

國立交通大學

資訊科學與工程研究所

博士論文

籃球影片之語義標注與摘要擷取之研究

A STUDY ON SEMANTIC ANNOTATION AND
SUMMARIZATION OF BASKETBALL VIDEO

研究生：陳俊旻

指導教授：陳玲慧博士

中華民國一百零三年七月

籃球影片之語義標注與摘要擷取之研究

A STUDY ON SEMANTIC ANNOTATION AND
SUMMARIZATION OF BASKETBALL VIDEO

研究生：陳俊旻

Student: Chun-Min Chen

指導教授：陳玲慧博士

Advisor: Dr. Ling-Hwei Chen



國立交通大學資訊學院
資訊科學與工程研究所
博士論文

A Dissertation Submitted to
Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy
in
Computer Science

July 2014

Hsinchu, Taiwan, Republic of China

中華民國一百零三年七月

籃球影片之語義標注與摘要擷取之研究

研究生：陳俊旻

指導教授：陳玲慧博士

國立交通大學資訊學院
資訊科學與工程研究所

摘要

運動影片在我們的休閒娛樂中，扮演了重要角色，然因運動影片的資訊量很大，除了需要的頻寬與傳輸時間多，觀眾亦需耗費大量的時間觀賞，為了節省不必要的時間成本與能源成本，影片精華檢索、影片摘要、以及影片慢動作重播偵測已成為一個熱門的研究題目。目前大多數方法，皆對影片中的每一張畫面分析，然而語義事件只發生在有計分板的畫面，慢動作重播則只出現在沒有計分板的畫面，從不相關的畫面中擷取語義事件或慢動作重播，反而降低方法的準確度與執行效率，且現存的方法多針對足球影片而設計，對籃球影片之探討相對較少，為了解決現存方法所遇到的各式挑戰，本論文將以籃球影片為例，提出一個新穎的運動影片分析架構，讓一般民眾得以有效率的查詢賽事精華，也讓專業人士能夠用來延伸到其他相關應用(自動影片精華產生、運動員動作分析、球隊戰術分析等)。在此架構中，首先提供一個影片畫面分割方法，將運動影片分成有/無計分板兩類。接著，對有計分板的畫面提出一語義事件偵測方法，對無計分板畫面提出一慢動作重播偵測方法。

關於語義事件偵測的相關研究，現存的方法，多使用影片本身的影像或聲音作為特徵，然而僅使用影片內容作為特徵，往往會發生一些語義鴻溝，也就是較低階的影片特徵，和較高階的語義事件，兩者之間的差距。雖然近來有些方法，參考網路轉播文字作為外部知識以彌補語義鴻溝，但從網路轉播文字中擷取語義事件，並標注在運動影片上，仍然存在許多困難與挑戰。在此論文中，我們將討論相關的困境，並提出兩個方法來解決。

關於慢動作重播偵測的研究，現存方法大致可以分為兩類。慢動作重播前後，常常有製播單位後製加上的特效畫面，第一類方法都是基於這些特效的位置，來偵測慢動作重播，但籃球影片較為複雜，此假設在籃球影片未必恆成立。第二類方法是分析慢動作片段的特徵，利用這些特徵將慢動作重播片段和一般片段作區分，但由於某些用於足球的特徵並不適用於籃球，此類方法在籃球應用上仍有改進空間。籃球是世界上最重要的運動之一，但在偵測籃球影片慢動作重播上，仍有許多挑戰尚待解決。本論文將提出一個新的方法，偵測籃球影片中的慢動作重播，提供一個重要的運動影片分析素材。

實驗結果顯示，本論文所提出的架構與方法，可行性與有效性皆可得到良好的驗證，基於提出的架構與方法皆沒有使用籃球限定的特徵，我們期望本論文可以被延伸應用於其他類型的運動影片。

A STUDY ON SEMANTIC ANNOTATION AND SUMMARIZATION OF BASKETBALL VIDEO

Student: Chun-Min Chen

Advisor: Dr. Ling-Hwei Chen

Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University

ABSTRACT

Semantic event and slow motion replay extraction for sports videos have become hot research topics. Most researches analyze every video frame; however, semantic events only appear in frames with scoreboard, whereas replays only appear in frames without scoreboard. Extracting events and replays from unrelated frames causes defects and leads to degradation of performance. In this dissertation, a novel framework will be proposed to tackle challenges of sports video analysis. In the framework, a scoreboard detector is first provided to divide video frames to two classes, with/without scoreboard. Then, a semantic event extractor is presented to extract semantic events from frames with scoreboard and a slow motion replay extractor is proposed to extract replays from frames without scoreboard.

As to semantic event extraction, most of existing researches focus on analyzing audio-visual features of video content as resource knowledge. However, schemes relying on video content encounter a challenge called semantic gap, which represents

the distance between lower level video features and higher level semantic events. Although the multimodal fusion scheme that conducts webcast text as external knowledge to bridge the semantic gap has been proposed recently, extracting semantic events from sports webcast text and annotating semantic events in sports videos are still challenging tasks. In this dissertation, we will address the challenges in the multimodal fusion scheme. Then, we will propose two methods to overcome the challenges.

As to slow motion replay detection, many methods have been proposed, and they are classified into two categories. One assumes that a replay is sandwiched by a pair of visually similar special digital video effects, but the assumption is not always true in basketball videos. The other analyzes replay features to distinguish replay segments from non-replay segments. The results are not satisfactory since some features (e.g. dominant color of sports field) are not applicable for basketball. Most replay detectors focus on soccer videos. In this dissertation, we will propose a novel idea to detect slow motion replays in basketball videos.

The feasibility and effectiveness of all the above proposed methods have been demonstrated in experiments. It is expected that the proposed sports video analysis framework can be extended to other sports.

誌 謝

首先，我想對我的指導老師陳玲慧教授獻上我最誠摯的感謝，在她亦師亦母指導之下，無論是學術研究的方法、解決問題的能力、待人處事的態度等，都令我獲益匪淺；更感謝她的支持，讓我得以兼顧對籃球的熱愛以及學術研究的熱誠，我很幸運能夠遇到這麼好的老師。

接著我要感謝自動化資訊處理實驗室的諸多伙伴，因為各位學長學弟們的陪伴，讓我的研究生涯充實有趣而不孤單。感謝井民全學長和郭萱聖學長，在我剛進入實驗室的階段，能有兩位非常好的楷模與榜樣。感謝李惠龍學長、楊文超學長、歐占和學弟，從資格考一起共同奮戰，到整個學術生涯過程中的提點與幫忙。感謝林懷三學弟口試期間的熱心協助。

再來，我要感謝交通大學男子籃球隊伙伴們的支持，讓我在研究疲累之餘，能有一個避風港，和各位學長學弟們一同在球場上奔馳，是我的榮幸，也是我這輩子最珍貴的回憶。感謝我最好的朋友們：偉益、金煌、志瑋、秉澄、信華，在我脆弱的時候陪伴我度過難關。感謝在我生命中，每一位曾經給予我幫助與鼓勵的朋友，謝謝你們，讓我成為一個更好的人。

最後，也是最重要地，我要感謝一直以來無條件支持我的家人：父親國淇、母親淑真、哥哥俊宇、姊姊韻如，永遠作我堅強的後盾，讓我毫無後顧之憂地追求目標、挑戰人生、享受生活，謹以最感恩的心，將此篇論文獻給我最親愛的家人。

TABLE OF CONTENTS

CHINESE ABSTRACT.....	i
ENGLISH ABSTRACT.....	iii
ACKNOWLEDGMENT (IN CHINESE).....	v
TABLE OF CONTENTS.....	vi
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
CHAPTER 1.....	1
INTRODUCTION.....	1
1.1 Motivation.....	1
1.2 Related Work.....	2
1.2.1 Semantic Event Extraction Challenges.....	2
1.2.2 Slow Motion Replay Detection Challenges.....	5
1.3 Synopsis of the Dissertation.....	6
CHAPTER 2.....	8
A NOVEL FRAMEWORK FOR SPORTS VIDEO ANALYSIS.....	8
2.1 Video Frames Partition.....	8
2.1.1 Context-Based Static Region Detection.....	11
2.1.2 Scoreboard Selection.....	13
2.1.3 Experimental Results.....	14
2.2 Overview of the Framework.....	15
2.3 Summary.....	16
CHAPTER 3.....	17
A NOVEL APPROACH FOR SEMANTIC EVENT EXTRACTION FROM SPORTS WEBCAST TEXT.....	17
3.1 Introduction.....	17
3.2 Proposed Method.....	20
3.2.1 Unrelated Words Filtering.....	22
3.2.1.1 Stop Words.....	23
3.2.1.2 The Proposed Interactive System for Establishing Sports Stop Word List and Event Keyword List.....	24
3.2.1.3 The Proposed Unrelated Words Filtering Procedure.....	26
3.2.2 Event Clustering.....	27
3.2.3 Hierarchical Search System.....	30

3.3 Experimental Results	34
3.4 Summary	42
CHAPTER 4	43
ANNOTATING WEBCAST TEXT IN BASKETBALL VIDEOS BY GAME CLOCK RECOGNITION AND TEXT/VIDEO ALIGNMENT.....	43
4.1 Introduction.....	43
4.2 Proposed Method	47
4.2.1 Video Frames Partition.....	47
4.2.2 Semantic Event Extraction from Scoreboard Frames.....	48
4.2.2.1 Clock Digit Locator.....	49
4.2.2.2 Clock Digit Template Collection.....	50
4.2.2.3 Clock Digit Recognition.....	51
4.2.2.4 Text/Video Alignment	53
4.3 Experimental Results	54
4.4 Summary	56
CHAPTER 5	58
A NOVEL METHOD FOR SLOW MOTION REPLAY DETECTION IN BROADCAST BASKETBALL VIDEO	58
5.1 Introduction.....	58
5.2 Proposed Method	61
5.2.1 Video Frames Partition.....	61
5.2.2 Feature Extraction and Replay Detection	62
5.3 Experimental Results	80
5.4 Summary	87
CHAPTER 6	88
CONCLUSIONS AND FUTURE WORKS.....	88
REFERENCES	90
PUBLICATION LIST.....	92
VITA	93

LIST OF TABLES

Table 3.1 Average number of sports event categories in 25 basketball training data and 20 soccer training data.....	35
Table 3.2 Mappings of basketball event categories from pLSA to the proposed method.....	37
Table 3.3 Mappings of soccer event categories from pLSA to the proposed method.....	39
Table 3.4 Occurrences of exception basketball events from 41 testing games.	41
Table 3.5 Occurrences of exception soccer events from 48 testing games.	41
Table 4.1 Semantic events extraction results of the proposed method.....	55
Table 5.1 Replay detection results for MNS.....	81
Table 5.2 Replay detection results for MNS with self pruning.	82
Table 5.3 Replay detection results for MNS by methods in the first category.....	82
Table 5.4 Total replay detection results with fixed $TH_{slv} = 30$	84
Table 5.5 Total replay detection results with fixed $TH_{smoothness} = 85\%$	84
Table 5.6 Total replay detection results with $TH_{smoothness}=0.85$ and $TH_{slv} = 25$	85
Table 5.7 Total replay detection results with $TH_{smoothness}=0.85$ and $TH_{slv} = 30$	86
Table 5.8 Total replay detection results by methods in the first category.....	86

LIST OF FIGURES

Fig. 2.1 Examples of scoreboard frames and non-scoreboard frames.....	10
Fig. 2.2 Block diagram of scoreboard template extraction.	10
Fig. 2.3 Example of pixel-based frame difference accumulation.....	12
Fig. 2.4 Scoreboard template extraction for 3 different broadcasters with extracted positions marked by white rectangle.	15
Fig. 2.5 The proposed framework.....	16
Fig. 3.1 An example of basketball webcast text.	21
Fig. 3.2 Block diagram of the proposed method.	21
Fig. 3.3 An example to illustrate description and word.....	23
Fig. 3.4 The block diagram of the interactive pre-training system.	26
Fig. 3.5 Block diagram of unrelated words filtering procedure.	27
Fig. 3.6 An example to illustrate the concept of the proposed hierarchical search system.	30
Fig. 3.7 An example to illustrate the data structure for hierarchical search.	32
Fig. 4.1 Two examples of overlaid scoreboard with game clock in basketball video.	45
Fig. 4.2 General definitions of game clock patterns.....	49
Fig. 4.3 An example of locating game clock digits (10:30).	50
Fig. 4.4 An example of text/video alignment.	54
Fig. 4.5 Examples of basketball games playing without game clock.....	56
Fig. 5.1 Examples of game-related segments.....	63
Fig. 5.2 Block diagram of slow motion replay detection.	64
Fig. 5.3 An example of comparison between a game-related segment and a replay segment.	65
Fig. 5.4 The two global features of each MNS in a basketball video.	68
Fig. 5.5 An example of the DH_1 sequence of a game-related segment misclassified as replay.	69
Fig. 5.6 Histogram of σ'_{DF_1} from the preliminary replays in ten experimented basketball videos.....	69
Fig. 5.8 Examples of still shots of the product and slogan in TV commercials.....	71
Fig. 5.9 Examples of abrupt transition detection results and the corresponding cut scenes of non-replay and replay.....	77

CHAPTER 1

INTRODUCTION

1.1 Motivation

Thanks to the rapid growth of computer science and network technology, people now are capable of using mobile devices, e.g. notebook, tablet, smart phone, to acquire sports videos anytime and anywhere. Since substantial number of sports videos are produced and broadcasted every day, it is nearly impossible to watch them all. Most of the time, people prefer to watch highlights of sports videos or retrieve only partial video segments that they are interested in. Many websites, such as ESPN, NBA, and Yahoo Sports, already make this kind of online service available. These online services are made by professional film editors and sports reporters by exhaustedly watching sports videos personally, so people or fans can see the unified version. However, these services may not please all fans. For example, fans, who want to practice certain sports skills or imitate specific sports stars cannot take advantage of the unified version highlight, and have to download the whole game and search for certain moves made by certain players. It is quite inconvenient. Therefore, sports video analysis, such as semantic event extraction [1]-[9] and slow motion replay detection [10]-[18], has become a valuable and hot research topic.

1.2 Related Work

Many research efforts have been spent on sports video analysis. However, some challenges still remain to be solved and will be presented in the following.

1.2.1 Semantic Event Extraction Challenges

Some semantic event extraction researches [1]-[3] use video content as resource knowledge. Chen and Deng [1] analyzed video features (e.g. color, motion, shot) to extract and index events in a basketball video. Hassan et al. [2] extracted audio-visual (AV) features and applied Conditional Random Fields (CRFs) based probabilistic graphical model for sports event detection. Kim and Lee [3] built an indexing and retrieving system for a golf video by analyzing its AV content. However, schemes relying on video content encounter a challenge called semantic gap, which represents the distance between video features and semantic events. Recently, some researches [4]-[9] use a multimodal fusion of video content and external resource knowledge to bridge the semantic gap. Webcast text, one of the most powerful external resource knowledge, is an online commentary posted with well-defined structure by professional announcers. It focuses on sports games and contains detail information (e.g., event description, game clock, player involved, etc.). The multimodal fusion scheme, which analyzes webcast text and video content separately and then does

text/video alignment to complete sports video annotation or summarization, has been used in American football [4], soccer [6]-[8], and basketball [7]-[8].

For webcast text analysis, Xu et al. [8] apply probabilistic latent semantic analysis (pLSA), a linear algebra-probability combined model, to analyze the webcast text for text event clustering and detection. Based on their observation, the descriptions of the same event in the webcast text have a similar sentence structure and word usage. They use pLSA to first cluster the descriptions into several categories and then extract keywords from each category for event detection. Although they extend pLSA for both basketball and soccer, there are two problems in the approach: 1) the optimal number of event categories is determined by minimizing the ratio of within-class similarity and between-class similarity. In fact, there are more event categories for a basketball or soccer game. For example, in a basketball game, many events, such as timeout, assist, turnover, ejected, are mis-clustered into wrong categories or discarded as noises. This may cause side effects degrading and limiting the results of sports video retrieval; 2) after keywords extraction, events can be detected by keywords matching. In Xu et al.'s method, they use the top ranked word in pLSA model as single-keyword of each event category. But in some event categories, the single-keyword match will lead to horrible results. For example, in their method for a basketball game, "jumper" event represents those jumpers that

players make. Without detecting “makes” as a previous word of “jumper” in description sentences, the precision of “jumper” event detection is decreased from 89.3% to 51.7% in their testing dataset. However, the “jumper” event actually is an event that consists of “makes jumper” event and “misses jumper” event. The former can be used in highlights, and the latter can be used in sports behavior analysis and injury prevention. Accordingly, using single-keyword match is insufficient and some important events will be discarded.

In the multimodal fusion scheme, text/video alignment has a great impact on performance, and it can be achieved through scoreboard recognition. A scoreboard is usually overlaid on sports videos to present the audience some game related information (e.g., score, game status, game clock) that can be recognized and aligned with text results. For sports with game clock (e.g., basketball and soccer), event moment detection can be performed through video game clock recognition Xu et al. [6]-[8] used Temporal Neighboring Pattern Similarity (TNPS) measure to locate game clock and recognize each digit of the clock. A detection-verification-redetection mechanism is proposed to solve the problem of temporal disappearing clock region in basketball videos. However, recognizing game clock in a frame which has no game clock is definitely unnecessary. The cost of verification and redetection could have been avoided. Moreover, the clock digit characters cannot be located on a

semi-transparent scoreboard.

1.2.2 Slow Motion Replay Detection Challenges

As to slow motion replay detection, many methods have been proposed, and they can be classified into two categories. The first category [10]-[15] is to locate positions of specific production actions called special digital video effects (SDVEs) or logo transitions, and bases on these positions to detect replay segments. However, in this category, they all made an imperfect assumption that a replay is sandwiched by either two visually similar SDVEs or logo transitions, the assumption is not always true in basketball videos. In fact, a basketball video segment bounded by paired SDVEs is not always a replay. Moreover, the beginning and end of a basketball replay can have some combinations: 1) paired visually similar SDVEs; 2) non-paired SDVEs; 3) a SDVE in one end and an abrupt transition in the other. So, previous work in this category cannot be applied to basketball videos with replays having combinations (2) and (3).

The second category [16]-[18] analyzes features of replays to distinguish replay segments from non-replay segments. Farn et al. [16] extracted slow motion replays by referring to the dominate color of soccer field; however, it is not applicable in basketball videos since the size of basketball court is relatively smaller and its

textures are more complicated. Wang et al. [17] conducted motion-related features and presented a support vector machine (SVM) to classify slow motion replays and normal shots. The precision rates of two experimented basketball videos are 55.6% and 53.3% with recall rates 62.5% and 66.7%, respectively. Han et al. [18] proposed a general framework based on Bayesian network to make full use of multiple clues, including shot structure, gradual transition pattern, slow motion, and sports scene. The method is suffered from the inaccuracy of the used automatic gradual transition detector. Their experiments show precision rate 82.9% and recall rate 83.2%.

The existing two category methods are generic but not satisfactory for basketball videos. Moreover, most previous researches analyze every video frame to detect replays, but detecting replays in video frames that are surely non-replay degrades both performance and detection rate.

1.3 Synopsis of the Dissertation

Semantic event and slow motion replay extraction for sports videos have become hot research topics. Most researches analyze every video frame; however, semantic events only appear in frames with scoreboard, whereas replays only appear in frames without scoreboard. Extracting events and replays from unrelated frames causes defects and leads to degradation of performance. To tackle the above-mentioned

challenges, a novel framework combining semantic event extraction and slow motion replay detection is proposed in this dissertation. In the framework, a scoreboard detector is first provided to divide video frames to two classes, with/without scoreboard. Then, a semantic event extractor is presented to extract semantic events from frames with scoreboard and a slow motion replay extractor is proposed to extract replays from frames without scoreboard.

The rest of the dissertation is organized as follows. Chapter 2 presents an overview of the proposed framework for sports video analysis. Under the framework, some sports video analysis schemes are proposed and discussed in Chapter 3 to Chapter 5. Chapter 3 describes an unsupervised approach to extract semantic events from sports webcast text. The text/video alignment and event annotation method is proposed in Chapter 4. Chapter 5 provides a slow motion replay detection method for broadcast basketball video. Some conclusions and future research directions are given in Chapter 6.

CHAPTER 2

A NOVEL FRAMEWORK FOR SPORTS VIDEO ANALYSIS

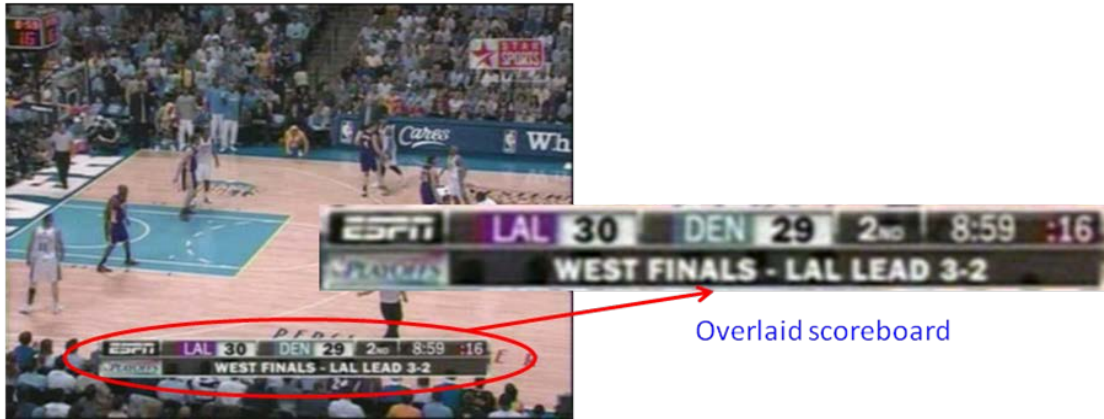
In this chapter, we will propose a novel framework to analyze sports videos. One of the main novelties is to refer to scoreboard information. It is observed that sports video frames can be partitioned into two categories according to the existence of scoreboard. Frames with scoreboard existence are called scoreboard frames, and others are called non-scoreboard frames. In general, semantic events appear during playing of a sports game, which consists of scoreboard frames only. Slow motion replays appear during temporal pausing of a sports game, which consists of non-scoreboard frames only. The phenomenon is dominant and used to skip large amount of unnecessary processing frames before semantic resource extraction. Accordingly, the performance and the detection rate can be assured. The chapter is organized as follows. In Section 2.1, a video frame partition method to divide frames into scoreboard frames and non-scoreboard frames is introduced. An overview of the proposed framework will be presented in Section 2.2. Note that extracting semantic events from scoreboard frames and extracting slow motion replays from non-scoreboard frames will be provided in the latter chapters.

2.1 Video Frames Partition

As can be seen from Fig. 2.1, in basketball videos, all frames can be broadly

classified into two categories, scoreboard frames and non-scoreboard frames. Scoreboard frames present basketball game with scoreboard overlaid on them, while non-scoreboard frames present the rest, e.g., sideline interview, slow motion replay, etc. Since semantic events only appear in scoreboard frames, whereas replays only appear in non-scoreboard frames. It is beneficial to filter out unnecessary processing frames in each semantic resource extraction step. So, an automatic scoreboard template extractor is first proposed to extract scoreboard template and scoreboard position. Then, the video frame partitioning can be done by simple template matching.

It can be seen from Fig. 2.1(a), a scoreboard is a large, still, and rectangular area which consists pixels that change very infrequently. Based on this fact, an automatic scoreboard template extractor is proposed. First, a context-based static region detector is provided to extract few static regions called scoreboard candidates. Then a scoreboard selection method is used to get the right scoreboard. The block diagram of the scoreboard template extraction is shown in Fig. 2.2.



(a) Scoreboard frame.



(b) Non-scoreboard frame (sideline interview).



(c) Non-scoreboard frame (TV commercial).



(d) Non-scoreboard frame (slow motion replay).

Fig. 2.1 Examples of scoreboard frames and non-scoreboard frames.

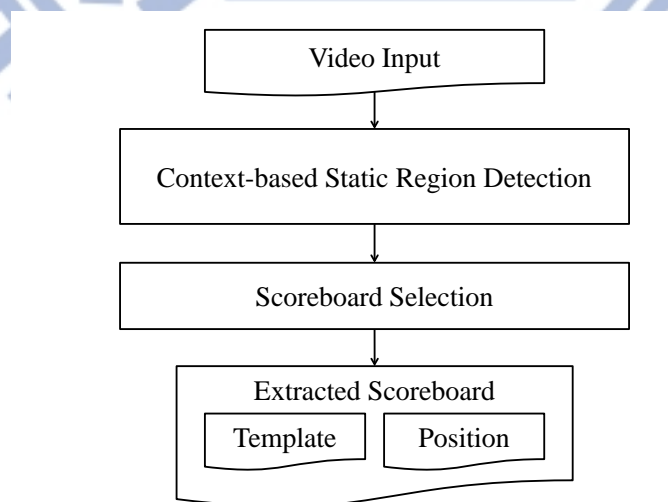


Fig. 2.2 Block diagram of scoreboard template extraction.

2.1.1 Context-Based Static Region Detection

As to context-based static region detection, a sports video is considered as an input frame sequence. Let f_i be the i -th input frame and K be the total frame number. For each frame f_i , the pixel-based frame difference between f_i and its previous frame f_{i-1} is first calculated as follows:

$$Df_i(x, y) = |f_i(x, y) - f_{i-1}(x, y)|, \quad 2 \leq i \leq K$$

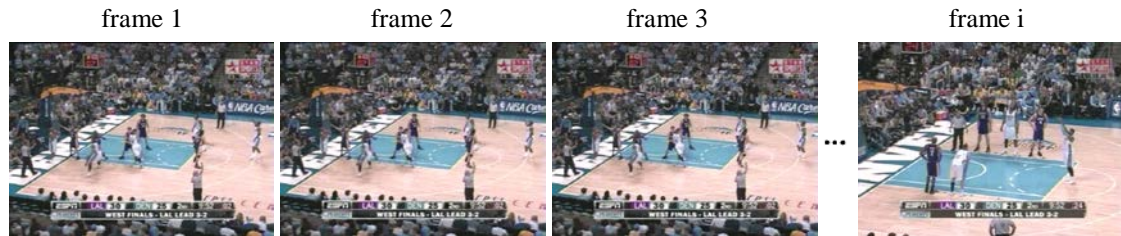
Where $f_i(x, y)$ represents the color value of pixel (x, y) at frame f_i . Then, an accumulated difference frame, ADf_i , is created by

$$ADf_i(x, y) = \sum_{j=2}^i Df_j(x, y), \quad 2 \leq i \leq K$$

Fig. 2.3 shows an example. As time goes by, the accumulated difference at each pixel can be considered as the change degree at that position.

After binarizing the accumulation result, each white point represents the position that changes more frequently and each black point represents the opposite. Then, we do region growing on black points of each binarized accumulated difference frame to find the largest connected component, which satisfies two constraints, as a potential scoreboard candidate. One constraint is about size. Since a scoreboard should be large enough to present score information, the width of the bounding box of the connected component should be at least $1/12$ frame width and the height should be at least $1/18$ frame height. The other constraint is about shape. The shape of the connected

component should be near rectangular, that is, the ratio of the connected component area and its bounding box area should be at least 0.9.



(a) Video frame sequence.



(b) Pixel-based frame difference.



(c) Accumulation of neighboring frame pair differences.



(d) Binarized results.

Fig. 2.3 Example of pixel-based frame difference accumulation.

For each binarized accumulated difference frame, if a potential scoreboard candidate is found, its position is then recorded. If the position is unchanged for consecutive frames, e.g. 300 frames, this means a potential scoreboard candidate is stable enough, and it can be considered as a scoreboard candidate. The context-based static region detector is applied repeatedly to the video frame sequence until few candidates are detected.

2.1.2 Scoreboard Selection

Some sports videos have overlaid rectangular logos made by the TV stations. The TV station logo is overlaid at the same position during the game while the scoreboard may disappear from time to time (see Fig. 2.1). Thus the logo is possibly detected as a scoreboard candidate. Fortunately, a TV station logo is never larger than a scoreboard, thus the scoreboard selection will prune smaller size candidates. Note that a scoreboard candidate consists of two parts, position and template. Now, we have located the scoreboard position. For template, since the scoreboard may disappear from time to time, extracting a template from a scoreboard candidate position cannot guarantee a right one. To solve this problem, for each scoreboard candidate sc extracted from f_i , the temporal change of the candidate sc , $TC(sc)$, is evaluated by

$$TC(sc) = \sum_{s=-2}^2 \sum_{x=0}^{M_c-1} \sum_{y=0}^{N_c-1} |f_i(x, y) - f_{i-s}(x, y)|,$$

where M_c and N_c represent the width and height of sc , $f_i(x, y)$ represents the color value of pixel (x, y) at frame f_i , and s represents temporal frame offset. Then, the scoreboard selection will take the one with the least temporal change as the scoreboard template.

According to our experiments, four scoreboard candidates are enough to extract the right scoreboard template. After scoreboard template extraction, the video frames partition can be done by matching every frame with scoreboard template at the scoreboard position.

2.1.3 Experimental Results

Our experiments are conducted by 10 NBA basketball games from 3 different broadcasters, i.e., ESPN, TNT, NBA TV. The data are recorded from TV in MPEG-2 format with resolution 480×352 . All 10 scoreboard templates are extracted successfully. It can be seen from Fig. 2.4, the proposed scoreboard template extractor works great for the 3 different broadcasters. Due to the effective results for different style scoreboards, it is believed that the proposed scoreboard template extractor can be generalized to other sports. Note that a scoreboard contains rich information in a sports video, so the proposed scoreboard template extractor is applicable as a pre-process in scoreboard information extraction and sports event detection as well.



(a) Game match broadcasted by ESPN. (b) Game match broadcasted by TNT. (c) Game match broadcasted by NBA TV.

Fig. 2.4 Scoreboard template extraction for 3 different broadcasters with extracted positions marked by white rectangle.

2.2 Overview of the Framework

The proposed framework is shown in Fig. 2.5. It can be seen from Fig. 2.5, the existing methods for semantic event extraction and replay detection can easily apply to the framework. Contrary to previous works, in the framework, scoreboard frames and non-scoreboard frames will be separately processed in semantic event extraction and slow motion replay detection. Since scoreboard only covers a small part of a video frame, conducting this slight-cost partitioning task before semantic resource extraction improves a lot of performance in both time complexity and detection accuracy.

In this dissertation, some sports video analysis schemes are proposed under the framework. A novel approach for webcast text analysis is presented in Chapter 3. Semantic event annotation through video clock recognition is provided in Chapter 4. A novel method for slow motion replay detection is described in Chapter 5.

Accordingly, the framework of the dissertation is presented in Fig. 2.5 as well. Detail techniques will be discussed in the following chapters.

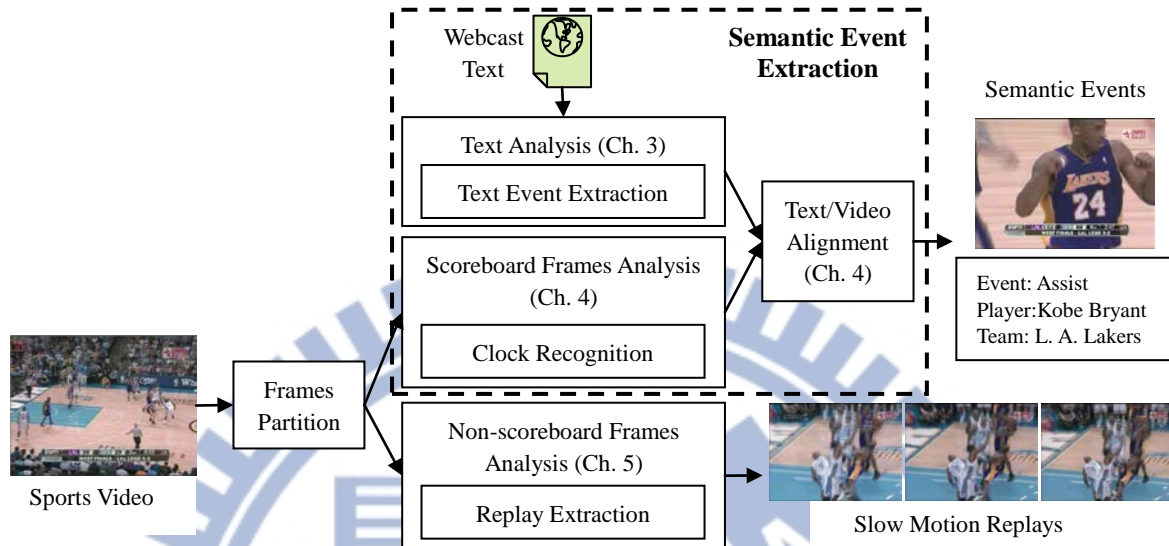


Fig. 2.5 The proposed framework.

2.3 Summary

In this chapter, a novel framework for sports video analysis, which provides flexibility to combine different schemes of event extraction and those of replay detection, is proposed. The novelty of video frames partition prevents semantic resource extraction from a lot of unnecessary processing frames, so the performance and detection rate can be increased. The framework is also capable of acquiring both two valuable semantic resources in one time.

CHAPTER 3

A NOVEL APPROACH FOR SEMANTIC EVENT EXTRACTION FROM SPORTS WEBCAST TEXT

In this chapter, we will propose an unsupervised approach to extract semantic events from sports webcast text. First, unrelated words in the descriptions of webcast text are filtered out, and then the filtered descriptions are clustered into significant event categories. Finally, the keywords for each event category are extracted. The extracted significant text events can be used for further video indexing and summarization. Furthermore, we also provide a hierarchical searching scheme for text event retrieval.

3.1 Introduction

In video summarization and retrieval, a source video is first clipped into smaller videos representing significant events through a preprocessing, called semantic event detection, which detect events occurred in a video and annotates events with appropriate tags. With finer results of the preprocessing, video summarization and retrieval can be completed efficiently and correctly. Most of existing event detection schemes use video content as their resource knowledge. However, the schemes relying on video content encounter a challenge called semantic gap, which represents the distance between low level video features and high level semantic events. In sports video, two kinds of external knowledge can be used to bridge the gap.

One of the external knowledge is Closed-Caption (CC) [19]. CC is the transcript of speech and sound, and it is helpful for semantic analysis of sports videos. It is mainly used in aid of listening and language learning, but only available in certain videos and certain countries. Because CC completely records the sound in video, it contains a lot of redundant information and usually lacks of structure. The other external knowledge is webcast text. Comparing to CC, webcast text is the online commentary posted by professional announcers and focuses more on sports games. It contains more detail information (e.g., event name, time, player involved, etc.), which is difficult to extract from video content itself automatically. Xu and Chua [5] first use webcast text as external knowledge to assist event detection in soccer video. They proposed a framework that combines internal AV features with external knowledge to do event detection and event boundary identification. But the proposed model is inapplicable to other team sports. Xu et al. [8] apply probabilistic latent semantic analysis (pLSA), a linear algebra–probability combined model, to analyze the webcast text for text event clustering and detection. Based on their observation, the descriptions of the same event in the webcast text have a similar sentence structure and word usage. They use pLSA to first cluster the descriptions into several categories and then extract keywords from each category for event detection. Although they extend pLSA for both basketball and soccer, there are two problems in the approach.

- 1) The optimal numbers of event categories are nine for basketball and eight for soccer in the results, which is determined by minimizing the ratio of within-class similarity and between-class similarity. In fact, there are more event categories for a basketball or soccer game. For example, in a basketball game, many events, such as timeout, assist, turnover, ejected, are mis-clustered into wrong categories or discarded as noises. This may cause side effects degrading and limiting the results of video retrieval.
- 2) After keywords extraction, events can be detected by keywords matching. In Xu et al.'s method, they use the top ranked word in pLSA model as single-keyword of each event category. But in some event categories, the single-keyword match will lead to horrible results. For example, in their method for a basketball game, “jumper” event represents those jumpers that players make. Without detecting “makes” as a previous word of “jumper” in description sentences, the precision of “jumper” event detection is decreased from 89.3% to 51.7% in their testing dataset. However, the “jumper” event actually is an event that consists of “makes jumper” event and “misses jumper” event. The former can be used in highlights, and the latter can be used in sports behavior analysis and injury prevention. Accordingly, using single-keyword match is insufficient and some important events will be discarded.

To treat the above-mentioned problems, we propose a method to analyze sports webcast text and extract significant text events. An unsupervised scheme is used to detect events from the webcast text and extract multiple keywords from each event. A data structure is used to store these multiple keywords and to support a hierarchical search system with auto-complete feature for event retrieval. The word “hierarchical” means that a user can get more specific results by querying more keywords and the word “auto-complete” means that the system can give suggested keywords during the query step.

3.2 Proposed Method

Webcast text comprises knowledge which is closely related to the game and is easily retrieved from websites. As can be seen in Fig. 3.1, it contains time tags, team names, scores, and event descriptions. The format is so organized that we can follow the time flow and understand how the game goes on. Among this well-organized text, it is apparent that event descriptions relate to semantic events the most. Our goal is to analyze event descriptions and automatically extract significant events from them.

1st Quarter Summary			
TIME	NEW ORLEANS	SCORE	DENVER
12:00	Start of the 1st Quarter		
12:00	Jumpball: Tyson Chandler vs. Nene Hilario (Chris Paul gains possession)	0-0	
11:33	Peja Stojakovic misses 10-foot two point shot	0-0	
11:32	New Orleans offensive rebound	0-0	
11:32	shot clock violation	0-0	
11:19		0-2	Carmelo Anthony makes 10-foot jumper (Kenyon Martin assists)
11:03		0-2	Dahntay Jones personal foul (Chris Paul draws the foul)
10:53	Kenyon Martin blocks Peja Stojakovic's jumper	0-2	
10:52		0-2	Kenyon Martin defensive rebound
10:42		0-4	Kenyon Martin makes 20-foot jumper (Carmelo Anthony assists)
10:23	David West makes 12-foot two point shot	2-4	
10:08		2-6	Nene Hilario makes driving layup (Carmelo Anthony assists)
10:08	Peja Stojakovic shooting foul (Nene Hilario draws the foul)	2-6	
10:08		2-6	Nene Hilario misses free throw 1 of 1
10:07	David West defensive rebound	2-6	

Fig. 3.1 An example of basketball webcast text.

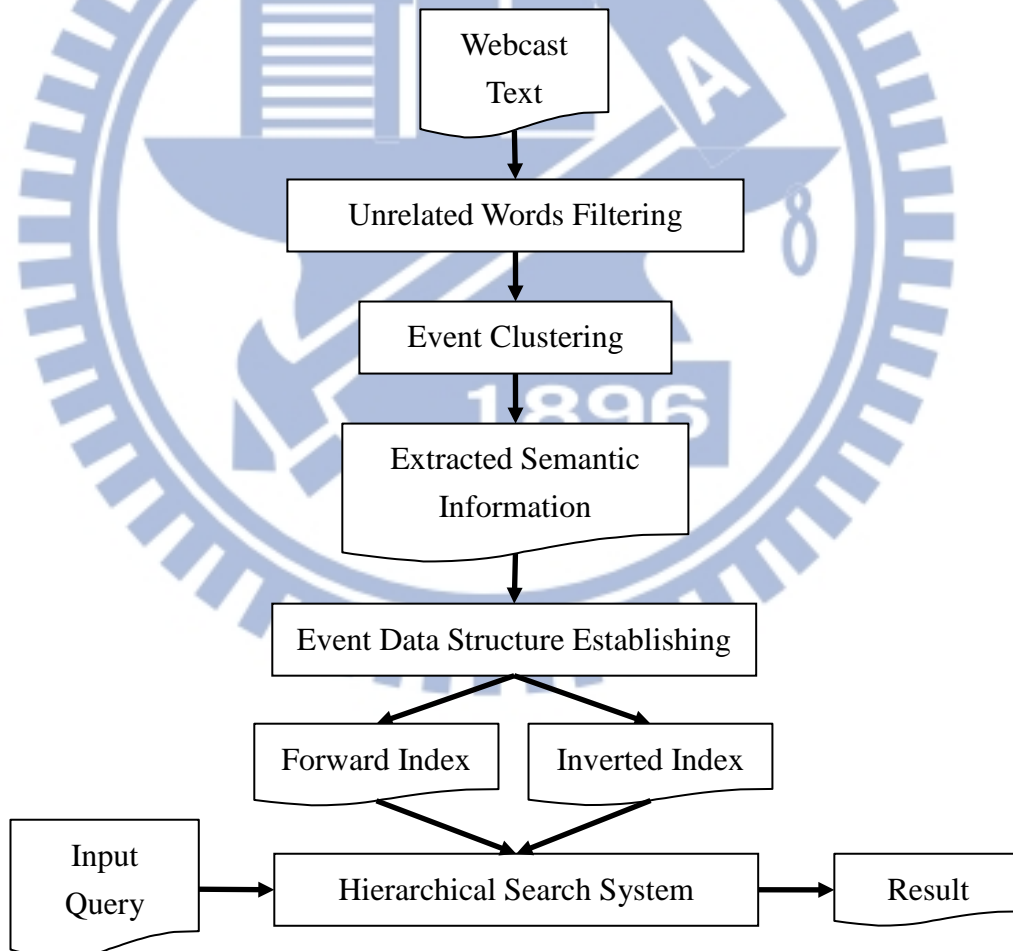


Fig. 3.2 Block diagram of the proposed method.

The block diagram of the proposed method is presented in Fig. 3.2. It can be seen that we first filter out unrelated words of webcast text and then cluster them into significant events. We store the extracted semantic information with a pair of index tables and build a hierarchical retrieval system by manipulating the two tables. The detail of each block will be described in the following subsections.

3.2.1 Unrelated Words Filtering

In webcast text, each description can be considered as an event. It contains many words and may include player name, team name, movement name, and whether the player or the team makes the movement or not. An example is given in Fig. 3.3, a player named “Peja Stojakovic” failed to make a movement called “10-foot two point shot.”

The number of descriptions in each basketball game is more than four hundred. The descriptions are readable and can be easily categorized into several events by human eyes. But the task is not effortless for computer machines. According to our observations, words in each description consist of three mutually disjoint word sets: 1) stop words, 2) event keywords, and 3) names. Stop words are unrelated to event and should be discarded. Event keywords are closely related to event and should be kept for event detection. Names including team names and player names should be

preserved for event annotation. Our objective is to extract event keywords and use these keywords to do event clustering. To achieve the objective, based on a reference stop word list and an online name information, an interactive system is first provided to establish a sports stop word list and an event keyword list. The system will be explained in Sections 3.2.1.1 and 3.2.1.2. According to these two lists, for each webcast text, an unrelated word filtering procedure described in Section 3.2.1.3 is next provided to filter out stop words and to preserve name words. The remaining keywords are then used for event clustering, which will be described in Section 3.2.2.

	12:00	Jumpball: Tyson Chandler vs. Nene Hilario (Chris Paul gains possession)	0-0
Description	11:55	Peja Stojakovic misses 10-foot two point shot	0-0
	11:32	New Orleans offensive rebound	0-0
	11:32	shot clock violation	0-0
	11:19		0-2
	11:03		0-2
Word	10:55	Kenyon Martin blocks Peja Stojakovic's jumper	0-2
	10:52		0-2

Fig. 3.3 An example to illustrate description and word.

3.2.1.1 Stop Words

In information retrieval, there are some words that occur very frequently (e.g. some articles, prepositions, pronouns, be-verbs) and are useless in document matching. These words are called stop words [20]. Due to the uselessness of stop words, filtering out them during both index step and query step can reduce the index size and query

processing time. This technique has been used in search engines and can be implemented through predefining a stop word list. For the variety of applications, there is no standard stop word list. Many reference stop word lists [21]-[22] have been proposed by using techniques about statistics and probability.

From Fig. 3.1, it can be seen that descriptions contain articles (e.g. “the”), prepositions (e.g. “of”), range of shot (e.g. “10-foot”), and points of shot (e.g. “two point”). Some words are details of events which decrease the connections between similar events. With the aid of reference stop lists, articles and prepositions can be easily filtered out from descriptions. However, the range of shot and points of shot are exceptions in reference stop lists. Moreover, in soccer webcast text, due to the relatively larger ground, there are more unrelated words to describe locations where an event happens. For example, right wing, left wing, inside the box, outside the box, left corner, right corner, etc. Accordingly, it is hard to automatically generate a sports stop word list for all kinds of sports. So we will provide an interactive system to establish a sports stop word list.

3.2.1.2 The Proposed Interactive System for Establishing Sports Stop Word List and Event Keyword List

As mentioned previously, an interactive system is proposed to establish the sports stop word list and the event keyword list for sports webcast text. First, webcast

text descriptions of several games are taken as training inputs, next some unrelated words are filtered out according to a reference stop word list [21] and a name word list (e.g., online box score in basketball and online player statistics in soccer). And then the system interacts with sports professionals, who will divide the remaining words into a black list and a white list. The black list contains stop words for sports, and the white list contains sports event keywords. Finally the black list is merged into the reference stop word list to get the sports stop word list. The block diagram of the interactive system is presented in Fig. 3.4.

Our training webcast text is conducted by 41 basketball games and 48 soccer games. After the reference stop words filtering and the name words filtering, the remaining words needed to interactively ask professionals are less than 100 in basketball and less than 200 in soccer. The responses from professionals may take just few minutes.

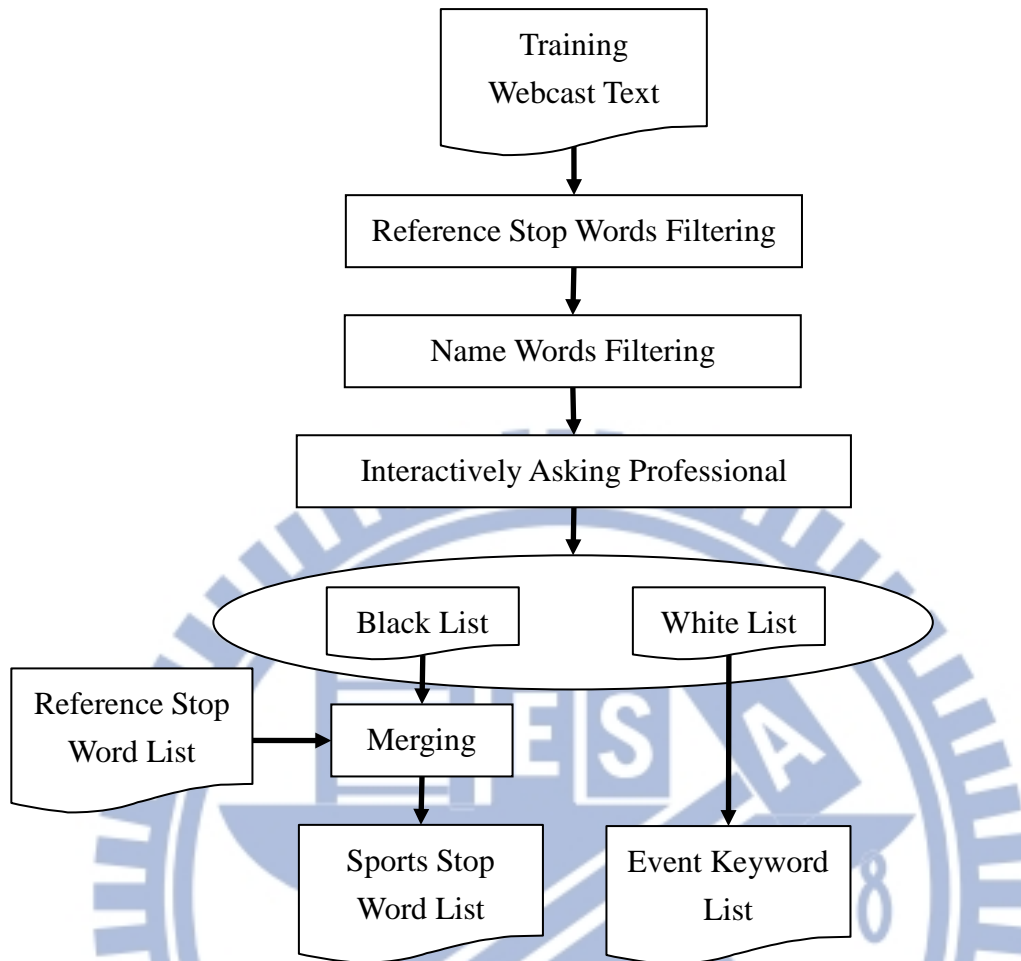


Fig. 3.4 The block diagram of the interactive pre-training system.

3.2.1.3 The Proposed Unrelated Words Filtering Procedure

Fig. 3.5 shows the block diagram of the proposed unrelated words filtering procedure. For a webcast text, the sports stop word list is first used to filter out unrelated words. Next the event keyword list is used to extract event keywords. Then the words with uppercase beginning in the remaining words are considered as reserved names for further indexing. According to our experiment results, the unrelated words filtering works well both in basketball and soccer.

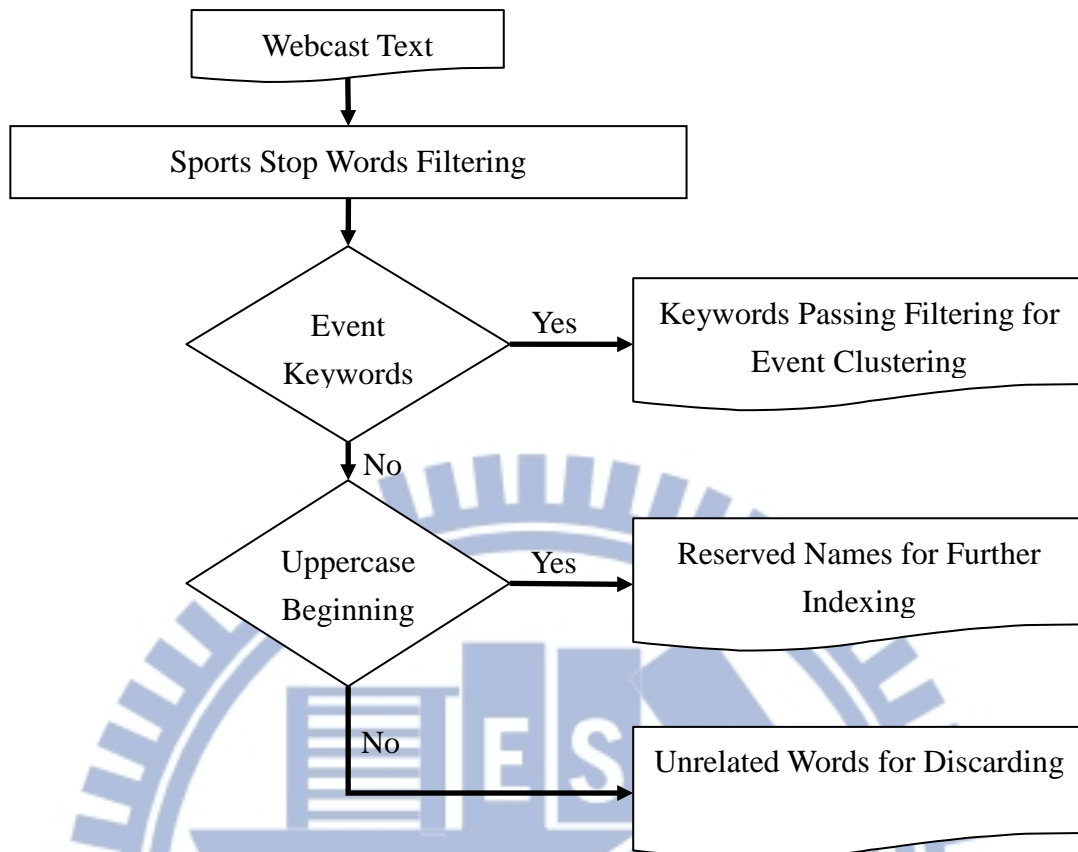


Fig. 3.5 Block diagram of unrelated words filtering procedure.

3.2.2 Event Clustering

After filtering, each description is reduced and almost exactly describes an event; for example, “misses shot” represents a missed shot. So a matching function is provided to cluster these filtered descriptions into event categories.

Filtered descriptions can be represented as $FD = \{ fd_1, fd_2, \dots, fd_N \}$, and event categories can be represented as $C = \{ C_1, C_2, \dots, C_K \}$, where N denotes the number of descriptions in a game and K denotes the number of categories that the clustering step produces. Since a filtered description consists of some words, it can be considered as a set of words. Note that the number of keywords of an event category

is not restricted to be single in our method. The matching function is defined as

$$Text_Match(x, y) = \begin{cases} 1, & \text{if } x = y \\ 0, & \text{otherwise,} \end{cases} \quad (3.1)$$

where x and y are two sets of words. Each filtered description, fd_i , can be clustered

into one category based on the following function

$$Clustering(fd_i) = \arg \max_m \{Text_Match(fd_i, Keywords(C_m)), m = 1, \dots, K\}, \quad i = 1, \dots, N, \quad (3.2)$$

where $Keywords(C_m)$ denotes the multiple-keywords set of category C_m . $Clustering(fd_i)$

$= j$ means that description fd_i is clustered into category C_j . In order to avoid zero

matching in (2), a flag function to examine whether the situation happens is defined as

$$Flag(fd_i) = \max_m \{Text_Match(fd_i, Keywords(C_m)), m = 1, \dots, K\}, \quad i = 1, \dots, N. \quad (3.3)$$

The detail of the proposed clustering algorithm is given below.

Clustering Algorithm

Step0: Initialization: Given $FD = \{fd_1, fd_2, \dots, fd_N\}$.

Set $K = 1$, $Clustering(fd_1) = 1$, $Keywords(C_1) = fd_1$, $i = 2$.

Step1: // Cluster the description fd_i according to Functions (3.1), (3.2), and (3.3).

// The procedure includes the following pseudo code

For $m = 1$ to K , use Function (1) to calculate $TMfd_{im}$

$$TMfd_{im} = \text{Text_Match}(fd_i, \text{Keywords}(C_m));$$

Let

$$\text{Flag}(fd_i) = \max_{m=1, \dots, K} \{TMfd_{im}\};$$

if ($\text{Flag}(fd_i) = 0$) then begin

// fd_i cannot be clustered into any existing class

// create a new class for fd_i

$K = K + 1;$

$\text{Keywords}(C_K) = fd_i;$

$\text{Clustering}(fd_i) = K;$

else

// fd_i is clustered into one of the existing classes

Use Function (2) to calculate $\text{Clustering}(fd_i)$ as

$$\text{Clustering}(fd_i) = \arg \max_m \{TMfd_{im}\};$$

end

Step2: If any of the descriptions in FD is not clustered yet, set $i = i + 1$ and go to

Step1 for next iteration. Otherwise, end of iterations.

Once the clustering algorithm is completed, the filtered descriptions are clustered into event categories, and keyword extraction is done by using each keyword set as

multiple keywords of the event. At the meantime, semantic event detection is accomplished. Then two data structures are built to recommend users for further queries and to support the hierarchical search.

3.2.3 Hierarchical Search System

Fig. 3.6 gives an example to show the concept of the proposed hierarchical search system. First, a user can query by one word to get rough results. Then he can continually query by more words to get into deeper levels for finer results. Here we implement the system by establishing a pair of index tables and manipulating them back and forth.

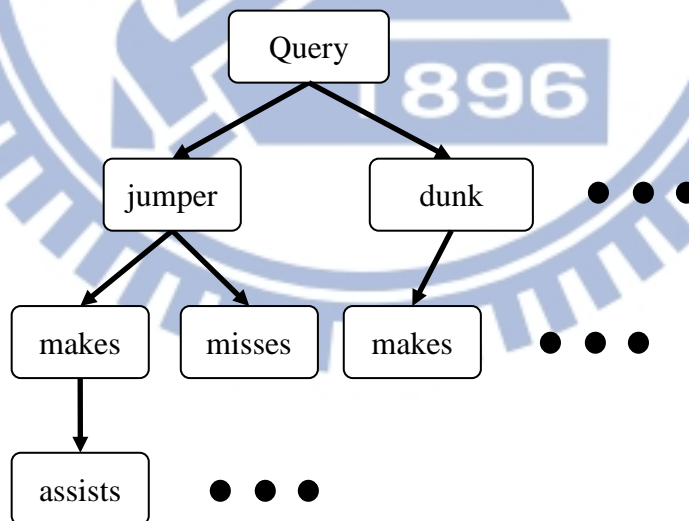


Fig. 3.6 An example to illustrate the concept of the proposed hierarchical search system.

Here we build a forward index table and an inverted index table. The former records mappings from descriptions to event keywords, and the latter stores mappings from keywords to descriptions. Note that the forward index table is established automatically after applying the unrelated words filtering procedure. Based on the forward index table, the inverted index table can be established by sequentially scanning event keyword set of each description. An example is given in Fig. 3.7 to do clearer explanation. Suppose we have five descriptions as shown in Fig. 3.7(a). After applying unrelated words filtering procedure to each description, we can obtain Fig. 3.7(b). By scanning each row in Fig. 3.7(b), for each row, we can obtain a description index (DI) and the corresponding event keyword set (EKS). Then DI is linked to each keyword in EKS. After scanning all rows sequentially in Fig. 3.7(b), Fig. 3.7(c) is established. Both inverted index table and forward index table are referred to achieve the hierarchical search system. The inverted index table is used for returning query results by intersecting those description sets mapped by query keywords. The forward index is originally just an intermediate, but reused in our method for providing suggested query keywords, i.e. auto-complete feature.

Webcast Text

Index of Description	Description
D1	Peja Stojakovic misses 10-foot two point shot
D2	David West misses jumper
D3	Peja Stojakovic makes 19-foot two point shot
D4	Trevor Ariza makes 19-foot jumper
D5	David West makes 17-foot jumper (Chris Paul assists)

(a) Descriptions and their indices.

Forward Index

Index of Description	Event Keyword Set
D1	misses, shot
D2	misses, jumper
D3	makes, shot
D4	makes, jumper
D5	assists, makes, jumper

(b) Mappings from description indices to event keywords.

Inverted Index

Keywords	Indices of Description Set
assists	D5
jumper	D2, D4, D5
makes	D3, D4, D5
misses	D1, D2
shot	D1, D3

(c) Mappings from keywords to description indices.

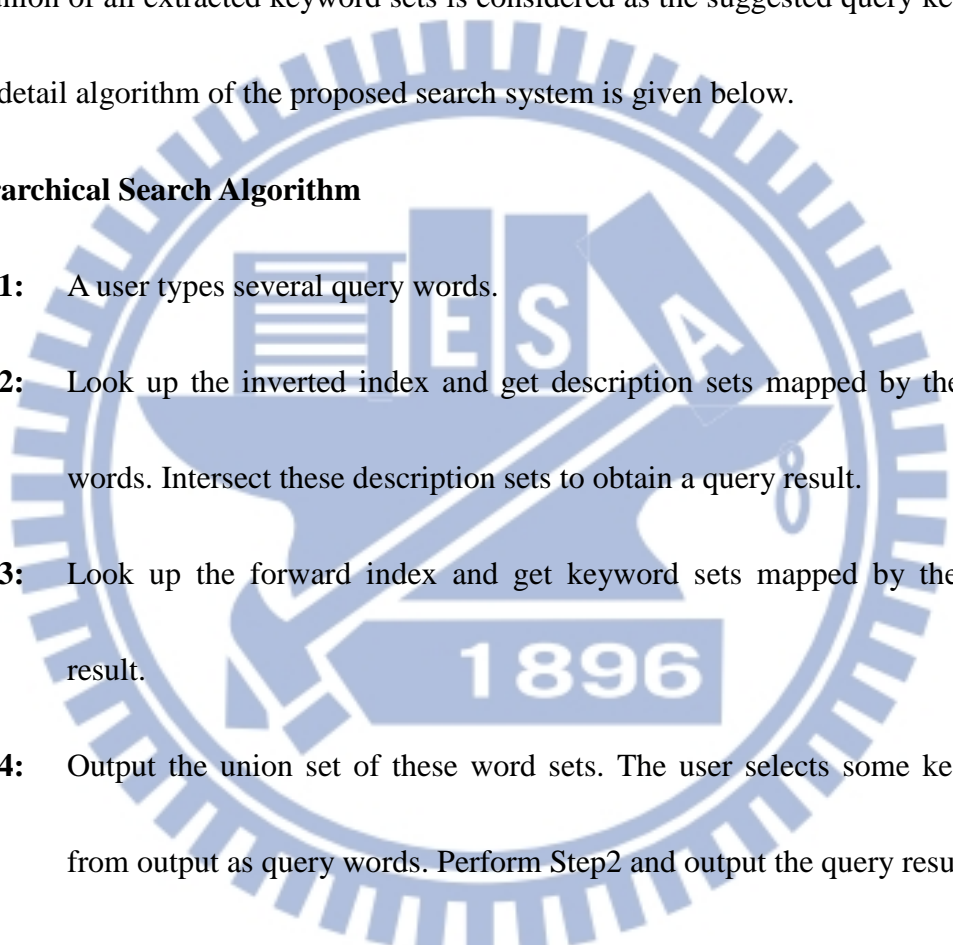
Fig. 3.7 An example to illustrate the data structure for hierarchical search.

In our system, a query is considered as a set of multiple words. The hierarchical feature means that a user can get more general results by querying fewer words or get more specific result by querying more words; for example, the results of querying “jumper” are those descriptions having the keyword “jumper”, and the results of querying “jumper makes” are those descriptions having both “jumper” and “makes.”

The query result is the intersection of description sets obtained through the keywords of query in the inverted index list. For providing suggested query keywords, the resulting intersection set is then used as another query for the forward index list. The keyword set of each description in the resulting intersection set are extracted. Finally, the union of all extracted keyword sets is considered as the suggested query keywords.

The detail algorithm of the proposed search system is given below.

Hierarchical Search Algorithm

- 
- Step1:** A user types several query words.
- Step2:** Look up the inverted index and get description sets mapped by the query words. Intersect these description sets to obtain a query result.
- Step3:** Look up the forward index and get keyword sets mapped by the query result.
- Step4:** Output the union set of these word sets. The user selects some keywords from output as query words. Perform Step2 and output the query result.

Here, we use Fig. 3.7 as an example to do explanation. Assume that a user types a query {jumper}, the system will look up the inverted index list and get a temporary result set {D2, D4, D5}. Then, the system will look up the forward index list and recommend the user {assists, jumper, makes, misses}, i.e. the union set of {jumper, misses}, {jumper, makes}, and {assists, jumper, makes}. If the user changes his query

to {jumper, makes}, the system will return {D4, D5}, i.e. the intersection set of {D3, D4, D5} and {D2, D4, D5}. Therefore, a powerful hierarchical search system with query recommendation function is built.

3.3 Experimental Results

In most search systems, statistical analysis such as receiver operating characteristic (ROC) analysis or recall-precision is used to evaluate the performance. Through the analysis, the system degradation caused by misclassification can be estimated. However, as mentioned in Section 3.2.2, we cluster descriptions by an exactly matching function, so there is no misclassified event in our system. This means that both precision and recall rates of the proposed method are 100%.

Researches aimed at detecting text events from webcast text are few. Xu and Chua [5] modeled webcast text as external knowledge in detecting events from football and soccer. The evaluation of the fusion video event detection was presented, but that of webcast text analysis alone was not. Xu et al. [8] proposed a framework to analyze webcast text and videos independently and align them through game time. According to the framework, the performance of video event detection mainly depends on webcast text analysis. Here we compare our method with Xu et al.'s work.

Our experiments are conducted by 25 NBA 2009-2010 games and 41 NBA

2008-2009 postseason games. The former are used as training database, and the latter are used as testing database to examine the reliability of the proposed method. We also collect 68 UEFA Champions League 2010-2011 soccer games, where 20 of them are used as training database and the other 48 are used as testing database. The webcast text from 134 games is acquired from ESPN website. As can be seen in Table 3.1, hundreds of descriptions in a game are clustered into, in average, 44 semantic event categories for basketball and 20 semantic event categories for soccer.

Table 3.1 Average number of sports event categories in 25 basketball training data and 20 soccer training data.

	Mean	Variance	Standard deviation
Basketball	44.08	9.08	3.01
Soccer	19.85	5.40	2.32

From Xu et al.'s previous work, the pLSA, the optimal number of event categories is nine for basketball and eight for soccer. The top three keywords of each category are selected by a conditional probability. They use the top ranked keyword as single keyword during event detection. We map the top three results of pLSA to our multiple keywords categories in Table 3.2 and Table 3.3. In Table 3.3, because “attempt” is chosen as a member of black list in the interactive system, we use “shot” as the single-keyword match for mappings from soccer events in pLSA to those in the proposed method. The words “missed” and “misses” refer to the same verb (e.g., miss)

and have the same meaning in descriptions. We consider these two words as the same and use “missed(misses)” as their common representative. In order to achieve fine performance in detecting semantic events, Xu et al. not only use keywords detection in description sentences, but also analyze context information in them. For example, in basketball, the top ranked keyword “jumper” is detected as “Jumper” event only if its previous word is “makes,” and other sentences containing word “jumper,” e.g., Kenyon Martin misses 22-foot jumper, are discarded. However, these discarded events are actually semantic events and can be valuable for further research, e.g., sports posture analysis, injury prevention, special highlight, etc. It can be seen from Table 3.2 and Table 3.3 that every category of pLSA is mapped to several different semantic events of the proposed method. These several events are related but somehow different. For example, in basketball, “jumper misses” describes that a jumper is missed while “jumper makes” describes that a jumper is made successfully. In soccer, “blocked shot” describes that a shot attempt is blocked by an opponent while “missed(misses) shot” describes that a shot attempt is missed by the kicker himself. Hence, misclassifying or discarding these events decreases the precision and recall rates. However, in our method, the precision and recall rates are both 100%. With the support of hierarchical search system, we can query multiple keywords for more specific events, which is even better than pLSA with context information. Table

3.2 and Table 3.3 also show those semantic event categories which are unavailable in Xu et al.'s method, but can be detected in our method, e.g., steal, timeout, turnover for basketball and injury, blocked, penalty for soccer. These semantic events are important for special highlights or injury prevention, and should not be ignored or misclassified. So, the proposed method is superior to pLSA.

Table 3.2 Mappings of basketball event categories from pLSA to the proposed method.

Xu et al.'s Method (pLSA)		Proposed Method
Category	Ranked Keywords	(Categories with Multiple Keywords)
Shot	shot	makes shot, misses shot
	pass	
	bad	
Jumper	jumper	jumper misses, jumper makes, assists jumper
	foot	makes
	misses	
Layup	layup	layup makes, layup misses, driving layup makes,
	driving	assists layup makes
	blocks	
Dunk	dunk	dunk makes, assists dunk makes, dunk makes
	makes	slam, driving dunk makes, dunk misses
	misses	

Table 3.2 Mappings of basketball event categories from pLSA to the proposed method (continued).

Xu et al.'s Method (pLSA)		Proposed Method
Category	Ranked Keywords	(Categories with Multiple Keywords)
Block	blocks shot assists	blocks layup, blocks jumper, blocks driving layup, blocks hook shot, blocks shot, blocks dunk, blocks layup, blocks jumper, blocks driving layup, blocks hook shot, blocks shot, blocks dunk
Rebound	rebound defensive offensive	defensive rebound, offensive rebound
Foul	foul draw personal	draws foul shooting, draws foul personal, draws foul offensive, ball draws foul loose, foul technical, defense foul illegal person, draws flagrant foul type
Free throw	throw free makes	free makes throw, free misses throw
Substitution	enters game timeout	enters
N/A		bad pass, bad pass steals, bad lost steals, full timeout, official timeout, turnover, traveling, ejected, double dribble, defense illegal, clock shot violation

Table 3.3 Mappings of soccer event categories from pLSA to the proposed method.

Xu et al.'s Method (pLSA)		Proposed Method
Category	Ranked Keywords	(Categories with Multiple Keywords)
Corner	corner	corner, assisted corner saved shot, corner goal penalty shot, corner saved shot, assisted corner goal, assisted corner goal shot, assisted corner
	conceded	missed(misses), corner goal shot, corner missed(misses) shot, assisted corner
	bottom	missed(misses) shot, corner free kick missed(misses) shot, assisted corner saved, corner free goal kick shot
Shot	attempt	blocked shot, assisted missed(misses) shot, assisted blocked shot, assisted goal saved shot, missed(misses) shot, assisted corner saved shot,
	right	assisted shot, corner goal penalty shot, corner saved shot, assisted corner goal shot, corner goal shot, corner missed(misses) shot, goal saved shot,
	footed	free kick shot, assisted goal shot, free kick missed(misses) shot, assisted corner missed(misses) shot, corner free kick missed(misses) shot, goal penalty saved shot, corner free goal kick shot, goal penalty shot
Foul	foul	foul, card foul yellow, foul penalty, card foul
	dangerous	
Card	for	
	yellow	card foul yellow, card yellow
	shown card	

Table 3.3 Mappings of soccer event categories from pLSA to the proposed method
(continued).

Xu et al.'s Method (pLSA)		Proposed Method
Category	Ranked Keywords	(Categories with Multiple Keywords)
Free kick	kick	free kick, free kick shot, free kick
	free	missed(misses) shot, corner free kick
	wins	missed(misses) shot, corner free goal kick shot
Offside	offside	offside
	ball	
	tries	
Substitution	substitution	replaces substitution, injury replaces substitution
	replaces	
	lineups	
Goal	goal	assisted goal saved shot, corner goal penalty shot, assisted corner goal, assisted corner goal
	shot	shot, corner goal shot, goal saved shot, assisted goal shot, assisted goal saved, goal penalty saved
	box	shot, goal saved, goal, corner free goal kick shot, goal penalty shot, assisted goal
	N/A	injury, assisted missed(misses), assisted blocked, penalty, assisted

Here we want to examine the reliability of the proposed method. For basketball, 25 NBA 2009-2010 games are taken as training data. After processing all the training data and gathering the extracted semantic events, we collect the union of these semantic events as a sample set with cardinality 82. Then we process the testing data, which are collected from 41 NBA 2008-2009 postseason games, and examine whether all the semantic events extracted from testing data are listed in the sample set or not.

For soccer, we use 20 UEFA Champions League soccer games as training data and 48 UEFA Champions League soccer games as testing data. According to our examination, with sparse exceptions, almost all the semantic events extracted from testing data can be found in the sample set. Table 3.4 and Table 3.5 show all exception events which are quite rare. These exceptions may be caused by different writing styles or some rarely happened events, and can still be collected in an interactive way if necessary. Therefore, the proposed method is very stable.

Table 3.4 Occurrences of exception basketball events from 41 testing games.

Exception events	18679 basketball descriptions	
	Number	Percentage
10 second	3	0.02%
backcourt	7	0.04%
called full timeout	1	0.01%
driving dunk misses	2	0.01%
dunk misses slam	2	0.01%
away ball draws foul	5	0.03%
misses pointer	7	0.04%
flagrant free misses throw	1	0.01%
blocks driving dunk	1	0.01%

Table 3.5 Occurrences of exception soccer events from 48 testing games.

Exception events	5727 soccer descriptions	
	Number	Percentage
card	6	0.10%
corner penalty saved shot	2	0.03%
missed(misses)	3	0.05%
goal shot	1	0.02%
assisted corner missed shot	1	0.02%
missed shot	1	0.02%
shot	4	0.07%
corner missed(misses)	3	0.05%
corner saved	2	0.03%
assisted corner	1	0.02%
blocked	1	0.02%

3.4 Summary

In this chapter, we have proposed an unsupervised approach for semantic event extraction from sports webcast text and made some contributions: 1) detecting semantic events from webcast text in an unsupervised manner; 2) requiring no additional context information analysis; 3) preserving more significant events in sports games; 4) extracting multiple keywords from event categories to support hierarchical searching; 5) providing auto-complete feature for finer retrieval. According to experimental results, the proposed method extracts significant semantic events from basketball and soccer games and preserves those events that are ignored or misclassified by previous work. The extracted significant text events can be used for further video indexing and summarization. Furthermore, the proposed method is reliable.

CHAPTER 4

ANNOTATING WEBCAST TEXT IN BASKETBALL VIDEOS BY GAME CLOCK RECOGNITION AND TEXT/VIDEO ALIGNMENT

In this chapter, we will propose a text/video alignment and event annotation method. As mentioned in Chapter 2, semantic events appear in scoreboard frames only. Thus, the proposed semantic event extraction method focuses on analyzing scoreboard frames. For each scoreboard frame, location of each clock digit is first located. A digit templates collection scheme is provided to collect digit character templates. With clock digit locations and digit templates, a two-step strategy is proposed to recognize game clocks on the semi-transparent scoreboard in scoreboard frames. With the game clock recognized from sports video, the alignment work is done by finding every match for game clock extracted from webcast text and annotating the corresponding event description on video frames.

4.1 Introduction

In the world, substantial number of sports videos are produced and broadcasted through television program or Internet streaming. It is nearly impossible to watch all sports videos. Most of the time, fans prefer to watch highlights of sports videos or retrieve only partial video segments that they are interested in. Therefore, sports video summarization and retrieval have become valuable and hot research topics. In these

topics, automatic semantic event detection and video annotation are essential works.

Most of existing researches [1]-[3] use video content as resource knowledge. However, schemes relying on video content encounter a challenge called semantic gap. Recently, some researches [4]-[9] use a multimodal fusion of video content and external resource knowledge to bridge the semantic gap. The multimodal fusion scheme, which analyzes webcast text and video content separately and then does text/video alignment to complete sports video annotation or summarization, has been used in American football [4], soccer [6]-[8], and basketball [7]-[8].

In the scheme, text/video alignment, which consists of event moment detection and event boundary detection, has a great impact on performance. It can be achieved through scoreboard recognition. As can be seen in Fig. 4.1, a scoreboard is usually overlaid on sports videos to present the audience some game related information (e.g., score, game status, game clock) that can be recognized and aligned with text results. For sports with game clock (e.g., basketball and soccer), event moment detection can be performed through video game clock recognition. Xu et al. [6]-[8] used Temporal Neighboring Pattern Similarity (TNPS) measure to locate game clock and recognize each digit of the clock. A detection-verification-redetection mechanism is proposed to solve the problem of temporal disappearing clock region in basketball videos. However, recognizing game clock in a frame which has no game clock is definitely

unnecessary. The cost of verification and redetection could have been avoided. Moreover, the clock digit characters cannot be located on a semi-transparent scoreboard.



(a) Transparent scoreboard.



(b) Non-transparent scoreboard.

Fig. 4.1 Two examples of overlaid scoreboard with game clock in basketball video.

According to our observation, two main problems of detecting game clock in basketball videos are the temporal disappearance and the temporal pause of game clock. The temporal disappearance of game clock may be caused by slow motion replays, shot transition effect or TV commercials, etc. The temporal pause of game

clock may be due to some basketball events, e.g., timeout, substitution, foul, etc. These two problems make game clock recognition of basketball videos much harder than that of soccer videos. Furthermore, in order not to let scoreboard cover details of video frames, more and more sports videos use transparent scoreboard overlay. The transparency of scoreboard is another serious problem for game clock location and recognition.

As to event boundary detection, some researchers used hidden Markov model (HMM) [7] and conditional random field model (CRFM) [8]. However, not all events have obvious temporal patterns for start and end boundaries due to the complicated camera motions and play ground textures of sports videos. In Xu et al.'s experiments [8], boundary detection accuracy (BDA) are relatively low for foul and substitution events in basketball. Because foul event is short and followed by some other events (e.g. free throw, throw in) without obvious temporal transition patterns, and substitution event is loose of structure. Even if boundaries are labeled manually, results may still be subjective.

To treat the above-mentioned problems, based on the sports video analysis framework proposed in Chapter 2, we present a text/video alignment and event annotation method.

4.2 Proposed Method

In the proposed method, a video frame partition method (see Chapter 2) is referred to divide frames into scoreboard frames and non-scoreboard frames. For each scoreboard frame, location of each clock digit is first located. A digit templates collection scheme is provided to collect digit character templates. With clock digit locations and digit templates, a two-step strategy is proposed to recognize game clocks on the semi-transparent scoreboard in scoreboard frames. With the game clock recognized from sports video, the alignment work is done by finding every match for game clock extracted from webcast text (see Chapter 3) and annotating the corresponding event description on video frames.

4.2.1 Video Frames Partition

As can be seen from Fig. 2.1, in basketball videos, all frames can be broadly classified into two categories, scoreboard frames and non-scoreboard frames. Scoreboard frames present basketball game with scoreboard overlaid on them, while non-scoreboard frames present the rest, e.g., sideline interview, slow motion replay, etc. Since semantic events only appear in scoreboard frames, it is beneficial to filter out unnecessary processing frames in each semantic resource extraction step. So, an automatic scoreboard template extractor is needed.

As shown in Fig. 2.1(a), a scoreboard is fixed rectangular area with pixels changing infrequently. Based on this fact, our previous work (see Chapter 2) presented an automatic scoreboard template extractor. Here, we adapt this extractor to get the scoreboard template and position. After scoreboard template extraction, the video frames partition can be done by matching every frame with scoreboard template at the scoreboard position.

4.2.2 Semantic Event Extraction from Scoreboard Frames

In this study, a multimodal fusion scheme is conducted for semantic events extraction from scoreboard frames. Using our previous work in Chapter 3, text events with game clock can be extracted from webcast text. Then, a text/video alignment and event annotation method is proposed by recognizing game clocks of scoreboard frames.

As to game clock recognition, location of each clock digit is first located. A digit templates collection scheme is provided to collect digit character templates. With clock digit locations and digit templates, a two-step strategy is proposed to recognize game clocks on the semi-transparent scoreboard in scoreboard frames. Here, without loss of generality, four game clock patterns can be defined in Fig. 4.2.

Game Clock Patterns	$X_1X_2:X_3X_4$	$X_2:X_3X_4$	$X_3X_4.X_5$	$X_4.X_5$
Meaning of Each Digit	X_1 :TEN-MINUTE, X_2 :MINUTE, X_3 :TEN-SECOND, X_4 :SECOND, X_5 :TENTH-SECOND			

Fig. 4.2 General definitions of game clock patterns.

4.2.2.1 Clock Digit Locator

Based on the fact that there are 30 frames and 300 frames taken in one second and ten seconds, for each scoreboard frame f_i , the pixel-based frame difference between f_i and f_{i-30} and that between f_i and f_{i-300} are first calculated as follows:

$$Df_{i,30}(x, y) = |f_i(x, y) - f_{i-30}(x, y)|.$$

$$Df_{i,300}(x, y) = |f_i(x, y) - f_{i-300}(x, y)|.$$

Where (x, y) is a point of the scoreboard area. Then, two accumulated difference frame, $ADf_{i,30}$ and $ADf_{i,300}$, are created by

$$ADf_{i,30}(x, y) = \sum_{j=31}^i Df_{j,30}(x, y).$$

$$ADf_{i,300}(x, y) = \sum_{j=301}^i Df_{j,300}(x, y).$$

The accumulated difference at each pixel can be considered as the change degree at that position. Since SECOND digit changes every 30 frames and TEN-SECOND digit changes every 300 frames, two approximated areas of SECOND digit and TEN-SECOND digit can be located by observing $ADf_{i,30}$ and $ADf_{i,300}$ in scoreboard frame sequences. Based on positions and sizes of these two areas, a complete game

clock area is located.

Note that each game clock pattern consists of a separation mark (colon or dot). It is observed from the vertical projection histogram of the game clock area that the separation mark is located at the lowest local peak, and all clock digits are separated from each other by a local valley. Based on the information and the width of the detected SECOND digit area, each clock digit area can be located. An example is shown in Fig. 4.3.

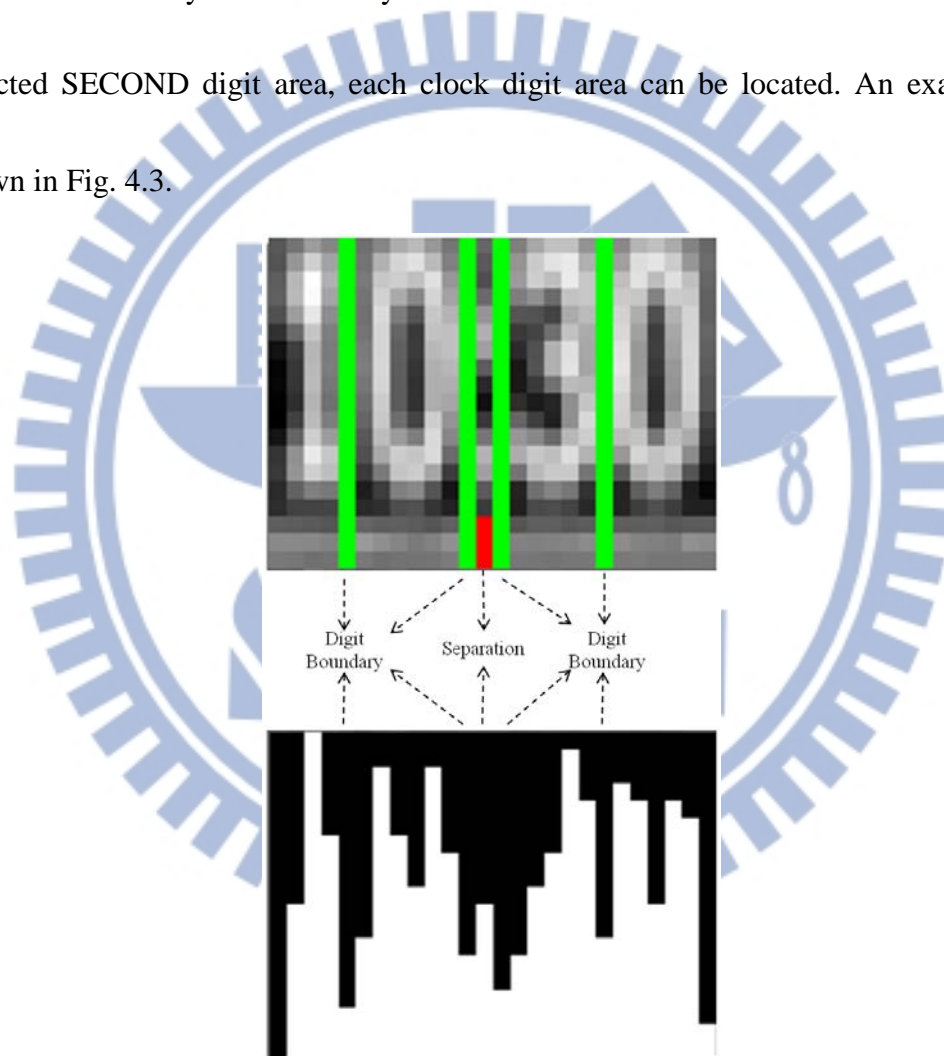


Fig. 4.3 An example of locating game clock digits (10:30).

4.2.2.2 Clock Digit Template Collection

After locating each clock digit area, a simple idea is proposed to collect clock

digit templates. Every time TEN-SECOND digit changes, it means SECOND digit area has been through a complete cycle from 9 to 0, i.e. 9, 8, 7, ..., 0. Therefore, once a pattern change of TEN-SECOND digit area is detected, a set of digit templates can be collected by sampling from SECOND digit area every second. After collecting eleven samples in consecutive eleven seconds as a candidate template set, a verification step is provided to examine whether the set is correct. For each candidate template set, if the first template is exactly the same as the eleventh one, then the candidate is considered as a correct digit templates set since all members in the set complete a cycle. Otherwise, keep collecting another candidate set until a correct one is verified.

4.2.2.3 Clock Digit Recognition

After locating each clock digit and collecting a correct digit templates set, a two-step strategy is proposed to recognize game clocks on the semi-transparent scoreboard.

First, for each clock digit, a local strategy is proposed to narrow the number of digit templates used in pattern matching. While applying template matching to recognize each digit, only the following three candidates should be considered: 1) current digit character; 2) next digit character; 3) possible digit character derived from

frame number difference. For example, if we try to update SECOND digit of “10:30”, the first candidate is “0” itself. The second candidate is “9”. Assume it has been 60 frames since the last recognized result of SECOND digit, and frame rate of video is 30. It is possible that two seconds has been collapsed since the last update, so the third candidate is derived to be “8”. Note that narrowing the number of candidates not only prevents possible errors from matching other digit templates, but also provides a mechanism to correct the recognition result in later frames.

After applying local strategy in pattern matching for each clock digit, a global strategy is proposed to verify the overall game clock recognition result. For example, if we recognize the clock time as $mn:st$, the overall recognition result is

$$T = (m \times 10 + n) \times 60 + s \times 10 + t.$$

For each recognition result of frame f_k , $T(k)$, and a new candidate result recognized from a later frame f_l , $T(l)$, the verification equation is defined by

$$T(k) = T(l) + \text{Round}((k - l) \div 30).$$

If the equation holds, the new candidate result is regarded as a right one, and the frame f_l is recognized as game clock $T(l)$. Note that the application here focuses on semantic event extraction, so recognition result for every single frame is not important. Instead, recognizing when a right game clock is updated is more valuable for text/video alignment and event annotation.

4.2.2.4 Text/Video Alignment

Based on the recognized game clock, a text/video alignment is presented to do sports video annotation. The alignment consists two parts. First, through the recognized game clock in video frames, the corresponding target frame of each event extracted from webcast text (see Chapter 3) is located, this is called event moment detection. Second, the time period for each event is determined, this is called event boundary detection.

As to event boundary detection, since many basketball events do not have obvious boundary patterns, even manually labeling event boundary is very subjective, the result may vary from person to person. In fact, a sports event should have redundancies before and after the event moment to explain the cause and result. Therefore, here, we can set a general interval for all kinds of basketball events. For example, ten seconds before the event moment to five seconds after the event moment. The basketball events extracted in the interval are treated as successful text/video alignment. An example of text/video alignment is presented in Fig. 4.4.



Fig. 4.4 An example of text/video alignment.

4.3 Experimental Results

Our experiments are conducted by 11 NBA 2008-2009 postseason games. The basketball videos are captured by TV card. The webcast text is acquired from ESPN website. After annotating all basketball videos, the experimental result is evaluated by watching them with human eyes. An event is detected as a hit if the manually generated event boundary is covered by the proposed method. As can be seen in Table 4.1, the detection rate of annotation result without video frames partition is horrible. On the contrary, the detection rate of the proposed method reaches 100%. The main reason is that video frames partition can prevent unnecessary game clock recognition from frames without a scoreboard and raise the digit recognition rate. This also solves the challenge of discontinuity in basketball games.

Table 4.1 Semantic events extraction results of the proposed method.

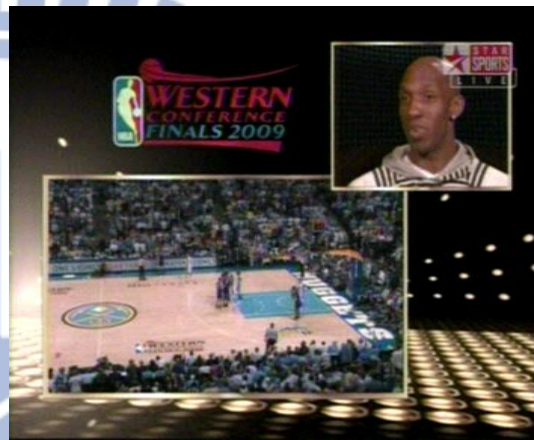
NBA 2009 postseason games	Annotation without video frames partition	Annotation with video frames partition
	Correctly detected number / Total event number (Detection rate)	
CLE vs. DET on Apr 25	91/150 (60.7%)	150/150 (100%)
ATL vs. MIA on May. 2	50/161 (31.1%)	161/161 (100%)
ATL vs. CLE on May. 8	61/159 (38.4%)	159/159 (100%)
LAL vs. HOU on May. 9	81/149 (54.4%)	149/149 (100%)
BOS vs. ORL on May. 9	57/169 (33.7%)	169/169 (100%)
DEN vs. LAL on May. 20	126/169 (74.6%)	169/169 (100%)
DEN vs. LAL on May. 22	144/181 (79.6%)	181/181 (100%)
LAL vs. DEN on May. 24	97/178 (54.5%)	178/178 (100%)
LAL vs. DEN on May. 26	132/179 (73.7%)	179/179 (100%)
DEN vs. LAL on May. 28	136/182 (74.7%)	182/182 (100%)
LAL vs. DEN on May. 30	69/213 (32.4%)	213/213 (100%)
Average	55.3%	100%

We check the results and find that the missing events are due to a special circumstance. As can be seen from Fig. 4.5, sometimes a basketball game is playing

but the game clock is not shown on video frames. For example, 1) late start after broadcasting TV commercials; 2) picture-in-picture interviewing a player; 3) some other statistic numbers; 4) trailers produced by the broadcasting company. Since no game clock is available for recognizing in these frames, the missing errors are acceptable.



(a) Late start of a quarter from 11:56 instead of 12:00.



(b) Picture-in-picture interviewing a player.



(c) Some other statistic numbers.



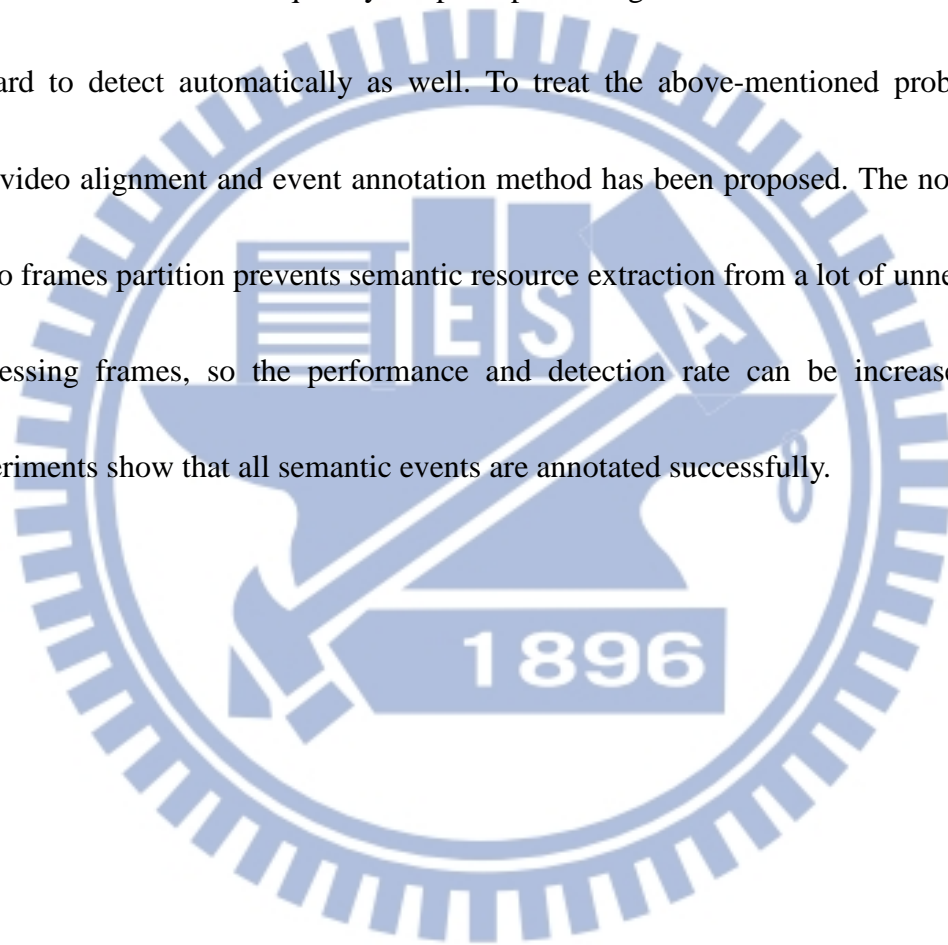
(d) Trailer of TV program.

Fig. 4.5 Examples of basketball games playing without game clock.

4.4 Summary

Using webcast text as external knowledge in multimodal fusion framework for

sports video annotation is a trend recently. Semantic text events are extracted in our previous work (see Chapter 3). The other challenging task is to annotate semantic events by game clock recognition and text/video alignment. Different from game clock recognition of soccer videos, game clock recognition of basketball videos is much harder due to the frequently temporal pause of game clock. The event boundary is hard to detect automatically as well. To treat the above-mentioned problems, a text/video alignment and event annotation method has been proposed. The novelty of video frames partition prevents semantic resource extraction from a lot of unnecessary processing frames, so the performance and detection rate can be increased. Our experiments show that all semantic events are annotated successfully.



CHAPTER 5

A NOVEL METHOD FOR SLOW MOTION REPLAY DETECTION IN BROADCAST BASKETBALL VIDEO

In this chapter, we will propose a method to detect slow motion replays in basketball videos. The existence of scoreboard is referred to filter large amount of non-replay frames, this improves detection accuracy. After video frames partition, every consecutive non-scoreboard frame sequence bounded by scoreboard frames are considered as a non-scoreboard segment. Characteristics of replays and non-replays are observed to create features, which can be used to detect replays and prune non-replays from non-scoreboard segments.

5.1 Introduction

Slow motion replays present detail processes of sports events, and they have been widely referred by professionals for athlete performance analysis, professional training, and injury prevention. Slow motion replays also provide resources for sports video analysis such as highlight generation [23], video summarization [24]-[26], and event detection [8]. Therefore, slow motion replay extraction has become a valuable and hot research topic.

Some methods [23][12][15]-[18] for slow motion replay detection have been proposed, they can be classified into two categories. The first category [23][12][15]

assumes that a replay is sandwiched by either two special digital video effects (SDVEs) or two logo transitions. Based on the assumption, Pan et al. [23] build a hidden Markov model (HMM) to detect slow motion replays. Some methods [12][15] first locate SDVEs or logo transitions, and then consider those segments sandwiched by SDVEs or logo transitions as slow motion replays. They either assume that the two SDVEs or logo transitions before and after a replay are identical [12] or visually similar [15]. However, these assumptions are not always true in basketball videos. In fact, production effects used in basketball videos are various and complicated. The beginning and end of a basketball replay have some combinations: 1) paired visually similar SDVEs; 2) non-paired SDVEs; 3) a SDVE in one end and an abrupt transition in the other. Furthermore, a basketball video segment bounded by paired SDVEs is not always a replay. So, previous work in this category cannot be applied to basketball videos.

The second category [16]-[18] analyzes features of replays to distinguish replay segments from non-replay segments. Farn et al. [16] extracted slow motion replays captured from both standard cameras and high speed cameras. The extractor refers to the dominate color of soccer field; however, it is not applicable in basketball videos since the size of basketball court is relatively smaller and its textures are more complicated. Wang et al. [17] conducted motion-related features and presented a

support vector machine (SVM) to classify slow motion replays and normal shots. The approach experimented on soccer and basketball videos. But the precision rates of two experimented basketball videos are 55.6% and 53.3% with recall rates 62.5% and 66.7%, respectively. Han et al. [18] proposed a general framework based on Bayesian network to make full use of multiple clues, including shot structure, gradual transition pattern, slow motion, and sports scene. Since they considered gradual transition as a feature clue, the method is suffered from the inaccuracy of their used automatic gradual transition detector. Their experiments performed improvements in replay detection with precision rate 82.9% and recall rate 83.2%, but the recall rate is still not high enough for sports highlight generation.

Basketball is one of the most important sports in the world, yet challenges of slow motion replay detection in basketball videos still remain. The first category methods are not applicable for basketball videos due to the improper assumption. The second category methods are applicable for basketball videos, but there is room for improvement in both precision rate and recall rate. Moreover, most previous researches analyze every video frame to detect replays, but detecting replays in video frames that are surely non-replay degrades both performance and detection rate.

5.2 Proposed Method

In this study, we propose a novel method to tackle above-mentioned challenges and detect slow motion replays in basketball videos. First, video frames partition proposed in Chapter 2 is referred to filter out video frames that have no chance of being replays. After filtering, video frames without scoreboard existence, called non-scoreboard frames, are grouped into several non-scoreboard segments. Then, characteristics of replays and non-replays are both observed to create features, where the former is for detecting replays and the latter is for pruning non-replays.

5.2.1 Video Frames Partition

As can be seen from Fig. 2.1, in basketball videos, all frames can be broadly classified into two categories, scoreboard frames and non-scoreboard frames. Scoreboard frames present basketball game with scoreboard overlaid on them, while non-scoreboard frames present the rest, e.g., sideline interview, slow motion replay, etc. Since scoreboard frames which are definitely non-replays usually occupy nearly half of a broadcast basketball video, it is beneficial to filter out scoreboard frames from detecting slow motion replays. So, an automatic scoreboard template extractor is needed. As shown in Fig. 2.1(a), a scoreboard is fixed rectangular area with pixels changing infrequently. Based on this fact, our previous work (see Chapter 2)

presented an automatic scoreboard template extractor. Here, we adapt this extractor to get the scoreboard template and position. After scoreboard template extraction, the video frames partition can be done by matching every frame with scoreboard template at the scoreboard position.

5.2.2 Feature Extraction and Replay Detection

After video frames partition, every consecutive non-scoreboard frame sequence bounded by scoreboard frames can be considered as a non-scoreboard segment. Characteristics of replays and non-replays are observed to create features, which are used to detect replays and prune non-replays.

According to our observation, there are three possible components in a non-scoreboard segment: 1) slow motion replay; 2) TV commercial; 3) game-related segment which shows game-related information with background around the court but is not a replay. Some game-related segment examples are given in Fig. 5.1.

Non-scoreboard segments can be classified into three classes by their time duration:

Short, medium, and long. The composition of each class is different from the others.

Short non-scoreboard segment (SNS), which is less than 6 seconds, is only caused by temporal disappearing of scoreboard for showing some game-related information.

Medium non-scoreboard segment (MNS), which is between 6 seconds and 30 seconds,

occurs in temporal pause of the game, and it is either game-related segment or slow motion replay. Long non-scoreboard segment (LNS), which is more than 30 seconds, occurs in long pause of the game, and it is a combination of TV commercial, half-time report, game-related segment with or without replay in it. To treat different compositions and characteristics of each non-scoreboard segment class, different strategies are proposed to detect slow motion replays.



Fig. 5.1 Examples of game-related segments.

A SNS can be always detected as a non-replay, since a slow motion replay is never less than 6 seconds. A MNS is either slow motion replay or game-related segment, so a binary classifier is given to classify it as a replay segment or a game-related segment. Replay detection for LNS is more complicated than that for MNS because of the existence of TV commercials. TV commercials are various, and they possibly contain features that are ambiguous with replay. Hence, each LNS is first cut into sub segments, then replays are detected from the sub segments. Note that our replay detection results of MNS are great that some of the information can be

referred to aid replay detection for LNS. More replay detection details for MNS and LNS are presented in the following paragraphs. The block diagram of slow motion replay detection is shown in Fig. 5.2.

For MNS, most histogram differences of neighboring frames in a game-related segment are similar. On the contrary, the histogram differences of neighboring frames in a replay segment are various (see Figs. 5.3(a)-5.3(c)). The variety of differences in a replay segment is caused by two reasons. One is the camera flash which appears frequently in replays (see Fig. 5.3(d)). The other is due to the smaller court of basketball videos, the background sometimes changes between in-court view and out-court view in a replay, and the histogram difference of two neighboring frame becomes larger (see Fig. 5.3(e)).

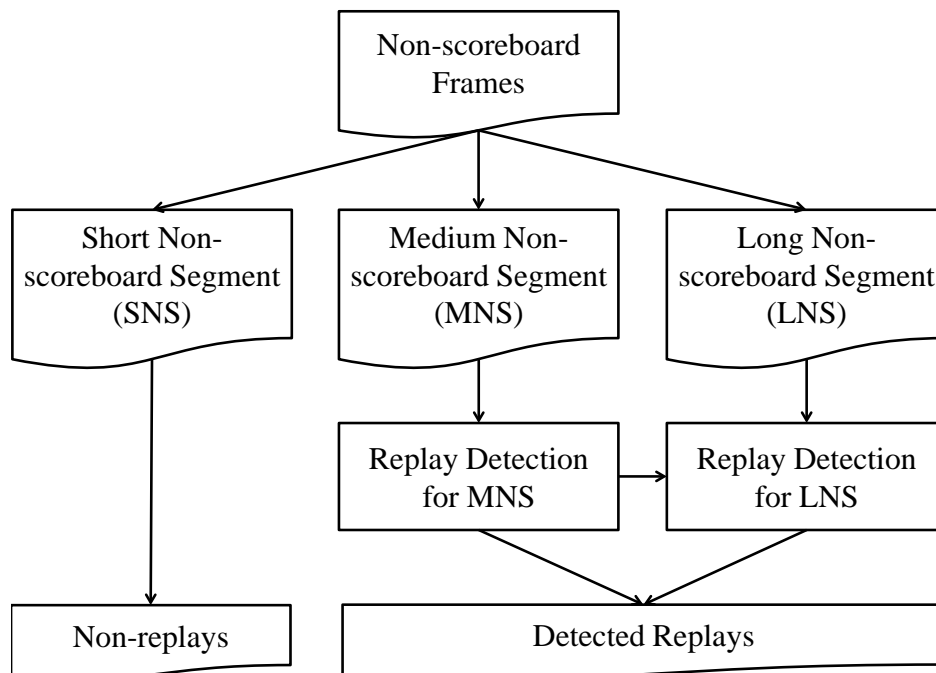
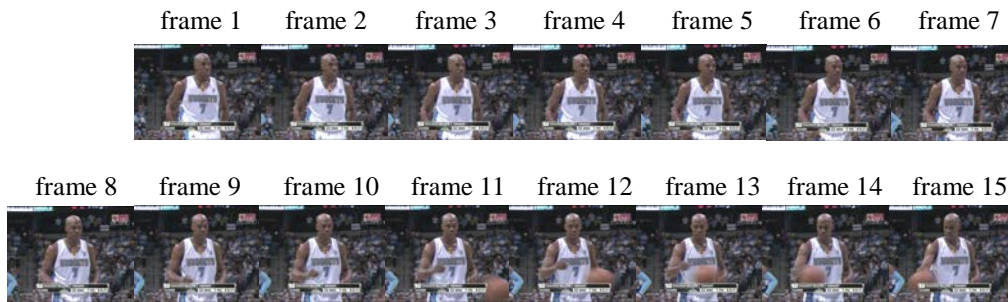
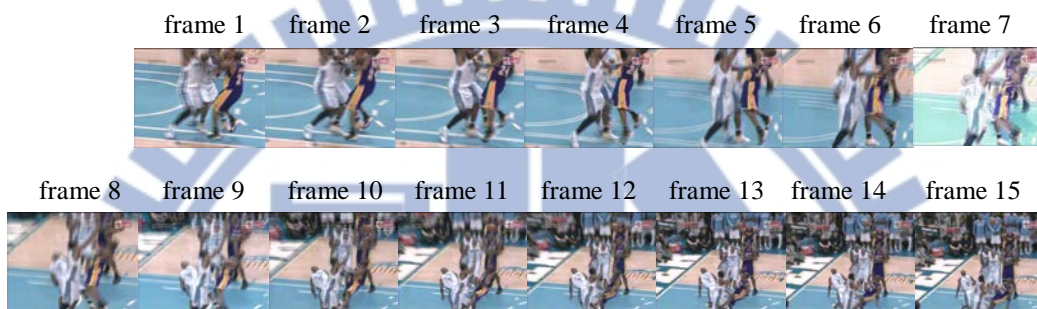


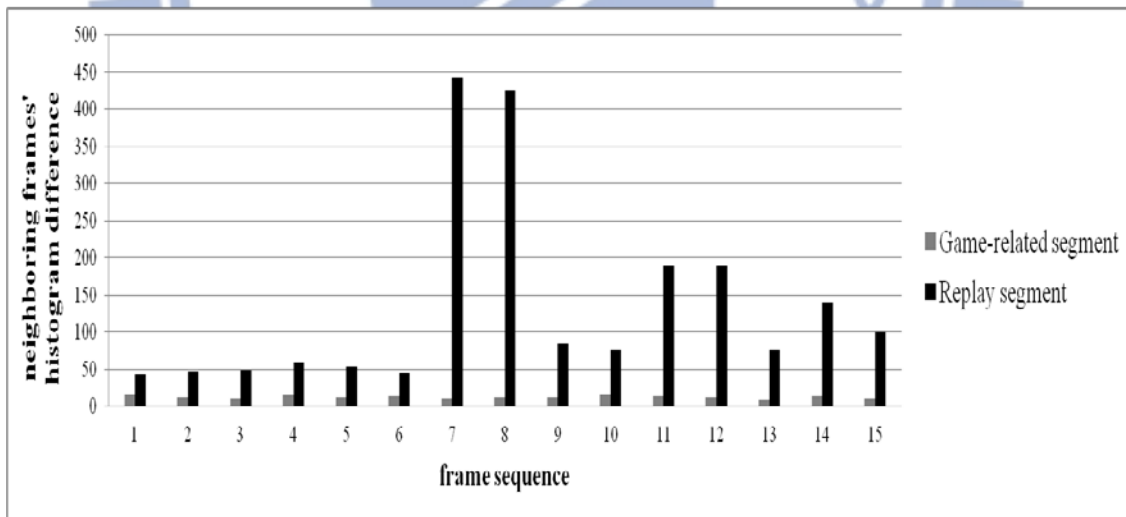
Fig. 5.2 Block diagram of slow motion replay detection.



(a) An example of game-related segment.

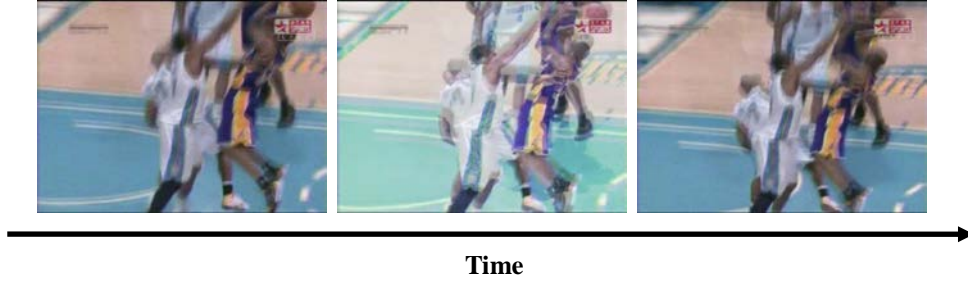


(b) An example of replay segment.



(c) Comparison between differences of neighboring frames in (a) and (b).

Fig. 5.3 An example of comparison between a game-related segment and a replay segment.



(d) Frames 6~8 of the replay segment in (b) with a camera flash.



(e) Frames 10~13 of the replay segment in (b) with background changing from in-court view to out-court view.

Fig. 5.3 An example of comparison between a game-related segment and a replay segment (continued).

Based on the different characteristics, a binary classifier is used. Given a non-scoreboard segment (NS) with the frame sequence $(nf_1, nf_2, \dots, nf_{K_n})$, K_n is the total frame number of NS. Let $(nh_1, nh_2, \dots, nh_{K_n})$ be the corresponding color histograms of K_n frames in NS. Two histogram-based frame differences are defined by

$$DH_1(nh_i) = \frac{1}{8 \times 8 \times 8} \sum_{r=0}^7 \sum_{g=0}^7 \sum_{b=0}^7 |nh_i(r, g, b) - nh_{i-1}(r, g, b)|,$$

$$DH_{15}(nh_i) = \frac{1}{8 \times 8 \times 8} \sum_{r=0}^7 \sum_{g=0}^7 \sum_{b=0}^7 |nh_i(r, g, b) - nh_{i-15}(r, g, b)|.$$

Note that here we quantize each of r , g , and b into 8 levels, and DH_1 is used to

measure the histogram difference of two neighboring frames, DH_{15} is for two frames with distance 15. After calculating $DH_1(nh_i)$ and $DH_{15}(nh_i)$ for each frame in NS, let $\sigma_{DH_1}(\text{NS})$ and $\sigma_{DH_{15}}(\text{NS})$ represent the standard deviations of sequences $DH_1(nh_i)$ and $DH_{15}(nh_i)$, respectively. Then these two standard deviations are considered as global variation features of a NS. The two global variation features of all MNSs are used in binary classifier to classify each MNS as a preliminary replay or a game-related segment. According to our preliminary experiments for MNS classification, the average precision rate of correctly classifying segments as replays in ten experimented basketball videos is 94% with average recall rate 100%. In order to explain the performance of the binary classifier, the two global features of each MNS in one of the experimented basketball videos are shown in Fig. 5.4. From this figure, we can see that replay segments and game-related segments can be well-separated based on these two global features.

Note that the misclassification is due to that few game-related segments consist of several still shots (see Fig. 5.5) with few near abrupt transitions and are misclassified as replays (i.e. false alarms). Since the differences of neighboring frames in a replay segment are always diverse, another variation feature can be used to prune this kind of misclassification.

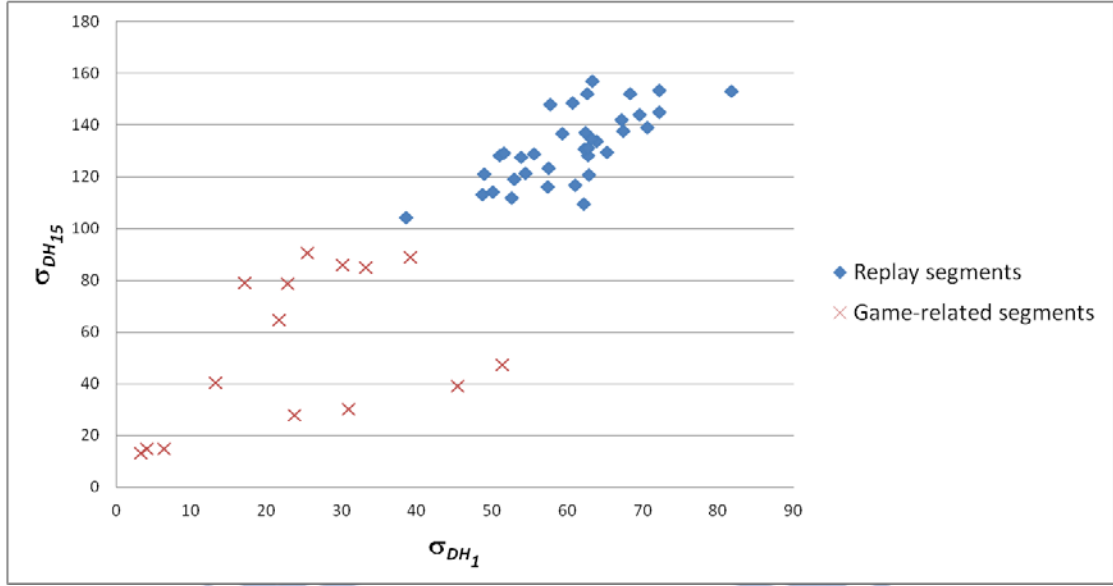


Fig. 5.4 The two global features of each MNS in a basketball video.

In order to decrease the effects of few near abrupt transitions in a game-related segment, a mean filter is used to skip larger neighboring frame differences, pixel-based difference of a neighboring frame pair is defined by

$$DF_1(nf_i) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} |nf_i(x, y) - nf_{i-1}(x, y)|, \quad 2 \leq i \leq K_n,$$

where M and N represent the width and height of a frame, and $nf_i(x, y)$ represents the color value of pixel (x, y) at frame nf_i . Let $\mu_{DF_1}(\text{NS})$ be the mean value of all $DF_1(nf_i)$ in the NS. And let $\sigma'_{DF_1}(\text{NS})$ represent the standard deviation of those $DF_1(nf_i)$ less than $\mu_{DF_1}(\text{NS})$, and it is considered as another variation feature of NS. Hence, for each MNS detected as a replay with small σ'_{DF_1} , it should be pruned. To determine the threshold automatically, a self-training mechanism is provided. For each MNS detected as a replay in ten experimented basketball games, σ'_{DF_1} is calculated and the

histogram of σ'_{DF_I} is established and shown in Fig. 5.6.

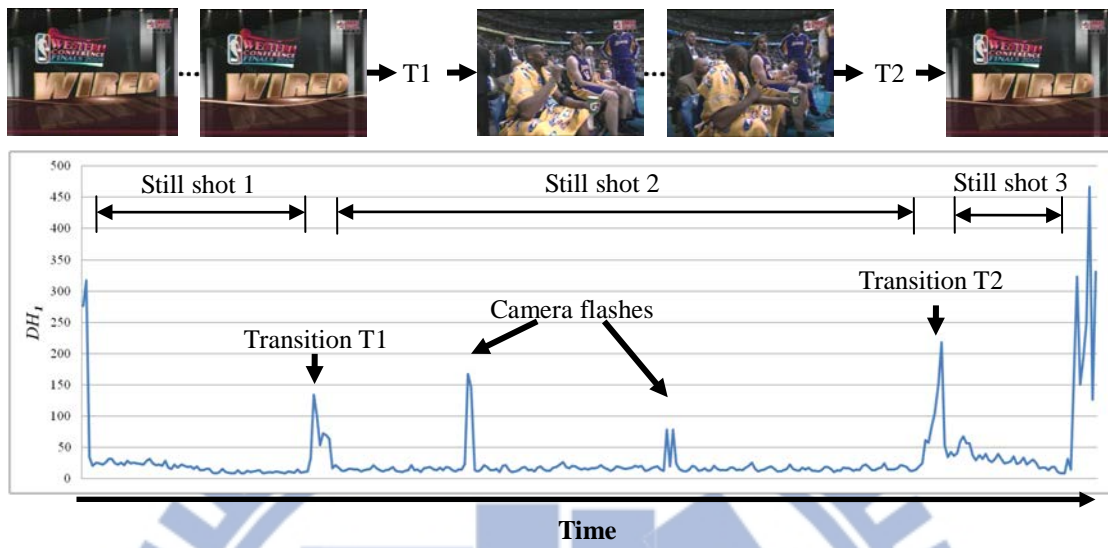


Fig. 5.5 An example of the DH_I sequence of a game-related segment misclassified as replay.

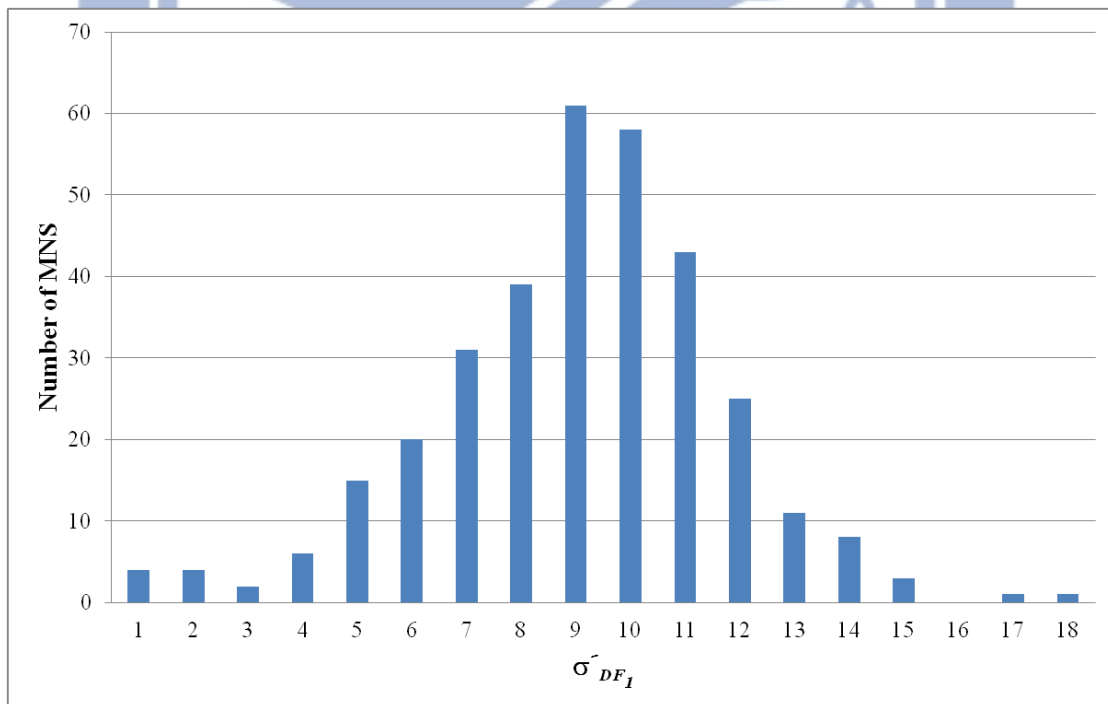


Fig. 5.6 Histogram of σ'_{DF_I} from the preliminary replays in ten experimented basketball videos.

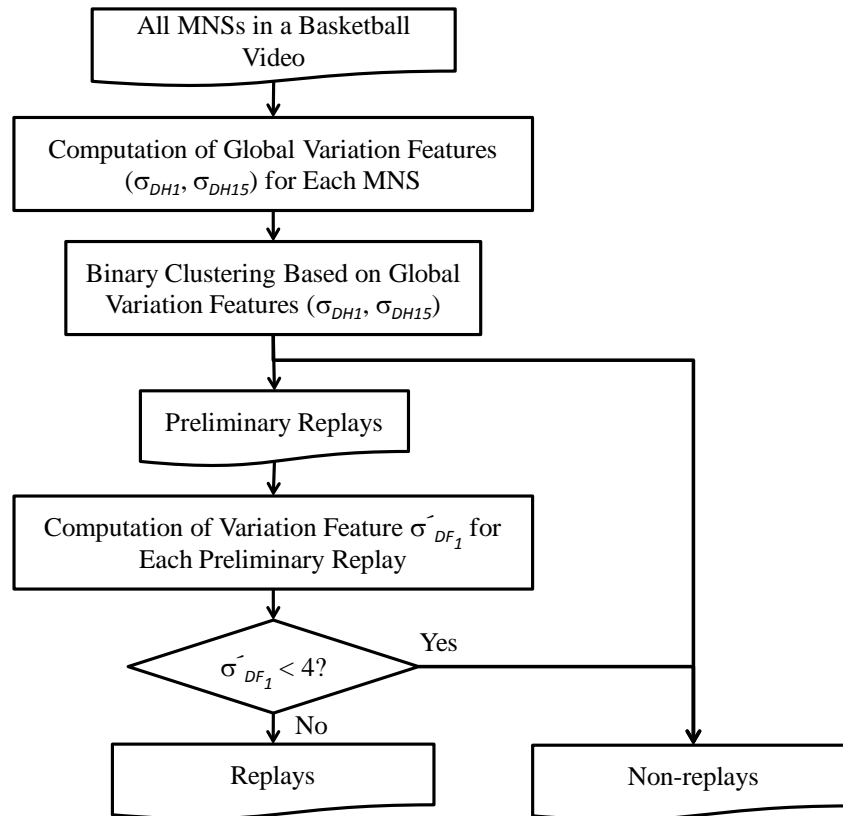


Fig. 5.7 Block diagram of replay detection for MNS.

As to LNS, the task becomes more complicated due to that the major portion of LNS is TV commercial, and the features of TV commercials can be ambiguous with replays. Since a LNS consists of thousands of frames, our strategy is to cut each LNS into several sub segments, called sub-LNSs. Then, instead of directly detecting replays from sub-LNS, detection results for MNS and characteristics of TV commercials are referred to build some pruning rules. After pruning non-replays, the rest sub-LNSs are considered as detected replays.

According to the structure of typical commercial block [27], TV commercials are

always grouped into blocks, and several monochrome black frames are inserted to separate each of them. Mostly, the last few seconds of each TV commercial consist of still shots of the product and slogan to impress viewers (see Fig. 5.8). However, there are neither monochrome black frames nor still shots in slow motion replays. So, each LNS can be cut by consecutive runs of low differences of neighboring frames without affecting the completeness of replays.

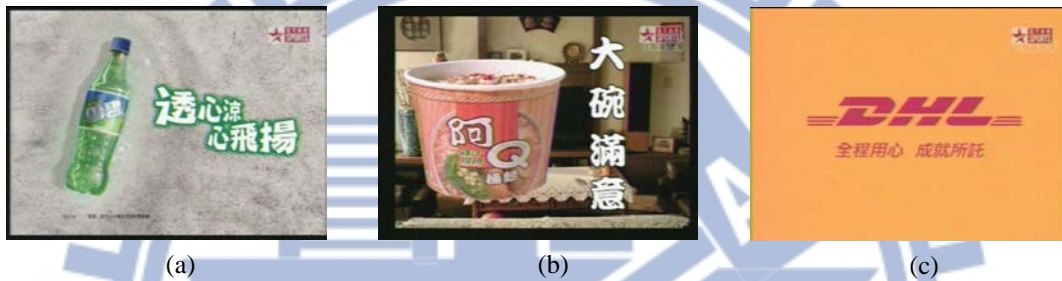


Fig. 5.8 Examples of still shots of the product and slogan in TV commercials.

Given the frame sequence of a LNS = $(lf_1, lf_2, \dots, lf_{K_l})$, where K_l is the total frame number of the LNS. Pixel-based difference of a neighboring frame pair $DF_1(lf_i)$ is calculated and recorded. Consecutive frames with $DF_1(lf_i)$ less than 20 are considered as a still run. All still runs with length more than 20 are used to cut the LNS into several sub-LNSs, each of which is bounded by a pair of still runs with length > 20 .

As mentioned earlier, a slow motion replay is never less than 6 seconds, so each sub-LNS less than 6 seconds can be detected as a non-replay. For those sub-LNSs more than 6 seconds, the binary classifier is not applicable because variation features of TV commercials can be ambiguous with replays. Since the detection results for

MNS are great, they can be considered as a pre-trained model to build some pruning rules. In the pre-trained model, let R-MNS and G-MNS represent the set of detected replay segments and the set of detected game-related segments, respectively. All frames of R-MNS are grouped as a replay frame sequence (*RFS*); likewise, all frames of G-MNS are grouped as a game-related frame sequence (*GFS*). The proposed pruning rules are given below.

Rule 1: Global Variation Pruning

Inspired by replay detection for MNS, the differences of neighboring frames in a replay segment are diverse. The observation is already proved by the great recall rate in the preliminary experiments. Based on the two global variation features used in binary classifier for MNS, the center feature vector ($C\sigma_{DH_1}(R-MNS)$, $C\sigma_{DH_{15}}(R-MNS)$) of R-MNS is denoted as

$$C\sigma_{DH_1}(R-MNS) = \frac{\sum_{NS \in R-MNS} \sigma_{DH_1}(NS)}{\text{Number of NS in R - MNS}}, \quad (5.1)$$

$$C\sigma_{DH_{15}}(R-MNS) = \frac{\sum_{NS \in R-MNS} \sigma_{DH_{15}}(NS)}{\text{Number of NS in R - MNS}}. \quad (5.2)$$

The radius of global variation features of R-MNS is defined by

$$r_{\text{variation}}(R-MNS) = \max_{NS \in R-MNS} \left\{ \sqrt{(\sigma_{DH_1}(NS) - C\sigma_{DH_1}(R-MNS))^2 + (\sigma_{DH_{15}}(NS) - C\sigma_{DH_{15}}(R-MNS))^2} \right\}. \quad (5.3)$$

For each sub-LNS, sLNS, its Euclidean distance from its two global variation features

$(\sigma_{DH_1}(sLNS), \sigma_{DH_{15}}(sLNS))$ to the center feature vector of R-MNS can be calculated

as

$$UD(sLNS) = \sqrt{(\sigma_{DH_1}(sLNS) - C\sigma_{DH_1}(R - MNS))^2 + (\sigma_{DH_{15}}(sLNS) - C\sigma_{DH_{15}}(R - MNS))^2}. \quad (5.4)$$

If $UD(sLNS) > r_{variation}(R-MNS)$, it is not similar to any of R-MNS. The dissimilarity

is caused by either too larger global variations or too small ones. It is more reasonable

to prune those dissimilar sub-LNS with small global variations. A threshold TH_1 is set

as

$$TH_1 = \min_{NS \in R-MNS} \{\sigma_{DH_1}(NS)\} \quad (5.5)$$

Hence, for each dissimilar sub-LNS with $\sigma_{DH_1}(sLNS) < TH_1$, it should be pruned as

non-replay.

Rule 2: Color Pruning

The color distribution of a TV commercial is various; however, the color

distribution of *RFS* or *GFS* is more related to game itself. So a sub-LNS should be

pruned if its color distribution is neither similar to that of *RFS* nor similar to that of

GFS.

Given $RFS = (rf_1, rf_2, \dots, rf_{K_r})$ and $GFS = (gf_1, gf_2, \dots, gf_{K_g})$, where K_r is the total

frame number in *RFS* and K_g is the total frame number in *GFS*. Let $RHS = (rh_1,$

$rh_2, \dots, rh_{K_r})$ be the corresponding quantized histogram sequence calculated from *RFS*

and $GHS = (gh_1, gh_2, \dots, gh_{K_g})$ be the corresponding quantized histogram sequence calculated from GFS . Then, mean color histograms of RHS and GHS can be calculated by

$$\mu_{RHS} = \frac{1}{K_r} \sum_{i=1}^{K_r} rh_i, \quad (5.6)$$

$$\mu_{GHS} = \frac{1}{K_g} \sum_{i=1}^{K_g} gh_i. \quad (5.7)$$

The maximum differences are considered as the radiuses of RHS and GHS and calculated by

$$r_{RHS} = \max_{1 \leq i \leq K_r} \left\{ \frac{1}{8 \times 8 \times 8} \sum_{r=0}^7 \sum_{g=0}^7 \sum_{b=0}^7 |rh_i(r, g, b) - \mu_{RHS}(r, g, b)| \right\}, \quad (5.8)$$

$$r_{GHS} = \max_{1 \leq i \leq K_g} \left\{ \frac{1}{8 \times 8 \times 8} \sum_{r=0}^7 \sum_{g=0}^7 \sum_{b=0}^7 |gh_i(r, g, b) - \mu_{GHS}(r, g, b)| \right\}. \quad (5.9)$$

Given a sub-LNS, sLNS, with frame sequence $= (slf_1, slf_2, \dots, slf_{K_s})$, where K_s is the total frame number of sLNS. Let $(slh_1, slh_2, \dots, slh_{K_s})$ be the corresponding quantized histogram sequence calculated from the sub-LNS. Mean color histogram of the sLNS is defined by

$$\mu_H(sLNS) = \frac{1}{K_s} \sum_{i=1}^{K_s} slh_i. \quad (5.10)$$

Hence, for the mean color histogram of each sub-LNS, if its distance from μ_{RHS} is larger than r_{RHS} and its distance from μ_{GHS} is larger than r_{GHS} , the sub-LNS is considered as a non-replay.

Rule 3: Smoothness Pruning

In a slow motion replay, the pixel-based difference of each neighboring frame pair is usually larger to show the details of a sports event. However, a non-replay is normally much smoother to fulfill requirements of human visual perception. So, for each sub-LNS, if most differences of neighboring frame pairs are smoother than those of a replay, it should be pruned.

Let $RFS = (rf_1, rf_2, \dots, rf_{K_r})$, where K_r is the total frame number in RFS . Mean pixel-based neighboring frame difference is defined by

$$\mu_{DF_1}(RFS) = \frac{1}{K_r - 1} \sum_{i=2}^{K_r} DF_1(rf_i), \quad (5.11)$$

where $DF_1(rf_i)$ is the pixel-based difference of two neighboring frames rf_i and rf_{i-1} , and it is already calculated by formula (6) in earlier process (i.e. replay detection for MNS). Given a sub-LNS, sLNS, with frame sequence $= (slf_1, slf_2, \dots, slf_{K_s})$, where K_s is the total frame number of sLNS. The pixel-based frame difference $DF_1(slf_i)$ of two neighboring frames slf_i and slf_{i-1} is already calculated as well (i.e. sub-LNS cutting for LNS). For each sLNS, the smoothness feature is defined by

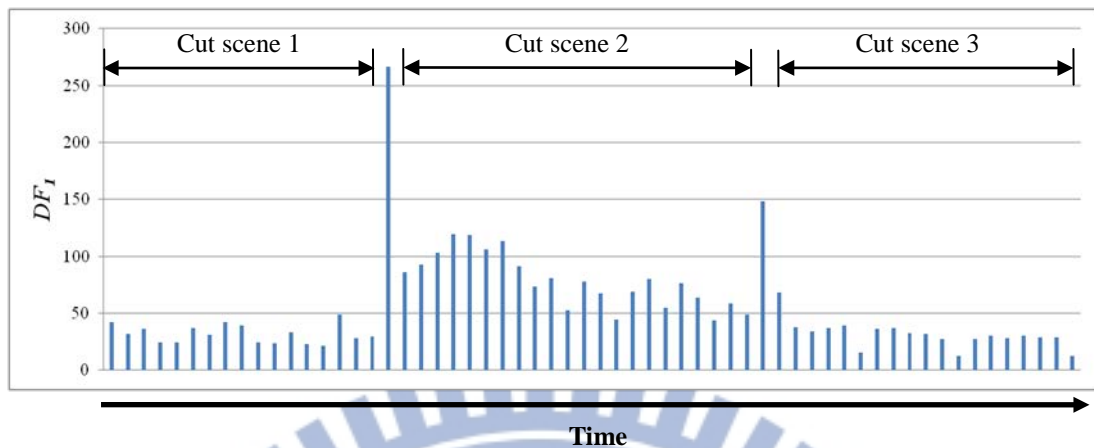
$$smoothness(sLNS) = \frac{1}{K_s - 1} \left\| \{slf_i \mid DF_1(slf_i) < \mu_{DF_1}(RFS)\} \right\|, \quad 2 \leq i \leq K_s. \quad (5.12)$$

If the smoothness feature is larger than a threshold, $TH_{smoothness}$, the sub-LNS is detected as a non-replay.

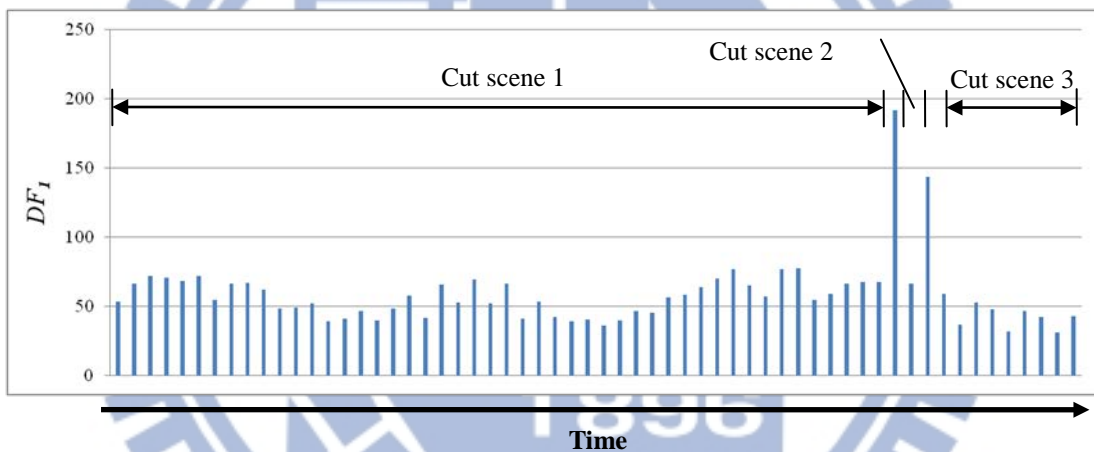
Rule 4: Scene Length Variation Pruning

From the structure of TV commercials [27], each individual TV commercial consists of many story scenes with abrupt scene changes. On the contrary, a replay always happens in the same scene, and there are rare abrupt scene changes in a replay. Note that some unexpectedly camera flashes may appear in replays to challenge the detection of abrupt transition, so the number of abrupt transitions is not a good pruning feature. Here, the length of each scene cut by abrupt transitions is observed instead. The length of each cut scene in a replay is various because the camera flashes appear unexpectedly; however, that in a commercial is relatively stable to show a story. Hence, for each sub-LNS, if lengths of its cut scenes are relatively stable, it should be considered as a commercial, i.e. non-replay.

Here, an abrupt transition detection method for each sub-LNS is provided by finding every frame slf_i with local maximum difference $DF_I(slf_i)$ which is two times larger than that of one of its neighboring frames, i.e., $DF_I(slf_i) > 2DF_I(slf_{i-1})$ or $DF_I(slf_i) > 2DF_I(slf_{i+1})$. Some examples of abrupt transition detection results are given in Fig.5.9. After abrupt transition detection, the scene length variation feature for each sub-LNS is defined by the standard deviation of the lengths of the cut scenes. If the scene length variation feature is less than a threshold, TH_{slv} , the sub-LNS is detected as a non-replay.



(a) Non-replay.



(b) Replay.

Fig. 5.9 Examples of abrupt transition detection results and the corresponding cut scenes of non-replay and replay.

After pruning non-replays by the four proposed rules, the rest sub-LNSs are considered as detected slow motion replays. The detail implementation steps of replay detection for LNS are given below.

Step0: Initialization: Given R-MNS, RFS , GFS , and all LNSs in a basketball game.

Cut each LNS into sub-LNSs. Consider sub-LNSs less than 6 seconds as non-replays and pass those more than 6 seconds to Step1.

Step1: // Global Variation Pruning Using Rule 1

For R-MNS

Calculate $(C\sigma_{DHI}(\text{R-MNS}), C\sigma_{DHI5}(\text{R-MNS}), r_{variation}(\text{R-MNS}),$ and TH_1 by formulas (5.1), (5.2), (5.3), and (5.5).

In LNS, for each unprocessed sub-LNS, sLNS

Calculate $UD(\text{sLNS})$ by formula (5.4).

if $(UD(\text{sLNS}) > r_{variation}(\text{R-MNS})$ and $\sigma_{DHI}(\text{sLNS}) < TH_1)$

Consider sLNS as a non-replay

else

Consider sLNS as a potential replay

end

Step2: // Color Pruning Using Rule 2

For *RFS* and *GFS*

Calculate $\mu_{RHS}, \mu_{GHS}, r_{RHS},$ and r_{GHS} by formulas (5.6), (5.7), (5.8), and (5.9).

In LNS, for each potential replay sLNS

Calculate $\mu_H(\text{sLNS})$ by formula (5.10).

if ($\|\mu_H(\text{sLNS}) - \mu_{RHS}\| > r_{RHS}$ and $\|\mu_H(\text{sLNS}) - \mu_{GHS}\| > r_{GHS}$)

Reconsider sLNS as a non-replay

end

Step3: // Smoothness Pruning Using Rule 3

For RFS

Calculate $\mu_{DF_1}(RFS)$ by formula (5.11).

In LNS, for each potential replay sLNS

Calculate $smoothness(\text{sLNS})$ by formula (5.12).

if ($smoothness(\text{sLNS}) > TH_{smoothness}$)

Reconsider sLNS as a non-replay

end

Step4: // Scene Length Variation Pruning Using Rule 4

In LNS, for each potential replay sLNS

Calculate scene length variation feature of sLNS.

if (scene length variation feature of sLNS $< TH_{slv}$)

Reconsider sLNS as a non-replay

end

Note that each rule corresponds to a specific feature for pruning non-replays. Since

our goal is to prune non-replays and to keep replays, changing the order of the four rules, i.e., Step1-Step4, comes out the same results.

5.3 Experimental Results

Our experiments are conducted by 10 NBA basketball games from 3 different broadcasters, i.e., ESPN, TNT, NBA TV. The length of each game match is about 150 minutes with TV commercials included. The data are recorded from TV in MPEG-2 format with resolution 480×352.

In slow motion replay detection, there are two kinds of resource videos. One is the commercial-free sports video which is only available for professional staff from the production room. The other is the broadcasted version with a lot of TV commercials, and it is more available for general audiences. From our experimental results, the proposed replay detection method can fulfill both kinds of potential users.

Table 5.1 and Table 5.2 show the replay detection results for MNS. In Table 5.1, some game-related segments (e.g., player's statistic, game series information, player's online comments) are misclassified into replays, the reason is that they have several still shots with few near abrupt transitions. This will increase the global variation features. It can be seen from Table 5.2, with the proposed automatic self pruning, the precision rate can be raised from 94% to 97%. The rare false alarms are acceptable

because they are all game-related video segments.

As to experimental comparison with replay detection methods in the first category, here, we assume that all methods in the first category can extract all segments sandwiched by paired SDEVs correctly, and consider the extracted segments as slow motion replays. Since some extracted segments sandwiched by paired SDEVs are not slow motion replays, only those real slow motion replay segments are considered as correctly detected ones. The results are shown in Table 5.3.

Table 5.1 Replay detection results for MNS.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	44	45	98%	44	100%
LAL-DEN (30/05/09)	33	36	92%	33	100%
BOS-ORL (09/05/09)	22	31	71%	22	100%
ORL-CLE (29/05/09)	44	46	96%	44	100%
CLE-ORL (31/05/09)	23	23	100%	23	100%
ORL-BOS (18/05/09)	26	27	96%	26	100%
LAL-ORL (15/06/09)	22	25	88%	22	100%
CLE-ATL (10/05/09)	23	23	100%	23	100%
HOU-LAL (18/05/09)	27	27	100%	27	100%
LAL-DEN (23/05/09)	48	49	98%	48	100%
Total	312	332	94%	312	100%

Table 5.2 Replay detection results for MNS with self pruning.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	44	44	100%	44	100%
LAL-DEN (30/05/09)	33	35	94%	33	100%
BOS-ORL (09/05/09)	22	24	92%	22	100%
ORL-CLE (29/05/09)	43	44	98%	44	98%
CLE-ORL (31/05/09)	23	23	100%	23	100%
ORL-BOS (18/05/09)	26	26	100%	26	100%
LAL-ORL (15/06/09)	22	24	92%	22	100%
CLE-ATL (10/05/09)	21	21	100%	23	91%
HOU-LAL (18/05/09)	27	27	100%	27	100%
LAL-DEN (23/05/09)	48	49	98%	48	100%
Total	309	317	97%	312	99%

Table 5.3 Replay detection results for MNS by methods in the first category.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	44	52	85%	44	100%
LAL-DEN (30/05/09)	33	39	85%	33	100%
BOS-ORL (09/05/09)	22	38	58%	22	100%
ORL-CLE (29/05/09)	43	44	96%	44	98%
CLE-ORL (31/05/09)	23	27	85%	23	100%
ORL-BOS (18/05/09)	26	28	93%	26	100%
LAL-ORL (15/06/09)	21	25	84%	22	95%
CLE-ATL (10/05/09)	23	23	100%	23	100%
HOU-LAL (18/05/09)	27	27	100%	27	100%
LAL-DEN (23/05/09)	48	55	87%	48	100%
Total	310	358	87%	312	99%

It can be seen from Table 5.3, the precision rate is decreased due to many game-related segments with paired SDEVs, e.g., player's statistics, game series information, sideline clips during timeout. Accordingly, the precision of our method is better. Since MNS is the only possible non-scoreboard segment in commercial-free sports videos, the proposed replay detector for MNS is applicable for commercial-free resources.

In our approach, $TH_{smoothness}$ and TH_{slv} have to be preset. As to $TH_{smoothness}$, a larger threshold means that the condition to prune non-replay is stricter. So, the precision rate is decreased while the recall rate is increased. On the other hand, a smaller threshold means that the condition is looser, so the precision rate is increased while the recall rate is decreased. This exactly illustrates the tradeoff phenomenon. TH_{slv} also has the tradeoff phenomenon for a similar reason. To show the tradeoff phenomenon, results of fixed $TH_{slv} = 30$ with various $TH_{smoothness}$ are shown in Table 5.4. Results of fixed $TH_{smoothness} = 85\%$ with various TH_{slv} are shown in Table 5.5 as well. By observing the trends in Table 5.4 and Table 5.5, two pairs of thresholds, $(TH_{smoothness} = 85\%, TH_{slv} = 25)$ and $(TH_{smoothness} = 85\%, TH_{slv} = 30)$, are chosen in our experiments. The results are presented in Table 5.6 and Table 5.7.

Table 5.4 Total replay detection results with fixed $TH_{slv} = 30$.

TH_{slv}	$TH_{smoothness}$	Precision	Recall
30	75%	90%	90%
	80%	89%	92%
	85%	88%	94%
	90%	86%	94%

Table 5.5 Total replay detection results with fixed $TH_{smoothness} = 85\%$.

$TH_{smoothness}$	TH_{slv}	Precision	Recall
85%	15	78%	97%
	20	83%	96%
	25	87%	96%
	30	88%	94%
	35	89%	91%

Table 5.6 and Table 5.7 present total replay detection results by combining results for MNS and LNS. As can be seen from Table 5.7, the use of a stricter threshold can raise precision rate from 87% to 88% with 2% degradation of recall rate. Since one of the most important goals of replay detection is for highlight generation, the high recall rates in both results show the great performance. As compared with previous researches for basketball videos [17]-[18], our method presents the superior performance.

We also compare our approach with methods in the first category. The results of methods in the first category are shown in Table 5.8. From this table, we can see that the recall rate (69%) are worse than ours (94%). The reason is that methods in the first category assume that all replays are sandwiched by paired SDVEs, this will cause

detection missing due to that replays are not always sandwiched by paired SDVEs.

Accordingly, as compared with previous researches [12][15], our method is superior.

Moreover, our precision rate and recall rate are also higher than those methods in the

second category [17]-[18].

Table 5.6 Total replay detection results with $TH_{smoothness}=0.85$ and $TH_{slv} = 25$.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	65	66	98%	66	98%
LAL-DEN (30/05/09)	47	56	84%	49	96%
BOS-ORL (09/05/09)	42	59	71%	44	95%
ORL-CLE (29/05/09)	61	70	87%	62	98%
CLE-ORL (31/05/09)	45	55	82%	47	96%
ORL-BOS (18/05/09)	43	47	91%	47	91%
LAL-ORL (15/06/09)	33	42	79%	34	97%
CLE-ATL (10/05/09)	26	26	100%	29	90%
HOU-LAL (18/05/09)	35	37	95%	36	97%
LAL-DEN (23/05/09)	66	77	86%	68	97%
Total	463	535	87%	482	96%

Table 5.7 Total replay detection results with $TH_{smoothness}=0.85$ and $TH_{slv} = 30$.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	62	63	98%	66	94%
LAL-DEN (30/05/09)	47	56	84%	49	96%
BOS-ORL (09/05/09)	40	54	74%	44	91%
ORL-CLE (29/05/09)	60	69	87%	62	97%
CLE-ORL (31/05/09)	43	48	90%	47	91%
ORL-BOS (18/05/09)	42	46	91%	47	89%
LAL-ORL (15/06/09)	32	41	78%	34	94%
CLE-ATL (10/05/09)	26	26	100%	29	90%
HOU-LAL (18/05/09)	34	36	94%	36	94%
LAL-DEN (23/05/09)	65	73	89%	68	96%
Total	451	512	88%	482	94%

Table 5.8 Total replay detection results by methods in the first category.

Match	Correctly Detected	Total Detected	Precision	Total Replays	Recall
DEN-LAL (28/05/09)	46	54	85%	66	70%
LAL-DEN (30/05/09)	36	42	86%	49	73%
BOS-ORL (09/05/09)	25	41	61%	44	57%
ORL-CLE (29/05/09)	48	49	98%	62	77%
CLE-ORL (31/05/09)	25	29	86%	47	53%
ORL-BOS (18/05/09)	26	28	93%	47	55%
LAL-ORL (15/06/09)	25	29	86%	34	74%
CLE-ATL (10/05/09)	23	23	100%	29	79%
HOU-LAL (18/05/09)	27	27	100%	36	75%
LAL-DEN (23/05/09)	53	60	88%	68	78%
Total	334	382	87%	482	69%

5.4 Summary

In this chapter, we have proposed a novel idea to detect slow motion replays in basketball videos. Video frames partition is referred to filter large amount of non-replay frames, which improves a lot of performance in both time complexity and detection accuracy. After filtering, a slow motion replay detector is proposed to detect replays from MNS and LNS with different strategies. From our results, the proposed replay detection method is applicable for both kinds of basketball videos whether TV commercials are contained or not. As compared with rarely discussed researches for basketball videos, our method shows the superior performance and great novelty.



CHAPTER 6

CONCLUSIONS AND FUTURE WORKS

In this dissertation, a novel framework for sports video analysis, which provides flexibility to combine different schemes of event extraction and those of replay detection, is proposed. Two semantic resource extraction schemes are introduced and incorporated in our framework to tackle challenges of sports video analysis. The novelty of video frames partition prevents semantic resource extraction from a lot of unnecessary processing frames, so the performance and detection rate can be increased. The framework is also capable of acquiring both two valuable semantic resources in one time. Conclusions of each proposed method and suggestions for future researches are given as follows.

In Chapter 3, we have proposed an unsupervised approach for semantic event extraction from sports webcast text and made some contributions: 1) detecting semantic events from webcast text in an unsupervised manner; 2) requiring no additional context information analysis; 3) preserving more significant events in sports games; 4) extracting multiple keywords from event categories to support hierarchical searching; 5) providing auto-complete feature for finer retrieval. The proposed method extracts significant semantic events from basketball and soccer games and preserves those events that are ignored or misclassified by previous work. The extracted significant text events can be used for further video indexing and

summarization. Future studies may be directed to extend our approach to other free-styled webcast text.

In Chapter 4, we have used webcast text as external knowledge in multimodal fusion scheme for sports video annotation. A game clock recognition method is proposed to tackle the problem of recognizing discontinuous game clock of basketball videos. The novelty of video frames partition prevents our method from a lot of unnecessary processing frames. Finally, all semantic events are annotated successfully.

In Chapter 5, we have proposed a method to detect slow motion replays in basketball videos. video frames partition is referred to filter large amount of non-replay frames. After filtering, a slow motion replay detector is proposed to detect replays from MNS and LNS with different strategies. The proposed replay detection method is applicable for both kinds of basketball videos whether TV commercials are contained or not.

Since no basketball specific feature is used in our methods, future studies may be directed to push forward the proposed sports video analysis framework to other sports videos.

REFERENCES

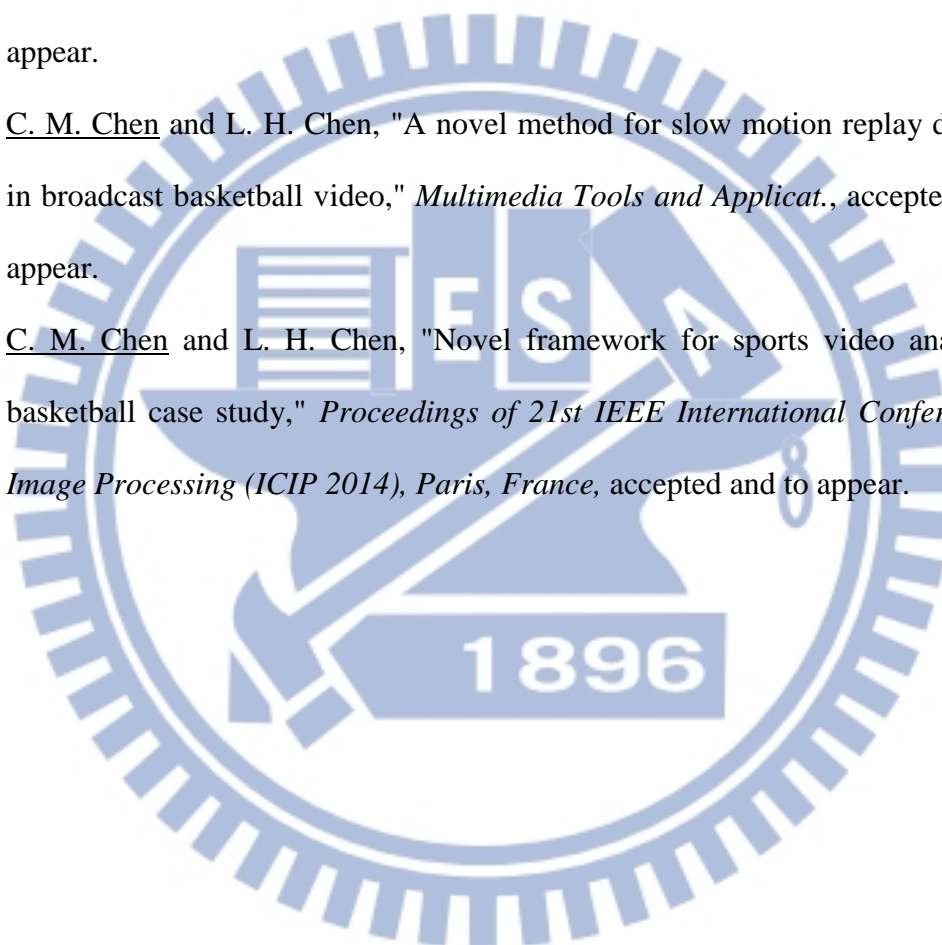
- [1] Y. H. Chen and L. Y. Deng, "Event mining and indexing in basketball video," in *Int. Conf. on Genetic and Evolutionary Computing (ICGEC)*, pp. 247-251, Aug. 29-Sep. 1, 2011.
- [2] E. Hassan, S. Chaudhury, M. Gopal, and V. Garg, "A hybrid framework for event detection using multi-modal features," in *Int. Conf. on Computer Vision Workshops (ICCV Workshops)*, pp. 1510-1515, 2011.
- [3] H. G. Kim and J. H. Lee, "Indexing of player events using multimodal cues in golf videos," in *IEEE Int. Conf. on Multimedia and Expo (ICME)*, pp. 1-6, 2011.
- [4] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, "Personalized abstraction of broadcasted American football video by highlight selection," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 575-586, 2004.
- [5] H. Xu and T. Chua, "Fusion of audio-visual features and external knowledge for event detection in team sports video," in *Proc. Workshop on Multimedia Information Retrieval (MIR '04)*, pp. 127-134, 2004.
- [6] C. Xu, J. Wang, K. Wan, Y. Li, and L. Duan, "Live sports event detection based on broadcast video and web-casting text," in *Proc. ACM Int. Conf. on Multimedia (MM '06)*, pp. 221-230, 2006.
- [7] C. Xu, J. Wang, H. Lu, and Y. Zhang, "A novel framework for semantic annotation and personalized retrieval of sports video," *IEEE Trans. Multimedia*, vol. 10, no. 3, pp.421-436, 2008.
- [8] C. Xu, Y. Zhang, G. Zhu, Y. Rui, H. Lu, and Q. Huang, "Using webcast text for semantic event detection in broadcast sports video," *IEEE Trans. Multimedia*, vol. 10, no. 7, pp. 1342-1355, 2008.
- [9] P. Lin, S. Li, T. Tsai, and Y. Tsai, "Tagging webcast text in baseball videos by video segmentation and text alignment," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 7, pp. 999-1013, 2012.
- [10] H. Pan, B. Li, and M. I. Sezan, "Automatic detection of replay segments in broadcast sports programs by detection of logos in scene transitions," in *Proc. ICASSP'02*, pp. 3385-3388, 2002.
- [11] L. Y. Duan, M. Xu, Q. Tian, and C. S. Xu, "Mean shift based video segment representation and applications to replay detection," in *Proc. ICASSP'04*, pp. 709-712, 2004.
- [12] Q. Huang, J. M. Hu, W. Hu, T. Wang, H. L. Bai, and Y. M. Zhang, "A reliable logo and replay detector for sports video," in *Proc. ICME'07*, pp. 1695-1698, 2007.

- [13] N. Nguyen and A. Yoshitaka, "Shot type and replay detection for soccer video parsing," in *Proc. ISM'12*, pp. 344-347, 2012.
- [14] X. L. Zhang and M. Zhi, "Slow motion replay detection of tennis video based on color auto-correlogram," in *Proc. ICDIP'12*, pp. 83341C, 2012.
- [15] F. Zhao, Y. Dong, Z. Wei, and H. L. Wang, "Matching logos for slow motion replay detection in broadcast sports video," in *Proc. ICASSP'12*, pp. 1409-1412, 2012.
- [16] E. J. Farn, L. H. Chen, and J. H. Liou, "A new slow-motion replay extractor for soccer game videos," *International Journal of Pattern Recognition and Artificial Intelligence*, vol.17, no. 8, pp.1467-1481, 2003.
- [17] L. Wang, X. Liu, S. Lin, G. Xu, and H. Y. Shum, "Generic slow-motion replay detection in sports video," in *Proc. ICIP'04*, pp. 1585-1588, 2004.
- [18] B. Han, Y. Yan, Z. H. Chen, C. Liu, and W. G. Wu, "A general framework for automatic on-line replay detection in sports video," in *Proc. MM'09*, pp. 501-504, 2009.
- [19] N. Nitta, N. Babaguchi, and T. Kitahashi, "Generating semantic descriptions of broadcasted sports video based on structure of sports games and TV programs," *Multimedia Tools and Applicat.*, vol. 25, no. 1, pp. 59-83, Jan. 2005.
- [20] C. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, pp. 27-28, 2008.
- [21] [Online]. Available: <http://armandbrahaj.blog.al/2009/04/14/list-of-english-stop-words/>
- [22] [Online]. Available: <http://www.textfixer.com/resources/common-english-words.txt>
- [23] H. Pan, P. van Beek, and M. I. Sezan, "Detection of slow motion replay segments in sports video for highlights generation," in *Proc. ICASSP'01*, pp. 1649-1652, 2001.
- [24] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Trans. Image Processing*, vol. 12, no. 7, pp. 796-807, 2003.
- [25] F. Chen and C. De Vleeschouwer, "Automatic production of personalized basketball video summaries from multi-sensored data," in *Proc. ICIP'10*, pp. 565-568, 2010.
- [26] F. Chen and C. De Vleeschouwer, "Formulating team-sport video summarization as a resource allocation problem," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 193-205, 2011.
- [27] B. Satterwhite and O. Marques, "Automatic detection of TV commercials," *IEEE Potentials*, vol. 23, no. 2, pp. 9-12, 2004.

PUBLICATION LIST

We summarize the publication status of the proposed methods and our research status in the following.

1. C. M. Chen and L. H. Chen, "A novel approach for semantic event extraction from sports webcast text," *Multimedia Tools and Applicat.*, accepted and to appear.
2. C. M. Chen and L. H. Chen, "A novel method for slow motion replay detection in broadcast basketball video," *Multimedia Tools and Applicat.*, accepted and to appear.
3. C. M. Chen and L. H. Chen, "Novel framework for sports video analysis: a basketball case study," *Proceedings of 21st IEEE International Conference on Image Processing (ICIP 2014), Paris, France*, accepted and to appear.



VITA

Chun-Min Chen was born in Taipei, Taiwan, Republic of China on September 19, 1983. He received the B.S. degree in Computer Science and Information Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan in 2005. He received the Ph.D. degree in College of Computer Science, National Chiao Tung University, Hsinchu, Taiwan in 2014. His research interests include image processing, image/video retrieval, and pattern recognition.

