EVENT DETECTION IN TENNIS MATCHES BASED ON VIDEO DATA MINING

Ming-Chun Tien[2], *Yi-Tang Wang*[2], *Chen-Wei Chou*[2],
Kuei-Yi Hsieh[1], *Wei-Ta Chu*[3], *and Ja-Ling Wu*[1,2]

[1]Dept. of CSIE [2]GINM
National Taiwan University

[3]Dept. of CSIE
National Chung Cheng University

ABSTRACT

This paper proposes a mining-based method to achieve event detection for broadcasting tennis videos. Utilizing visual and aural information, we extract some high-level features to describe video segments. The audiovisual features are further transformed to symbolic streams and an efficient mining technique is applied to derive all frequent patterns that characterize tennis events. After mining, we categorize frequent patterns into several kinds of events and therefore achieve event detection for tennis videos by checking the correspondence between mined patterns and events. The experimental results show that the proposed approach is a promising way to detect events in broadcasting tennis video.

Index Terms— Event detection, data mining, tennis videos

1. INTRODUCTION

There have been significant amounts of studies on sports video analysis in recent years. Most of them centered on topics like scene classification, event detection, structure analysis, summarization or highlight extraction for different sports. In this paper, we focus on tennis video analysis and thoroughly investigate automatic event detection. According to tennis regulations, tennis events can be explicitly categorized into the following five types:

1) Fault: A player fails in his/her first serve, and the camera immediately switches out of the court view.
2) Double fault: A player consecutively fails in two serves. In double fault, the camera doesn't switch out of the court view after the first failed serve, and the player successively fails the second serve.
3) Ace or unreturned serve: A player successfully serves, and his/her opponent fails to return the ball. In ace cases, the opponent is not able to touch the ball and therefore fails to return. In unreturned serve, the opponent barely touches the ball but is still unable to successfully return (the returned ball touches net or is out-of-court).
4) Baseline rally: A player successfully serves and the opponent successfully returns. They then stroke around the baseline until one of them fails to return.
5) Net approach: A player successfully serves and the opponent successfully returns. One or both of them once approach the net to stress his/her opponent.

Kijak et al.[1] modeled shot transition patterns to detect specific scenes, such as rally and replay. Kolonias et al.[2] proposed a generic architecture to describe the evolution of tennis matches; however, only rough event detection results were reported. Studies in [3] [4] took advantages of tennis heuristics and player's spatial information to perform event detection, but only visual information was utilized. Some of recent studies presented convincing event models, but no comprehensive result was reported. In this paper, we will automatically extract real-world audiovisual features, and comprehensively detect tennis events on the basis of data mining techniques.

The rest of this paper is organized as follows. Section 2 describes the extraction of audiovisual features. In Section 3, an efficient mining technique is introduced and applied to do event detection. Section 4 shows the experimental results and Section 5 concludes this paper.

2. FEATURE EXTRACTION

In broadcasting tennis videos, the camera always switches to court view when two players combat against each other. We can segment the video into plays according to the view changes of the camera. Some audiovisual features could be extracted to describe the characteristics of each play.

2.1. Play segmentation

Tennis videos are composed of court view shots (plays) and non-court view shots (breaks). We apply a typical shot change detection method based on histogram difference to segment videos into shots. Since a court view shot usually contains a large number of court pixels, the dominant color ratio (DCR)[5] is utilized as the descriptor to extract court view shots. For each court view shot, the techniques of line detection and camera calibration [6] are exploited to locate the court position in video frames. There are two reasons to find the court location:

1) Only using DCR for court view shot detection will result in many false alarms. If we could locate the court position in a shot, we could confirm that a shot is really a court view shot (a play).

2) With the obtained court information, we can investigate more about the real-world situation from visual appearances.

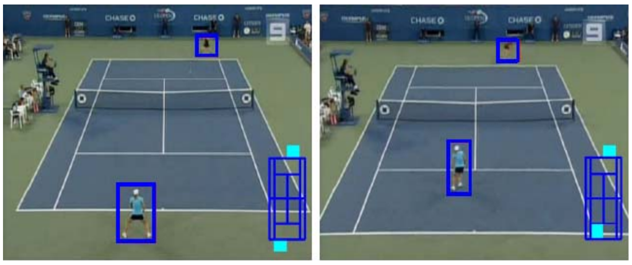

Fig. 1 Sample results of player detection.

2.2. Audiovisual features extraction

By integrating spatial information, temporal information, and audio effects, we can automatically characterize each play on the basis of the following four audiovisual features:

- Moving distance of the player (D_m)

We detect and track the player to find how the players move in a play. The idea of player detection is to find a non-dominant-color region surrounded by dominant-color areas. Fig.1 shows the result of player detection. After mapping the players' positions to the real-world coordinates and calculating their moving distances, the feature D_m is derived from averaging the moving distances of two players.

- Relative position between the player and the court (D_r)

The results of court detection and player detection help us to project the player's position onto a virtual map, which describes where the player is in the court. Fig. 2 shows the virtual map, which has been partitioned into region one and region two. The court area in the top part of a video frame is partitioned symmetrically. If a player ever moves to region one, set D_r to 1, otherwise, set D_r to 0.

- Applause/cheer sound effects (D_a)

Appearance of audio effect implies some special events. For instance, audiences are often kind to give applauses or cheers after good plays, such as aces or baseline rallies. On the other hand, audiences often keep quiet if the player invokes a fault or a double fault. In this work, we extract several audio features including energy, band energy ratio, zero-crossing rate, frequency centroid, bandwidth, and mel-frequency cepstral coefficient (MFCC), and apply an HMM-based (hidden Markov models) method to detect applause/cheer sound effects. If audio effect occurs after a play, set D_a to 1, otherwise, set D_a to 0.

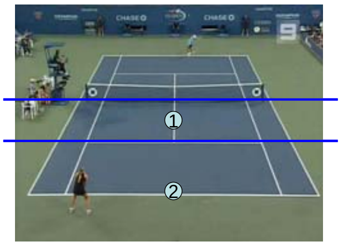

Fig. 2 Players relative positions in the court.

- Length of the play (D_g)

It's apparent that different events have different lengths. For example, the length a double fault is longer than that of a fault. The length of the play presenting a rally is more likely longer than that of an ace. Therefore, we take the length of a play into account in event detection, which is denoted as D_g in this paper.

3. EVENT DETECTION

The five tennis events defined in Section 1 take place frequently in a tennis tournament. We treat the event detection problem as a data mining problem in this paper. Symbols derived from audiovisual features are used to represent each play in the video, and a data mining algorithm is utilized to find frequent patterns from the symbolic streams. We manually categorize the frequent patterns into several events and apply the correspondence between patterns and events to the test videos to automatically detect tennis events.

3.1. Generating symbolic streams

We take each play in the video as a time unit and transform the extracted features of each time unit into symbolic streams according to the mapping given in Table1. As shown in Table1, each feature D_* has its corresponding symbol set; for example, the symbol set of D_a is $\{a,b,c\}$. Fig. 3 shows some examples of symbolic streams. For each time instant (play) i, let S_i be the symbolic stream, representing the features derived from the video, at this particular time instant i. By this way, for a given video, a series of symbolic streams (denoted as $S = S_1, S_2, S_3, ..., S_n$) can be obtained.

3.2. Mining of frequent patterns

We define a **pattern** as $\boldsymbol{p} = p_1\ p_2\ \ldots p_m$, where m is the number of symbols used to represent a symbolic stream S_i (m=4 in this paper), and p_j is a subset of the underlying symbol set with respect to feature D_*. If p_j matches all the symbols in the underlying symbol set, we use the "don't care" character $*$ to denote $p_{j.}$ Let $|p_j|$ be the number of "none don't care" (non-$*$) symbols in the set p_j. The length of a pattern $\boldsymbol{p}$ is defined as $\sum | p_j |$, and a pattern with length k is called a *k-pattern.* Moreover, we define

Features	D_m			D_r		D_a		D_g		
Scale	*Short*	*Medium*	*long*	D_r=0	D_r=1	D_a=0	D_a=1	*Short*	*Medium*	*long*
Symbol	*a*	*b*	*c*	*d*	*e*	*u*	*v*	*x*	*y*	*Z*

Table 1 Data transformation: mapping between features and symbols.

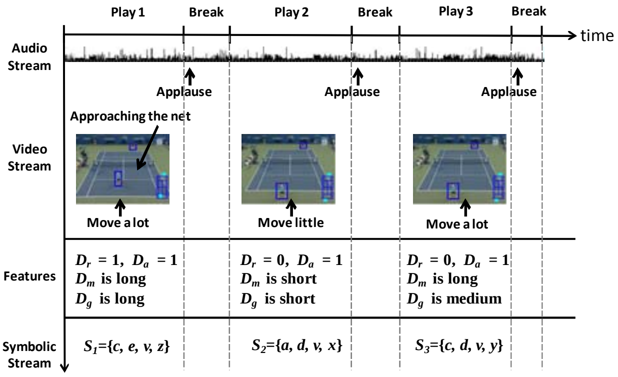

Fig. 3 Examples of symbolic streams.

subpattern of a pattern $\boldsymbol{p} = p_1\ p_2\ \dots p_m$ as a pattern $p' = p_1'\ p_2' \dots p_m'$ such that $p_j' \subseteq p_j$ for every j where $p_j' \neq *$. Due to a strong correlation between frequencies of patterns and their subpatterns, the traditional Apriori-Algorithm may reduce the search space in mining slowly. Consequently, the Max-subpattern Tree introduced in [7] is adopted to efficiently find frequent patterns in *S*.

Follow the definitions given in [7], let F_1 be the set of frequent 1-patterns. A **candidate frequent max-pattern**, C_{max}, is the maximal pattern which can be derived from F_1. For example, if the frequent 1-pattern set is {*a****, b****, *d***, ***v*, ****z*}, C_{max} will be {*a,b*}*d*vz*. The **maximal subpattern** of two patterns p^1 and p^2 is denoted by $MS(p^1, p^2)$ and defined as follows: $MS(p^1, p^2)$ is a common subpattern of both p^1 and p^2, in addition, none of other common subpattern has the length longer than $MS(p^1, p^2)$. For example, if p^1= {*a,b*}*d*vz* and p^2=*adguz*, $MS(p^1, p^2)$ will be *ad**z*. Based on the above-mentioned definitions, our mining algorithm can be presented as follows.

1. Scan *S* once to find the set of frequent 1-patterns (F_1), by accumulating the frequent count for each 1-pattern and selecting among them whose frequent count is no less than the given threshold, *Th1*. Form the candidate frequent max-pattern C_{max} from F_1 and take C_{max} as the root of the Max-subpattern Tree.

2. Scan *S* once. For each symbolic stream S_i, insert $MS(S_i, C_{max})$ into the Max-subpattern Tree with its count=1 if it is not already there; otherwise, increase the count of $MS(S_i, C_{max})$ by one. The detail of the insertion algorithm can be found in [7].

3. Obtain the set of frequent *k*-patterns from the Max-subpattern Tree :

```
for k=2~ length of $C_{max}$
{
    • Derive candidate patterns of length k from frequent patterns of length k-1.
    • Scan the Maxsubpattern Tree to find frequency_count of these candidate patterns and eliminate the non-frequent ones. The frequency_count of each node is calculated by summing the count values of the node itself and its ancestor in the Max-subpattern Tree. If the derived frequent k-pattern set is empty, return.
}
```

We go through each frequent pattern derived from the mining algorithm and manually map all frequent patterns to corresponding events. Frequent patterns mapped to the same event are merged into a set. Finally, we could categorize all frequent patterns into several sets and each set represents a specific event. According to the relationship between patterns and events, we can achieve event detection for the test videos.

4. EXPERIMENTAL RESULTS

Three broadcasting videos of tennis tournament are used to evaluate the proposed methods. The evaluation data are captured from different broadcasting channels with significant variation in broadcasting styles and audio conditions. On the basis of these data, we evaluate both the performance of feature extraction and event detection.

4.1 Performance of Feature Extraction

From Section 3.1, the features D_m and D_r are directly related to the effectiveness of player detection. We randomly select five plays from each type of events and manually judge whether the player's position is correctly detected or not. The accuracy of player detection is about 0.98 in this experiment, which implies the accuracy of D_m and D_r is 0.98. For the aural feature D_a, we compare the number of detected ones and the ground truth. The overall precision and recall rates for this part are 0.98 and 0.89, respectively. The performance of extracting the length of play, D_g, is directly related to the correctness of court view detection. If court view can be correctly detected, calculating D_g is trivial. With the prescribed dominant color-based descriptor and the court detection filtering, very promising court view detection performance, say 0.95, for both precision and recall rates, is achieved.

As the evaluation results reported above, one can see that the proposed methods effectively extract high-level features. Although errors in feature extraction would lead to event detection errors, this promising feature extraction performance makes automatic tennis video analysis realistic. The performance of event detection reported in the next section is based on the aforementioned automatic feature extraction and event detection processes.

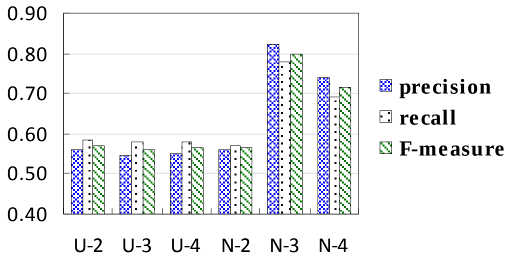

Fig. 4 Performance of the mining-based event detection method in different quantization levels.

4.2 Performance of Event Detection

In the evaluation, we put the whole broadcasting tennis match video, which may consist of commercials and other unrelated segments, to the developed system without any manual preprocess. This system detects all possible events in plays and report comprehensive evaluation results. Transforming features to symbols will result in information loss since we quantize D_m and D_g to several levels. To evaluate the influence of quantization on event detection, we detect events based on different quantization settings. Fig. 4 shows the performance of the mining-based event detection. "U-*i*" and "N-*i*" represent that the event detection is based on uniformly and nonuniformly quantizing features, into *i* levels, respectively. As shown in Fig. 4, the nonuniformly three-level quantization brings the best performance in terms of precision, recall, and F-measure. Table 2 illustrates the performance of detecting different events based on nonuniformly three-level quantization.

Basically, the mining-based method considers the frequency of occurrence and finds hidden patterns, which may represent the characteristics of events. Only the patterns that frequently occur would be found in the mining algorithm. For specific plays that rarely happen, the mining-based method would miss in detection. Since the training data contain a lot of baseline rally events, the performance for detecting baseline rallies is the best. The approach could work better if more training data were provided. We have especially low recall rate in ace/unreturned serve detection. The main reason comes from the miss of applause/cheer detection. Applause or cheer sounds may be interfered by the anchorperson's speech, and the sound effects are not always spirited for ordinary plays. Moreover, there are relatively fewer aces or unreturned serves in tennis matches, and therefore, the detection performance is worse than that of other events.

	Number of plays	Precision	Recall
Fault/Double fault	103	0.91	0.82
Ace/Unreturned serve	62	0.65	0.53
Baseline rally	184	0.85	0.83
Net approach	39	0.72	0.85
Total performance	388	0.82	0.78

Table 2 The performance of detecting different events based on nonuniformly three-level quantization.

5. CONCLUSION

We have presented an approach that utilizes visual and aural cues to perform event detection in tennis videos from the perspectives of video mining. To characterize events, we extract high-level features from audio and video streams. Based on the extracted features, the event detection problem is converted to a data mining problem. A Max-subpattern Tree is utilized during the mining process to achieve the effectiveness of mining frequent patterns. In the evaluation, we show that the proposed methods effectively extract high-level features, which provides robust foundation for event detection. The evaluation also demonstrates the superiority of the mining-based approach. In the future, we will further investigate about the potentials of using mining-based event detection method on other kinds of sports videos. More elaborate audiovisual features would be designed and extracted to enhance the event detection performance.

6. REFERENCES

[1] E. Kijak, G. Gravier, L. Oisel and P. Gros, "Audiovisual integration for tennis broadcast structuring," *Multimedia Tools and Applications*, vol. 30, pp. 289-311, 2006.

[2] I. Kolonias, W. Christmas and J. Kittler, "Automatic evolution tracking for tennis matches using an HMM-based architecture," *in Proceedings of IEEE Workshop on Machine Learning for Signal Processing*, pp. 615-624, 2004.

[3] N. Rea, R. Dahyot and A. Kokaram, "Classification and representation of semantic content in broadcast tennis videos," *in Proceedings of IEEE International Conference on Image Processing*, vol. 3, pp. 1204-1207, 2005.

[4] J. Han, D. Farin and P.H.N.d. With, "Multi-level analysis of sports video sequences," *in Proceedings of SPIE Conference on Multimedia Content Analysis, Management, and Retrieval,* 2006.

[5] A. Ekin and A.M. Tekalp, "Robust dominant color region detection and color-based applications for sports video," *in Proceedings of IEEE International conference on Image Processing*, vol. 1, pp. 21-24, 2003

[6] D. Farin, S. Keabbe, P.H.N.d.With and W. Effelsberg, "Robust camera calibration for sport videos using court models," *in SPIE Storage and Retrieval Methods and Application for Multimedia*, vol. 5307, pp. 80–91, 2004.

[7] J. Han, G. Dong and Y. Yin, "Efficient mining of partial periodic patterns in time series database," *in Proceedings of the 15th International Conference on Data Engineering*, pp. 106-115, 1999.