conference on*

Visual Database System II Budapest, Hungary, 1992, pp. 113-127.
[8]. Zhi-Cheng Zhao, An-Ni Cai, Shot Boundary Detection Algorithm in Compressed Domain Based on Adaboost and Fuzzy Theory, *Advances in Natural computation,* 2006, pp. 617-626.
[9]. R. C. Nelson, R. P. Polana, Qualitative Recognition of Motion Using Temporal Texture, *CVGIP: Image Understanding,* 56, 1, 1992, pp. 78-89.
[10]. Tamura H., Mori S., Yamawaki T., Textural features corresponding to visual perception, *IEEE Transactions on Systems, Man and Cybernetics,* 8, 6, 1978, pp. 460 - 473.
[11]. Dugad R., Ratakonda K., Ahuja N., Robust Video Shot Change Detection, in *Proceedings of the IEEE Workshop on Multimedia Signal Processing,* 1998, pp. 376-381.