ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 1, No. 2, 2006, *pp*. 101-109

Retrieval of Sports Video Clips Using Audio-Visual Features and Text Information *

Yaqin Zhao [1 +], Xin He [1], Xianzhong Zhou [2]

[1] School of Automation, Nanjing University of Science & Technology, Nanjing 210094
[2] School of Management and Engineering, Nanjing University, Nanjing 210093

(*Received October 18, 2005, accepted February 12, 2006*)

Abstract. Video clip retrieval plays a critical role in the content-based sports video retrieval. This paper proposes a content-based retrieval strategy of sports video clip in which visual and auditory features and text information are extracted to locate similar video clips. Because in sports game play scenes are concerned and interested for most audiences, a long sport video is first automatically segmented into play segments and time-out segments based on audio feature in compressed domain. And then we obtain similar video clips based on sliding shot window algorithm and equivalence relation theory. Finally similar video clips are ranked by visual factor and order factor in terms of longest common sequence (LCS). The experimental results showed that the proposed method could effectively and efficiently retrieve similar video clips corresponding with a query clip in a sports video of TV program.

Keywords: content-based video clip retrieval, sliding shot window, equivalence relation, matching function

1. Introduction

The research interest of the digital video retrieval has increased enormously over the last decade, due to the remarkable increase of video data. To access the useful information and reduce the transmission cost, the research of video retrieval methods has become a popular research topic [1,2]. Sports video of TV programs is an important resource for the sports fans or the special sports analysis experts. There are lots of related works that are concerned with the parse of sports video such as Y. Gong et al. [3] that presented a method to automatic parse the soccer programs using domain knowledge, J. Assfalg [4] proposed a technique for highlight extraction in soccer videos, CHEN JY [5] described an automatic audio classification and segmentation for soccer video structuring. But most of the recent reported works related to sports video focus on semantic and structural parse [3-5], a few researches concentrate on the content-based retrieval of sports video image [6]. In particular, the retrieval of sports video clip barely starts. In fact, from the point of view of users, they usually query a video clip from a sport video, not a video image. Furthermore, from the angle of information quantity, video clip has more semantic information than single image, and search results based on video clip are more significant.

Two major concerns in video clip retrieval are respectively automatic segmentation and retrieval of similar video clips from video database, and similarity ranking of similar video clips. Yoshinori Ohno et al. [7] that describes a system to track soccer players and a ball by using color information from video images, Anil Jain [8] presented a retrieval method of video clip, etc. But these methods only focus on visual feature, not truly constitute a content-based multimedia research method. The retrieval of sports video clips is quite difficult because a great lot of similar play shots or clips appear periodically or semi-periodically in a continuous game video. In this paper, we introduce a retrieval method of sports video clip using audio-visual features. Usually, one is interested in some play segments, not those time-out segments. In order to reduce search space, in the method a long sports video is first segmented into some shorter video segments that are play segments and time-out segments, then matching function is defined in terms of equivalence relations corresponding with shots of query video clip. Several similar video clips are automatically segmented using sliding shot window algorithm. Longest common subsequence (LCS) is used to define order factor. Finally, both visual factor and order factor are considered for ranking similar clips according to similar degree of

* The research is supported by the Natural Science Foundation of Jiangsu province in China BK2004137).
+ Corresponding author. *E-mail address*: yaqinzhao@163.com.

query clip and similar clips. Fig. 1 shows the block diagram of the proposed sports video clip retrieval method.

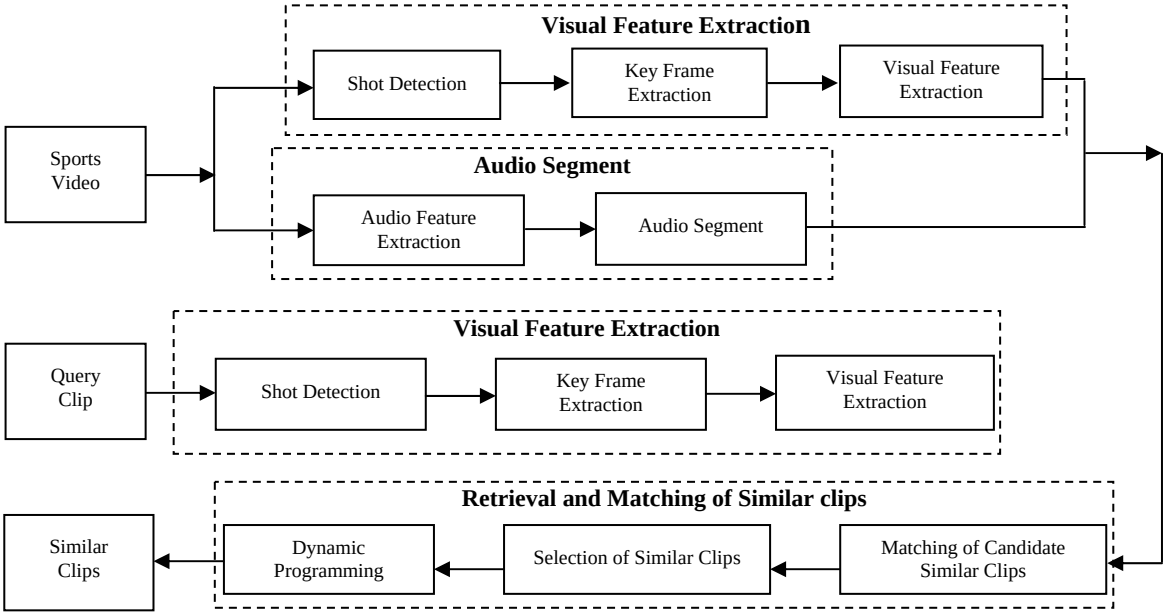

Fig.1. Block Diagram of the Proposed Method

2. Automatic Segment of Sports Video

In a long game video, some shots or clips are not concerned to audiences, such as time-out segments and commercials. Therefore a long game video is first automatically segmented into play segments and time-out segments in terms of audio features in order to reduce search time. Here, a time-out segment denotes video segment of a break by one coach or between two games, namely rest-time segments of players. Other video segments are called play segments. In time-out, some advertisements or wonderfully shots are usually inserted, sometimes cheering acts of rooters, or commentaries in studio. It is obvious that these video segments have few common visual features. However, we can find that music usually accompany in audio information of time-out segments, such as volleyball game, basketball game and soccer. Therefore, we segment a video where music appears in a period of time.

2.1. Audio Feature Extraction

MPEG audio feature is extracted using the method in [9]. Audio data is divided into short-time frames sequence of about 20ms. For each frame, root mean square of each sub-band vector is first computed using the equation $M(i) = (\sum_{t=1}^{32} (S_t[i]^2)/32)^{1/2}, \quad i = 1,2,\cdots 32$.

Where S_t is 32-dimension sub-band vector, M denotes the feature of current frame. The objective pf audio segment is for computational reduction, thus coarse segment cannot influence final results. Four feature of compressed domain are used as expressing dynamic and static characteristics of audio. They are four descriptive operators of audio signal on compressed domain, and are computed as follow.

(1) Centroid: $Centroid = \sum_{i=1}^{32} iM[i] / \sum_{i=1}^{32} M[i]$, the *centroid* shows basic audio frequency band.

(2) Rolloff: $Rolloff = \arg(\sum_{i=1}^{R} M[i] = 0.85 \sum_{i=1}^{32} M[i])$, the *rolloff* denotes the frequency that energy of audio signal attenuate to 3dB.

Spectral Flux is defined as the normalized difference of the vector M between two adjacent frames. It denotes dynamic feature of audio signal.

Root Mean Square: $RMS = \sqrt{\frac{\sum_{i=1}^{32} (M[i]^2)}{32}}$, it shows the intension of current audio frame.

2.2. Audio Segment

According to 2.1, each short-time audio frame is represented with a 4-dimension feature vector f_t, and then we compute logarithm of Euclidian distance between two adjacent feature vector f_t and f_{t+1} using (1).

$$d_t = \lg\left(\sum_{i=1}^{4}(f_{ti} - f_{(t+1)i})^2\right) \tag{1}$$

For the d_t sequence, the difference df_t is calculated using (2).

$$df_t = \frac{1}{k+1}\left|\sum_{i=t-k}^{t} d_i - \sum_{i=t}^{t+k} d_i\right| \tag{2}$$

If $df_t > \beta_1$, it indicates that feature vector change abruptly, namely transformation of audio signal occurs, so we segment the audio stream from this place, where β_1 is the predefined fixed segment threshold. In play segments, variation of distances is relatively regular in a long period of time. Because of appearance of music or no applause in time-out segments, audio variation is obviously different from play segments, namely variation of distance don't follow the rule of play segments in a certain period of time.

3. Retrieval of Candidate Similar Clips

In the section we introduce how to retrieve candidate similar video clips corresponding with a query clip. Candidate similar clips are automatically segmented according to visual features. Similar video shots or clips appear periodically or semi-periodically in a game video, thus we define matching function of two clips in order to prevent that excess candidate similar clips are segmented from a long game video. A sliding shot window is defined to decide the locations of candidate similar clips.

3.1. Visual Feature Extraction

Shot segmentation is the first stage to the following higher-level video structure analysis. In our approach, color and texture features are extracted from image frames. The normalized HSV (Hue-Saturation-Value) color histograms of frames are computed as visual features. The color coordinates of HSV color space are uniformly quantized into 8 (Hue), 8 (Saturation) and 8 (Value) bins, respectively, resulting in a total of 256 quantized color bins. The color similarity between two frames is defined as,

$$FFSim_{color}(f_i, f_j) = \sum_{l=1}^{bins} \min(Hf_{il}, Hf_{jl}) \tag{3}$$

Where Hf_{il} and Hf_{jl} denote respectively normalized histogram of the frames f_i and f_j. Texture feature is represented by coarseness、contrast and directionality defined in [10]. Texture dissimilarity is the sum of Euclidean distance of three normalized feature values. The similarity between two frames is defined as,

$$FFSim(f_i, f_j) = FFSim_{color}(f_i, f_j) + (1 - FFDSim_{texture}(f_i, f_j)). \tag{4}$$

Where $FFDSim_{texture}$ denotes texture dissimilarity.

Key frames are next extracted from the shots. The first frame in a shot is first selected as a key frame, and then the similarity between current frame and previous key frames is computed. If the similarity is smaller than the predefined threshold, the current frame is regarded as a key frame. The algorithm terminates until the end frame of a shot. Let the set of extracted key frames from a shot be $KF = \{kf_1, kf_2, \cdots, kf_{N_k}\}$, where N_k is the number of key frames in the set. The similarity between current and previous key frames is defined as:

$$FKSim(f_i, KF) = \max(FFSim(f_i, kf_n)) \quad (kf_n \in KF) \tag{5}$$

The similarity between two shots is defined as:

$$Shsim(S_i, S_j) = \max_{k \in KF_i, l \in KF_j}(FFSim(k, l)) \tag{6}$$

Where KF_i, KF_j are respectively the sets of key frames of the shots S_i and S_j, the frames k and l are arbitrary frames in the sets of key frames.

3.2. Retrieval of Similar Clips

Our objective here is to locate sequences often with shots arranged in different orders. This is quite common in sports video. To do this, we first need to find out the similarity correspondence by comparing each shot of query clip with that of game video. Retrieval of similar clips is based on sliding shot window and

equivalence relations corresponding with shots of the query clips. Shot is the unit of sliding shot window, and the size of window denotes the number of shots in a window as shown by Fig.2.

Because a video clips is composed of continuous shots representing same semantic content, these shots are similar each other. Hence it is usual that several shots of query clip are similar to one shot of sliding window, and vice versa. In fact, the number of one to one correspondence shots can really reflect similar degree. Fig.3 depicted many-many correspondence between query shots and sliding window shots. The number of one-one correspondence shots is actually four, as depicted by Fig.4. Therefore, we define matching function of clips as follow.

Definition 1. Equivalence Relation Corresponding with Query shot.

Let V_q denotes a query clip, and k is the number of its shots. And let V_w denotes the clip in a sliding window, and l is the size of window. For one query shot S_{qi}, equivalence relation corresponding with S_{qi}, is defined as $R_i = \{\{[S_{wj}]_{R_i}\}, \{V_w - [S_{wj}]_{R_i}\}\}$, where $\{[S_{wj}]_{R_i}\} = \{S_{wj} \mid ShSim(S_{qi}, S_{wj}) \geq \beta_2\}$, i=1,2 $\cdots$,k, j=a,a+1,$\cdots$, a+l-1, $ShSim(S_{qi}, S_{wj})$ is similarity of two shots, β_2 is similarity threshold. Obviously, $\{[S_{wj}]_{R_i}\} \bigcap \{V_w - [S_{wj}]_{R_i}\} = \varnothing$ and $\{[S_{wj}]_{R_i}\} \bigcup \{V_w - [S_{wj}]_{R_i}\} = V_w$ hold.

Such equivalence relations are different as usual. Usual equivalence relation is defined on a universe, and thus one object is sure to belong to equivalence class of itself. Here, the definition of equivalence relation involves shots in two video clips (correspond to two universe sets). The objects in the equivalence relation corresponding with one query shot S_{qi} are shots in sliding window, not in query clips. From the angle of similarity, the shots in $\{[S_{wj}]_{R_i}\}$ are equivalent each other. But S_{qi} is not included in its corresponding relation R_i. The number of objects in $\{[S_{wj}]_{R_i}\}$ shows the number of shots of one-many correspondence. And the number of shots with same equivalence relation in V_q indicates many-one correspondence.

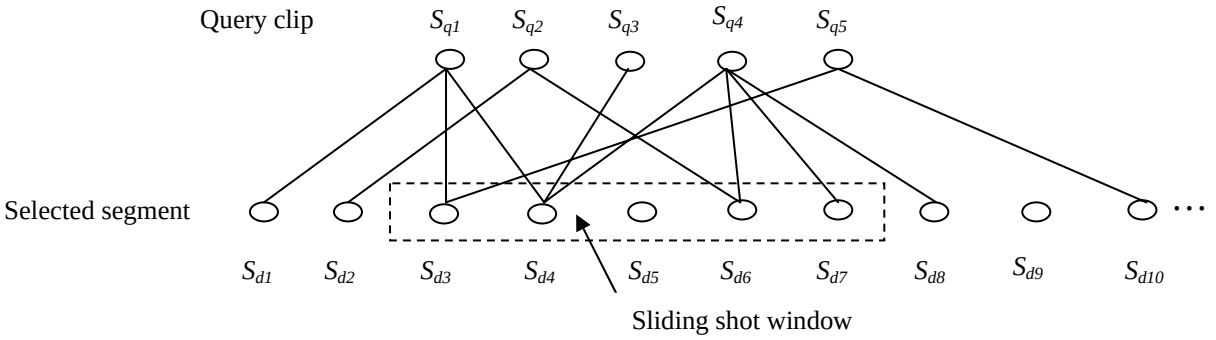

Fig.2. Correspondence between query shots and selected segment shots

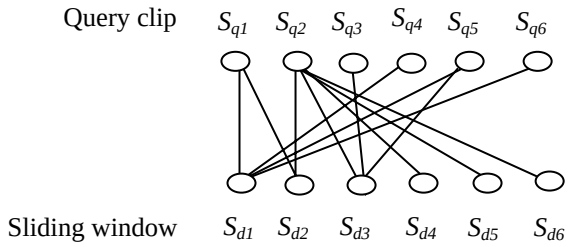

Fig.3 Many-many correspondence

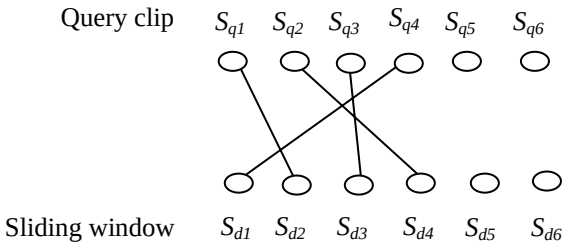

Fig.4 One-one correspondence

Definition 2. Matching Function

Let V_q denotes a query clip, and k is the number of its shots. And let V_w denotes the clip in a sliding window, and l is the size of window. For $\forall S_{qi} \in V_q$, suppose R_i denote equivalence relation corresponding with S_{qi}. And then matching function of V_q and V_w is defined as:

$$r_{qw} = \frac{\sum_{i=1}^{k} \delta_i}{k} \tag{7}$$

Where $\delta_i = \begin{cases} 1 & if\ \{[S_{wj}]_{R_i}\} \subset \{[S_{wj}]_{R_n}\}\ and\ \{[S_{wj}]_{R_i}\} \neq \{[S_{wj}]_{R_n}\} \quad for\ all\ n = 1,2,\cdots i-1,\ i = 1,2,\cdots,k \\ 0 & ortherwise \end{cases}$.

$\sum_{i=1}^{k} \delta_i$ denotes the number of one to one correspondence shots of two clips. For example, if $\sum_{i=1}^{k} \delta_i = 4$ in Fig.2, and then the matching degree r_{qw}=4/6. How to apply matching function to decide candidate similar clips based on sliding shot window? The algorithm is described as below.

Algorithm 1: the method for automatic segment of candidate similar clips based on sliding shot window

Input: shot sequence of the query clip $V_q = \{S_{q1}, S_{q2}, \cdots, S_{qk}\}$, and shot sequence of sports video $V_d = \{S_{d1}, S_{d2}, \cdots, S_{qn}\}$.

Output: shot sequences of m candidate similar video clips $V_{ss} = \{\{S_{11}, S_{12}, \cdots, S_{1s_1}\}, \{S_{21}, S_{22}, \cdots, S_{2s_2}\}, \cdots, \{S_{m1}, S_{m2}, \cdots, S_{ms_m}\}\}$.

(1) Initialize, and input shot sequence of the query clip V_q, and shot sequence of sports video $V_d = \{S_{d1}, S_{d2}, \cdots, S_{qn}\}$, set the size of window as l=(1.1~1.3) k;

(2) If shot sequence of V_d is vacant, stop. Otherwise go to (3);

(3) Mark shots in V_d that have at least one corresponding similar shots in V_q with M_{sj}, and obtain a similar shot sequence $MShot = \{M_{s1}, M_{s2}, \cdots M_{sb}\}$;

(4) Compute matching degree

(4.1) If similar shot sequence *Mshot* is vacant, stop. Otherwise employ a sliding shot window in the position of current $M_{sj} (1 \le j \le b)$;

(4.2) Compute matching degree between query clip V_q and the clip in current sliding shot window, If $r_{qw} \ge \beta_m$, go to (4.3), otherwise $j = j + 1$, go to (4.1);

(4.3) Move the sliding shot window over the sequence of shots in game video at one shot increment. At each window position, we compute matching the degree between two clips;

(5) Decide similar clips

(5.1) Plot the variation curve of matching degree at each window position, and abscissa of the curve is sequence number of shots, as shown in Fig.5.

(5.2) Use the threshold β_2 to remove those local maxima where the number of matching shots is not significant enough to conclude a similar video clip, $j = j + 1$, go to (4.1).

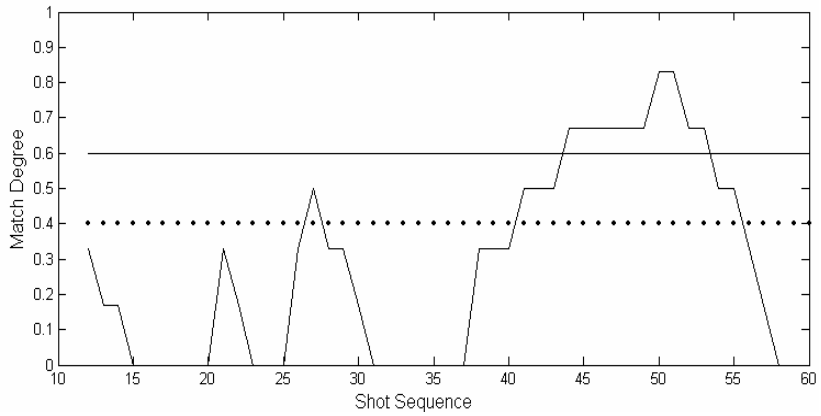

Fig. 5. Variation curve of matching degree r_{qw}

Where β_m is a threshold of matching degree, this means that there is a significant portion of one to one correspondence shots between query clip and the clip in sliding shot window, then we proceed next step. In this way the disturbance of single similar shot is eliminated, here we set $\beta_m = 0.3$. Note that the value of β_m are only relative with computational speed, does not influence retrieval results. By adjusting dynamically the

threshold β_2, we can obtain similar clips that meet different matching degree requirements. As shown in Fig.5, two similar clips are obtained when $\beta_2 = 0.4$; only one similar clip is obtained when $\beta_2 = 0.6$. In this paper, we set $\beta_2 = 0.6$ because similar shots are abundant in sports video.

4. Selection of Similar Clips

For retrieval of sports video, we should extract the meaningful texts appeared in the frames within interesting events and other situations. Fig. 6 displays meaningful texts in basketball games, and Fig.7 shows those in football games. A text segmentation algorithm is designed based on edge detection for extracting and recognizing Chinese captions in sports video programs [2]. Some of candidate similar clips are similar to query clip in visual contents, but semantic information of their representing is completely different from that of query clip. We can discard these fake similar clips by extracting meaningful text information in image frame. For example, in basketball games a foul penalty throw usually is accompanied with texts that show name, jersey number, team badge, and technic statistics of the throwing player.

a. A text example of foul penalty throw

b. A text example before goal throw c. A text example after goal throw

Fig.6. Examples of meaningful texts in basketball games

Fig.7. An example of meaningful texts in football games

5. Similarity Model Between Two Video Clips

After several similar clips are automatically segmented from sports video, we rank them in terms of similarity. Many factors influence similarity of two clips, such as visual factor, temporal order factor and interference factor, etc. Visual similarity denotes similar degree of low-level visual feature of two clips; temporal order similarity represents that one similar clip has higher similarity if its shots are consistent with those of query clip in temporal order; interference factor denotes that unmatched shots influence similarity of two clips. Let $V_q = \{S_{q1}, S_{q2}, \cdots, S_{qk}\}$ and $V_s = \{S_{da}, S_{d(a+1)}, \cdots, S_{dr}\}$ are respectively the shot sequences of query clip and similar clip, For computational convenience, $V_s = \{S_{da}, S_{d(a+1)}, \cdots, S_{dr}\}$ is denoted as $V_s = \{S_{d1}, S_{d2}, \cdots, S_{dm}\}$, where $m = r - a + 1$. Let us define $MaxS_{qs}(i) = MaxS_{qs}(S_i, S_j) = \max_{1 \le j \le s} ShSim(S_i, S_j)$ and $MaxS_{sq}(j) = MaxS_{sq}(S_i, S_j) = \max_{1 \le j \le s} ShSim(S_i, S_j)$. Thus $MaxS_{qs}(S_i, S_j)$ is the maximum similarity of the query shot S_i with any shots of similar clip V_s, and $MaxS_{sq}(S_i, S_j)$ is the maximum similarity of the shot S_j of similar clip V_s with any query shots. Visual factor is defined as,

$$VisionFactor = \frac{1}{2}\left(\overline{MaxS_{qs}} + \overline{MaxS_{sq}}\right). \tag{8}$$

Where $\overline{MaxS_{qs}} = \dfrac{\sum_{i=1}^{k} MaxS_{qs}(i)}{k}$, and $\overline{MaxS_{sq}} = \dfrac{\sum_{j=1}^{s} MaxS_{sq}(j)}{s}$.

In these candidate similar clips, some shots of them can be inconsistent with those of query clip in temporal order. We compute longest common sequence (LCS) of the query clip and one similar clip based on dynamic programming technique in order to rank similar clips. Here we are only interested in finding the length of LCS rather than the explicit construction of LCS sequence. If the *k*th shot in V_s is similar to a shot in V_q, we mark it as "1"; otherwise we mark it as "0" as shown in Equation (9),

$$Sim(k) = \begin{cases} 1 & if\ \ kth\ shot\ is\ similar \\ 0 & if\ \ kth\ shot\ is\ not\ similar \end{cases}. \quad (9)$$

We denote the length of continuous similar accumulative shot sequence as *LC*. It is expressed by using *Sim(k)* as shown in Equation (10),

$$CS(k) = \begin{cases} CS(k-1) + Sim(k) & if\ \ Sim(k) = 1 \\ 0 \quad if\ \ Sim(k) = 0 \end{cases} \quad k = 1,2,\cdots s. \quad (10)$$

The length of the continuously similar shots is expressed as shown in Equation (11),

$$LCS(k) = \begin{cases} CS(k) & if\ \ Sim(k+1) = 0 \\ 0 \quad if\ \ Sim(k+1) = 1 \end{cases} \quad k = 1,2,\cdots,s \quad (11)$$

Then order factor is defined as,

$$OrderFactor = \sum_{k=1}^{s} LCS(k) \quad (12)$$

The similarity between query clip and one similar clip is computed using (13),

$$Similarity(V_q, V_s) = \omega_V VisionFactor + \omega_O OrderFactor \quad (13)$$

Where ω_V and ω_O denote weights of visual factor and order factor. Note that the computation of longest common sequence involves both temporal order and interference factor.

6. Experimental Results

For one experiment, we kinescoped one quarter of one basketball game video that contains 10662 frames. The query clip is a foul penalty throw of one team. After audio segment, 4 play segments and 3 time-out segments were distinguished. Afterwards, 10 candidate similar clips were automatically segmented from play segments based on sliding window and equivalence relation. We discarded 3 candidate clips in terms of text information, and 7 similar clips were selected. Similarity ranking of similar video clips is based on different similarities between query clip and similar clips. Fig. 8 shows 7 similar video clips and their ranking. The first video clip is query clip, and similar video clips are arrayed in column. Video clip of each column represents a similar clip. From left to right, the similarities between similar video clips and query clip are in descending sort.

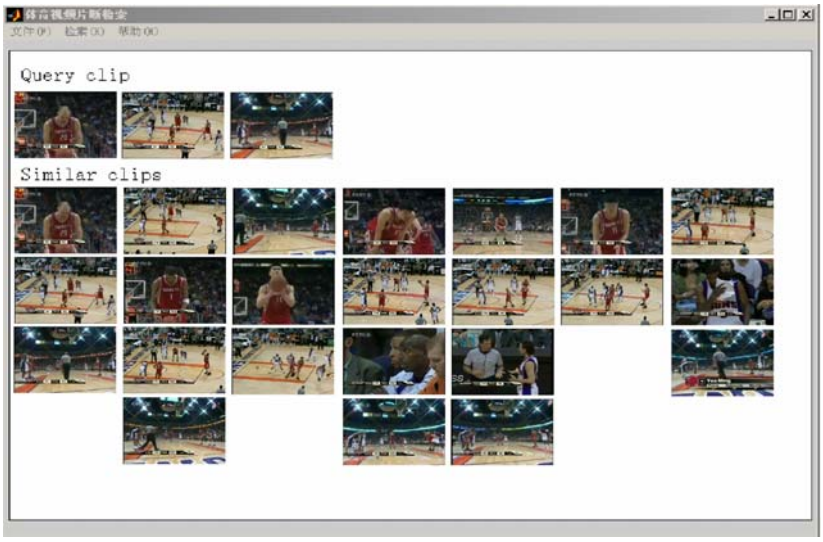

Fig.8. Retrieval results of foul penalty throw

In recent years, there were some researches of sports video clip retrieval, but most of them were focused on a certain sports item. General methods of video clip retrieval of sports were hardly reported. Anil Jain [8] proposed a retrieval method using visual feature (denoted *Algorithm 1*) for sports video. Our retrieval method is based on visual-audio features and text information. For validating performance of our method, we compare our method with *Algorithm 1*. Furthermore, in the *Algorithm 1* many to many relations of shots in two clips are not considered for similarity computation of two clips. Our video data comprise two spots

video, one quarter of one basketball game and one half of one football game. 4 query video clips are respectively a video clip of goal throw of one team, a video clip of foul penalty throw of one team in basketball game, a video clip of corner kick and a video clip of free kick next to goal box in football game, as displayed in Fig. 9-12. The experimental results of *Algorithm 1* are listed in Table 1. The experiments obtained 64.2% precision and 81.6% recall, and this shows that only use visual features is not sufficient to retrieve similar clips. Table 2 shows the retrieval results of our method using the combination of visual and audio and text information. Our method proposed in this paper improves the precision to 81.6%, and the recall to 91.1%. This shows that our method is effective in the sports video clip retrieval. Background of the second shot is in basketball query clip 1, whereas those of 5 other similar clips are auditoria, therefore our method miss 5 similar clips. 4 fault clips were obtained due to the close-up shots of foul players. Because the football video has little meaningful text information in corner kick and free kick, some fault clips can not be discarded using text content. In the same way, similar disadvantages also exist in *Algorithm 1*.

Fig.9. A goal throw of one team

Fig.10. A foul penalty throw of one team

Fig.11. A corner kick

Fig.12. A free kick next to goal box

Table 1. Experimental results for *Algorithm 1* [8]

Query video clips	Number	Correct	Fault	Miss	Precision (%)	Recall (%)
Basketball video clip 1	21	16	9	5	64	76.2
Basketball video clip 2	7	5	3	2	62.5	71.4
Football video clip 1	5	4	2	1	66.7	80
Football video clip 2	7	7	4	0	63.6	100
Average	40	31	17	9	64.2	81.6

Table 2. Experimental results of our proposed method

Query video clips	Number	Correct	Fault	Miss	Precision (%)	Recall (%)
Basketball video clip 1	21	18	4	5	81.8	78.5
Basketball video clip 2	7	7	1	0	87.5	100
Football video clip 1	5	5	2	0	71.4	100
Football video clip 2	7	6	1	1	85.7	85.7
Average	40	36	8	6	81.6	91.1

7. Conclusions

This paper presents a retrieval method of TV sports video clips using audio-visual features and text Information. A long match video first is segmented automatically into shorter segments. Afterwards, we

define equivalence relations corresponding with query shots, and matching function of two clips to locate candidate similar clips. Fake similar clips are discarded in terms of meaningful text information. Visual factor and order factor are considered for similarity ranking of similar video clips. Several TV sports video is applied to validate the performance of the method. The results showed that our method could effectively retrieve similar video clips with a query clip from a TV sports video.

8. Reference

[1] Huang-Chia Shih and Chung-Lin Huang, Content-Based Scalable Sports Video Retrieval System, pp.1553-1556.

[2] Huayong Liu. Content-Based TV Sports Video Retrieval Based on Audio-Visual Features and Text Information, in Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, 2004.

[3] Y. Gong, L. T. Sin, C. H. Chuan, et al, Automatic Parsing of TV Soccer Programs, ICMCS'99, 1995, pp. 167-174.

[4] J. Assfalg, M. Bertini, et al, Highlight Extraction in Soccer Videos, in proceedings of the 12th International conference on image analysis and processing, 2003.

[5] J. Chen, Y. Li, S. Lao, et al, Unif ied BSU-Based Framework for Sports Video Content Analysis, Mini- Micro Systems, 26(2005)2, 272-275.

[6] Y. Peng, Chong-Wah Ngo, Q. Dong, et al, An Approach for Video Retrieval by Video Clip, 14(2003)8, 1409-1417.

[7] Yoshinori Ohno, Jun Miura and Yoshiaki Shirai, Tracking Players and a Ball in Soccer Games, In Proc. of the 1999 IEEE Int'l Conf. on Multisensor Fusion and Integration for Intelligent Systems, August 1999.

[8] Anil Jain, Aditya Vailaya, Wei Xiong, Query by Video Clip, Multimedia Systems, 7(1999), 369-384.

[9] X. Zhao, F. Wu, Y. Zhuang, Audio clip retrieval and relevance feedback based on the audio representation of fuzzy clustering, Journal of Zhejiang University (Engineering Science), 37(2003)3, 264-268.

[10] H. Tumura, Mori s, T. Yamawaki, Texture features corresponding to visual perception, IEEE Transactions on Sys, Man, and Cyb, 8(1978)6, 735-750.