SETIT 2009
5th International Conference: **S**ciences of **E**lectronic, **T**echnologies of **I**nformation and **T**elecommunications
March 22-26, 2009 – TUNISIA

A New Method for Soccer Video Summarizing Based on Shot Detection, Classification and Finite State Machine

Youness TABII* and Rachid OULAD HAJ THAMI*

*ENSIAS, University Mohamed V Laboratory SI3M, BP 713, ENSIAS Soussi, RABAT, Morocco

youness.tabii@gmail.com

oulad@ensias.ma

Abstract: This paper presents a novel method for the automatic extraction of summaries in soccer video based on shot detection, shot classification and FSM. Our method consists of four stages (Figure 1): playfield segmentation, shot detection, shot classification, soccer video word extraction and finding out the appropriate sub-words which present summaries using the Finite State Machine (FSM) and domain knowledge. The segmentation of the playfield is a pre-processing step; we make use of it in the next stages. In shots detection, we use the Discrete Cosine Transform Multi-Resolution (DCT-MR) to extract the different types of shot transition. A statistical method is used to classify shots in three major classes (long shot, medium shot and close-up shot). Then, we extract the word presenting the soccer video using shot detection and classification. Finally, we look for sub-words that can be function as summaries (or highlights) using the Finite State Machine (FSM). Experimental results demonstrate the effectiveness and efficiency of the proposed method.

Key words: Soccer video, segmentation, binarization, FSM, domain knowledge.

INTRODUCTION

The objective of soccer video analysis is: (1) to extract events or objects in the scene; (2) to produce general summaries or summaries for the most important moments in which TV viewers may be interested. The playfield segmentation, events and objects detection play an important role in achieving the above described aims. The analysis of soccer video is very useful for game professionals because it enables them to see which team is better in terms of ball possession or to detect which strategy is useful for each team in a specific moment.

Related works in the literature of sports video analysis have dealt with soccer and various sports games. In [KOU 06], the authors present an algorithm to detect shot changes using the discreet cosine transform (DCT). They calculate the DCT of the luminance matrix by block of 8x8, then the two distances between neighboring pixels (vertical and horizontal distance). The only threshold for the shot changes detection is that the average of vertical and horizontal distances is superior to 1/2.

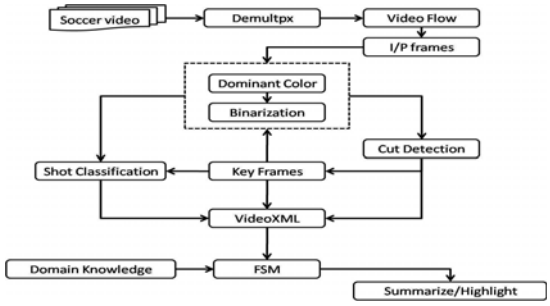


Figure 1: *Stages for soccer video summarizing extraction.*

In [XIE 04], the authors propose a method to segment soccer video into "play" or "break" segments in a HMM (hidden Markov model) framework. The low level features adopted are video dominant color and motion activity. Assflag and al. [ASS 03] use playfield zone classification, camera motion analysis and players' position to infer highlights of non-broadcast soccer video by FSM (finite state machine). Although good results are reported, the FSM-based method requires that researchers make good rules for different kinds of events by hand. Furthermore, the soccer programs from different cameraman and director hold different styles. The appearance of the

same soccer events may vary a lot with the change of camera station. This will depress the extensibility of event detection models. In [QIX 05], the authors present a method for exciting event detection in broadcast soccer video based on mid-level visual description and incremental SVM learning is investigated.

In order to achieve a reliable video description, the primary requirement is to structure the video into elementary shots. This video partitioning step enables us to provide content-based browsing of the video. Secondary, classification of these shots, would facilitate higher level tasks such as video editing or retrieval.

For ease of reference, we have to provide brief definition of the different kinds of shots boundaries. A cut is an abrupt transition between two shots that occurs between two adjacent frames. A fade is a gradual change in brightness, either starting or ending with a black frame. A dissolve is similar to a fade except that it occurs between two shots. The images of the first shot get dimmer and those of the second shot get brighter until the second shot replaces the first one. Other types of shot transitions include wipes and computer generated effects such as morphing.

1. COLOR DOMINANT EXTRACTION

The playfield usually has a distinct tone of green that may vary from stadium to stadium. But in the same stadium, this green color may also change due to the weather and lighting conditions. Therefore, we do not assume any specific value for the dominant color of the field (Figure 2).

Figure 2: *Lighting conditions***.**

We compute the statistics of the dominant field color in the HSV space by taking the mean value of each color component around its respective histogram peaks i_{peaks}. An interval [i_{min} , i_{max}] is defined around each i_{peaks}[KON 03].

$$\sum_{i=i_{\min}}^{i_{peak}} H[i] \le 2H[i_{peak}] \quad and \quad \sum_{i=i_{\min}-1}^{i_{peak}} H[i] > 2H[i_{peak}] \qquad (1)$$

$$\sum_{i=i_{peak}}^{i_{\max}} H[i] \le 2H[i_{peak}] \quad and \quad \sum_{i=i_{peak}}^{i_{\max}+1} H[i] > 2H[i_{peak}] \qquad (2)$$

$$colormean = \frac{\sum_{i=i_{\min}}^{i_{\max}} H[i] * i}{\sum_{i=i_{\min}}^{i_{\max}} H[i]} \qquad (3)$$

H is the histogram for each color component (H,S,V) using the following quantization factor: 64 hue, 64 saturation, 128 intensity. Finally, the mean color is then converted into (R_{mean}, G_{mean},B_{mean}) space so as to determine the playfield surface :

$$G(x,y) = \begin{cases} 1 \quad if & \begin{cases} I_G(x,y) > I_R(x,y) + K(G_{peak} - R_{peak}) \\ I_G(x,y) > I_B(x,y) + K(G_{peak} - B_{peak}) \\ |I_R - R_{peak}| < R_t \\ |I_G - G_{peak}| < G_t \\ |I_B - B_{peak}| < B_t \\ I_G < G_{th} \end{cases} \\ 0 \quad otherwise \end{cases} \qquad (4)$$

G(x,y) is the binarized frame. In our system and after a number of tests, the new thresholds are: R_t = 12, G_t = 18, B_t = 10, K = 0.9 and G_{th} = 85. After the binarization process we use the Coarse Spatial Representation (CSR). The CSR divide the binary frame into sub-block of N x N, where N = 32. Then, we perform control within each block N x N to see if there is a majority of green pixels (the white in the binary frame) (TWP = 0.75, Equation (5)).

$$B(i,j) = \begin{cases} 1 \quad if \quad \sum_{y=N_j}^{N_{j+1}} \sum_{x=N_i}^{N_{i+1}} G(x,y) > TWP * N^2 \\ 0 \quad otherwise \end{cases} \qquad (5)$$

We get a spatial organization of the frame by block of green / non green. This organization provides a convenient way to differentiate between types of frames (Figure 3).

Figure 3: *Binarization and Coarse Spatial Representation***.**

2. SHOT DETECTION

The shot is often used as a basic unit for video analysis and indexing. A shot may be defined as a sequence of frames captured by "a single camera in a single continuous action in time and space". The extraction of this unit (shot) still presents problems for sports video.

In this section, we present the method we used in our framework (Figure 1) for shot detection in soccer video. This method is based on the use of mutli-resolutions of the Discrete Cosine Transform (DCT-MR).

As we know the DCT operates on *A* block of *M* * *N* frame samples or residual values after prediction and creates *B*, which is an *M* * *N* block of coefficients

(Equation 6).

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\left(\frac{\pi(2m+1)p}{2M}\right) \cos\left(\frac{\pi(2n+1)q}{2N}\right)$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}} & p = 0 \\ \sqrt{\frac{2}{M}} & 1 <= p <= M-1 \end{cases} \qquad (6)$$

$$\alpha_q = \begin{cases} \frac{1}{\sqrt{N}} & q = 0 \\ \sqrt{\frac{2}{N}} & 1 <= q <= N-1 \end{cases}$$

Where M and N are respectively the row and the column size of the block **A**.

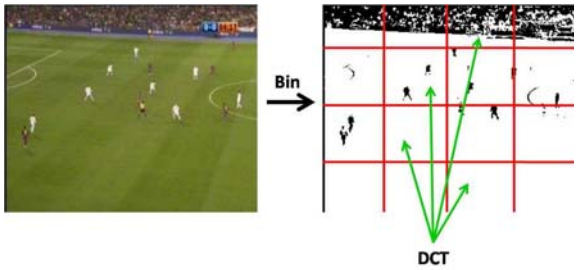


Figure 4: *Discrete Cosine Transform Mutli-Resolution***.**

We define the DCT-MR as follows: we divide the frame into blocks of 2^R*2^R, where R is the Resolution (in this work R=*{*1,2,3,4,5*}*), then we compute the DCT for each block in this frame (Figure 4).

In our case, for each I/P frame, we compute the DCT with different resolutions, from the lowest resolution to the highest (*R* = *{*1,2,3,4,5*}*). We make use one prefixed threshold ***Thv*** for different resolution to detect various transitions between shots. After the calculations of DCT-MR, we compute the vertical and the horizontal distance between adjacent pixels (Figure 5).

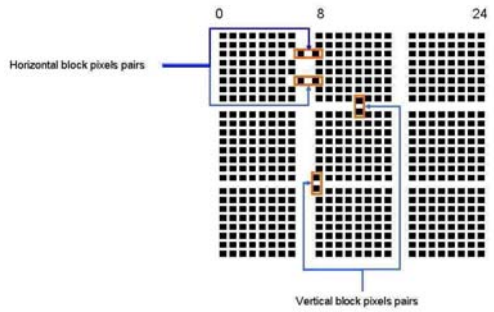


Figure 5: *Horizontal and vertical adjacent pixels***.**

The vertical distance between adjacent pixels of frames can be defined as:

$$distV = \sum_{i=1}^{\frac{w-R}{R}} \sum_{j=1}^{h} \frac{pixel_{Rij} - pixel_{R(i+1)j}}{h(w-R)/R} \qquad (7)$$

Similarly, the horizontal distance is:

$$distH = \sum_{i=1}^{w} \sum_{j=1}^{\frac{h-R}{R}} \frac{pixel_{iRj} - pixel_{iR(j+1)}}{w(h-R)/R} \qquad (8)$$

Thus, the mean distance for all adjacent pixels of video frame with width *w* and height *h* is calculated as:

$$distHV = \frac{distV + distH}{2} \qquad (9)$$

Where **R** is the resolution, ***h*** and ***w*** are the height and the width respectively of video frame. For all frames we compute distH, distV and distHV using equations: 7; 8; 9.

For abrupt transition, our algorithm computed the DCT with the lowest resolution (R=1), which makes our algorithm very fast in detecting this kind of scene transition. For other types of transitions we need to do calculations for the next levels of resolution.

3. KEY FRAMES EXTACTION

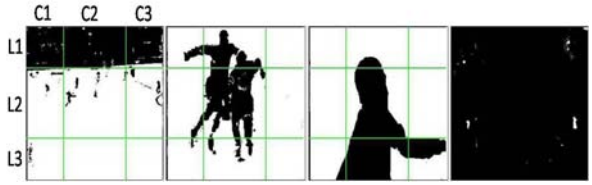
After the shots' detection step, we choose the key frames representing each shot. These key frames enable us to classify the shots according to the four classes (Long shot, Medium shot, Close-up shot, Out-field shot; see section SHOT CLASSIFICATION). We used a simple selection algorithm of key frames. The algorithm is based on the extraction of the following frames: the first and the second at the beginning of the shot, four in the middle and three at the end of the shot. Thus, we have 9 frames which represent each shot. We make use these key frames for shots classification.

4. SHOT CLASSIFICATION

When combined with other features, Shots' classifications usually offer interesting information for the semantics of shots and cues. In our case of soccer video, the shots are classified in four classes:

1. **Long Shot (LS):** A long shot displays the global view of the soccer playfield.
2. **Medium Shot (MS):** A medium shot is where a whole human body is usually visible.
3. **Close-up Shot (CpS):** A close-up shot shows the above waist view of the player.
4. **Out Field Shot (OFS):** An outfield shot shows the viewers, coach . . . etc.

The classification of shots in one of the aforementioned classes is based on spatial features. We use the Golden Section algorithm [EKI 03], which divides the key frames into 3:5:3 proportions in both directions (Figure 6).



(a) **(b)** **(c)** **(d)**

Figure 6: *3:5:3 division frame format.*

Once the key frames are divided into a 3:5:3 format, the following proposal is considered: the cameraman always tries to follow all the actions: goals, players, referee … etc. In order that TV viewers could see the action better, the cameraman puts the action in the middle of the screen. This simple proposal allows us to extract the differences between the four classes considered. These differences are, for instance, the long shot (Figure 6(a)) characterized by a superior black lines (superior blocks are black), the medium shot (Figure 6(b)) in which the two columns (left column and right column) are white and the superior blocks are averagely 50% black and the close-up shot (Figure 6(c)) with a middle column is averagely 70%. The out playfield shot (Figure 6(d)) is almost black. Using the above proposal, we defined a simple algorithm for the classification of shots:

$$Linemean\ (i) = \frac{\sum G(x,y,i)}{\sum G(x,y)} \qquad (10)$$

$$Columnmean\ (j) = \frac{\sum G(x,y,j)}{\sum G(x,y)} \qquad (11)$$

$$where \qquad \{i,j\} = \{1,2,3\}$$

Linemean(i) and *Colomnmean(j)* present the rate of green color (white in binary frame) in each line and each column. To classify all shots in the four defined classes, we put prefix thresholds for each class, so on, for each line and each column.

4.1. video XMLisation

Shot detection, shot classification and key frames extraction are very important steps for soccer video summarizing. To give meaning of these three steps, we present the video in a XML file format for future uses (section VIDEO SUMMARIZING). The structure adopted in our algorithm is:

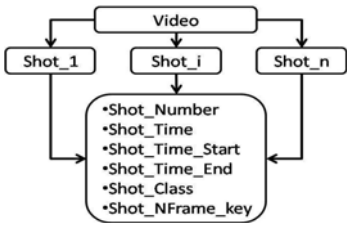


Figure 7: *Adopted XML video format.*

In this structure we make used time as a new factor, we will use this factor in FSM to generate the video summarize.

```
<?xml version="1.0" encoding="utf-8" ?>
- <Video>
    <File_Name>BD/11.mpg</File_Name>
    <Time>5.608000e+001</Time>
    <Shot_Samples>13</Shot_Samples>
  - <Shots>
      <!-- Shots description  -->
    - <shot>
        <Shot_Number>1</Shot_Number>
        <Shot_Time>2.840000e+000</Shot_Time>
        <Shot_Time_Start>0</Shot_Time_Start>
        <Shot_Time_End>2.840000e+000</Shot_Time_End>
        <Shot_Class>Long</Shot_Class>
        <Shot_NFrame_key>9</Shot_NFrame_key>
      </shot>
    - <shot>
        <Shot_Number>2</Shot_Number>
        <Shot_Time>4.800000e+000</Shot_Time>
        <Shot_Time_Start>2.840000e+000</Shot_Time_Start>
        <Shot_Time_End>7.640000e+000</Shot_Time_End>
        <Shot_Class>Medium</Shot_Class>
        <Shot_NFrame_key>9</Shot_NFrame_key>
      </shot>
    - <shot>
```

Figure 8: *XML video file Snippet.*

Figure 8 shows a snippet from XML file generated after shot detection, classification and key frames extraction.

5. VIDEO SUMMARIZING

In this section, we present some observations on soccer video that explore the interesting relations between syntactic structure and the semantic of the video.

5.1. Domain knowledge

We define a complete set of rules which will present the semantic states in a soccer game:

1. **Rule 1:** TV producers tend to stay in long view in order to keep the audience informed of the status of the entire field.
2. **Rule 2:** Close-up shots and outfield shots are merged in the same class.
3. **Rule 3:** In the case of an important action, the TV producers interrupt the long view by zoom-ins or close-ups to follow the players' action.
4. **Rule 4:** After the action, TV producers show the TV viewers replays and slow motions (such as how the action is, its consequences, etc).
5. **Rule 5:** If there is a goal, goto Rule 3. TV producers show also the audience in the stadium within considerable time range to show celebration.
6. **Rule 6:** In case of an important moment, both close-up and medium shots have a significant time which exceeds the case where the match is peaceful.
7. **Rule 7:** Each summary should not exceed 1 min. We took summaries between 30 second and 1 min.

These rules show the typical production style (or **production language**) and editing patterns that help the TV viewers understand and appreciate the game.

5.2. Finite state machine of soccer video

In this section we present the finite state machine which modelizes soccer video. This finite state machine used with domain knowledge to generate

dynamic summarize of video soccer (Figure 9).

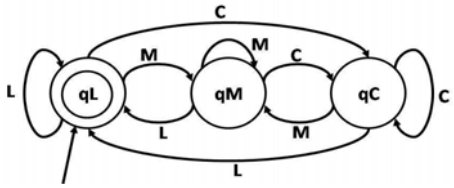


Figure 9: *Adopted Finite State Machine for Soccer video (FSM-SC).*

Formally, a finite state machine is a 5-tuple $M = \{P; Q; \delta; q_0; F\}$, where P is the alphabet, Q is a set of state, δ is a set of transition rules: $\delta \subseteq (Q * \sum \cup \{\varepsilon\} * Q)$, $q_0 \in Q$ is a set of initial (or starting) states and $F \in Q$ is a set of final states. In our case of soccer video, P is the alphabet which presents the summary, $Q = \{qL;qM;qC\}$, where qL is a Long shot, qM is a Medium shot and qC is a Close-up shot, δ is the domain knowledge, $q_0 = \{qL\}$, and $F = \{qL\}$.

After shot detection, classification we make a word that represent the soccer video (a soccer match is a set of long, medium and close-up shots, for example LLMCCCCCMLLLMCMLL...), and our FSM-SC (Finite State Machine for Soccer video) seeking this word to find the sub-words that can be an important moment (highlight) or can be as summary (Figure 10).

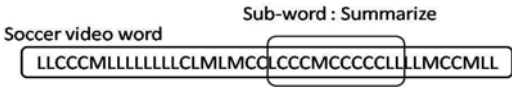


Figure 10: *Soccer video word.*

6. EXPERIMENTAL RESULTS

Five clips from five matches of soccer video used in our experiment. All clips are in MPEG format, 352*x*288 size, 1150*kbps*, 25 frames per second; dominant color ratio is computed on I- and P-frames. Table 1 show a brief description of soccer video clips.

Match name	length	Long shots	Medium shots	Close-up shots	Goals
M_1	40’35’’	115	47	143	1
M_2	20’41’’	62	23	57	3
M_3	30’30’’	78	37	86	3
M_4	22’24’’	67	31	74	3
M_5	32’53’’	90	41	105	1
Total	147’03’’	412	179	465	11

Table 1: *soccer clips description (manual description: 1056 shots).*

Where:

M_1: Match: FC Barcelona vs Real Madrid.

M_2: Match: Cameroon vs Tunisia.

M_3: Match: FC Barcelona vs Real Betis.

M_4: Match: Real Madrid vs FC Mallorca.

M_5: Match: AC Milan vs FC Barcelona.

Resolution	Total	Correct	False alarm	Recall %	Precision %
R=1	1056	428	76	40.53	84.92
R=2	628	237	41	62.97	85.25
R=3	391	189	21	80.87	90.00
R=4	202	158	14	95.83	91.67
R=5	98	98	6	100	94.23

Table 2: *Results of shot detection algorithm.*

Tables 2 shows the result we obtained for detection of shot transition, with the threshold is ***Thv*** = 0*:*12 for different resolutions.

Shot class	Total	Correct	False alarm	Recall %	Precision %
Long	412	401	23	97.33	94.57
Medium	179	160	38	89.38	80.80
Close-up	465	459	18	98.70	96.22
Total	1056	1020	79	96.59	92.81

Table 3: *Results of shot classification algorithm.*

Table 3 presents results of shot classification algorithm. The classification rate for medium shot is low. This is may be due to the prefixed thresholds In other words, the features are less discriminant for these types of shots. However our algorithm works satisfactorily.

Match	Number of summaries	Highlights	Goals	Nothing
M_1	11	3	1	7
M_2	7	2	3	2
M_3	9	2	3	4
M_4	9	3	3	3
M_5	9	3	1	5

Table 4: *Result of summarize extraction using FSM and domain knowledge.*

After shot detection and classification, we generate the XML file, and we use it with domain knowledge as argument of FSM to generate summaries of soccer video. Table 4 shows the obtained results of FSM, where *Highlight* presents the number of important moment detected in clip, *Goal* is the number of goal detected in clip and *Nothing* presents summary of the clip in peaceful moments.

7. Conclusion

In this paper, we presented new algorithms for soccer video summarizing based on shot detection and shot classification. First, we use shot detection with DCT multi-resolution; second, the classification of shots is based on a statistical and spatial method; third, we summarizing the soccer video using finite state machine and the domain knowledge that plays the role of matching features.

The algorithm leaves much more for improvement and extension: there are other relevant low-level features that might provide complementary information and may help improve performance, such as camera motion, edge, higher-level object detectors and audio features.

ACKNOWLEDGMENT

This work has been supported by Maroc-Telecom.

REFERENCES

[ASS 03] J. Assfalg, M. Bertini, C. Colombo, A. D. Bimbo, and W. Nunziati, *Semantic annotation of soccer videos: automatic highlights identification*, Computer Vision and Image Understanding, Special isssue on video retrieval and summarization, 92,Issue 2/3, November-December 2003.

[EKI 03] A. Ekin, A. Tekalp, and R. Mehrotra. *Automatic soccer video analysis and summarizartion*. IEEE, Symp, 2003.

[KON 03] W. Kongwah, Y. Xin, Y. Xinguo, and X. Changsheng. *Real-time goal-mouth detection in mpeg soccer video*. ACM, USA Berkley-California, 2003.

[KOU 06] H. Koumaras, G. Gardikis, G. Xilouris, E. Pallis, and A.Kourtis. *Shot boundary detection without threshold parameters*. Journal of Electronic Imaging, 15(2), April-Jun 2006.

[QIX 05] Y. Qixiang, H. Qingming, G.Wen, and J. Shuqiang. *Exciting event detection in broadcast soccer video with mid-level description and incremental learning*. *MM'05, Singapore. ACM*, p 455-458, November 6 -11, 2005.

[XIE 04] L. Xie, P. Xu, S.-F. Chang, A. Divakaran, and H. Sun. *Structure analysis of soccer video with domain knowledge and hidden markov models*. Pattern Recognition Letters, 2004.