International Journal of Innovative Computing, Information and Control
Volume **10**, Number **2**, April **2014**

ICIC International ©2014 ISSN 1349-4198
pp. 481–494

VIDEO FRAMES SIMILARITY FUNCTION BASED GAUSSIAN VIDEO SEGMENTATION AND SUMMARIZATION

Yanjiao Zhang[1,2], Zhicheng Wei[1,*] and Yanling Wang[1,2]

[1]Department of Information Technology
[2]College of Physics Science and Information Engineering
Hebei Normal University
No. 20, Road East of 2nd Ring South, Yuhua District, Shijiazhuang 050024, P. R. China
*Corresponding author: weizhicheng@hebtu.edu.cn

Received January 2013; revised May 2013

Abstract. *By exploiting gaussian theory and split-merge approach, this paper proposed a novel method for video segmentation and summarization. Firstly, a video sequence is segmented into unequal sized shots using a gaussian method. It depends on a video frames similarity function (VFSF) which describes the similarity of each set of three adjacent frames in a video sequence. Secondly, a tridiagonal matrix is introduced to decrease the computing capacity of VFSF. It is demonstrated that the VFSF based gaussian method can detect shot boundaries with high accuracy. At last, the split-merge approach is employed to extract key frames for generating a video summarization, which also performs on the basis of VFSF values. According to the content changes of shots, split and merge processes are implemented in succession to divide the detected shots with large content changes into clusters for potential key frames extraction. The final key frames are obtained through reducing redundant potential key frames to better represent a video. The experimental results show that the key frames that are extracted with the proposed method are representative of the whole video and are effective for generating a video summarization.*

Keywords: Video frames similarity function (VFSF), Tridiagonal matrix, Gaussian method, Video segmentation, Split-merge approach, Key frame extraction, Video summarization

1. **Introduction.** A common fundamental step of video indexing, called video segmentation, is to divide the video stream into a set of segments (*shots*). A shot is described as an unbroken sequence of frames captured from one camera [1]. In general, it consists of frames that have consistent visual characteristics, such as color, motion, object and texture.

In order to segment a video sequence into shots, we need to define a dissimilarity or similarity methods among frames. In previously reported work, dissimilarity or similarity methods have mainly focused on pixel-based approaches [2] and histogram-based approaches [3,4]. Being sensitive to objects and camera motions, the pixel-based approaches are not very effective in segmenting a video for its density of motions. Histogram-based approaches exploiting global information, such as intensity histograms or color histograms algorithms [5-8], provide a better tradeoff between accuracy and speed. Although the performance of histogram-based approaches is sufficient for segmenting scenes with abrupt changes, it fails to discriminate the differences between consecutive frames with low similarity values as being due to motion when dissolve exists. Noticing the weaknesses of the above-mentioned methods, many researchers have suggested the use of other methods based on more complex features. Huang and Liao [9] proposed to exploit motion vectors as basic feature to depict visual content complexity of a given video, which improves the

detection performance but it is not an effective solution in the cases where strong and abrupt lighting changes involved in the video sequence. Mutual information (MI) in gray space of images is presented by Butz and Thiran in [10] that uses affine image registration for compensation of camera panning and zooming to detect shot boundary, leading to expensive computation. In addition, one problem in most existing segmentation procedures is that they are lack of adaptivity with the threshold confirmed through experiment. A shot is detected when a certain dissimilarity or similarity value between continuous frames exceeds a threshold, which would influence the segmentation performance to a great extent.

Currently, video browsing and retrieval are mainly achieved by exploiting key frames which provide an applicable video summarization. Key frames that represent the salient message of the original video can be extracted after shots are segmented. Much research work has been done on key frame extraction. Tonomura *et al.* [11] proposed to choose the first frame of each shot as the key frame, which is the simplest method. In [12], Ueda *et al.* described each shot using its first and last frames. Both of [11] and [12] neglect the visual content complexity of a video shot. More sophisticated approaches exploiting color and motion features [13-15] have been proposed to extract one or more key frames from each shot. In [16], Zhang *et al.* used sequential comparison of color to measure video content complexity and extract key frames. They also introduced dominant or global motion to improve the effect of the color features. However, these existing methods either are computationally expensive or cannot effectively capture the major visual content.

This paper proposed an improved method that is more accurate and effective than all the existing video segmentation and summarization methods. In terms of segmentation, the application of gaussian theory based on VFSF describing video sequence in a concise way that is to simplify the similarity of each set of three sequential frames to point values, makes each video sequence segmented into unequal sized shots. By avoiding distortion of the visual content in a video, this approach enables video segmentation performance to reach a more reasonable result than the results from employing the equal-sized segmentation method used in, for instance, [17]. In addition, the utilization of tridiagonal matrix decreases computing capacity to a great degree, for the less important elements in similarity matrix are neglected when segmenting.

Then, split-merge method is developed to deal with key frames extraction related to the different degrees of content changes, which can automatically estimate the number of key frames should be selected to generate a video summarization. Divided into some clusters through split and merge processes, the shot with large content change provide some potential key frames. Ultimately, those redundant key frames among potential key frames are discarded. This enables key frames to extract more intelligently. Therefore, the final key frames will better capture the visual content of the whole video than some published methods [11,12] that select key frames artificially.

The remainder of the paper is organized as follows. In Section 2, a brief description of the tridiagonal matrix as well as the gaussian probability plot and "3σ *Rule*" are presented. The proposed VFSF-based gaussian video segmentation method is described in Section 3. Section 4 addresses the method used for key frames extraction. Experimental results over a large of data set and comparison with existing methods are given in Section 5 and concluding remarks are in Section 6.

2. **Background and Definitions.**

In this section, definitions and examples of techniques and procedures used in this paper are discussed.

Tridiagonal Matrix. In linear algebra, the nonzero elements of a tridiagonal matrix are on the main diagonal, the first diagonal below the main diagonal, and the first diagonal

above the main diagonal. For instance, the following matrix A is tridiagonal:

$$A = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{pmatrix} \tag{1}$$

a_{ij} $(i = 1, 2, 3, 4; j = 1, 2, 3, 4)$ is an nonzero element. In particular, if a_{ij} is nonzero and real, then matrix A is a real tridiagonal matrix. The eigenvalues are real too.

When tridiagonal matrices are applied to many linear algebra algorithms, less computational effort is required than for a general matrix. If the $n \times n$ matrix M is a real tridiagonal matrix, its determinant can be computed as the following:

$$\det [M]_{\{1,\cdots,n\}} = a_{n,n} \det [M]_{\{1,\cdots,n-1\}} - a_{n,n-1}\, a_{n-1,n} \det [M]_{\{1,\cdots,n-2\}} \tag{2}$$

where $\det [M]_{\{1,\cdots,i\}}$ refers to the ith principal minor, the submatrix obtained from the first i rows and columns of matrix M. Consequently, the cost of computing the determinant of a tridiagonal matrix is linear n, while for a general matrix the cost is cubic.

Gaussian Probability Plot. In order to estimate whether a data set is approximately gaussian distributed or not, the graphical method gaussian probability plot is very effective. The gaussian probability plot consists of two axes: horizontal axis which refers to gaussian order statistic medians and vertical axis which indicates ordered response values.

Given a set of ordered data points with index $i = 1, \cdots, n$, the values of two axes are calculated like this:

$$P(Z < z_i) = \begin{cases} 1 - 0.5^{1/n}, & i = 1 \\ 0.5^{1/n}, & i = n \\ \dfrac{i - 0.3175}{n + 0.365}, & \text{otherwise} \end{cases} \tag{3}$$

Then, plotted against a theoretical gaussian distribution, the sample values are regarded as a function of the corresponding gaussian order statistic medians. Another way to think about this is that the data are plotted against what we would expect to see if it was strictly consistent with the gaussian distribution. If the data are distributed according to a gaussian distribution, they should lie close to a straight line. In other words, the further the points apart from this line, the greater the indication of departure from gaussianity.

"3σ Rule". In a gaussian probability distribution, the probability of a variable X, whose gaussian distribution density curve with mean value μ and standard deviation σ is shown in Figure 1, lying outside the interval between $\mu - 3\sigma$ and $\mu + 3\sigma$ is less than 3‰. The practical requirement that a variable falls within the interval between $\mu - 3\sigma$ and $\mu + 3\sigma$ is known as "3σ Rule".

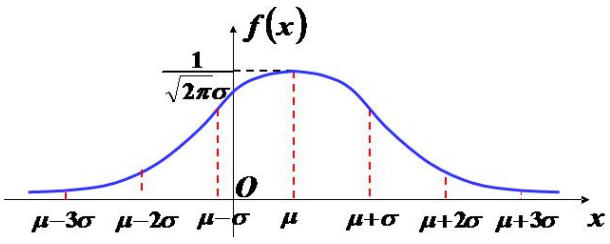


FIGURE 1. Gaussian distribution density curve for variable X

3. **VFSF-Based Gaussian Video Segmentation.** Before segmenting a video sequence, we first use a video frames similarity function (VFSF) to model the input video data, describing each video stream in a concise way and simplifying the similarity to point values in the time domain. Then, based on the VFSF results, a gaussian method is developed to divide video sequence into segments. Compared with the equal-sized segmentation method, such as used in [17], VFSF based gaussian video segmentation method avoids distortion of the visual content in a video with its obtained unequal sized shots.

3.1. **Video frames similarity function (VFSF).** A VFSF model is defined in this section to describe the similarity of each three given successive frames in time domain. Please refer our previous work [18] for detailed description.

Let us consider a video sequence with W frames in HSV color space, building a 128 bins histogram to represent each video frame. Firstly, the similarity between two consecutive frames is computed at the pixel level [19] using a class set D_{ij} and its size is matrix SN.

$$D_{ij} = \{p | C(t-1, p) = i, C(t, p) = j\} \tag{4}$$

where $C(t, p)$ is the value of frame t in the pixel p $(1 \leq p \leq F)$, and $(1 \leq i, j \leq 128)$. F is the total pixels of each frame.

Then we get a 128×128 matrix SN that describes the similarity of two successive frames, t and $t-1$ at pixel level. Simplifying SN into a tridiagonal matrix that keeps the most important information, a matrix PN is obtained as below

$$PN(i,j) = \begin{cases} d_{ij} \in SN(i,j), & i = j \text{ or } i = j \pm 1 \\ 0, & \text{other} \end{cases} \tag{5}$$

Inspired by the idea of extremum, we calculate the sum of the minimum and maximum between $PN(t+1)$ and $PN(t)$ respectively, receiving $Min(t)$ and $Max(t)$.

Finally, the VFSF model is defined as

$$V(t) = \begin{cases} Min(t)/Max(t), & 1 < t \leq W - 1 \\ 0, & t = 1 \end{cases} \tag{6}$$

In Figure 2, we give a diagrammatic representation of the process of VFSF model, in which the color value of three consecutive frames, the similarity matrix SN of two successive frames and its corresponding tridiagonal matrix PN and the final VFSF values are displayed orderly. Simplified matrix PN persists the foremost information about similarity between consecutive frames, facilitating the computing of VFSF model. Besides, the VFSF value of the three given frames with similar visual content in Figure 2 is high close to 1. It just conforms to the VFSF principle that the more similar consecutive frames are, the higher the VFSF values will be.

3.2. **Gaussian method for video segmentation.** In our approach, a small value of the VFSF indicates a large video content difference among three consecutive frames, which is what we expect to be regarded as the segment frame. Unlike some previously proposed threshold-dependent methods used to identify segment frames, we apply the "3σ Rule" of the gaussian probability distribution to determine how small VFSF values should be in order to be considered as segment frames.

In our previous work [18], for a set of VFSF values with W frames that have been first estimated to have a gaussian distribution, the mean μ and standard deviation σ are calculated. We choose the frames whose VFSF values are less than $\mu - 3\sigma$, composing a set *seg*. If the distance between two adjacent frames in *seg* is smaller than a threshold β, only the later frame is retained.

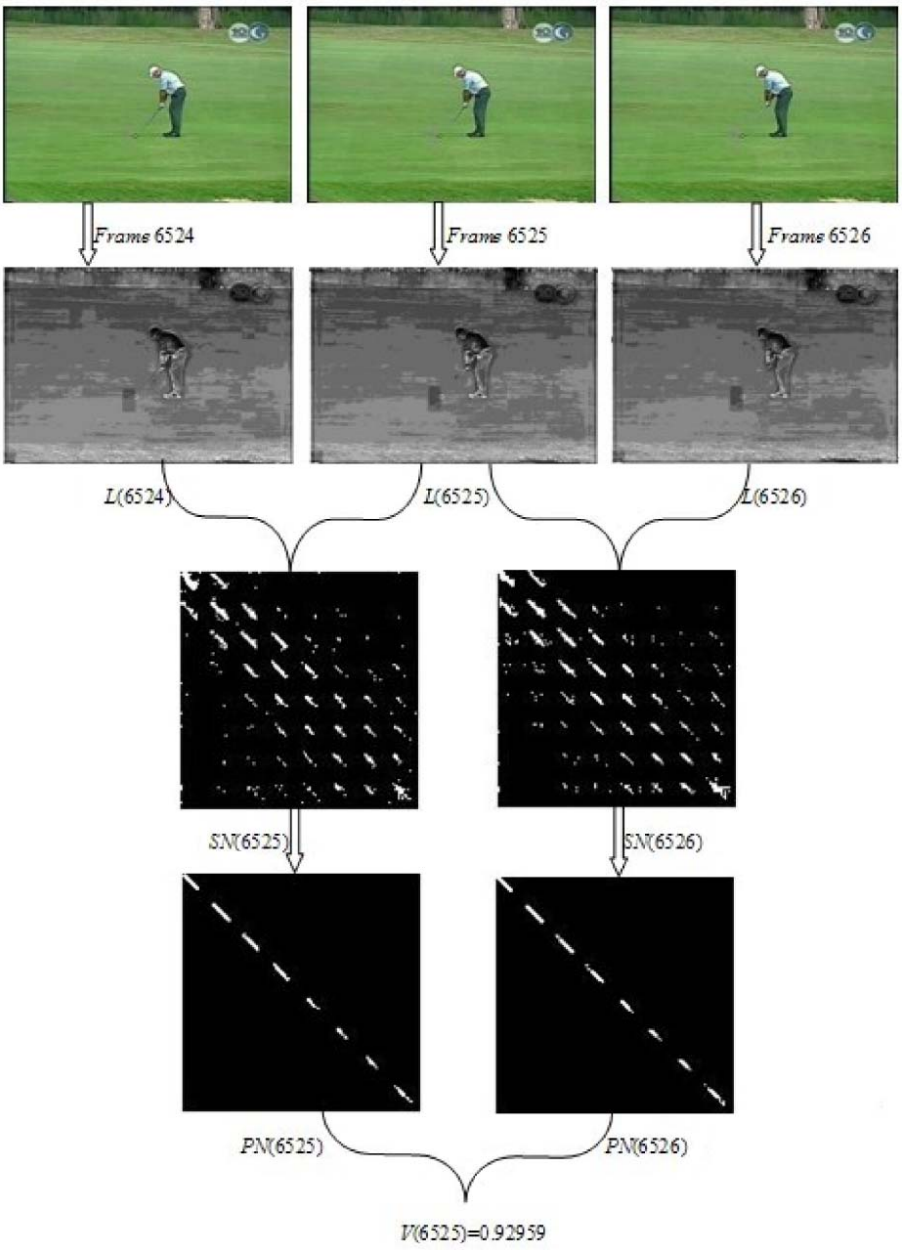
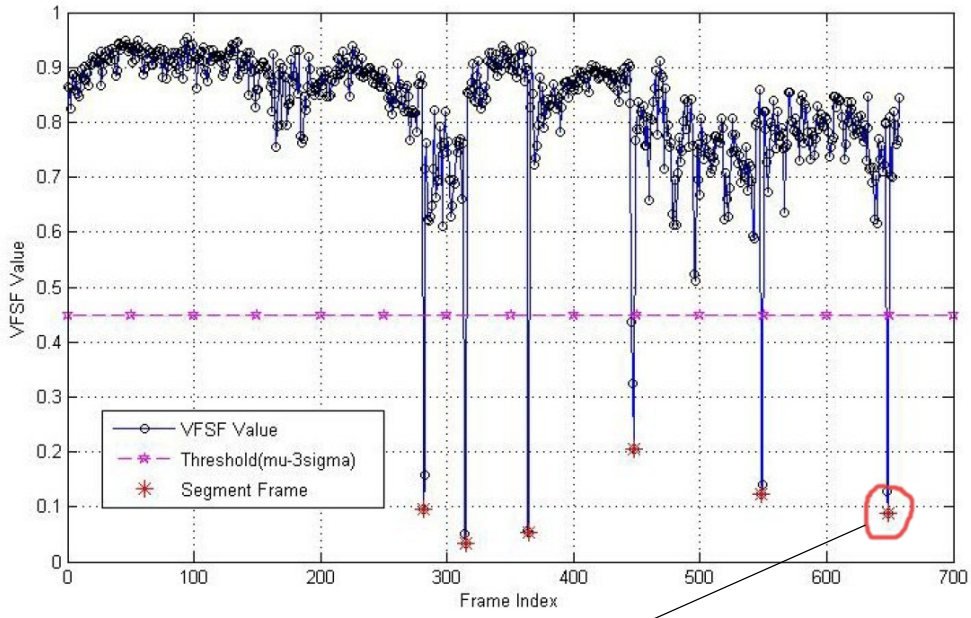


FIGURE 2. VFSF structural model

Keeping with the idea of using the smallest VFSF values to segment a given video stream, we select the frame whose VFSF value is smaller for each pair of segment frames to create a new set seg'.

$$seg' = \{t \in seg | \min(V(seg(i)), V(seg(i+1)))\} \tag{7}$$

Consequently, the video sequence is divided into U unequal sized segments, which are referred as "shots". Figure 3(a) shows an example taken from a football video sequence using the above gaussian method to identify segment frames, from which we can see that all frames with large visual content difference (corresponding to low VFSF values) are detected and regarded as segment frames as well. Frame 649, intuitively seen located in shot boundary, is precisely detected as a segment frame by our method and it is amplified, framed by a rectangle, to be displayed as well as its neighbor frame 648 and 650 in Figure 3(b).



(a) All segment frames of a football video

(b) One segment frame (frame 649) from all segment frames

FIGURE 3. Gaussian approach to segment for a football video

4. **Key Frames Extraction.** In consideration of the drawback of some existing key frames extraction approaches, this section aims to solve the artificial selection of key frames problem by exploiting a split-merge method based on VFSF results for every shot to determine how many key frames should be selected so to better represent the video content for indexing. The proposed method takes the different degrees of content changes in a given video into consideration, conquering the limitation that the amount of key frames is artificially ascertained based on segmented shots which generally select one or two key frames from each shot.

Now, we have segmented the initial video sequence, which we refer as Y, into U shots, as Equation (8)

$$Y = \{Q_1, Q_2, \cdots, Q_U\} \tag{8}$$

with Q_i $(1 \leq i \leq U)$ being the ith shot of video sequence Y.

We can then extract key frames from each shot to generate a video summarization for video indexing. Using the VFSF values to describe each shot, we formulate Q_i as

$$Q_i = \begin{cases} \{V(t)|t \in [1, seg'(1)]\}, & i = 1 \\ \{V(t)|t \in [seg'(i-1)+1, seg'(i)]\}, & 2 \leq i \leq U-1 \\ \{V(t)|t \in [seg'(U-1)+1, W-1]\}, & i = U \end{cases} \tag{9}$$

In order to estimate whether a visual content in a given shot changes significantly or not, we first calculate the standard deviation σ_Q which is compared with a predefined threshold θ. If $\sigma_Q < \theta$, all the frames in this shot share significant similarity and any

frame can represent the shot. In this instance, we select the middle frame as the key frame.

However, if $\sigma_Q > \theta$, which indicates the content in the shot changes largely, the situation becomes complicated. For such shots, we employ a split-merge algorithm to select potential key frames. Steps of the split-merge algorithm are summarized as follows:

i. Split process
   1) Initialize shot Q_i to one cluster c_1;
   2) Calculate this cluster's standard deviation σ;
   3) Compare with the threshold θ;
   4) If $\sigma > \theta$, this cluster is split equally into two clusters;
   5) For each of the two clusters, repeat beginning at step 2);
   6) If $\sigma < \theta$, break.

ii. Merge process
   1) Calculate the standard deviation σ for two contiguous clusters;
   2) If $\sigma < \theta$, these two clusters are merged into one, return to step 1);
   3) If $\sigma > \theta$, break.

Finally, through the split-merge process, shot Q_i is broken into K clusters and K potential key frames are obtained with one frame (the middle frame) chosen from each cluster. Those key frames are inevitably redundant for a long-distance shot with many frames.

Reducing the number of key frames to represent the shot becomes our next procedure after extracting the potential key frames from shot Q_i. In this procedure, we calculate VFSF values for each three consecutive potential key frames, estimating their levels of similarity and comparing computed VFSF values with the threshold θ. If $V > \theta$, these three frames are similar enough. In this case, we retain only the middle potential key frame and compare it with the next potential key frame in sequence. On the other hand, if these three frames are not sufficiently similar to discard any frame, all three frames are taken as key frames and the last frame of the three frames is further compared with the next potential key frame. An example of a football video shot exploiting our algorithm is shown in Figure 4, in which the given shot is separated into two clusters. Consequently, two potential key frames (frame 316 and 341) are acquired. Considering that the two frames share similar vision content, frame 341 (framed by a rectangle) is intended to be the final key frame to compose video summarization.

FIGURE 4. A football video: Potential key frames selected from each cluster of a given shot

5. **Experimental Results.** In this section, we evaluate the performance of the proposed method. Video segmentation and extraction of key frame are separately discussed owing to the differences in their calculation procedures. Representatives of several video types (see Table 1) have been tested in our experiments, including news, sports, downtown,

micro film and the entertainment field. In the last column of Table 1, GT is the number of precise segment locations determined by human observers for each video sequence, which will be stated in detail and used for experiments in the following.

TABLE 1. Video set used in the experiments

Video	Duration(s)	Frame Size	Total Frames	GT
golf	51	352×240	533	4
football	57	448×336	589	7
downtown	64	352×240	706	2
aquarium	69	352×240	802	3
news	147	640×480	2249	28
micro film	995	640×360	8845	106

5.1. **The performance of video segmentation.** Inspired by the receiver operating features described in [20], we estimate the performance of video segmentation making use of *recall* and *precision*. Let us define two parameters first. N_{GT} is used to designate the precise locations of segments determined by human observers for each video sequence and N_D is the number of detections (correct or false). The *recall* method which is also known as the sensitivity defines the ratio of correct experimental segments (obtained by our method) over the total number of true segments. The *precision* is the ratio of correct experimental segments over the experimental segments. We formally define these quantities as

$$\text{Recall} = \frac{|N_{GT} \cap N_D|}{|N_{GT}|} \tag{10}$$

$$\text{Precision} = \frac{|N_{GT} \cap N_D|}{|N_D|} \tag{11}$$

where $|N_{GT}|$ represents the cardinality of set N_{GT}.

In order to demonstrate the efficiency of the proposed method, we compare our approach to the MI algorithm proposed in [21] and CHD (color histogram differences) algorithm [22] in each of the experiments listed in Table 1. The major reason that we choose MI and CHD methods is that our method and MI, CHD all use the feature of adjacent frames to detect shot boundaries and extract key frames. The MI algorithm uses mutual information (MI) and joint entropy to describe the difference between two consecutive frames to detect shot boundaries, after which MI is used as well for key frames extraction. When determining the segment frames is implemented in the MI algorithm, the size of temporal window parameter N_W will influence the detection results to some extent. As one of the most reliable variants of histogram-based detection algorithms, the basic idea of CHD method is that the color content does not change rapidly within but across segments. It compares its differences between color histogram of contiguous frames to a threshold, determining whether this difference is large enough to be detected as a segment part. Thus, the selection of the threshold will affect the shot boundary detection performance as well. Figure 5 provides us the video frames similarity of a football video calculated by applying our method (VFSF) and MI, CHD method respectively. Comparing those three approaches in Figure 5, we can see that VFSF method can distinguish the segment frames more obviously than the MI approach and CHD approach. In addition, in the condition of consecutive frames with much more similar contents (like the frames from frame 0 to frame 261 in Figure 5), our method provides consistently high quality values close to 1 while the performance of MI and CHD methods are poor in this situation.

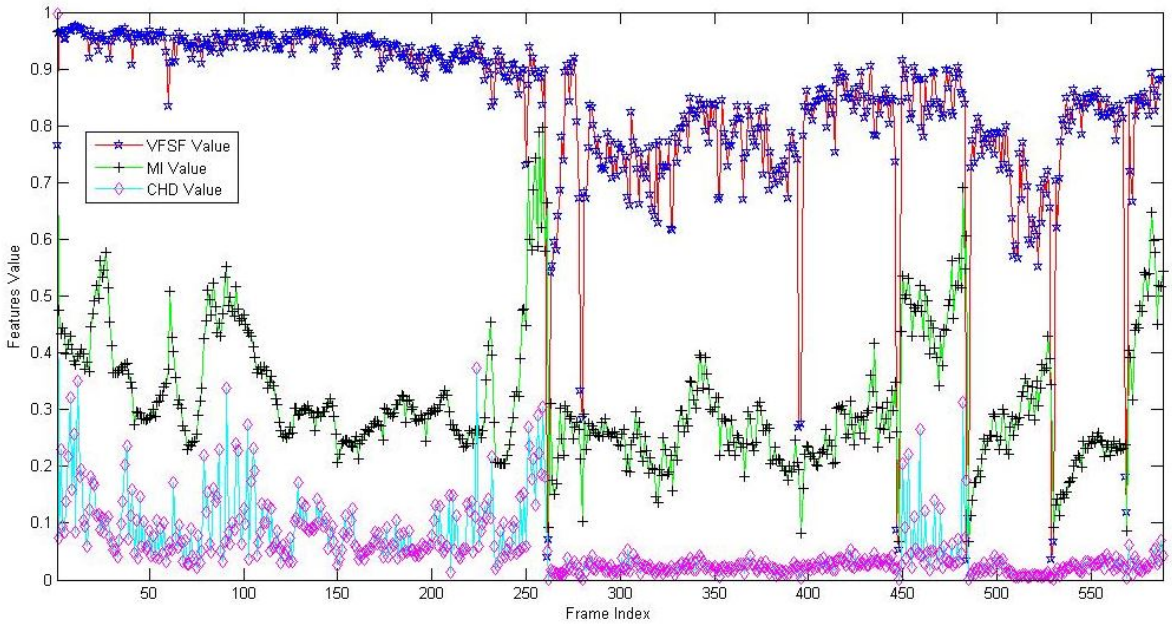


FIGURE 5. Comparison results between the VFSF and MI, CHD methods for calculating the video frames similarity of a football video sequence

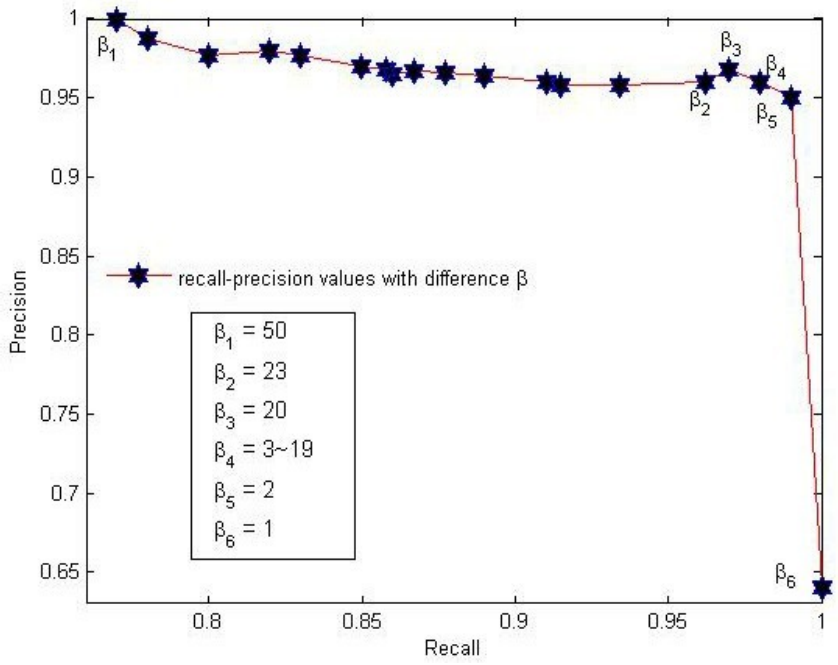
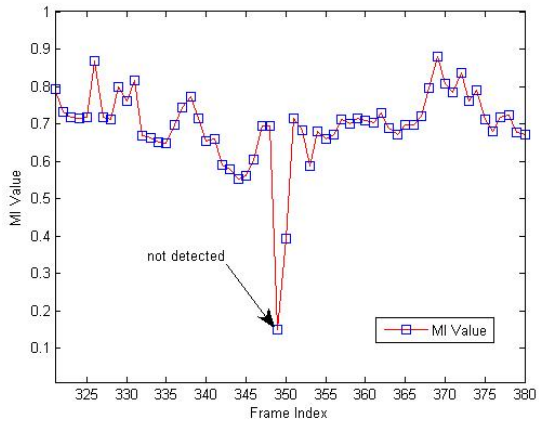
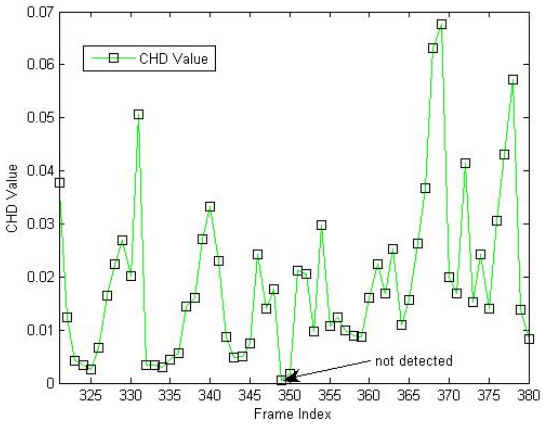


FIGURE 6. Recall-precision curve for the gaussian video segmentation approach performed with a threshold β lying in the range $[1, 50]$. When the value of β is in the range $[3, 19]$, the experimental results share the same recall-precision which is the best.

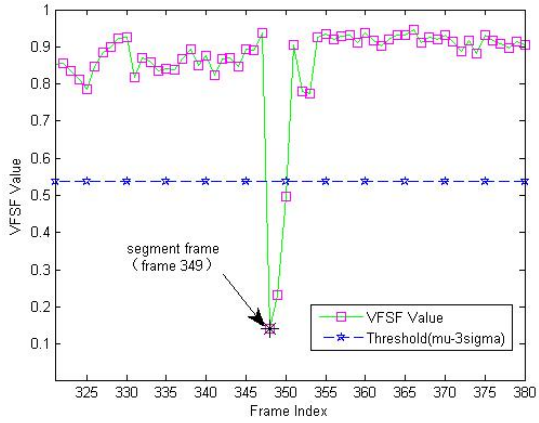
In our experiments of using the gaussian method to segment, we first need to determine the best value of the threshold β. Therefore, different β values are tested, winning the recall-precision graph shown in Figure 6. In the following experiments of testing the performance with all videos, we selected a threshold $\beta = 10$. The segmentation estimation results of the proposed method for several videos are shown in Table 2 in the columns



(a)



(b)



(c)

(d)

FIGURE 7. Boundary detection within a shot for a case with a change between images with very similar background color and the same object but difference locations of the object: (a) MI method; (b) CHD method misses this change; (c) GVFSF method can detect this change; (d) shot change in frame 349 (the position that GVFSF method can detect while MI method and CHD method do not spot).

TABLE 2. Video segmentation results with fixed threshold

Videos	GVFSF method		MI method		CHD method	
	Recall	Precision	Recall	Precision	Recall	Precision
football	1.00	1.00	0.86	1.00	0.86	1.00
aquarium	1.00	1.00	0.67	0.40	0.67	1.00
downtown	1.00	1.00	1.00	1.00	1.00	0.67
movie	0.98	0.96	0.98	0.95	0.89	0.96
golf	0.75	1.00	0.75	1.00	1.00	0.80
news	0.96	1.00	0.93	0.72	0.93	0.79

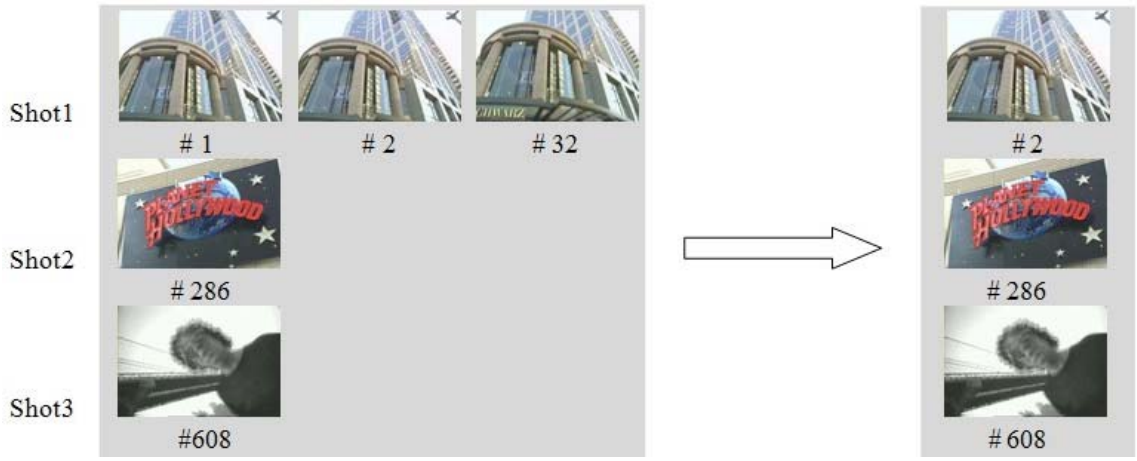


FIGURE 8. Key frame extraction for a downtown video sequence which is segmented into three shots: (a) five extracted potential key frames; (b) final three key frames obtained by merging two redundant frames (#1 and #32) in shot 1.

labeled as GVFSF (gaussian VFSF). Although MI method and CHD method show a high shot detection rate for most tested videos to a certain degree (see Table 2), they are generally lower than that of our method in which nearly 100% accuracy rate is obtained with relative short segments. We note in particular that in the case of shot changes across images with very similar background color distributions and with the same object but different locations of the object (Figure 7(d)), the GVFSF method can detect this change correctly and accurately (Figure 7(c)) while the MI method (Figure 7(a)) and CHD method (Figure 7(b)) all fail to spot the change. This can be seen clearly from Section 3.2 that our method successfully detects shot boundaries globally and is not affected by local information.

5.2. **The performance of key frames extraction and video summarization.** Applying our key frames extraction algorithm to all tested video sequences (Table 1), we have demonstrated its effectiveness especially for shots with large visual content changes.

In our tests, we experimentally selected the threshold $\theta = 0.1$. We first chose only one frame from shots without significant visual content changes and used the split-merge method to find more potential key frames for shots with large content changes. An example of extracting potential key frames from a downtown video is illustrated in Figure 8(a). Segmented into three shots, five potential key frames including three that are similar in visual content are extracted from the first shot. The procedure of reducing the number

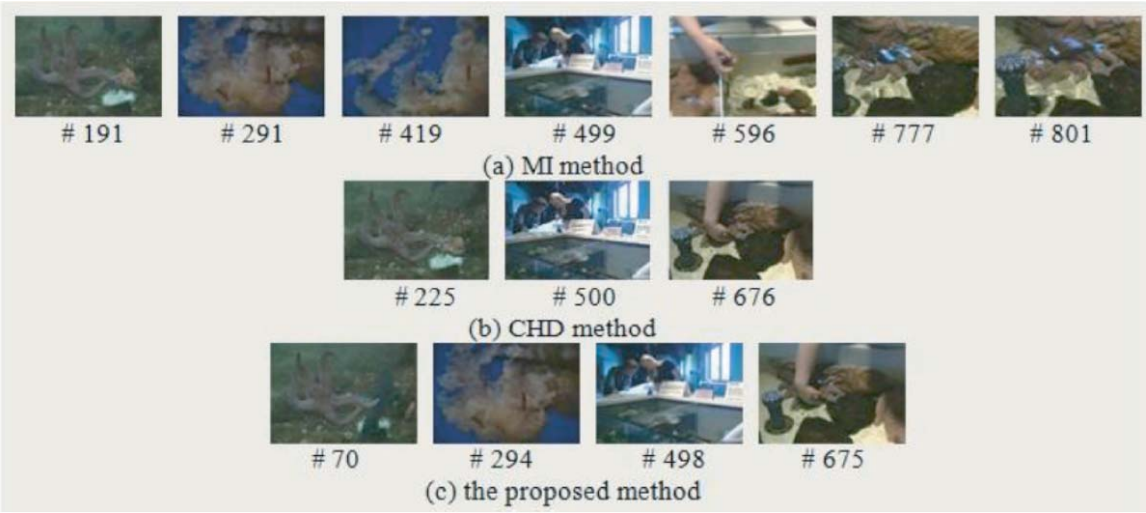


FIGURE 9. Key frames extracted from an aquarium video sequence: (a) key frames extracted by MI method; (b) key frames extracted by CHD method; (c) key frames extracted by the proposed method

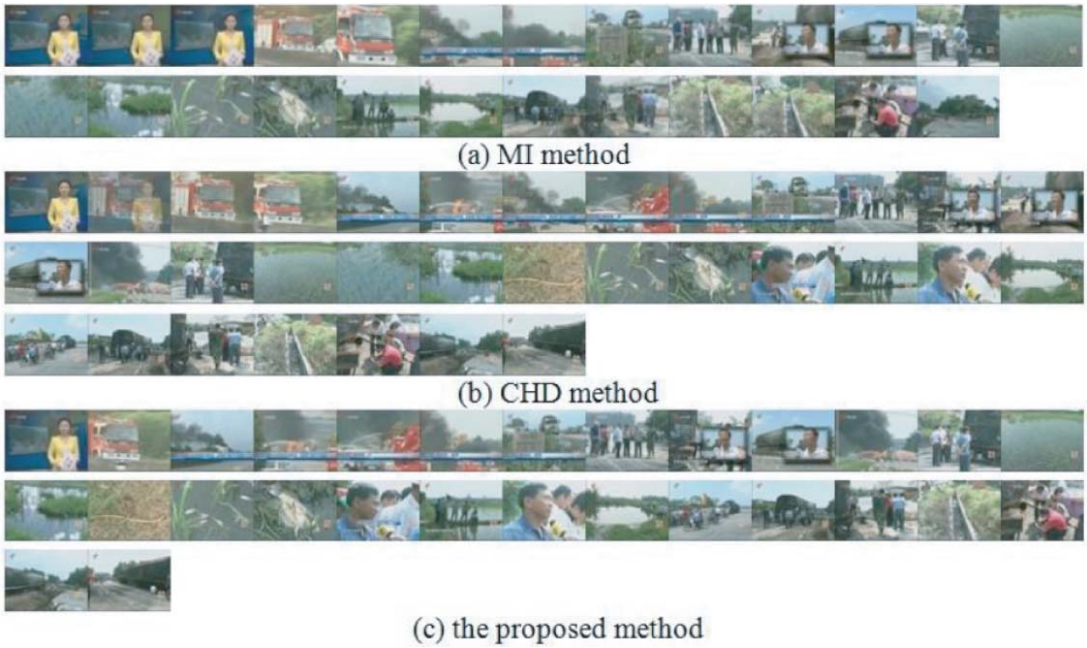


FIGURE 10. Video summarization result of a news video employing the proposed method in contrast to the results of MI and CHD methods

of potential key frames is then performed, and its results are shown in Figure 8(b). Two redundant frames in the first shot are successfully identified and discarded.

By extracting the final key frames from a set of potential key frames, we can generate a video summarization facilitating retrieval and indexing for a long video. The performance of key frames extraction for an aquarium video employing MI, CHD methods and the proposed method respectively can be seen in Figure 9. Some of the extracted key frames employing MI method, as shown in Figure 9(a), are redundancy, such as frame 291 and frame 419, while the result of using the proposed approach (Figure 9(c)) is ameliorative. In most previous key frames extraction algorithms, there is only one key frame within each shot, just like applying CHD method in Figure 9(b) which omits a part of information that

aquatic plants move about in the aquarium affecting the overall video retrieval. Superiorly, the extracted key frames using the proposed method overmatch the results of MI and CHD algorithm with neither redundancy frames nor omissive frames (frame 294 in Figure 9(c) is selected as the final key frame). Figure 10 demonstrates the final video summarization of a news video employing the proposed method in contrast to the results of MI and CHD methods, from which we can see obviously that the summarization of proposed approach not only models the whole process of the news event that a benzene tank car outbroke of a fire and the report about environment pollution caused by extinguishment but also captures the emphasis very well to facilitate video indexing for viewers.

6. **Conclusions.** In conclusion, a novel video summarization method using VFSF-based gaussian theory and split-merge is presented, and the experiments on test videos show that the proposed method can adaptively provide video segmentation with higher recall accuracy for finding representative content in comparison with the similar MI and CHD methods. In addition, a further examination about the split-merge performance for key frame extraction was performed on different types of videos, which shows the proposed summarization approach not only can effectively eliminate the redundancy in key fames but also find the most representative frames to generate a video summarization. The key frames extraction of the proposed method is based on the number of clusters and depended on visual content complexity of a shot. In consideration of the easy implement, the proposed approach is reasonable for the application of on-line video indexing, cutting down the time of viewers for information retrieval.

Acknowledgment. This work is supported by Foundation of Hebei Human Resources and Social Security Department (No. 2011226026), Hebei Science and Technology (Grant No. 11963546D) and National Natural Science Foundation of China (No. 61102103). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] X. U. Cabedo and S. K. Bhattacharjee, Shot detection tools in digital video, *Proc. of Non-Linear Model Based Image Analysis*, pp.121-126, 1998.

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

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

[4] I. K. Sethi and N. Patel, A statistical approach to scene change detection, *Proc. of SPIE*, vol.2420, pp.329-338, 1995.

[5] A. Dailianas, R. B. Allen and P. England, Comparison of automatic video segmentation algorithms, *Proc. of SPIE Photonics East'95: Integration Issues in Large Commercial Media Delivery Systems*, vol.2615, pp.2-16, 1995.

[6] G. Ahanger and T. Little, A survey of technologies for parsing and indexing digital video, *J. Vis. Commun. Image Represent.*, vol.7, no.1, pp.28-43, 1996.

[7] N. V. Patel and I. K. Sethi, Video shot detection and characterization for video databases, *Pattern Recognit.*, vol.30, no.4, pp.583-592, 1997.

[8] S. Tsekeridou and I. Pitas, Content-based video parsing and indexing based on audio-visual interaction, *IEEE Trans. on Circuits Syst. Video Technol.*, vol.11, no.4, pp.522-535, 2001.

[9] C.-L. Huang and B.-Y. Liao, A robust scene-change detection method for video segmentation, *IEEE Trans. on Circuits. Syst. Video Technol.*, vol.11, no.12, pp.1281-1288, 2001.

[10] T. Butz and J. Thiran, Shot boundary detection with mutual information, *Proc. of 2001 IEEE Int. Conf. Image Processing*, vol.3, pp.422-425, 2001.

[11] Y. Tonomura, A. Akutsu, K. Otsuji and T. Sadakata, VideoMAP and VideoSpaceIcon: Tools for anatomizing video content, *Inter CHI'93 Conference Proceedings*, Amsterdam, The Netherlands, pp.131-136, 1993.

[12] H. Ueda, T. Miyataka and S. Yoshizawa, Impact: An interactive natural-motion-picture dedicated multimedia authoring system, *Proc. of Human Factors in Computing Systems CHI'91*, pp.343-350, 1991.

[13] Y. Zhuang, Y. Rui, T. Huang and S. Mehrotra, Adaptive key frame extraction using unsupervised clustering, *Proc. of ICIP'98*, vol.1, pp.866-870, 1998.

[14] S. Ju, M. J. Black, S. Minneman and D. Kimber, Summarization of video-taped presentations: Automatic analysis of motion and gestures, *IEEE Trans. on Circuits Syst. Video Technol.*, vol.8, no.5, pp.686-696, 1998.

[15] C. Toklu and S. P. Liou, Automatic keyframe selection for content-based video indexing and access, *Proc. of SPIE*, vol.3972, pp.554-563, 2000.

[16] H. J. Zhang, J. Wu, D. Zhong and S. W. Smoliar, An integrated system for content-based video retrieval and browsing, *Pattern Recognit.*, vol.30, no.4, pp.643-658, 1997.

[17] J. Jiang, X.-P. Zhang and A. C. Loui, A new video similarity measure based on video time density function and dynamic programming, *Proc. of ICASSP*, Prague, Czech Republic, 2011.

[18] Y. Zhang, Z. Wei, Z. Zhao, X. Song and L. Fu, A gaussian video summarization method using video frames similarity function, *ICIC Express Letters*, vol.7, no.7, pp.1997-2003, 2013.

[19] J. Jiang and X.-P. Zhang, A content-based rapid video playback method using motion-based video time density function and temporal quantization, *Proc. of Int. Work. on Social, Adaptive and Personalized Multimedia Interaction and Access (SAPMIA 2010) in Conjunction with ACM Multimedia*, 2010.

[20] P. Browne, A. F. Smeaton, N. Murphy, N. O'Connor, S. Marlow and C. Berrut, Evaluation and combining digital video shot boundary detection algorithms, *The 4th Irish Machine Vision and Information Processing Conf.*, Belfast, Ireland, 2000.

[21] Z. Cerneková, I. Pitas and C. Nikou, Information theory-based shot cut/fade detection and video summarization, *IEEE Trans. on Circuits and Systems for Video Technology*, vol.16, no.1, pp.82-91, 2006.

[22] R. Lienhart, Comparison of automatic shot boundary detection algorithms, *Proc. of SPIE Storage and Retrieval for Image and Video Databases VII*, San Jose, CA, USA, vol.3656, pp.290-301, 1999.