Unsupervised sports video scene clustering and its applications to story units detection

Weigang Zhang[*a], Qixiang Ye[b], Liyuan Xing[c], Qingming Huang[c] and Wen Gao[a,b]

[a] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, 150001
[b] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China
[c] Graduate School of Chinese Academy of Sciences, Beijing, 100080, China

ABSTRACT

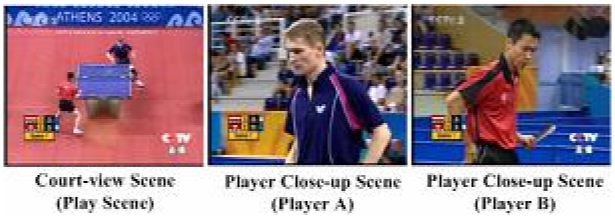
In this paper, we present a new and efficient clustering approach for scene analysis in sports video. This method is generic and does not require any prior domain knowledge. It performs in an unsupervised manner and relies on the scene likeness analysis of the shots in the video. The two most similar shots are merged into the same scene in each iteration. And this procedure is repeated until the merging stop criterion is satisfied. The stop criterion is defined based on a *J value* which is defined according to the Fisher Discriminant Function. We call this method *J-based Scene Clustering*. By using this method, the low-level video content representation—shots could be clustered into the mid-level video content representation—scenes, which are useful for high-level sports video content analysis such as play-break parsing, story units detection, highlights extraction and summarization, etc. Experimental results obtained from various types of broadcast sports videos demonstrate the efficacy of the proposed approach. Moreover, in this paper, we also present a simple application of our scene clustering method to story units detection in periodic sports videos like archery video, diving video and so on. The experimental results are encouraging.

Keywords: Sports video analysis, unsupervised scene clustering, story units detection

1. INTRODUCTION

In recent years, more and more sports video data is being produced, distributed and made available all over the world. Therefore, there is an emerging need for efficient management including abstracting, personalizing, indexing, and retrieving. Furthermore, sports video appeals to large audiences and many sports content producers pay much attention to its management. And there is a growing interest in the research of sports video management algorithms.

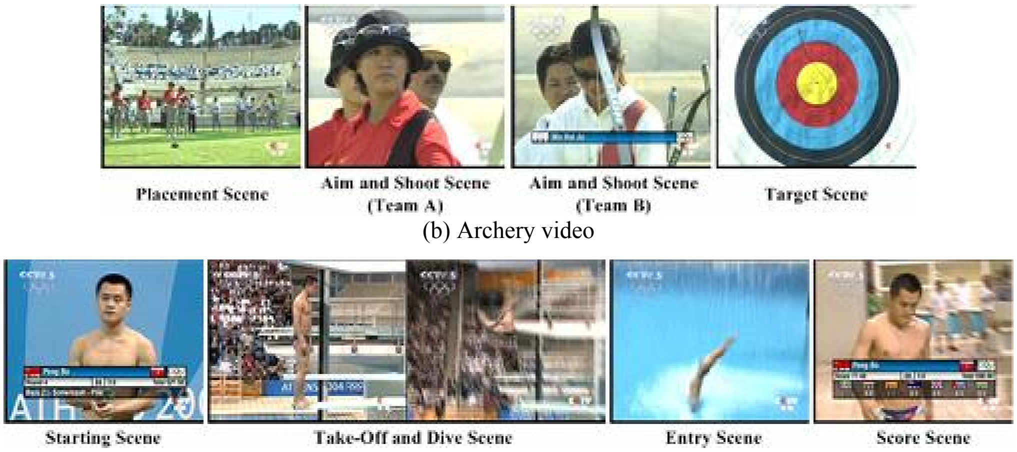
Compared with other videos such as news video and movie, sports video has its own special characteristics. A sports game usually occurs in a specific field and has well-defined temporal structures. In addition, sports video is usually taken by some fixed cameras in the field that results in some recurrent distinctive scenes (or views) throughout the video. For example, in a table tennis video, there are three dominant scenes—court-view (play scene), player close-up scene of player A and player close-up scene of player B, as shown in Fig. 1a. In an archery team events video, there are four dominant scenes—placement scene, aim and shoot scene of Team A, aim and shoot scene of Team B and target scene (In archery individual events, there is only one aim and shoot scene), as shown in Fig. 1b. In a diving video, there are also four dominant scenes—player starting position scene, take-off and dive scene, entry scene and score scene, as



(a) Table tennis video

[*] Further author information: {wgzhang, qxye, lyxing, qmhuang, wgao}@jdl.ac.cn

Visual Communications and Image Processing 2005, edited by Shipeng Li,
Fernando Pereira, Heung-Yeung Shum, Andrew G. Tescher, Proc. of SPIE Vol. 5960
(SPIE, Bellingham, WA, 2005) · 0277-786X/05/$15 · doi: 10.1117/12.631389



(b) Archery video

(c) Diving video

Figure 1: Typical dominant scene examples in sports video.

shown in Fig. 1c. These scenes usually have consistent visual attributes that do not vary largely from game to game and each comprises several shots whose visual contents are similar. They are quite useful for high-level sports video content analysis such as play-break parsing, story units detection, highlights extraction and summarization, etc. For example, in story units detection of sports video, these recurrent dominant scenes are the main proportion of a story unit. So if we obtain these dominant scenes, label and arrange them along the original video timeline, we can detect the story units easily. When the story units are obtained, they could be used to index and summarize the sports video.

As other generic videos, a common first step of sports video analysis is to segment the video into a series of low-level content representation units—shots, each comprising a sequence of interrelated consecutive frames taken contiguously by a single camera and representing a continuous action in time and space. Although this effort provides better access than unstructured raw video stream, a one-hour video may have hundreds of shots and the granularity for shot is still too small. It's not suitable for users to effectively manage and personalize the video at the shot level. Furthermore, users are generally more interested in the underlying events and story units in a video. So the shots should be further grouped into the mid-level content representation units—scenes, which are useful for high-level content analysis. Especially in sports video which has well-defined content structures and domain-specific rules, a fixed number of dominant scenes appear periodically over the video footage. We should take full advantage of these dominant scenes and use them to help to perform sports video content analysis and semantic indexing.

In the past a few years, some conventional algorithms such as *k*-means clustering and hierarchical clustering have been exploited to cluster shots into scenes[1, 2]. These methods, however, require some prior knowledge to obtain good clustering results. For example, the number and initial centroids (prototypes) of clusters are required by *k*-means clustering algorithm to obtain good results in a small number of iterations[3]. It should be noted that the estimation of correct number of clusters and the decision of good cluster centroids had been longstanding problems in the cluster analysis [3, 4]. They are usually data or application dependent. By restricting to the popular sports video, there are also a few existing works to address the algorithms of scene clustering [5, 6, 7]. Lu et.al[5] formulate video scene clustering as a problem of graph partitioning and present a scene clustering method which does not need the number of scene clusters as input. They also present an unsupervised dominant scene clustering method for sports video based on traditional *k*-means clustering algorithm[6], in which recursive peer-group filtering (PGF) scheme is used to solve the problem of scene centroid initialization and time-coverage criterion is used to solve the problem of scene number estimation. To speed up the clustering process and minimize the space requirement, they apply PCA and LDA techniques to reduce the feature dimension. Tao et.al[7] use *k*-means clustering to obtain shot mosaic clusters and use the results to mine "play" shots.

In this paper, we present a new clustering approach for scene analysis in sports video, which does not need any prior knowledge and performs in an unsupervised manner. By recurrent merging procedures and an appropriate merging stop criterion, this method produces the final scene clustering results automatically. In this approach, the scene merging stop criterion is provided based on a *J value* which is defined according to Fisher Discriminant Function. This not only determines the best merging stop point, but also makes the final scene results more consistent with human perception. We call this method *J*-based scene clustering. Moreover, in this paper, we also present a simple application of our scene clustering method to detect story units, which are useful for sports video abstracting and indexing.

The rest of the paper is organized as follows. In section 2, we provide the flow chart of the proposed system. In section 3, we describe the *J*-based scene clustering method in detail. Section 4 is devoted to show the scene clustering experimental results on various types of broadcast sports videos. In section 5, we show how this method applies to story units detection in sports video and the detection results are also presented in this section. Conclusions are drawn in section 6.

2. FLOW CHART OF THE PROPOSED APPROACH

As shown in Fig. 2, the sports video is first segmented into a series of shots. Then several key-frames (In our approach, we choose five key-frames for each shot) are selected from each shot for shot content representation. After that the *J*-based scene clustering is performed and the scene clustering results are obtained. Finally, the story units of the sports video are detected by using the scene clustering results with additional domain knowledge.

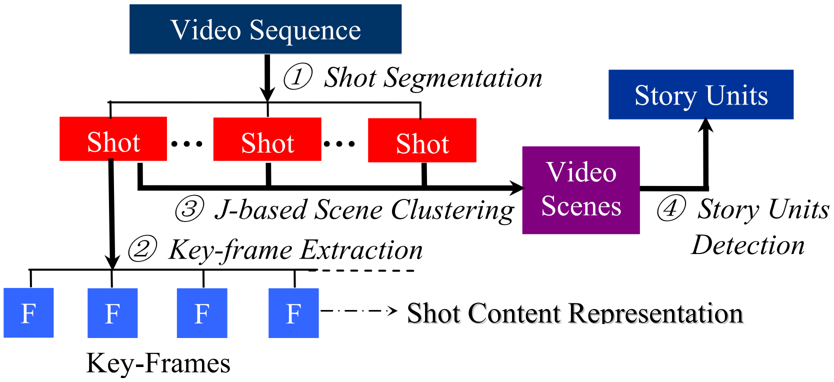


Figure 2: The flow chart of *J*-based scene clustering and its applications.

3. *J*-BASED SCENE CLUSTERING

3.1 Shot segmentation and representation

Shot segmentation is the first stage to the following higher level video structure analysis. In our approach, by a common color histogram based shot segmentation algorithm[8], the sports video is first segmented into N shots $S=\{s_1,s_2,\cdots,s_N\}$. For the simplicity of computation, five key-frames are then extracted from each shot at an equal interval, which are used to represent the visual content of their corresponding shot, $K^i=\{k_1^i,k_2^i,k_3^i,k_4^i,k_5^i\}$, where k_3^i is the center frame of shot i.

3.2 Scene likeness of shots

Each shot is considered as the fundamental processing unit and the normalized HSV (Hue-Saturation-Value) color histograms of its key-frames are computed as features to evaluate the scene likeness. In this paper, the color coordinates of HSV color space are uniformly quantized into 16 (Hue), 4 (Saturation) and 4 (Value) bins, respectively, resulting in a total of 256 quantized color bins[9].

The distance between two shots (or scenes) is used to represent their scene likeness. The smaller the distance is, the more similar the two shots (or scenes) are (the higher their scene likeness is). The distance of shot s_i and shot s_j is

$$SD(s_i,s_j)=1/2\times\left(Min\{d(k_m^i,k_n^j)\}+\hat{M}in\{d(k_m^i,k_n^j)\}\right),m,n\in\{1,2,\cdots,5\}; \qquad (1)$$

$$d(k_m^i, k_n^j) = \sum_{b=1}^{B} \left| H_m^i(b) - H_n^j(b) \right| \Big/ B; \tag{2}$$

Where Min and $\hat{Min}$ denote the minimum and the second minimum distance between the key-frames of shot s_i and s_j, respectively. $d(k_m^i, k_n^j)$ is the HSV histogram distance of key-frame k_m^i and k_n^j. H_m^i is the normalized 256-bin HSV color histogram of key-frame m in shot i. And B, equal to 256, is the total number of quantized colors. When calculating the shot-scene distance or scene-scene distance, the weighed mean feature value of all shots in a scene cluster is used to represent the scene's content.

3.3 *J*-based scene clustering

At the beginning, each shot is initialized as a scene. The likeness of each two initial scenes is calculated by formula (1). Then these initial scenes $S^c = \{s_1^c, s_2^c, \cdots, s_N^c\}$ are merged into disparate scene clusters. In each iteration the two scenes whose distance is the smallest are merged into one scene cluster. This procedure is repeated until the merging stop criterion is satisfied. Then the merging process is stopped and the final scene clustering results are obtained.

Before we provide the merging stop criterion, we first define a *J value* based on Fisher Discriminant Function. The *J* value is the total scene cluster scatter, which describes the ratio of intra-cluster scatter to inter-cluster scatter of the scenes in the merging process. When the scene number is K_l in the merging process, the *J* value is defined as

$$J_l = \frac{\sum_{c=0}^{K_l} J_w^c}{J_t} = \frac{\sum_{c=0}^{K_l} \sum_{i=0}^{N_c} \left\| \vec{s}_i^{\,c} - \vec{s}_{mean}^{\,c} \right\|}{\sum_{i=0}^{N} \left\| \vec{s}_i - \vec{s}_{mean} \right\|} \tag{3}$$

where J_t is the total inter-cluster scatter of the initial scene sequence, J_w^c is the intra-cluster scatter of scene cluster c. N is the total scene number in the initial scene sequence, N_c is the shot number of scene cluster c. $\|\bullet\|$ represents the Euclidean distance. $\vec{s}_i^{\,c}$ and $\vec{s}_{mean}^{\,c}$ denote the feature value of shot i and the mean feature value of all shots in scene cluster c, respectively. $\vec{s}_i$ denotes the feature value of the initial scene cluster i and $\vec{s}_{mean}$ denotes the mean feature value of all initial scenes in which each has only one shot.

At the beginning of the procedure, the intra-cluster scatter of all initial scenes is 0 and J_l is 0.0. With the increment of intra-cluster scatter when two scenes are merged into one, J_l is increasing. If all the scenes are merged into one scene

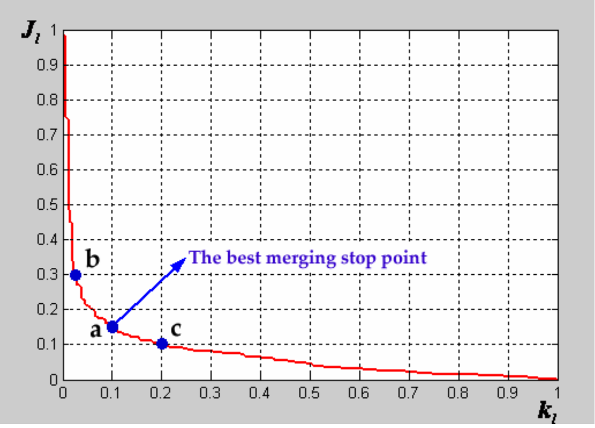
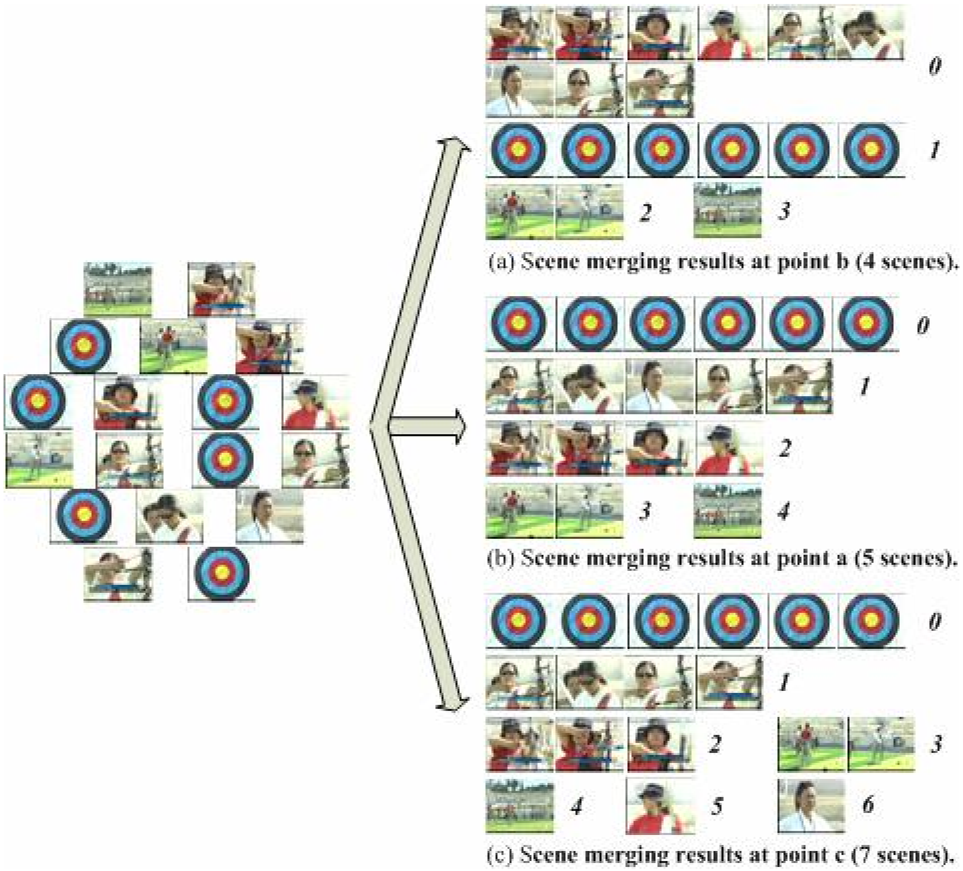


Figure 3: The relation curve of J_l and k_l on an archery video.



(a) Scene merging results at point b (4 scenes).

(b) Scene merging results at point a (5 scenes).

(c) Scene merging results at point c (7 scenes).

Figure 4: Comparison of scene merging results according to different points on the relation curve.

cluster, J_l will reach the maximum and is equal to 1.0. The smaller J_l is, the more similar the shots within each scene cluster are. Actually, it is expected that both J_l and the scene number are small. But in real situation, the value of J_l will increase with the decreasing of scene number. As a tradeoff between J_l and the scene number, we choose the point where $\boldsymbol{J_l + k_l}$ is the smallest as the best merging stop point, which is shown in Fig. 3. This is the defined stop criterion for the scene merging procedure. Here $k_l = K_l / N$ is the ratio of the scene number to the total number of scenes in the initial scene sequence in the merging process.

Some scene merging results according to different points on the relation curve of J_l and k_l are shown in Fig. 4. From the figure we can see that at point a the scene results are the best. So we choose the point a as the best merging stop point. And at point b and c, excessive merging and inadequate merging take place, respectively.

3.4 Algorithm description

The *J*-based scene clustering algorithm is described as follows:

1. Segment the sports video into shots, $S = \{s_1, s_2, \cdots, s_N\}$, N is the total shot number.
2. Extract five key-frames from shot i at an equal interval for shot content representation, $K^i = \{k_1^i, k_2^i, k_3^i, k_4^i, k_5^i\}$, where k_3^i is the center frame of shot i, and $i = \{1, 2, \cdots, N\}$.
3. Calculate the normalized 256-bin HSV histogram H_m^i (H-16, S-4, V-4) of key-frame m in shot i, $m = \{1, 2, \cdots, 5\}$ and $i = \{1, 2, \cdots, N\}$.

4. Each shot is initialized as a scene $S^c = \left\{ s_1^c, s_2^c, \cdots, s_N^c \right\}$ and the distance (scene likeness) of every two initial scenes $SD(s_i^c, s_j^c),\ i, j = \{1,2,\cdots,N\},\ i \neq j$ is calculated by Formula (1) and (2).

5. Merge the two most similar scenes whose distance is the smallest into one scene cluster, $s_l^c = (s_i^c, s_j^c)$.

   a). The weighed average HSV histogram feature value of the two merged scenes is computed to represent the visual content of the new scene, $H_m^l = (N_i * H_m^i + N_j * H_m^j)/(N_i + N_j), m \in \{1,2,\cdots,5\}$, N_i and N_j are the shot number of scene s_i^c and s_j^c, respectively.

   b). Recalculate the distance between the new scene s_l^c and other scenes.

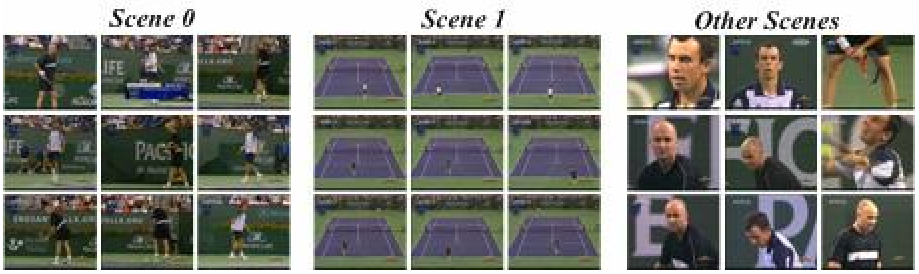
   c). Compute the value of $J_l + k_l$, where J_l is calculated according to Formula (3) and $k_l = K_l / N$, K_l is the scene number in the merging process.

6. Execute step 5 continuously without stop in the merging process to find the best merging stop point where $J_l + k_l$ reaches the minimum. Then restart step 5 and repeat it until $J_l + k_l$ reaches the minimum. Merging is stopped and the scene clustering results are obtained.

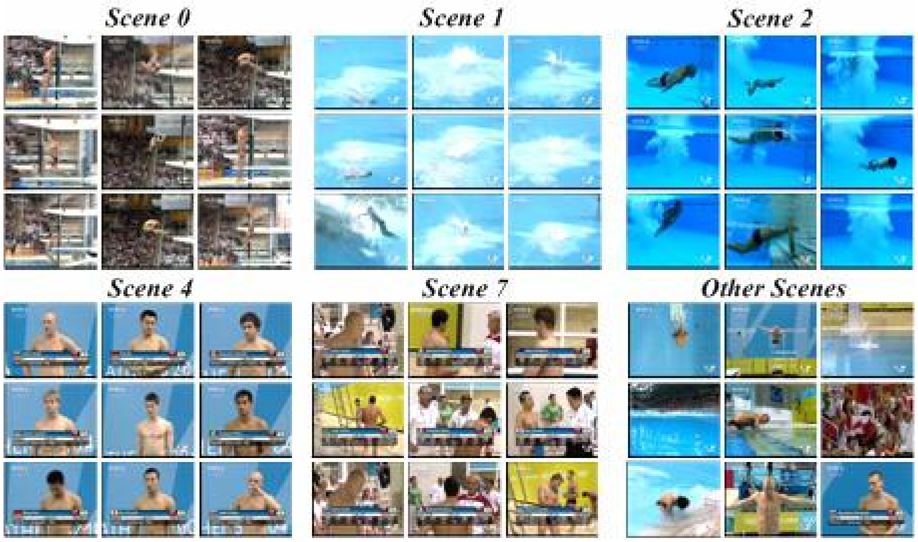
7. Sort the result scenes according to their shot number in a descending order, and output the ordered scene clustering results.

4. EXPERIMENTAL RESULTS OF SCENE CLUSTERING

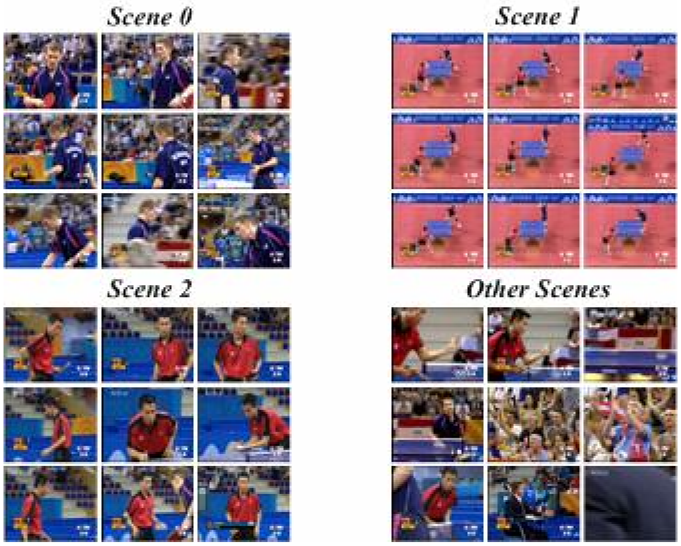
We have applied the proposed *J*-based scene clustering method to cluster scenes in several broadcast sports videos of different games, including archery, diving, tennis, table tennis and gymnastics, etc. All the test videos were compressed by MPEG-2, digitized at 25 frames/s (PAL) with a resolution of 352×288. The durations of the videos range from 6 to 31 minutes. And most of them are recorded from the living broadcast programs of Athens Olympic 2004.



(a) Tennis (11 scenes out of 36)



(b) Diving video (14 scenes out of 42)



(c) Table tennis video (9 scenes out of 9)

Figure 5: Scene clustering results of our proposed approach. The first 9 shots of each dominant scene and the first 1 shot of each scene in “Other Scenes” are shown in this figure. In (a), “Other Scenes” denotes scene 2 to 10. In (b), “Other Scenes” denotes scene 3, 5, 6 (replay scenes) and 8 to 13. In (c), “Other Scenes” denotes scene 3, 4, 5 (the first two shots are shown) and 6 to 8.

In the experiments of our *J*-based scene clustering method, the only input is a test sports video file and then our prototype system outputs the entire scene clustering results automatically. It performs in a completely unsupervised manner. Furthermore, in our approach, most of the processing time is consumed by calculating the histogram distance (scene likeness) of different shots (or scenes) and the scene clustering is completely in real-time.

Some scene clustering results obtained from various test sports videos are shown in Fig. 5. In these experimental results, the scenes which contain more than five shots occupy almost 90% content of the video. Compared with the dominant scene clustering results presented in Lu et.al’s work[6], our clustering results are more consistent with the original scene structure of the sports video and more applicable for high-level content analysis. In their work, only the first one or two dominant scenes are obtained and the other shots are simply considered as another dominant scene. Thus, these dominant scenes contain little structure information of the sports video and they are not suitable for high-level content analysis.

5. APPLICATIONS TO STORY UNIT DETECTION

As audiences remember the video content in terms of story units rather than the visual appearance variations in shots or scenes, it is a necessity to organize the video content in terms of single story unit that represents the conceptual chunks in audiences’ memory. These story units can further be used for video summarization or indexed by semantic objects to facilitate video browsing.

In many periodic sports videos, such as archery, diving, gymnastics, weightlifting and so on, there is a typical structure as shown in Fig. 6. These sports videos have a fixed order of scenes and show the characteristic of periodicity. According to the structure layers of these sports videos, we can segment and index them at different levels to meet the different demands. In this paper, the story unit is defined at the Player-Level. And each story unit is a complete competition interval which starts with the preparation scene of a player and ends with a score result scene. For instance, in diving video, a story unit begins with the player standing on the dive platform or springboard, then follows with the actions of taking-off, diving and entering water, and ends with the score of the player announced, as shown in Fig. 1c.

When scene clustering is completed, we can use the clustering results to detect story units in sports video with additional domain knowledge. Before the clustering results are applied to detect story units, they are ordered and weighed according to the number of shots they contain at first. Then, the n highest weighed scenes are selected as dominant scenes and labeled by $\{0,1,\cdots,n\}$. Here scene 0 has the most shots and n varies with the type of sports game. Other scenes and their shots are considered as noises and discarded. When a scene is labeled, its shots are also labeled as the same. After that we can obtain a labeled shot sequence along the original video timeline.

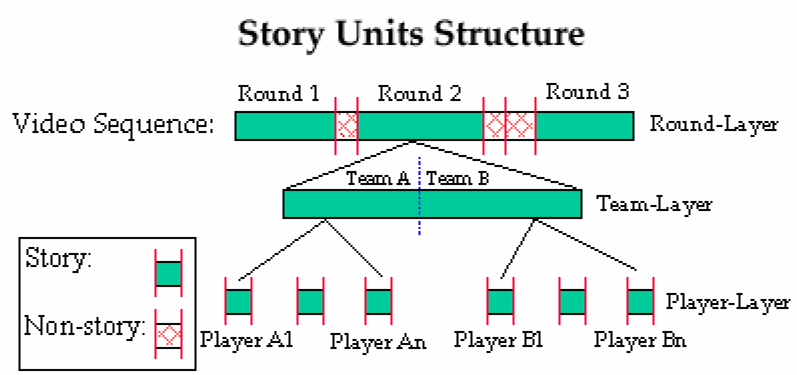


Figure 6: A typical story unit structure of sports video (For individual events, there is no Team-Layer).

We observe that besides the periodicity, such periodic sports videos have another characteristic that they have some key scenes in which each shot contains the key action content of the sports and denotes that a story takes place. For example, in an archery video, the target scene is the key scene which consists of the score shots of all the players, as shown in Fig. 1b. In a diving video, the take-off and dive scene as well as the entry water scene are the key scenes, as shown in Fig. 1c. As we know, in diving sports, the take-off action, dive action and entry water action of the athlete are the highlights of this game. The shots in these key scenes are usually taken by the main camera in the production of sports video. Their visual attributes do not vary largely with the change of story units of different players. Moreover, the key actions of all athletes will be captured in the video production. So the key scenes usually contain the most shots. Based on these observations, if we obtain these key scenes and their shots, the corresponding story units can be detected. In fact, as the proposed scene clustering method groups the shots into scenes based on their scene likeness analysis and the shots contained in each key scene usually have high scene likeness, so our method has the ability to cluster all the key shots into the corresponding key scenes. In our experiments on archery videos, scene 0 which contain the most shots is always the target scene. And in the experiments on diving videos, scene 0 and 1 which contain the most and second most shots are always the take-off and dive scene and entry water scene, respectively.

We use the obtained key scenes and their shots to detect story units on archery videos and diving videos. For an archery video, scene 0 (target scene) is the key scene. We select the top four scenes (scene 0, 1, 2 and 3) as dominant scenes. As each shot in scene 0 is the end shot of a story unit in archery video, so according to these shots in scene 0 and other shots in scene 1, 2 and 3, the story units can be detected easily. For a diving video, scene 0 (take-off and dive scene) and scene 1 (entry water scene) are the key scenes. We know that the entry water action always follows the take-off and dive action. Therefore, if a shot is included in scene 0 and its next shot is included in scene 1, they indicate a story occurs. So we can detect the story units according to these consecutive key shots in diving videos. If we find a pair of consecutive key shots, their corresponding story unit is detected.

We have done some experiments of story units detection on several broadcast sports videos of Athens Olympic 2004. The test set includes four videos: two archery videos (team events) and two diving videos. Their durations range from 14 min to 1 hour. We adopt the measure method used in information retrieval systems to evaluate the performance of our story units detection method. And the used story unit detection precision (P) and recall (R) are defined as below:

$$Precision = \frac{\#\ of\ correctly\ detected\ story\ units}{\#\ of\ detected\ story\ units} \quad (4)$$

$$Recall = \frac{\#\,of\,correctly\,detected\,story\,units}{\#\,of\,actual\,story\,units} \tag{5}$$

Another commonly used metric F_1 that combines precision and recall is also used for the evaluation. F_1 is high only when the precision and recall are both high.

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6}$$

The experimental results are shown in Table 1. As the proposed scene clustering approach shows good performance and the key scenes are well clustered, the results of story units detection is good. This shows that detecting story units with the scene clustering results is an effective way in sports video.

Table 1: Experimental results of story units detection

Sports video	# of frames	# of scenes / shots	# of shots in scene 0/1	# of actual story units	# of detected story units	# of correctly detected story units	Performance		
							P	R	F1
Archery1	20,657	19/140	37	37	37	37	1.00	1.00	1.00
Archery2	32,144	36/228	58	58	58	58	1.00	1.00	1.00
Diving1	46,480	42/337	60/47	28	28	27	0.96	0.96	0.96
Diving2	103,023	88/728	123/101	66	66	63	0.95	0.95	0.95

6. CONCLUSIONS

In this paper, an unsupervised sports video scene clustering approach is proposed, which can group the low-level video content representation—shots, into the mid-level video content representation—scenes. The scenes comprising similar shots are obtained by merging continuously. In each merging procedure, the two most similar shots (their scene likeness is the highest) are merged into the same scene. We also provide a merging stop criterion to decide when the scene merging process terminates and then the optimum scene clustering results are obtained. This criterion is defined based on a *J* value which is derived from Fisher Discriminant Function. By this *J*-based scene clustering method, the obtained scene results are more consistent with human perception. They also contain more structural information of the sports video which can be utilized for high-level content analysis. In this paper, we also apply the proposed method to detect story units in sports video and the experimental results are encouraging. All the works in this paper are a part of our sports video analysis project SPISES[10]. In the future, we will improve this scene clustering method and develop a more effective story units detection approach for sports video indexing, browsing and summarization.

ACKNOWLEDGEMENTS

This work is partly supported by NEC Research China on "Context-based Multimedia Analysis and Retrieval Program" and "Science 100 Plan" of Chinese Academy of Sciences.

REFERENCES

1. Yeung, M., B.-L. Yeo, and B. Liu, "Extracting Story Units from Long Programs for Video browsing and Navigation," *in Proc. IEEE Conf. on Multimedia Comput. and Syss*. 1996.
2. A. Hanjalic and HJ Zhang, "An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis," *IEEE Transactions on Circuits and Systems for. Video Technology*, vol. 9, no. 8, Dec. 1999.
3. J. Pena, J. Lozano, and P. Larranaga, "An empirical comparison of four initialization methods for the k-means algorithm," *Pattern Recognition Letters*, vol. 20, pp. 1027--1040, 1999.

4. Fraley, C. and AE Raftery, “How Many Clusters: Which Clustering Method? Answers via Model-Based. Cluster Analysis,” *Computer Journal*, 4, 578–588.
5. Hong Lu and Yap-Peng Tan, “An Efficient Graph Theoretic Approach to Video Scene Clustering,” *the Fourth Pacific-Rim Conference on Multimedia*, Dec 2003.
6. Hong Lu and Yap-Peng Tan, “Unsupervised clustering of dominant scenes in sports video,” *Pattern Recognition Letters,* vol. 24, no. 15, pp. 2651-2662, Nov. 2003.
7. Tao Mei, Yu-Fei Ma, He-Qin Zhou, Wei-Ying Ma, Hong-Jiang Zhang, “Sports Video Mining with Mosaic,” *MMM 2005*: 107-114.
8. Zhang H J, Kankanhalli A, Smoliar S W, “Automatic Partitioning of Full-Motion Video,” *ACM/Springer Multimedia Systems*, 1993, 1(1) :10～28.
9. Manjunath, B.S., Ohm, J.-R., Vasudevan, V.V., Yamada, A., “Color and texture descriptors,” *IEEE Trans. CSVT*, Volume: 11 Issue: 6 , Page(s): 703 -715, June 2001
10. SPorts vIdeo Summarization and Enrichment System, http://www.jdl.ac.cn/en/project/spises/SPISES.htm.