INTEGRATED ANALYSIS OF AUDIOVISUAL SIGNALS AND EXTERNAL INFORMATION SOURCES FOR EVENT DETECTION IN TEAM SPORTS VIDEO

Huaxin Xu

(*B.Eng, Huazhong University of Science and Technology*)

Submitted in partial fulfillment of the

requirements for the degree

of Doctor of Philosophy

in the School of Computing

NATIONAL UNIVERSITY OF SINGAPORE

2007

Acknowledgments

The completion of this thesis would not have been possible without the help of many people to whom I would like to express my heartfelt gratitude.

First of all, I would like to thank my supervisor, Professor Chua Tat-Seng, for his care, support and patience. His guidance has played and will continue to play a shaping role in my personal development.

I would also like to thank other professors that gave valuable comments on my research. They are Professor Ramesh Jain, Professor Lee Chin Hui, A/P Leow Wee Kheng, Assistant Professor Chang Ee-Chien, A/P Roger Zimmermann, and Dr. Changsheng Xu.

Having stayed in the Multimedia Information Lab II for so many years, I am obliged to labmates and friends for giving me their support and for making my hours in the lab filled with laughters. They are Dr. Yunlong Zhao, Dr. Huamin Feng, Wanjun Jin, Grace Yang Hui, Dr. Lekha Chaisorn, Dr. Jing Xiao, Wei Fang, Dr. Hang Cui, Dr. Jinjun Wang, Anushini Ariarajah, Jing Jiang, Dr. Lin Ma, Dr. Ming Zhao, Dr. Yang Zhang, Dr. Yankun Zhang, Dr. Yang Xiao, Renxu Sun, Jeff Wei-Shinn Ku, Dave Kor, Yan Gu, Huanbo Luan, Dr. Marchenko Yelizaveta, Dr. Shiren Ye, Dr. Jian Hou, Neo Shi-Yong, Victor Goh, Maslennikov Mastislav Vladimirovich, Zhaoyan Ming, Yantao Zheng, Mei Wang, Tan Yee Fan, Long Qiu, Gang Wang, and Rui Shi.

Special thanks to my oldest friends - Leopard Song Baoling, Helen Li Shouhua and Andrew Li Lichun, who stood by me when I needed them.

Last but not least, I cannot express my gratitude enough to my parents and my wife for always being there and filling me with hope.

Contents

Summary

Event detection in team sports video is a challenging semantic analysis problem. The majority of research on event detection has been focusing on analyzing audiovisual signals and has achieved limited success in terms of range of event types detectable and accuracy. On the other hand, we noticed that external information sources about the matches were widely available, e.g. news reports, live commentaries, and Web casts. They contain rich semantics, and are possibly more reliable to process. Audiovisual signals and external information sources have complementary strengths - external information sources are good at capturing semantics while audiovisual signals are good at pinning boundaries. This fact motivated us to explore integrated analysis of audiovisual signals and external information sources to achieve stronger detection capability. The main challenge in the integrated analysis is the asynchronism between the audiovisual signals and the external information sources as two separate information sources. Another motivation of this work is that video of different games have some similarity in structure yet most exiting systems are poorly adaptable. We would like to build an event detection system with reasonable adaptability to various games having similar structures. We chose team sports as our target domains because of their popularity and reasonably high degree of similarity.

As the domain model determines system design, the thesis first presents a domain model common to team sports video. This domain model serves as a “template” that can be instantiated with specific domain knowledge and keep the system design stable. Based on this generic domain model, two frameworks were developed to perform the integrated analysis, namely the late fusion and early fusion frameworks. How to overcome the asynchronism between the audiovisual signals and external information sources was the central issue in designing both frameworks. In the late fusion framework, the audiovisual signals and external information sources are analyzed separately before their outcomes get fused. In the early fusion framework, they are analyzed together.

Key findings of this research are (a) external information sources are helpful in event detection and hence should be exploited; (b) the integrated analysis performed by each framework outperforms analysis of any single source of information, thanks to the complementary strengths of audiovisual signals and external information sources; (c) both frameworks are capable of handling asynchronism and give acceptable results, however the late fusion framework gives higher accuracy as it incorporates the domain knowledge better.

Main contributions of this research work are:

- We proposed integrated analysis of audiovisual signals and external information sources. We developed two frameworks to perform the integrated analysis. Both frameworks were demonstrated to outperform analysis of any single source of information in terms of detection accuracy and the range of event types detectable.

- We proposed a domain model common to the team sports, on which both frameworks were based. By instantiating this model with specific domain knowledge, the system can adapt to a new game.

- We investigated the strengths and weaknesses of each framework and suggested that the late fusion framework probably performs better because it incorporates the domain knowledge more completely and effectively.

List of Tables

List of Figures

Chapter 1

INTRODUCTION

1.1 Motivation to Detecting Events in Sports Video

Rapid development in computing, networking, and multimedia technologies have resulted in the production and distribution of large amount of multimedia data, in particular digitized video. The whole video archive is a treasure for both entertainment and professional purposes. Consumption of this treasure necessitates efficient management of the archive. Although management by human labor has been a feasible solution and has been in practice for years, the need for *automatic management by computers* is getting imminent, because:

- the volume of video archive is growing fast towards being prohibitively huge, due to wide use of personal video capturing devices;
- convenient access to video archive by personal computing devices such as laptops, cell phones and PDAs makes user needs diverse, thus serving these needs goes beyond the capacity of human labor.

The earliest automatic management systems organized video clips based on manually entered text captions. The brief description by caption brought some benefits, namely requiring simple and efficient computation for retrieving video clips. However, beyond the limits of brief text description, such representation often could not distinguish different parts of a video clip, nor could it support detailed analysis of the video content. Therefore this scheme failed to serve humans' needs regarding "what is in the video". Subsequently content-based systems were developed. Early content-based systems indexed and managed video contents by low-level features, such as color, texture, shape and motion. Metric similarity based on these features enabled detection of shot boundaries [34], identification of key frames [34], video abstraction [37] and visual information retrieval with examples or sketches as queries [19]. These system essentially view video content in the perspective of "what it looks/sounds like". However, human users would like to access the content based on high-level information conveyed. This information could be *who*, *what*, *where*, *when*, *why*, and *how*. For example, human users may want to retrieve video segments showing Tony Blair [23], or showing George Bush entering or leaving a vehicle [23]. In other words, human users would like to index and manage the video based on "what it means", or *semantics*. Low-level processing cannot offer such capabilities; higher level processing that can provide semantics is demanded. Major research fields involving semantic analysis are listed below:

- *Object recognition* aims to identify an visible object such as a car, a soccer player, a particular person, or a textual overlay. This task may also involve the separation of foreground objects from background.
- *Movement/gesture recognition* detects movement of an object or of the camera from a sequence of frames. The system may compute metrics describing the movement, such as panning parameter of the camera [86], or classify the pattern of movement into a predefined category, such as the gesture of smile.
- *Trajectory tracking*, whereby the computer discovers the trajectory of a moving object, either in an offline or online fashion.
- *Site/setting recognition* determines if a segment of video is taken in a specific setting such as in a studio or more generally indoor, on a beach or more generally outdoor, etc.

- *Genre classification*, whereby the computer classifies the whole video clip or particular parts into a set of predefined categories such as commercial, news, sports broadcast, and weather report, etc.

- *Story segmentation* aims to identify temporal units that convey coherent and complete meaning from well structured video e.g. news [21]. In some video that are not well structured e.g. movie, a similar notation *scene segmentation* refers to identifying temporal units that are associated to a unified location or dramatic incident [90].

- *Event*[1] *annotation* finds video parts depicting some occurrence e.g. aircraft taking off and people walking, etc. Sometimes this task and object/setting recognition are collectively called concept annotation.

- *Topic detection and tracking* finds temporal segments coherent on a topic each, identifies the topics and reveals evolution among topics [46].

- *Identification of interesting parts*, wherein the computer identifies parts of predefined interest as opposed to those less interesting. The task can be further differentiated with regard to whether the interesting parts are categorized, e.g. highlight extraction (not categorized) vs. event recognition (categorized) in sports video analysis.

- *Theme-oriented understanding or assembling*, whereby the computer tries to understand the video in terms of overall sentiment being conveyed such as humor, sadness, cheerfulness, etc. Or the computer assembles a video clip that strikes human viewers with sentiments from shorter segments [65] [92].

The tasks listed above infer semantic entities from audiovisual signals embedded in the video. The semantic entities are at various levels. For example, events and themes are at a relatively higher level than objects and motions are. Inference of

[1] The term *event* here means differently than the other occurrences of “events” in the thesis. This “event” refers to anything that takes place.

higher level entities may need help from inference of lower level entities. Inference of semantic entities leads to development of further analysis, such as:

- *Content-aware streaming* wherein video is encoded in a way that streaming is viable with limited computing or transmitting resources. Usually encoding scheme is based on categorization of individual parts in terms of importance, which in turn involves knowledge of the video content to some extent.
- *Summarization* giving a shorter version of the original version and maintaining the main points and ambiance.
- *Question answering* answering users' questions with regards to some specific information, possibly accompanied with associated video content.
- *Video retrieval* providing a list of relevant video documents or segments in response to a query.

Sports video is a popular genre with large audience worldwide. Telecast of big sports events, such as the Olympic Games and the FIFA World Cup have billions of audience all over the world. Besides these global events, millions of people are also attracted to matches in renowned leagues such as the English Premier League (EPL) or in tournaments such as WTA tour. Sports video has a large production volume and occupies a significant portion of the whole video archive. Some games such as soccer and basketball are held in the form of leagues at regional and national, sometimes even international levels. Some other games such as tennis and golf are held in the form of tournaments. These leagues and tournaments have scheduled matches every week. These matches, along with those held in sporadic events over a wide range of games, may total hundreds a week. These matches are covered by dozens of sports channels and aired in thousands of hours of programs worldwide. The whole bulk of sports video is a treasure for both entertainment pursuers and sports professionals. For either group of users, the consumption of video content necessitates effective management of video, which can be facilitated by semantic analysis. Semantic analysis helps to parse the video content into meaningful units, index these units in a way similar to human understanding, and differentiate the contents with regards to importance or interestingness.

A suitable indexing unit for sports video would be an *event*. This is because: (a) events have distinct semantic meanings; (b) events are self-contained and have clear-cut temporal boundaries; and (c) events cover almost all interesting or important parts of a match. Event detection aims to find events from a given video, and this is the basis for further applications such as summarization, content-aware streaming, and question answering. This is the motivation for event detection in sports video.

1.2 Problem Statement

Generally, an event is something that happens (source: Merriam-Webster dictionary). In analysis of team sports video, event and event detection are defined as follows.

Definition 1 *Event*
An event is something that happens and has some significance according to the rules of the game.

Definition 2 *Event detection*
Event detection is the effort to identify a segment in a video sequence that shows the complete progression of the event, that is, to recognize the event type and its temporal boundaries.

In fact, as semantic meaning is differentiated for each event, "event recognition" may be a more accurate term. However, this thesis still follows the convention and uses "event detection". An event detection system should satisfy these requirements: 1) the events detected are a fairly complete coverage of happening that viewers deem important; and 2) the event segments cover most relevant scenes and not too lengthy with natural boundaries.

This thesis addresses the problem of detecting events in full-length broadcast team sports videos.

Definition 3 *Team sports*
Team sports are the games in which two teams move freely on a rectangular field and try to deliver the ball into their respective goals.

Examples of this group of sports are soccer, American football, and rugby league, etc. The reason why we choose this group of sports is: (a) they appeal to a large audience worldwide, and (b) they offer a balance between commonality and specialty, which serve our purpose of demonstrating the quality of our domain models well.

1.3 Summary of the Proposed Approach

The majority of research on event detection has been focusing on analyzing audiovisual signals. However, as audiovisual signals do not contain much semantics, such approaches have achieved limited success. There are a number of textual information sources such as match reports and real time game logs that may be helpful. This information is said to be *external* as it does not come with the broadcast video. External information sources may be categorized to *compact* or *detailed* regarding to the level of detail.

We proposed integrated analysis of audiovisual signals and external information sources for detecting events. Two frameworks were developed that perform the integrated analysis, namely the late fusion and early fusion frameworks.

The late fusion framework has two major steps. The first is separate analysis of the audiovisual signals and external information sources, each generating a list of video segments as candidate events. The two lists of candidate events, which may be incomplete and in general have conflicts on event types or temporal boundaries, are then fused. The audiovisual analysis consists of two steps: *global structure analysis* that helps indicate when events may occur and *localized event classification* that determines if events actually occur. The text analysis generates a list of candidate events called text events by performing information extraction on compact descriptions and model checking on detailed descriptions.

In contrast to the late fusion framework, the early fusion framework processes the audiovisual signals and external information sources together by a Dynamic Bayesian Network before any decisions are made.

1.4 Main Contributions

- We proposed integrated analysis of audiovisual signals and external information. We developed two frameworks to perform the integrated analysis. Both frameworks were demonstrated to outperform analysis of any single source of information in terms of detection accuracy and the range of event types detectable.
- We proposed a domain model common to the team sports, on which both frameworks were based. By instantiating this model with specific domain knowledge, the system can adapt to a new game.
- We investigated the strengths and weaknesses of each framework and suggested that the late fusion framework probably performs better because it incorporates the domain knowledge more completely and effectively.

1.5 Organization of the Thesis

The rest of the thesis is organized as follows.

1. Chapter 2 reviews related works, including those on event detection in sports video, on structure analysis of temporal media, on multi-modality analysis, on fusion of multiple information sources, and on incorporation of domain knowledge.

2. Chapter 3 describes properties of team sports video and common practices for both frameworks. This chapter describes the domain model, audiovisual signals and external information sources, steps for unit parsing, extraction of commonly used features, and the experimental data.

3. Chapter 4 describes in detail the late fusion framework with experimental results and discussions.

4. Chapter 5 describes in detail the early fusion framework with experimental results and discussions.

5. Chapter 6 concludes the thesis with key findings, conclusions and possible future works.

Chapter 2

RELATED WORKS

This Chapter reviews works on event detection from sports video (reported in Section 2.1) as well as other works on multimedia analysis in general (reported in Sections 2.2 - 2.5). The second group of related works may offer enlightenment to our problem. In particular, these include structure analysis on temporal media, multi-modality analysis, fusion of multiple information sources, and incorporation of domain knowledge.

2.1 Related Works on Event Detection in Sports Video

Semantic analysis of video of various sports has been actively studied, e.g. soccer [98], swimming [17], tennis [26], and others. As a basic and integral semantic entity, event in sports video serves as a suitable unit that facilitates higher level manipulation, e.g. annotation [17], browsing, retrieval and summarization. Much research effort has been made to detect events from sports videos [26] [112]. As detection of some other high-level entities may offer enlightenment to detection of events, this Section also includes reviews of such works as well, for example, on activity categorization, highlight extraction, atomic action detection, etc.

Compared to other video genres such as news and movie, sports video has well-defined content structure and domain rules:

- A long sports match is often divided into a few segments. Each segment in turn contains some sub-segments. For example, in American football, a match contains two halves, and each half has two quarters. Within each quarter, there are a number of plays. A tennis match is divided first into sets, then games and points.
- Broadcast sports videos usually have production artifacts such as replays, graphic overlays, and commercials inserted at certain times. These help mark the video's structure.
- A sports match is usually held on a pitch with specific layout, and captured by a number of fixed cameras. These result in some canonical scenes. For example, In American football, most plays start with a snap scene wherein two teams line up along the lines of scrimmage. In tennis, when a serve starts, the scene is usually switched to the court view. In baseball, each pitch starts with a pitching view taken by the camera behind the pitcher.

The above explanation suggests sports videos are characterized by distinct domain knowledge, which may include game rules, content structure and canonical scenes in videos. Modeling the domain knowledge is central to event detection. Actually an event detection effort is essentially an effort to establish and enforce the domain model.

2.1.1 Domain Modeling Based on Low-Level Features

Early works attempted to handcraft domain models as distinctive patterns of audiovisual features. The domain models were results of human inspection of the video content and were enforced in a heuristic manner.

Gong et al. [33] attempted to categorize activity in a soccer video to classes such as “top-left corner kick” and “shot at left goal”, which in a coarse sense can be viewed as event detection. They built models on play position and movement of each shot. The models were represented in the form of rules, e.g. “if the play position is near the left goal-area and the play movement is towards the goal, then it is a shot at left goal.” The play position was obtained by comparing detected and joined edges to templates known *a priori*. The play movement was estimated by minimum absolute difference (MAD) [27] on blocks. It is noteworthy that some categories of activity were at a lower level than events were, e.g. “in the left penalty area”. This seems to suggest that while play position and movement could describe spatial properties well, they were not capable of differentiating a wide range of events.

Tan et al. [86] detected events in basketball video such as fast breaks and shots at the basket. The model for fast break was “video segments whose magnitude of the directional accumulated pan exceeds a preset threshold”. And one model for shot at the basket was “video segments containing camera zoom-in right after an fast break or when the camera is pointing at one end of the court”. The camera motion parameters such as magnitude of pan or zoom-in were estimated from motion vectors in MPEG video streams. Some more descriptors could be further derived, such as the directional accumulated pan over a period of time and duration of a directional camera motion. Note that the method’s detection capability was also limited. Fast break and full court advance were differentiated by an ad hoc threshold. Some events that lack distinctive patterns in camera motion such as rebounds and steals could not be detected.

Li et al. [56] aimed to detect plays in baseball, American football and sumo wrestling videos. These three games have common characteristics in structure: important actions only occur periodically in game segments that are interleaved with less important segments. The game segments containing important actions are called plays. Recurrent plays are characterized by relatively invariant visual patterns for one game. This made play to be modeled as “starting with a canonical scene and ending with certain types of scene transitions", though the "canonical scenes" and "certain scene transitions" are game-specific. For baseball, the canonical starting scene was modeled as a pitching scene that conforms to certain spatial distribution of colors and spatial geometric structures induced by the pitcher and some other people (the batter, the catcher, and the umpire). For American football, the canonical starting scene was modeled as a snap scene that has dominant green color with scattered non-green blobs, and has little motion, plus parallel lines on a green background. For sumo wrestling, the canonical scene was one containing two symmetrically distributed blobs of skin color on a relatively uniform stage. Ending scene transitions could be something like a hard-cut in a temporal range. Heuristic search for these canonical scenes and scene transitions was performed to find starts and ends of plays. Though the method could reportedly find plays with over 90% F1 values, it could not differentiate events - plays characterized with certain outcomes.

Sadlier et al. [76] aimed to extract highlights from a wide range of sports videos: soccer, gaelic, rugby and hockey, etc. Since the task was to differentiate semantic significance, i.e. highlights vs. less interesting parts, we can also view it an event detection task in a coarse sense. Based on the assumption that commentators/spectators exhibit strong vocal reaction to momentary significance, the model here is that portions with high amplitude in soundtrack may be highlights. Highlights are those portions where sums of scalefactors from subbands 2 - 7 are large enough. These subbands account for the frequency range of 0.625kHz - 4.375kHz, which approximate the frequency range of human speech. Similar to Li et al. [56], the method could only tell highlights from less interesting parts, but could not differentiate events further, such as goals in soccer.

2.1.2 Domain Models Incorporating Mid-Level Entities

The reviews in 2.1.1 suggest that domain models based on low-level features were not descriptive enough. As events in games involve interactions among players or between a player and an object, it would be desirable to incorporate players and objects into the models. Given that players and objects have some semantic significance and they are not events yet, we call them *mid-level entities*. It is expected that mid-level entities would enrich models' descriptiveness, as events can be modeled by spatiotemporal relationships of mid-level entities. Besides players and objects, mid-level entities also include those that semantically abstract visual or audio content of a portion, e.g. replays and cheering.

Sudhir et al. [84] attempted to detect a rich set of tennis events: baseline-rallies, passing-shots, serve-and-volley, and net-game. Included in the domain model was a court model based on perspective geometry and an rule-based inference engine. The court model helped in transforming players' positions on the frame to the real world. And the transforming was performed over time. The inference engine then used this spatiotemporal information to tell the event. The rules in the inference engine were handcrafted like "if both players' initial and final positions in a play are close-to-baseline then this play is a baseline-rally". It can be seen that the rules made use of spatiotemporal relationships between players and baselines. Court lines on the frame were detected using a series of techniques: edge detection, line growing, and missing lines reconstruction. A point on the frame is projected to the real world court with the help of the court model. Players were tracked heuristically by template matching.

Nepal et al. [66] detected goals in basketball videos. The models involved two mid-level entities - cheering and scoreboard and one low-level cue - change in direction. Models were built on their temporal relationships and take on the form of rules. For example, one model was "goal $\rightarrow$ [10 seconds] $\rightarrow$ change in direction + [10 seconds] $\rightarrow$ cheering". All low-level cues and mid-level entities were detected heuristically. Specifically, cheering was found by looking for high energy segments in the soundtrack; scoreboard was found by looking for areas with sharp edges that entailed high AC coefficients in DCT blocks; and change in direction was found from motion vectors in a way similar to [56].

Yu et al. [110] aimed to detect atomic actions in soccer: passing and touching of the ball, and further to derive goals. Detection of passing and touching was based on ball trajectory and heuristic rules. Detection of goals involved detection of goalpost besides ball trajectory. Thus the ball, ball trajectory and the goalposts were the mid-level entities.

Bertini et al. [6] [17] built domain models of events in a rigorous fashion - they used finite state machines (FMM). The nodes represent states during the development of events, and the edges represent the transitions between the states[1]. Transitions are defined in terms of spatiotemporal relations between players and objects or between objects, for example, "ball moves away from goalpost". FMM may be superior to if-then rules as it is capable of describing more complex logic such as more diversions and/or loops, allowing it to enjoy some flexibility and maintain rigorousness.

Ekin et al. [30] and Duan et al. [26] used mid-level entities, namely audio keywords and shot types e.g. close-up or replay to describe games' temporal structures with regards to when events can possibly occur.

Mid-level entities also helped in enhancing robustness against variation in low-level features and in improving adaptability of high-level analysis, as in [26].

As expected, the incorporation of mid-level entities makes domain models superior to earlier ones. This is because models' expressiveness has been enhanced by spatiotemporal relationships of mid-level entities [84]; mid-level facilitates the modeling of hierarchical semantic entities [110]; mid-level entities help in describing video structures [26]; spatiotemporal relationships of mid-level entities make models more rigorous [6]; and abstraction brought by mid-level entities alleviates data sparseness problem and makes the systems more robust.

[1] The citation uses different terms from the original ones in the article to remain consistent with the other parts of the thesis. Original term referring to the edge is *event*, and the "event" of this thesis is referred to as *highlight*.

The following Section reviews briefly how typical mid-level entities are detected. They may be detected by heuristic or machine learning methods.

Camera motion parameters. Zhang et al.'s pioneering work on camera motion categorization [114] analyzed motion vectors in MPEG streams heuristically. They differentiated pans or tilts from modal motion vectors, and zooms from opposite motion vectors at the two ends of macroblock columns. To estimate quantitatively camera's rotational, zooming, and/or translational motion, a transformation matrix is usually built that links an image point and its correspondence resulting from the motion. This transformation matrix is made up of camera motion parameters. By determining the matrix with a number of point correspondences, the parameters are determined. Baldi et al. [15] and Assfalg et al. [6] attempted to track salient image locations e.g. corners in this framework. However, locating and matching a pair of salient image locations are difficult. To circumvent this difficulty, Tan et al. [86] used pairs of macroblocks in MPEG streams linked by a motion vector as samples of the transformation.

Graphic or textual overlay. Graphic or textual overlay are generally done by detecting high contrast areas, which is translated to high AC components in DCT blocks, e.g. Zhang et al. [115] and Nepal et al. [66]. For uncompressed video, general edge detection techniques were used, such as sobel filtering and radon transform [88]. Zhang et al. [113] [112] further recognized the content of the overlay by a series of techniques: segmentation of characters from background by binarization and grouping, and recognition of segments by Zernike moments. Babaguchi et al. [14] and Zhang et al. [112] utilized state transition graphs encoding game rules to further improve recognition accuracy.

Ball and ball trajectory. Early works on ball detection mainly relied on object segmentation subject to heuristic constraints, e.g. on color and shape [33]. The results had been generally poor. Yu et al. [109] [111] [110] also evaluated if a candidate ball trajectory conformed to characteristics of a ball trajectory. In this way, more constraints were put in effect. Verification of candidate trajectories was based on Kalman filter.

Court lines. Court lines are mostly detected as edges and would usually undergo growing and joining steps. Gong et al. [33] employed Gaussian-Laplacian edge detector. Differently, a heuristic method was reported by Sudhir et al. [83]. They formed lines by joining pixels that satisfy color criteria in a certain direction.

Salient objects. Most common objects in this group are goalposts, mid-field line and penalty-box in soccer. Detection of such objects are usually based on edge detection subject to color and shape constraints. Yu et al. [110] detected goalposts by a set of heuristic criteria on the directions, widths and lengths of edges. Wan et al. [89] applied Hough transform to edges and employed some postprocessing, including verification of goal-line orientation and color-based region (pole) growing.

Field zone. Gong et al. [33] recognized field zones by detecting line segments, joining and matching them to templates. Assfalg et al. [6] classified the field zone by naive Bayes classifiers based on attributes of the visible pitch region and lines, including region shape, region area, region corner position, line orientation and mid-field line position. Wang et al. [91] classified the field zone by a Competition Network and used these attributes: field-line positions, goalpost position and mid-field line position.

Players' positions. Sudhir et al. [84] proposed a method to detect and track players in tennis video. By compensating motion, they produced a residue of the current frame over the preceding one. Then they took largest connected blobs in dense areas of the residue as players. To track a player, they conducted a full search around the area where the player had last been detected using minimum absolute difference algorithm. Assfalg et al. [6] used adaptive template matching to find players' positions on the frame. Firstly candidate blobs were segmented from the pitch by color differencing, then templates were constructed with certain color distribution and shape adapted to the blobs' size. Finally they could tell by template matching if a blob was a player. Ekin et al. [30] developed a similar color-based template matching technique to detect the referee as well. Detecting players' positions usually comes with relating the positions to certain parts of a court or a pitch. This entails mapping a position between the coordinates system in the real world and that on the image domain. Such mapping is usually based on a camera geometry model. Sudhir et al. [83] mapped tennis court lines from the real world to the image domain, and told if a given player's position was close to a court line. Assfalg et al. [6] modeled the mapping by a homography matrix containing eight independent entries and determined the entries with four line correspondences.

Replay. Some works detected replays that are characterized by slow motion. Among them, Pan et al. [69] used HMM and Ekin et al. [30] used measure of fluctuations in frame difference, respectively. This algorithm did not give satisfactory boundaries of replays because it treated the boundaries as ordinary gradual transitions. Babaguchi et al. [12] and [70] detected replays by the editing effects immediately before and after replays. Babaguchi et al. [12] manually built models of such effects in terms of color and motion characteristics, and model-checked each frame. Pan et al. [70] used the relatively invariant logo as the editing effect. The method was to have several probabilistic measures of distance between a frame and the logo frame and fuse the measures by the Beyes's rule.

Audience. In view that audience is characterized by richness in edge, audience detection algorithms has generally been based on edge detection. Lee et al. [53] identified presence of audience in basketball video by detecting richness of edge from compressed MPEG stream: first DCT coefficients in one block of an I frame were projected in the vertical and horizontal directions by synthetic filters, then projected components in the same direction added up. If either sum was significant enough, the block would be declared as an edge segment and its direction was determined by comparing the vertical and horizontal sums. A frame's richness of edge was defined as the total length of edge segments. If a sequence of I frames had richness larger than a threshold, it would be declared to have audience scenes.

Cheering. Detection of cheering has been based on detection of high-energy segments in the soundtrack. Nepal et al. [66] used sum of scalefactors of all subbands in MPEG streams as the criterion of high energy.

Excited commentator's speech. Some works, e.g. Sadlier et al. [76] took an approach similar to that described in [66] to detect cheering, with scalefactors restricted in frequency range of human speech. Rui et al. [75] employed a more sophisticated approach. They first identified speech segments by heuristic rules involving Mel-scale Frequency Cepstrum Coefficients (MFCC) and energy in the frequency range of human speech. Then they did a classification using pitch and energy features and some machine learning algorithms (parametric distribution estimation, KNN, and SVM) to tell if a speech segment was excited. Tjondronegoro et al. [88] recognized help of lower pause rate and temporal constraints in detecting excitement besides pitch and energy features.

Game-specific sounds. Besides cheering and excited commentator's speech, there are other sounds that may serve as mid-level entities, e.g. batting sound in tennis and whistle in soccer. Rui et al. [75] detected batting sound by energy features and template matching algorithm. To detect whistles, Tjondronegoro et al. [88] used power spectral density (PSD) within whistle's frequency range and heuristic thresholds. Xu et al. [103] classified a number of game-specific sounds, including cheering, commentator's speech (excited and plain), various whistling, and etc. They used a series of SVM classifiers and a set of audio features: zero-crossing rate (ZCR), spectral power (SP), mel-frequency cepstral coefficients (MFCC), linear predication coefficients (LPC) and short time energy (STE).

Handcrafted domain models have been reported successful in their test scenarios, as they are precise, easy to implement and computationally efficient. However, models are laborious to construct and are seldom reusable, and they are not able to handle subtle events that do not have a distinctive audiovisual appearance, such as yellow/red card events in soccer. Because of these limitations, only a subset of events in a domain can be detected using this approach.

As more features are incorporated, representation of domain knowledge by handcrafted domain models may be inefficient and difficult. Instead, representation of domain knowledge by data driven techniques have been fostered. Zhou et al. [117] employed decision tree to model affinity between scenes of basketball video and to classify them by features' thresholds. They used low-level features from motion, color and edge. Rui et al. [75] moderated the probability that excited commentator's speech indicated baseball event by a confidence level. And the confidence level was derived from conditional probabilities of labeled data (baseball hits). Intille et al. [47] modeled and classified American football plays using trajectories of players and ball via Bayes networks; anyway, the mid-level entities were entered manually and were not automatically extracted from the video stream. Zhong et al.'s work [116] involved template adaptation. This was accomplished by clustering color-represented frames from the given video. Han et al. [35] used maximum entropy criterion to find appropriate distributions of events over the feature space. The features were a mixture of low-level audiovisual cues derived from color, edge, camera motion, and mid-level entities including player presence, words from closed caption and audio genre. Sadlier et al. [77] attempted to map each shot's features to whether the shot exhibits an event by SVM classifiers. They also employed a mixture of low-level features (speech band energy and motion activity) and some mid-level entities (detection of crowd, graphic overlay and orientation of field lines). These methods have a common characteristic: a data point is associated with a temporal unit of the stream, e.g. a video shot or an audio clip, and data points are processed independently with sequential relationships unconsidered.

Another group of works recognize the role of sequential relationships in indicating semantics and capture them by temporal models such as Hidden Markov Model (HMM). Assfalg et al. [7] modeled penalty, free-kicks and corner-kicks each with a HMM on features derived from camera motion. Leonardi et al. [54] built a controlled Markov chain model (CMC) to model goals. The model was two HMMs concatenated each having its own probability distributions; transition from the first one to the second was triggered by an external signal, a hard-cut in the scenario of the paper.

Besides aforementioned works that aimed to classify events modeled by individual HMMs, there are works trying to capture sequential relationships between events (or other semantic entities) as well. Xu et al. [102] and Xie et al. [97] [99] [101] modeled the individual recurrent events in a video as HMMs, and the higher-level transitions between these events as another level of Markov chain. Kijak et al. [51] attempted to describe tennis's complete structure of set - game - point by a hierarchical HMM with bottom-level HMM reflective of match progress (missed first serves and rallies) or editing artifacts (breaks and replays). Another purpose of using hierarchical HMM [51] was to identify boundaries of events, as the boundaries align with transitions of one-level-higher node. A majority of this group of works aim to discover recurrent structures from sports video, and will be reviewed in more detail later in “2.2 Related Works on Structure Analysis of Temporal Media”.

Note that the establishment of domain models by machine learning may be facilitated/supplemented by handcrafted domain models. The most obvious scenario is that of choosing a set of representative descriptors (including low-level cues and mid-level entities) and machine learning algorithms. Further scenarios could be adaptation of algorithms made specific to domain constraints, as exemplified in [75]. Such supplementation would poise the machine learning algorithms towards the essential problem and reduce the need for training samples. More detailed review on this aspect will be given in "2.5 Related Works on Incorporating Handcrafted Domain Knowledge to Machine Learnt Models".

2.1.3 Use of Multi-modal Features

As researchers began to realize that information from different modalities is complementary and with growing computing power, there is an increasing interest in multi-modal collaboration for event detection (actually this is true for virtually for all semantic analysis tasks). Commonly used modalities are video, audio and text. Sources for text are textual overlay [112], transcripts from automatic speech recognition (ASR) [5], closed caption [14] [8] [12] [9] [10] [62] [67] [13] [11] and the web [13]. Zhang et al.[112] and Ariki et al.[5] both aimed to detect baseball events. Zhang et al.[112] made use of both textual overlay and image information. Textual overlay showed score changes and the system inferred occurrences of events based on game rules. Image analysis found boundaries of pitching segments as a universal event container by algorithms similar to those described in [116]. Ariki et al.[5] adopted a similar approach; they used image and speech instead. Image analysis segments the video sequence to pitching segments; speech analysis located diverse events by keywords matching. Textual overlays and speech transcripts would provide rich semantics if accurately recognized, however, they are not always available. Babaguchi et al. presented a range of methods to use closed caption along with audio and visual streams in semantic analysis of American football video [14] [8] [12] [9] [10] [62] [67] [13] [11] [68]. Closed caption is human transcribed speech plus other relevant information such as time stamps and speaker identification. In sports video, it usually contains commentators' speech. It is generally reliable compared to machine recognized textual overlay or speech. Miyauchi et al. [62] first used textual cues from closed caption to roughly locate events, then performed a screening by a learning-based classifier working on audio features, and lastly identify events' boundaries by video cue. Nitta et al. [67] [68] segmented American football videos on accompanying closed caption streams, then refined boundaries by relating to video shots. Babaguchi et al. [13] summarized several methods of inter-modal collaboration. The paradigm was closed caption assuming primary role to indicate events and rough occurrence time, and audio-visual analysis assuming secondary role to refine events boundaries. Their work seemed to suggest that some assumptions were made: (1) closed caption contains sufficient detail, and (2) the temporal correspondence between closed caption and other modalities is relatively consistent. For the particular game of American football, these assumptions hold. Play in American football match is in intermittent segments and outcome of each segment is predicable, thus closed caption contains sufficient detail and has relatively consistent temporal lag behind the visual stream. However this may not be true for continuous games, such as soccer. During a soccer match, commentators may skip much detail due to unpredictability of match progress and the temporal lag may vary. Therefore a more useful approach would be one that assumes less. Babaguchi et al. [13] also suggested using external metadata on the Web. However, the described method assumed that the time recorded in the metadata was accurate and recognition of textual overlay for time was reliable, which limited the applicability of the method.

Table 2.1 gives a side-by-side view of some existing systems developed for detecting events in sports video.

Table 2.1: Comparing existing systems on event detection in sports video

	Domain	targets	Features	Algorithm	Accuracy	Pros	Cons
Gong et al.[33]	Soccer	Corner-kicks and shots	Edge, motion vectors	Modeling checking of edge to obtain play position and further events based on play position and motion.	40% 60% on segments	Easy to implement	Coarse models; limited range of event types.
Tan et al.[86]	Basketball	Fast break, full court advance, shot	Motion, shot category	Motion vector as samples to estimate camera motion parameters. Events are obtained by model checking on camera motion pattern and shot category.	70% 80%	Accurate and robust in estimating camera motion parameters.	Limited range of event types detectable
Duan et al.[26]	Soccer, tennis	Foul, free kick, penalty, corner kick, shot, goal (for soccer) game, deuce, point, serve, reserve, return, ace, fault, double fault, take the net, rally (for tennis)	Motion, color, texture, shot length	Use of mid-level semantic entities. Low-level to mid-level mapping by statistical learning; mid-level to high-level mapping by reasoning based on event models.	70% 100% on segments	Use of mid-level semantic entities alleviates data sparseness and enhances robustness and adaptability	Still not full range of event types
Li et al.[56]	Baseball, American football, sumo wrestling	A temporal unit called *play*	Color, spatial distribution of color, motion, shape	A play is modeled as starting with a canonical scene and ending with some cinematographic pattern	95% 98% on plays	Easy to implement and high accuracy	Cannot distinguish between event types within a play
Sadlier et al.[76]	Soccer, Gaelic football, rugby and hockey	To distinguish highlights from less interesting parts	Audio volume	By rule - highlights are associated with high audio volume	50% 75%	Easy to implement	Cannot distinguish between the event type of a highlight

Sudhir et al.[84]	Tennis	Baseline-rallies, passing-shots, serve-and-volley, and net-game	Edge, color	Objects on the frames are projected to real-world court by geometrical techniques. Event models are specified in terms of spatiotemporal relationships between objects. Events are detected by model-checking.	unreported	Detailed and explicit modeling of domain knowledge	Accuracy of events is subjected to accuracy of object detection
Nepal et al.[66]	Basketball	Goal	Motion, textual overlay, audio energy	Event models are specified in terms of crowd cheer, score board or change in motion. These mid-level semantic entities are obtained from the low-level features by heuristic rules.	50% 100%	Models are easy to understand and to implement.	Limited range of event types detectable
Yu et al.[110]	Soccer	Goal, touching, passing, just-missing	Motion, color, texture	Goals are obtained from mid-level reasoning based on ball, ball trajectory, and goalposts. Ball and Ball trajectory is obtained by Kalman filter-based method along with some heuristic constraints. Goalposts are detected by edge analysis.	77% 100% on segments	Reliable ball trajectory is desired as a descriptive cue which may facilitate detection of a wider range of event types.	Algorithm may not be robust enough.
Bertini et al.[17]	Swimming	Race start, race arrival, turning	Motion, edge, shape, color	Field zones are obtained from region and line-associated features by naïve Bayes classifiers. Events are detected by model checking in terms of motion pattern and field zone.	70% 85% on segments	Models are vigorously built and are easy to implement; generally accurate.	Models are laborious to manually build.

Assfalg et al.[6]	Soccer	Forward pass, shot on goal, turnover, corner-kick, free-kick, penalty, kickoff, counterattack	Motion, edge, shape, color	Field zones are obtained from region and line-associated feature by naïve Bayes classifiers. Players are obtained by matching to template specified in color-coherent blobs and particular shape. Players and objects on the image are projected to the real-world filed model to facilitate reasoning based on rules. Events are detected by model checking.	75% 100% on segments	Models are vigorously built and are easy to implement; generally accurate; position-based reasoning can be used in other field sports.	Models are laborious to manually build.
Zhou et al.[117]	Basketball	Left offense, right offense, left fast break, right fast break, left score, right score, left dunk, right dunk	Motion, color, edge	Decision tree as the overall framework	70% 89%	Principled approach that generates rules automatically and requires little feature selection	Limited range of event types detectable
Rui et al.[75]	Baseball	Highlight, baseball hit	Audio energy-related features, phoneme-level features, entropy-related features, prosodic features	Learning-based classifier (SVM, kNN, Gaussian fitting) to detect excited speeches; template matching to detect candidate hits; fusion (weighted, probabilistic) to generate highlights.	60% 70% on segments	Audio features are less expensive to compute; accuracy is acceptable; fusion schemes providing insights	Limited range of event types detectable

Zhong et al.[116]	Tennis, baseball	Serve (tennis), pitch (baseball)	Color, edge, motion	Model checking with components (objects) acquired in data-driven techniques	90% 95% on segments	Objects being acquired in data-driven techniques have reasonably high adaptability	Hard to detect event types with subtle audiovisual appearance
Han et al.[35]	Baseball	Home run, outfield hit, outfield out, infield hit, infield out, strike out, walk	Color, edge, motion, mel-cepstral coefficients (audio), keywords (textual)	Maximum entropy-based criterion to find appropriate distributions of events over the feature space	50% 87%	A principled data-driven approach to acquire (implicit) event models and relevant features	Accuracy is yet to be satisfactory
Sadlier et al.[77]	Soccer, gaelic football, hockey, rugby	Highlight	Audio energy, motion, edge as well as mid-level semantic entities - crowd, graphic overlay	SVM	Measured in curve of precision as recall increases: precision at 65% 74% for recall at 90%.	A principled approach targeting at a wide range of games	Ineffective encoding of domain knowledge
Zhang et al.[112]	Baseball	Score, last pitch	Textual overlay	Events are detected by reasoning on score as specified in domain knowledge	85% 100% on segments	Reliable in detecting score-related events	Limited range of event types detectable
[5]	Baseball	Pitch	Keywords from ASR, color	Events are detected from keywords using langauge model, and then corresponding video segments are identified by finding canonical scene.	84% 100%	Highly adaptable to detect occurrence of events given reliable ASR	Modeling of event boundaries may not be applicable to other events

Babaguchi et al.[10]	American football	Touchdown, conversion, field goal	Color, closed caption	Occurrence of events are detected from closed caption based on domain-specific language models; event boundaries are identified by matching images to templates.	75% 95% on segments	Closed caption provides is a more reliable information source than audiovisual signals (including ASR)	Closed caption content and format not standard, limiting the approach's applicability on a wider range of event types

2.1.4 Accuracy of Existing Systems

Accuracy wise, there is no simple scheme to compare existing systems. This is because they addressed different detection targets, worked on diversified data sets or domains, and were subjected to different scenarios. Anyway, a rough picture can still be derived. As an example system of handcrafted domain model based on low-level features, Tan et al. [86] attempted at two groups of events of basketball, (a) fast breaks and full court advances combined and (b) shots at the basket. From four minutes-long video clips, they reported an F1[2] value over 0.90. However, it is noteworthy that this method was capable of detecting only a few event types and left out a wide range of events such as rebounds and steals. Assfalg et al. [6] and Duan et al. [26] represented systems of handcrafted domain models involving mid-level entities. They could detect quite a range of soccer events and tennis events[3]. For soccer events, Assfalg et al. [6] tested on over 100 clips lasting from 15s to 90s. They reported F1 values of $0.65 \sim 0.96$. Duan et al. [26] tested on 3 full soccer matches and a couple of tennis video clips. They obtained F1 values of $0.67 \sim 0.95$ for soccer and $0.77 \sim 0.95$ for tennis. Note that this group of methods still fail to detect events that do not have distinctive audiovisual patterns, e.g. *substitution* in soccer. Decision tree-based method [117] could detect the most remarkable basketball events and reported F1 values of $0.78 \sim 0.81$ on a few dozens of pre-segmented clips. Maximum entropy-base method [35] detected a wide range of events[4] on 32 hours of videos and obtained F1 values of $0.48 \sim 0.88$. SVM-based method [77] differentiated video clips as "eventful" or "non-eventful", i.e. detected all events combined. They drew a plot of content rejection ratio (CRR) against event retrieval ratio (ERR) which are equivalent to precision against recall. The F1 values on dozens of clips were $0.75 \sim 0.81$ across a range of games (rugby, soccer, hockey, and gaelic football). HMM-based method [7] detected penalty,

[2] F1 is a notation borrowed from information retrieval community, defined as $F1 = 2 \cdot Precision \cdot Recall/(Precision + Recall)$

[3] Their target soccer events were goals, shots, penalties, free kicks, corner kicks, and foul and offside combined; target tennis events were serve, re-serve, return, ace, fault, double fault, take the net and rally.

[4] Their target events were home run, outfield hit, outfield out, infield hit, infield out, strike out, walk and junk.

free kick, and corner kick from dozens of soccer video clips and reported F1 values of $0.66 \sim 0.76$. Hierarchical HMM-based method [97] could achieve play - break segmentation accuracy of 0.75 on dozen-minute long soccer video. Collaborations based on audiovisual signals and textual information have reported F1 values of 0.90 on a wide range of events [68], and $0.85 \sim 0.98$ [5].

2.1.5 Adaptability of Existing Domain Models

Another aspect to look at in an event detection system is how much effort in the adaptation is required in order for the system to work on a different data set or further on a different domain, i.e., adaptability. Adaptability is more than about saving labor in building systems for different domains, it is also a test of domain model quality. Highly constrained domain model may produce good results on a small data set by taking advantage of biased distribution but would not generalize well. A quality domain model captures the essence of the domain knowledge that is coherent over large data sets. Therefore a quality domain model would maintain reasonably good results under fewer constraints. Adaptability measures this quality of generalization. Most existing systems worked on single domains, nevertheless there have been a few addressing a range of domains. Li et al. [56] proposed a model template for detecting plays in several sports: American football, baseball, and Wrestling. However, the model template served more as a guideline than as a concrete framework and the features were sport-specific. Moreover, it was only capable of differentiating plays from non-plays, rather than classifying among event types. Bertini et al. [17] used modal checking to detect events in soccer and swimming. The features were common across event types and sports, but the models were event type-specific, and required substantial human efforts to develop.

2.1.6 Lessons of Domain Modeling

To sum up, existing representation schemes of domain knowledge for sports video analysis need further improvement. Handcrafting domain models based on audio-visual descriptors (low-level cues and mid-level entities) has several drawbacks: (a) because the process is not principled, it can hardly be reproduced; (b) it may fail to reveal non-intuitive domain models, especially in high dimensionality scenarios; and (c) because the process is not principled, quality of output domain models is hard to evaluate. The major drawback of machine learning methods is that they may be misled by biased data sets. Training samples provided to machine learning techniques are oftentimes limited compared to those needed, hence they bring biases. Furthermore, handcrafting and machine learning techniques share a common drawback, namely they can only represent events with distinctive and consistent audiovisual patterns. Events that do not have this characteristic such as substitution in soccer is beyond their capability. This is because the information input to them - audiovisual signals - are at low level. The aforementioned review suggests that audiovisual signals-based domain models (including handcrafted and machine learnt) have limited detection capabilities in terms of accuracy and range of detectable events. Introduction of textual information may be a remedy, as text is more consistent in indicating semantics. Domain models that incorporate semantics derived from text and analysis of audiovisual signals may enlarge the range of detectable events and increase accuracy.

The aforementioned review of efforts in building domain models may suggest that a quality domain model should be one that possesses the following properties.

- It is *descriptive*, i.e. it is distinctive and consistent in indicating events;
- It *generalizes* well, i.e. maintains reasonably good results under relatively few constraints;
- It follows a *principled* manner to build and to enforce.

The review also suggests some guidelines may be helpful in building a quality domain model:

- Judicious choice and combination of handcrafted domain knowledge and machine learnt domain model;

- Use of both audiovisual signals (depicting low-level appearance) and textual information (capturing high-level semantics) from reliable sources.

2.2 Related Works on Structure Analysis of Temporal Media

Structure is an integral part of domain knowledge. It is a global representation and often provide coarse depiction of the media document. Structure analysis discovers underlying structure, which may pave the way for more detailed analysis. Our approaches address structure analysis of team sports video (readers are referred to "Section 1.3 Summary of Proposed Approach"). This Section gives a review of related works on structure analysis of temporal media. Such works include segmenting text by topics and segmenting video sequence to shots, stories or plays. Note that although individual works may be specific to different forms of media - video, audio or text, their generalization in mathematical terms may be helpful to our task.

Early works performed structure analysis in a heuristic way. Some text segmentation algorithms exploit a variety of linguistic features that mark topic boundaries, such as referential noun phrases [71]. In the case of video analysis, this means looking for canonical audiovisual cues for demarcation. Li et al. [56] tried to segment plays in American football video. They found starts of plays by snap scenes wherein two teams line up along the lines of scrimmage, and located ends of plays by shot cuts. Xu et al. [104] and Ekin et al. [29] devised rules to identify plays and breaks based on temporal arrangement of shot types (global view, medium view and close-up). Works on video segmentation with textual information have also exploited canonical cues. Merlino et al. [61] detected news stories from video by locating cue frames such as "Hello, I am <Person's name>" in closed caption and verifying by audiovisual cues such as silence and black frames. Haupmann et al.

[39] segmented news stories by detecting commercials and topic change markers “>>>” in closed caption.

There are more principled methods for video and text segmentation. Some works exploit the fact that segments are more cohesive within themselves than when they transit to another segment. This has been reflected in text segmentation and video shot segmentation works. In text, topic segments are lexically cohesive; and in video, shots are visually similar. As a conventional practice in text segmentation [42] [46] [32], a similarity value reflecting lexical cohesiveness is computed at each gap that could possibly be a segment boundary. A plot of this value shows how lexical cohesion change over time. A local minimum resulting from a sharp change in lexical cohesion is regarded as a boundary. Hearst et al. [42] and Galley [32] defined the lexical cohesion at a gap as the cosine similarity of two vectors representing the two windows before and after the gap. Hearst et al. [42] defined the vector element as count of a registered term in the window, and Galley [32] defined it as a score associated with a registered lexical chain. In video shot segmentation, similarity of two adjacent frames formed the plot and local minimums resulting from sharp changes in the plot were regarded as shot boundaries [50]. To detect gradual transitions, multi-resolution analysis were proposed that also computed similarity on larger scales. This approach has been applied to both text and video segmentation [57] [58].

Similarity-based segmentation may be extended to beyond adjacent units so that neighboring similar units may fall into the same segment. Yeung et al. [108] segmented video to stories by time-constrained clustering of visually similar shots. Shot similarity was defined based on difference metric reflecting color histogram intersection and pixel correlation between keyframes. Hierarchical clustering based on complete-link grouped visually similar shots to scenes. Sequential relations between shots rendered a scene transition graph, whose edges represented linking strength between scenes. Weak links were identified and regarded as story boundaries.

Similarity-based methods look at pair-wise similarities for abrupt changes that may mark boundaries. There is not a global picture of the whole media sequence. By contrast, Phung et al. [73] [72] [74] proposed a few functions to project all units into a plot and found segment boundaries based on the ebb and flow of each function. For their target genre of instructional video, they devised *thematic function* and *density function* to capture film makers' means of guiding viewers' attention. They hypothesized that these functions could reflect changes in topics and subtopics, respectively. Based on film-making grammar of instructional video, the expressive functions were defined as combinations of film-making artifacts such as shot length, motion activity, and frequency of narrator's appearance.

Segmentation can also be reviewed as a classification - whether a candidate point is a boundary? In this view, machine learning algorithms for classification were employed. For example, Hsu et al. [43] tried to compute the posterior probability of a point being a boundary given a number of features using maximum entropy model. The features were acoustics, speaker identification, presence of face(s), textual overlays, motion, A/V combination, and cue phrase. They reflected local characteristics surrounding the point under examination. Thus each point was viewed as an independent sample drawn from the distribution to be estimated. This practice was similar to that in [35]. The author also smoothed the posterior probabilities of candidate points by dynamic programming in the hope of modeling transition probabilities.

The aforementioned methods could not reflect recurrent nature of video structure. In recognition of HMMs' capability of modeling recurrence, they were employed. There have been three flavors of using HMMs. The first is modeling segments with individual HMMs. HMMs are first trained using full-fledged segments, they are then applied to classify fixed-length windows to labels and finally HMM labels are concatenated by dynamic programming [98] [101]. This practice has a weakness, namely that the test segments are temporally constrained and may not match models trained from full-fledged samples. The second approach was to model the whole video sequence using a HMM with certain states or state transitions marking critical points of structures [56] [55] [18]. Strictly speaking, this practice involves assigning labels to states, thus the states are no longer "hidden". The third approach modeled segments with individual HMMs, and also modeled transition among segments with a higher-level HMM, i.e. using a hierarchical HMM (HHMM). In this way, individual segments are free to end in the test sequence and states can be kept hidden [97] [101] [102] [51]. There are also other variants of HHMM that differ on whether learning is supervised or unsupervised and whether HHMM network structure is adaptable [97] [99].

2.3 Related Works on Multi-Modality Analysis

Humans understand multimedia documents by receiving information from multiple modalities. Imagine a silent telecast of a soccer match or merely soundtrack containing commentators' speech and stadium noise but with no pictures. Information of different modalities are complementary and analysis on multiple modalities is expected to outperform that on any single modality. Much effort has been devoted to analyzing multi-modal information in a range of tasks: annotation [2], retrieval [48] and search [41] [4] etc. Though these tasks all center around semantic analysis, they vary slightly in requirement or priority. To simplify discussion, we will focus on semantic inference from multi-modal information, i.e. annotation. Some example works in this field are [103] on video + audio (acoustic features) , [5] on video + audio (linguistic features from speech recognition) , and [112] on video + textual overlay.

Multiple modalities as a whole have a few properties that may influence the design of integrated analysis. First, the underlying modalities have diverse spatiotemporal layouts. For example, video frames (images) are available every 1/25 or 1/30 seconds, whereas speech occurs sporadically. Dictated by this property, features from different modalities may have different temporal granularity and may not have consistent spatiotemporal correlation. This gives rise to the issue of *asynchronism*. Second, features from different modalities may have varying degrees of correlation. For example, color histograms of a image may probably be more correlated than color histograms and acoustic features together in indicating a image frame's semantics. If features were to be grouped or consolidated to a smaller number to tackle the curse of dimensionality, highly correlated features may need to be put together. This gives rise to the issue of *feature consolidation* or *reduction* [93]. Third, different modalities may have varying descriptiveness regarding semantics. For example, reliable speech recognition is usually more descriptive than audiovisual signals when it comes to objects or events [18] [20]. This gives rise to the issue of *fusion of classifiers*. These issues have been the underlying factors that a multi-modal framework designer has to address. We sort our reviews of existing multi-modal frameworks by pipeline that reflects designers' overall thoughts regarding the listed issues. Note that most existing multi-modal frameworks involve fusion of features or classifiers. However, we will not focus on fusion schemes in this Section, as this is the topic of Section 2.4. Largely speaking, multi-modal frameworks fall in three types of pipelines - *intertwined analysis*, *fusion of parallel analysis*, and *unified analysis*.

Intertwined analysis. This category of methods is characterized by the fact that cues from single modalities only partially constrain the process of semantic inference. Often analysis on each modality is carried out in different step, therefore there would be multiple steps, hence the analysis is called "intertwined". Zhang et al. [112] involves both textual overlay and video analysis. Analysis on textual overlay detected score changes and inferred occurrence of tennis events based on domain models, in particular, transition graph model. In the locality provided by textual overlay detection, video analysis found boundaries of "pitch event segments" by matching images to templates of canonical pitching scenes. Ariki et al. also worked on baseball video and took a similar pipeline. They found rough locations of events by key phrases in automatic speech recognition (ASR) transcript, and refined the event boundaries by visual cues.

Babaguchi et al. [14], [8] [10] [68] proposed a family of methods to utilize closed caption and audiovisual signals to detect American football events. The paradigm was a series of analysis on closed caption and on image sequence. Analysis on closed caption found segments corresponding to events with the help of textual event models (event-indicating chains of key phrases). Video shots in temporal correspondence with these closed caption segments were regarded as candidate event shots. Image analysis further calculated distance of these video shots to pre-defined image models of events. Another group of papers by Babaguchi et al. [9] [13] [11] used gamestats on the Web in place of closed caption. Gamestats are more accurate in terms of indicating occurrence of events as well as their timing. Thus determination of event boundaries is reduced to reading match time on the textual overlay. They also developed a few spinoffs [62] [67] wherein mapping between closed caption and event labels was established by supervised learning. Fixed-length closed caption units were represented by key phrase vectors and were labeled as positive or negative instances of an event by k-Nearest Neighbors (KNN) in [62]. Closed caption segments broken up by speaker change or long pause were represented by speech related features such as speaker identification and number of sentences. Then closed caption segments were classified to one of the four scenes: live, others-game-related, commercial, and others-game-unrelated using a Bayesian network (BN). Finally a story was formed from a scene sequence [67]. Having rough locations of events or stories, the rest of the pipeline was largely the same as the paradigm - to refine boundaries with the help of video shots' boundaries. In view of possible accuracy loss incurred by KNN algorithm, there was also a screening step described in [62] that used short-term energy (STE) as the criterion.

Intertwined analysis captures correlations across multiple modalities in an explicit manner. This approach would preserve obvious correlations well, however, it may fail to discover correlations that are subtle or rest on statistical basis in large data sets. Due to this reason, this approach might not work well when the range of semantics of interest gets beyond a selected few. Judicious design of pipeline would help in taking advantage of reliable classifiers while containing poorly performing ones. Asynchronism is tackled by separate processing on asynchronous modalities.

Fusion of parallel analysis. This pipeline is characterized by a number of parallel analysis each of which works on a subset of all features available and provides individual results. All such results are then fused in a dedicated fusion step. In an effort to identify topical events from lecture videos showing slides, Syeda-Mahmood et al. [85] detected video and audio events from image and audio streams respectively, and used a probabilistic model to fuse them. The task of finding topical events was defined as finding a span of time when both slides and the lecturer's speech were on a certain topic. With slides used in the lecture available, video events were detected by identifying each slide being shown. A technique called region hashing was employed to recognize drawn objects on slides by their spatial layout, which in turn helped recognize the slide being shown. Audio events were detected by recognizing indicative words appearing in slides from the lecturer's speech. The fusion was essentially to combine curves representing probabilities of relevance as suggested by individual detectors. The authors defined integrated probability of relevance from the individual ones and found topical events by ebb and flow of the integrated value. Similarly, Rui et al. [75] arrived at the probability that a baseball video segment was exciting from two probabilities associated with excited commentator's speech and hit, respectively. They experimented with two fusion schemes, namely weighted sum with weights heuristically determined, and conditional fusion based on statistics of training data.

Fusion of parallel analysis may be useful for detecting a wide range of concepts over a large data set where a systematic approach is required and relevant features may be too bulky to be processed together. High-level feature detection as part of TRECVID organized by NIST [2] is such a task. It aims to detect dozens of concepts from over 300 hours of news video. Amir et al. [3] [4] build for each concept three SVM classifiers involving different language models and visual features. Decisions of the classifiers were later fused to generate the final list. Similarly, Hauptmann et al. [41] fuse decisions from ASR- and timing-based classifiers.

Fusion of parallel analysis is usually implemented by machine learning techniques. By splitting features to multiple classifiers, it allows a large number of features to be considered but at different times. Thus it is suitable for detecting a wide range of semantics and when little is known about salient features or patterns. This approach handles synchronism with varying degree of correlation by processing heterogeneous features separately. However, choosing a fusion scheme that is consistently effective is hard[40].

Unified analysis. In the pipeline of this type, all features across modalities are processed in a unified framework. The framework serves the fusing purpose and produces the final results.

A straightforward scheme of unification is to concatenate all features into a grand vector and apply some machine learning algorithms to this feature space [44] [80] [18]. Chaisorn [18] represented video shots by a vector composed of features describing image, textual overlay, audio, motion and temporal characteristics and performed shot type classification by decision tree. Hsu et al. [44] and Snoek et al. [80] experimented with inputting raw and wrapped multi-modal features to SVM classifiers. This approach preserves all potential correlations between features from different modalities. However, it has two major drawbacks. The first drawback is that the number of features may grow out of control as a result of modality merging. This is why the chosen features are usually of low dimension but highly informative. For example, [18] used number of faces in the image, audio genre, and whether textual overlay is at the center of the image; while [82] employed ratio of pixels classified to a concept. To reduce dimensionality while preserving correlations, Wu et al. [93] proposed a 2-step framework wherein features were first agglomerated to independent groups called modality (note the the term of modality is different than conventional definition), and then classifiers associated with different modalities were fused using SVM-based super-kernel fusion. The second drawback is that features may not be on the same numerical scale as a result of employing diverse underlying mechanisms and thus the feature space may be distorted unless a good normalization scheme is in place.

Features may also be assimilated in a sequential manner - one feature at a time. Some statistical models accommodate this learning scenario, including maximum entropy models and boosting. Maximum entropy model [16] [35] [43] [81] [45] [79] constructs an exponential log-linear function to approximate the posteriori probability of an event (i.e., presence of a semantic concept) given a number of binary features. The maximum entropy principle ensures that the estimated model describing posteriori probability agrees with the training data the best. The construction process includes two main steps - parameter estimation and feature induction. Parameter estimation indicates how much each active feature contributes to the model. Feature induction selects the candidate feature which, when adjoined to the set of active features, produces the greatest increase in likelihood of generating the training data. Maximum entropy method provides a nice scheme to select features and to fuse them. Boosting approaches have been successfully used to improve classification performance by fusing multiple weak classifiers [31]. When each feature was treated as a weak classifier, boosting was introduced to multimedia classification [87]. Different from maximum entropy models, boosting is not a generative model in that it does not estimate probabilistic models. However, boosting does employ a fusion formula that is a linear combination of features. Features' associated weights suggest their contribution to reducing training errors. Learning is an optimization process wherein features' weights are updated according to how much they could reduce the training errors. Both maximum entropy model and boosting are algorithmically optimization processes, thus they can be carried out in a sequential fashion. And sequentiality brings some implications with regards to feature unification. The two drawbacks of concatenated feature vector, namely excessively high dimensionality and distorted feature space are no longer problems. Besides, inducing one feature at a time also gives maximum entropy model another advantage - capability of selecting optimal features in maximum likelihood sense. Though boosting assigns different weights to features and this property was taken advantage of in selecting features [44], it is generally not recognized as capable of feature selection (in the maximum likelihood sense).

Another probabilistic approach to infer semantics from a whole bunch of features is graphical models based on Bayes rule. For example, Wu et al. [94] created an influence diagram to accommodate a wide range of features including contextual information (location, time, camera parameters), holistic perceptual features (derived from color, texture and shape), local perceptual features (SIFT-based features) and semantic ontology.

There is a body of works that establish correlation between modalities such as image and text for multimedia annotation or retrieval. Here is the classical scenario - correlation is established during training when both media content and textual labels are available and is applied on test data that has only media content. In a strict sense such works do not comply with our definition of “multi-modal analysis” as no text is available from test data. However, on consideration that establishing correlation between modalities may be helpful in truly multi-modal scenarios, we give a brief review of such works and cover them under “unified analysis”. This group of works is based on language models that rooted in the area of human language processing. The media content (images or video clips) may be described using a visual language of visterms (analogous to words), thus correlation between visterms and textual labels may be viewed as cross-lingual description. Mori et al. [63] established correlation between visterms and words by creating a co-occurrence table of visterms against words from the training set. Duygulu et al. [28] viewed this problem as analogous to that of machine learning, and used IBM translation models to solve it. Jeon et al. [49] viewed the problem as analogous to that of cross-lingual retrieval, and adapted relevance-based language models to compute the posteriori probability of the annotation given the image. They called the model Cross-Media Relevance Model (CMRM).

The last paragraphs on intertwined analysis, fusion of parallel analysis and unified analysis summarize multi-modal works with respects to the pipeline. Feature preparation wise, there are a few more issues.

One critical issue for video analysis is the judicious choice of features that can be effectively extracted and that are capable of distinguishing different semantic classes. Multiple modalities have expanded the pool of features by those derived from ASR, the Web, meta-data from capturing devices such as camera parameters or GPS. As feature pool grows big, abstracting low-level features to mid-level entities may be helpful in reducing dimensionality and in statistical modeling of semantic relationships at a higher level. Among popular features are "those related to people (face, anchor, etc), acoustic (speech, music, pitch, significant pause, etc), objects (image blobs, building, graphics, overlay text, etc), locations (indoor, studio, city, etc), genres (weather, sports, commercials, etc), and productions (camera operations, blank frames, etc)" [20].

Feature wrapping may be required by some algorithms such as maximum entropy models and boosting. These algorithms need homogeneous binary features yet raw multi-modal features usually have heterogeneous spatiotemporal characteristics. Raw features are wrapped by measuring the change over time, quantizing continuous values, or by telling logical predicates, e.g. if an anchor scene follows a significant pause [44]. Feature wrapping is a useful tool to tackle asynchronism between modalities. It even enhances performance of algorithms wherein it is not required, such as SVM [44].

Feature selection in a systematic manner is generally desirable as fewer features would enhance generalization and save computation. Maximum entropy models [35] and feature reduction techniques such as principle component analysis (PCA) and independent component analysis (ICA) [93] are standard means to this end. And they have been reported to enhance performance when other conditions were kept the same. For example, the same number of features picked by maximum entropy model outperformed those under boosting [44]. Nevertheless, it was also found that SVM with no feature pruning still outperformed some other algorithm with selected features, namely maximum entropy models [44].

2.4 Related Works on Fusion Schemes

Many multi-modal systems have a dedicated step to fuse features or results of classifiers. This is especially the case for the works under the categories of "fusion of parallel analysis" and "unified analysis". This Section will review works with regards to fusion schemes. The research scope to be reviewed may be wider than multimedia analysis, including data fusion, sensor fusion, etc. As asynchronism between features is an important concern in our problem, we will categorize works into two groups, namely with no and with synchronization issue, respectively.

2.4.1 Fusion Schemes with No Synchronization Issue

As fusion of features has been reviewed previously, this Section focuses on fusion of individual classifiers. This includes fusion schemes based on basis decisions, on confidence values associated with decisions, and on modification of classifiers.

Fusion based on decisions attempts to establish the correlation between desired final decision and basis decisions. This can be done in a direct or indirect manner, sometimes learning is involved. Linear combination is a direct scheme and by far the most straightforward. The central problem of using linear combination is how to determine the weights. The most intuitive way is to fix weights empirically, as Rui et al. did in [75]. More principled approaches may involve learning or optimization. Yan et al. [105] learned weights by the EM algorithm in a way that the overall probability of classifiers producing decisions for the training data gets maximized. Maison et al. [59] determined weights by minimizing the error rate on the training data. Adaboost [38] and rankboost [23] can also be regarded as fusion schemes by linear combination with learned weights. A generalization to linear combination is ensemble fusion [3], which first formulates a number of candidate normalization functions (for normalizing basis decisions so that they can be compared) and fusion functions, then optimizes fusion performance over these candidates. In recognition of non-linearity of the correlation, some researchers chose non-linear models to perform fusion. For example, stacking SVMs were used as a super-kernel non-linear fusion on basis decisions[41] [95]. Indirect approaches include Bayesian classifier that finds correlation between desired final decision and basis decisions from conditional probabilities [60].

Confidence based fusion schemes includes validity weighting [3], which assigns weights to classifiers reflecting the number of positive training samples they have.

Sometimes, fusion is performed on the model or parameters of classifiers rather than on basis decisions. Wu et al. [95] modeled SVM classifiers as kernel matrices and the fused one as a new SVM with a new kernel matrix. The parameters are linear combination of those from the original kernel matrices with the weights minimizing generalization error.

2.4.2 Fusion with Synchronization Issue

This Section reviews works on fusing features or decisions that are asynchronous.

One group of methods is to manipulate asynchronous probabilities directly. Syeda-Mahmood et al. [85] aimed to tell probability of a topical event occurring at a given instant from two probabilities suggested by video and audio detectors. The authors defined the fused probability as the sum of the basis probabilities minus their product (to prevent overflow). The definition was not standard, nevertheless it met the requirements of a probabilistic value and worked.

Another group of methods wrap features or decisions to be fused with temporal relation operators. Usually a fuzzy window is involved. Snoek et al. [81] needed to fuse features from closed caption, visual and acoustic modalities. They wrapped features using Allen time interval relations, i.e. *precedes*, *meets*, *overlaps*, etc, and formed binary predicates based on wrapped features. The predicates were fed into a maximum entropy classifier. Hsu et al. [45] used fuzzy windows when relating intermediate outcomes from different modalities.

The third group of methods use some meta-classifiers that ease the requirement on temporal correspondence. Xie et al. [100] proposed a layered dynamic mixture model to discover patterns from news video. First, signals within individual information streams were mapped to labels by supervised learning algorithms. These labels were asynchronous, even though they may refer to the same semantic topics. Then related labels from different streams were fused under a loose temporal bag referring to a topic. The fusion was conducted by the EM algorithm that maximized probabilities of labels being observed.

2.5 Related Works on Incorporating Handcrafted Domain Knowledge to Machine Learning Process

Machine learning may be a principled solution towards building domain models. Handcrafted or prior domain knowledge may facilitate this process. The facilitation may be in choosing learning algorithms, choosing features or tuning parameters.

Recent video retrieval systems have shown that it may be a feasible idea to classify queries into pre-defined classes and develop fusion models by taking advantage of the prior knowledge and characteristics of each query class. Yan et al. [105] considered four query classes of type named person, named object, general object and scene, and explored a 2-level hierarchical query-dependent fusion model that emphasizes text features. Chua et al. [24] further explored the use of external knowledge, specialized detectors and pseudo relevance feedback in a single-level query-dependent model with six query classes of person, sports, finance, weather, disaster and general.

Graphical models, in particular Bayesian networks, are a commonly used tool for representing knowledge. They are designed to reflect dependencies between variables, and they provide a way to describe strength of the dependencies by conditional probabilities. Intille et al. [47] used Bayesian networks to model relationships among play category, players' trajectories and ball trajectory in American football video. Wu et al. [94] tried to annotate photos by fusing contextual information (location, time, and camera parameters), visual content (holistic and local perceptual features) and semantic ontology with a graphical model. Prior knowledge helped to substantially reduce the hypothesis space to search for the right model, e.g. exposure time depend on time thus there is an arc connecting them.

Chapter 3

PROPERTIES OF TEAM SPORTS

3.1 Proposed Domain Model

As we have seen in review on event detection (Section 2.1), an effort to detect events is essentially that to establish and enforce a domain model. The domain model reflects on how we view the domain with regards to the task and how this view affects our choice of analysis techniques. We describe our domain model in this Chapter as the basis for detecting events. Note that what is described here is essentially a "template" domain model that needs to be instantiated with domain knowledge to cater to a particular game.

Common to all team sports videos is a distinct characteristic - alternating *advances* towards two opposite targets, e.g. goalposts.

Definition 4 *Advance*
An advance is a continued movement towards one end of the field.

Definition 5 *A draw*
Draw refers to the state when both teams are contending in the middle of the field and none of the teams are on the offensive.

Definition 6 *Break*
A break refers to the state when play is not going on, for example, before the match starts or after it ends, or when the ball gets out of bound and thus dead.

Definition 7 *Phase*
A phase is a segment of the match that corresponds to certain state of the game. A phase can be an advance, draw or break.

This characteristic is the basis of a generic model for different games. Recognizing the video structure in the perspective of *advance* is helpful to event detection, as this structure helps in locating events. For example, score-related events happen mostly at the ends of the *advances*, e.g., *goals* in soccer; launching of plays in the beginnings of *advances*, e.g., *punt-returns* in American football, and referee interventions between *advances*, e.g., *yellow-card* in soccer.

The proposed model comprises four parts describing different aspects of team sports - *temporal location specification*, *sequential relationship among events*, *semantic composition* and *audiovisual patterns.*

Temporal location specification
Events' temporal locations are specified under the structure of team sports video. This structure is described in the perspective of *advance.* Specifically, it is modeled as a finite state machine with the states of *advance in the left direction*, *advance in the right direction*, *draw* and *break* (Figure 3.1). *Draw* describes the state when no team is on the offensive; this happens in free-going games e.g. soccer. *Break* refers to the state when the ball is dead or the video is not showing on-going play. A video segment that stays on a particular state is called a *phase.*

Instances of each event type only occur at certain phases; the conditions of such phases are the events' *temporal location specification.* For example, the temporal location specification for *corner-kick* is: it only occurs in an *advance* that is preceded by an *advance* - *break* sequence, with the two *advances* in the same direction.

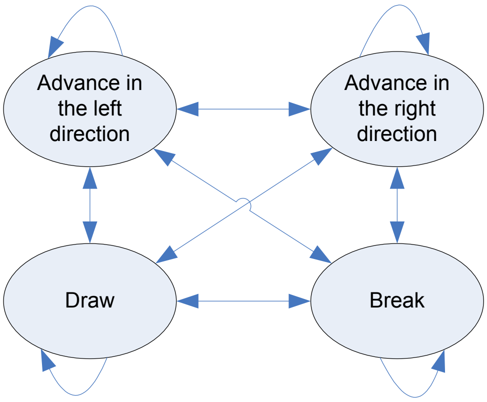


Figure 3.1: The structure of team sports video in the perspective of *advance*

Intermittent matches usually have substructures within a phase. For instance, in American football, an *advance* is composed of a series of plays (called downs). For these games, temporal locations may be specified on units at a level lower than phase.

Sequential relationship among events

Games are guided by rules and tactics, and thus events may exhibit certain sequential patterns. Capturing the sequential relationship among events may help in detecting events when some cues are missing or ambiguous.

Semantic composition

Semantically, an event is composed of a series of *actions*, wherein an action refers to a single interaction between players or between players and context during the development of the event.

A semantic composition model of an event type can be represented by an directed graph

$$\mathcal{G} := \langle \mathcal{A}, \mathcal{T} \rangle \tag{3.1}$$

where $\mathcal{A}$ is the set of nodes representing actions with attributes and T is the set

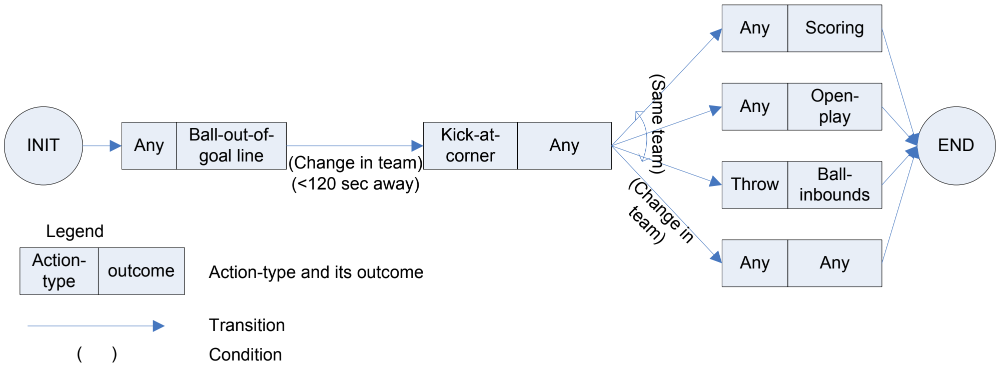


Figure 3.2: Semantic composition model of *corner-kick*.

of directed edges representing temporal transitions between actions. Each node $\phi \in \mathcal{A}$ has these attributes

$$\phi := \langle \mathsf{action\_type}, \mathsf{outcome}, \{\mathsf{pre\_tran}\}, \{\mathsf{post\_tran}\} \rangle \tag{3.2}$$

where action_type describes what action the player takes, outcome describes the state of the ball or of related players or of the overall match as a result of the action; $\mathsf{pre\_tran} := \langle \mathsf{pre\_node}, \mathsf{pre\_cond} \rangle$ describes a transition from the preceding node into the current node and the conditions to be met during the transition; and $\mathsf{post\_tran} := \langle \mathsf{post\_node}, \mathsf{post\_cond} \rangle$ describes a transition from the current into the next node; $\{\mathsf{pre\_tran}\}$ and $\{\mathsf{post\_tran}\}$ are the sets of pre- and post-transitions. The graph contains a number of paths connecting INIT and END. Any of these paths is a valid development of the event. For each specific event, the paths usually go through a common node, which helps to distinguish the event type from others, such as the node kick-at-corner in Figure 3.2. Due to its distinction and consistency, it is called the key action of the event type.

Audiovisual pattern

Structural units at various granularities, including phases, plays, and events, may have audiovisual patterns. This makes it possible for them to be detected in the feature space. Capturing these patterns may be done by a heuristic or machine-learning approach. Some event types have relatively distinctive and consistent audiovisual patterns, such as *goal* in soccer; while others do not, such as *offside* in soccer. Generally speaking, events' audiovisual patterns are less consistent than semantic composition models.

The proposed model can be enforced if the audiovisual signals and semantic-rich external information sources are available. Section 3.3 will describe these two forms of information.

3.2 Domain Knowledge Used in Both Frameworks

Domain knowledge instantiate a generic domain model and is the kernel that enables a system to cater to a new game. For our task of event detection, the domain knowledge involved is mainly about game rules, player database, event models, and special visual effects. Specifically,

- *Game rules* are the rules that regulate the match. Most importantly, we need to know the duration of the match and the field zones.
- *Player database* stores players' names, affiliation and positions, which helps in inferring possession or the direction of an *advance* from external information sources.
- *Event models* define the target event types in a particular game and specify these four aspects of events - temporal location specification, sequential relationship among events, semantic composition and audiovisual patterns.
- *Special visual effects* refer to production artifacts such as the channel logos used to indicate replays, and canonical views that are associated with certain content in the game, like the view of the goalpost in American football. Knowledge of the special visual effects facilitates audiovisual analysis.

Although the domain knowledge involved covers a wide range of aspects, much of it can be acquired automatically. We categorize domain knowledge into three types based on the amount of manual labor required during acquisition (See Figure 3.3).

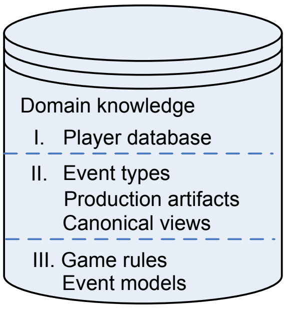


Figure 3.3: Various levels of automation in acquiring different parts of domain knowledge.

Acquisition of type I domain knowledge, the player database, can be fully automated. Player information can be extracted by issuing questions about teams and players to a FADA question answering engine [106] which searches on the league's and clubs' official websites [106].

Type II domain knowledge is automatically acquired, but needs human intervention to confirm or annotate. Among this type are target event types, production artifacts, and canonical views. Target event types are picked manually from a suggested list that is extracted from Web corpus in the same way as player information. Canonical views and production artifacts are acquired by automatic clustering of video contents and subsequent manual annotation.

Domain knowledge in type III, i.e., game rules and event models, is manually built. However, efforts in building event models can still be partially automated. Field zones are recognized by Bayesian classifiers. Action types and outcomes are acquired the same way target event types are acquired. The audiovisual patterns of some event types are acquired by machine-learning approaches, such as SVM.

Note that the domain knowledge in type III is the most stable part in a game, while that in type I and II is more “volatile”. Our data driven techniques for type I and II minimize ad hoc efforts in acquiring domain knowledge and thus enhance adaptability of the system.

For the testing games - soccer and American football, the domain knowledge acquired is as follows: soccer’s phase types are left-advance, right-advance, draw and break; target event types are goal, save, shot-off-target, penalty, free-kick, corner-kick, yellow-card, red-card, substitution and offside; field zones are left-lower-corner, left-upper-corner, left-third, middle-field, right-lower-corner, right-upper-corner and right-third.

American football has only two phases - left-advance and right-advance since time-outs are removed and there must be a team on the offensive at any moment; target event types are touchdown, conversion, field-goal, punt, punt-return, fumble-opponent, interception, touchback, kickoff, safety, fumble-own, incomplete-pass; field zones are left-lower-corner, left-upper-corner, left-twenty-yard-area, middle-field, right-lower-corner, right-upper-corner and right-twenty-yard-area.

3.3 Audiovisual Signals and External Information Sources

Enforcement of the domain model involves two sources of information - audiovisual signals and external information sources. This Section describes their properties and the main challenge in integrating them, namely the asynchronism between them.

3.3.1 Audiovisual Signals

Definition 8 *Audiovisual signals*
Audiovisual signals refer to the image stream and soundtrack produced by broadcasting professionals.

The image stream consists of evenly paced pictures (called frames) edited from one or more cameras; and the soundtrack is commentators' voice in the background of stadium noise and audience sound. Generally, sports TV producers observe certain rules-of-thumb in shooting and adopt some cinematographic techniques in editing [78], therefore the audiovisual signals may exhibit some video syntax.

Reviews on related works (Section 2.1) suggest that events can be indicated by audiovisual patterns, particularly in some constrained scenarios. This is because some level of correspondence exist between groups of events and audiovisual patterns. However, analysis solely based on audiovisual signals generally performs poorly. This seems to suggest the correspondence is not reliable, in other words, indicating audiovisual patterns are not really distinctive or consistent. In our task of detecting events from team sports video, reliability of audiovisual patterns vary from event to event. Some events virtually have no distinctive patterns. For example, an *offside* in soccer may appear like a regular cross. Similar drawback exists for actions. Most actions suffer from lack of reliable audiovisual patterns. This may be because actions are more specific with regards to semantics - they involve interactions between semantic entities (players or contextual objects). These semantic entities and their relations can hardly be mapped to consistent audiovisual patterns. Detecting the ball reliably is already a hard problem, let alone detecting it flying across the goal-line. Identifying actions probably needs more semantic-rich information than simply audiovisual signals can provide.

Despite the unreliability in indicating semantics, audiovisual signals are good at pinning time points. This would be beneficial to identifying boundaries of video structural units. First of all, audiovisual signals enable the indexing of individual frames. The frame, being the atomic temporal unit, can specify the boundaries precisely. Second, analysis on audiovisual signals could recognize artifacts such as camera motion and commentators' speech and hence able to find natural breakdown points as boundaries.

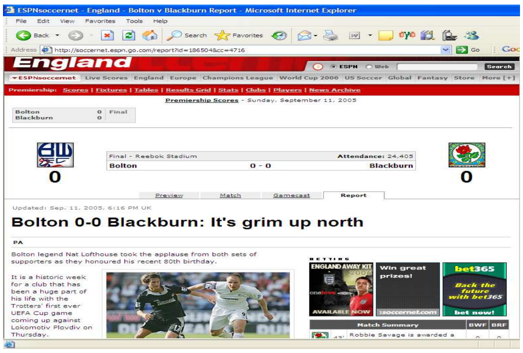
3.3.2 External Information Sources

Definition 9 *External information source*
External information sources refer to textual descriptions that do not come embedded with the audiovisual stream.

Textual information extracted from audiovisual stream such as speech recognition transcripts (ASR) is not considered as external information. Currently, external information sources are prevalent, it is generated by human and can be found on the Web, newspaper, TV or radio. Some typical external information sources are:

- match reports in newspapers and on the Web;
- studio reviews at half-time and end of the match;
- live commentary on the Web;
- live game logs on the Web, such as what ESPN provides for the English Premier League (EPL) [1].

Figure 3.4 - 3.7 show some snapshots of external information sources from the Web. Despite the various sport domains and formats, we can categorize the external information sources into two levels: compact and detailed.
• *Compact descriptions* are after-match summaries covering the full time match. Examples are match reports for soccer (Figure 3.4) and recaps in American football (Figure 3.5). A typical piece of compact description would document only a few key events pivotal to the course of the match. Documentation of an event would generally give names of players involved, their activities and outcomes. The temporal granularity of the compact descriptions is generally minute rather than second. The documentation of events is accurate in the sense that human users


ESPNsoccernet - England - Bolton v Blackburn Report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://soccernet.espn.go.com/report?id=186504&cc=4716
Go
England
ESPN Web Search
ESPNsoccernet Live Scores England Europe Champions League World Cup 2006 US Soccer Global Fantasy Store More [+]
Premiership: Scores | Fixtures | Tables | Results Grid | Stats | Clubs | Players | News Archive
Preview Match Gamecast Report
BETTING
ENGLAND AWAY KIT 2006
onelove
AVAILABLE NOW
Win great prizes!
soccernet.com
bet365
Back the future with bet365
bet now!

Premiership Scores - Sunday, September 11, 2005

Bolton	0	Final
Blackburn	0	

Final - Reebok Stadium		Attendance: 24,405
Bolton	0 - 0	Blackburn

0 0

Updated: Sep. 11, 2005, 6:16 PM UK

Bolton 0-0 Blackburn: It's grim up north

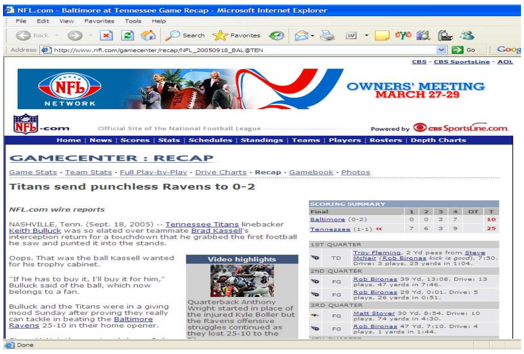
PA

Bolton legend Nat Lofthouse took the applause from both sets of supporters as they honoured his recent 80th birthday.

It is a historic week for a club that has been a huge part of his life with the Trotters' first ever UEFA Cup game coming up against Lokomotiv Plovdiv on Thursday.

Match Summary	BWF	BRF
43' Robbie Savage is awarded a		

Figure 3.4: Example of soccer match report (Source - http://soccernet.espn.go.com/).


NFL.com - Baltimore at Tennessee Game Recap - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.nfl.com/gamecenter/recap/NFL_20050918_BAL@TEN
Go
CBS - CBS SportsLine - AOL
NFL NETWORK
OWNERS' MEETING MARCH 27-29
NFL.com
Official Site of the National Football League
Powered by CBS SportsLine.com
Home | News | Scores | Stats | Schedules | Standings | Teams | Players | Rosters | Depth Charts
Video highlights
Done

GAMECENTER : RECAP

Game Stats · Team Stats · Full Play-by-Play · Drive Charts · **Recap** · Gamebook · Photos

Titans send punchless Ravens to 0-2

NFL.com wire reports

NASHVILLE, Tenn. (Sept. 18, 2005) -- Tennessee Titans linebacker Keith Bulluck was so elated over teammate Brad Kassell's interception return for a touchdown that he grabbed the first football he saw and punted it into the stands.

Oops. That was the ball Kassell wanted for his trophy cabinet.

"If he has to buy it, I'll buy it for him," Bulluck said of the ball, which now belongs to a fan.

Bulluck and the Titans were in a giving mood Sunday after proving they really can tackle in beating the Baltimore Ravens 25-10 in their home opener.

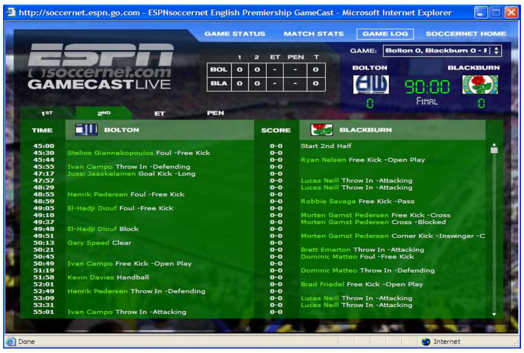
Quarterback Anthony Wright started in place of the injured Kyle Boller but the Ravens offensive struggles continued as they lost 25-10 to the

SCORING SUMMARY

Final	1	2	3	4	OT	T
Baltimore (0-2)	0	0	3	7		10
Tennessee (1-1) «	7	6	3	9		25

1ST QUARTER		
	TD	Troy Fleming, 2 Yd pass from Steve McNair (Rob Bironas *kick is good*), 7:50. Drive: 3 plays, 25 yards in 1:04.
2ND QUARTER		
	FG	Rob Bironas 39 Yd, 13:08. Drive: 13 plays, 47 yards in 7:46.
	FG	Rob Bironas 29 Yd, 0:01. Drive: 5 plays, 26 yards in 0:51.
3RD QUARTER		
	FG	Matt Stover 30 Yd, 8:54. Drive: 10 plays, 74 yards in 4:30.
	FG	Rob Bironas 47 Yd, 7:10. Drive: 4 plays, 1 yards in 1:44.

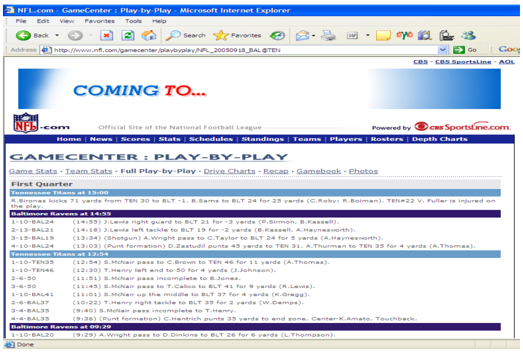
Figure 3.5: Example of American football recap (Source - http://www.nfl.com).


http://soccernet.espn.go.com - ESPNsoccernet English Premiership GameCast - Microsoft Internet Explorer
GAME STATUS
MATCH STATS
GAME LOG
SOCCERNET HOME
ESPN soccernet.com GAMECAST LIVE
GAME: Bolton 0, Blackburn 0 - F
BOLTON
BLACKBURN
90:00
FINAL
0
0
1ST 2ND ET PEN
Done
Internet

	1	2	ET	PEN	T
BOL	0	0	-	-	0
BLA	0	0	-	-	0

TIME	BOLTON	SCORE	BLACKBURN
45:00		0-0	Start 2nd Half
45:30	Stelios Giannakopoulos Foul -Free Kick	0-0	
45:44		0-0	Ryan Nelsen Free Kick -Open Play
45:55	Ivan Campo Throw In -Defending	0-0	
47:17	Jussi Jaaskelainen Goal Kick -Long	0-0	
47:57		0-0	Lucas Neill Throw In -Attacking
48:29		0-0	Lucas Neill Throw In -Attacking
48:55	Henrik Pedersen Foul -Free Kick	0-0	
48:59		0-0	Robbie Savage Free Kick -Pass
49:05	El-Hadji Diouf Foul -Free Kick	0-0	
49:10		0-0	Morten Gamst Pedersen Free Kick -Cross
49:37		0-0	Morten Gamst Pedersen Cross -Blocked
49:48	El-Hadji Diouf Block	0-0	
49:51		0-0	Morten Gamst Pedersen Corner Kick -Inswinger -C
50:13	Gary Speed Clear	0-0	
50:21		0-0	Brett Emerton Throw In -Attacking
50:45		0-0	Dominic Matteo Foul -Free Kick
50:49	Ivan Campo Free Kick -Open Play	0-0	
51:19		0-0	Dominic Matteo Throw In -Defending
51:58	Kevin Davies Handball	0-0	
52:01		0-0	Brad Friedel Free Kick -Open Play
52:49	Henrik Pedersen Throw In -Defending	0-0	
53:09		0-0	Lucas Neill Throw In -Attacking
53:31		0-0	Lucas Neill Throw In -Attacking
55:01	Ivan Campo Throw In -Attacking	0-0	

Figure 3.6: Example of soccer game log (Source - http://soccernet.espn.go.com/).


NFL.com - GameCenter : Play-by-Play - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.nfl.com/gamecenter/playbyplay/NFL_20050918_BAL@TEN
Go
CBS - CBS SportsLine - AOL
COMING TO...
NFL.com
Official Site of the National Football League
Powered by CBS SportsLine.com
Home | News | Scores | Stats | Schedules | Standings | Teams | Players | Rosters | Depth Charts
Done

GAMECENTER : PLAY-BY-PLAY

Game Stats · Team Stats · **Full Play-by-Play** · Drive Charts · Recap · Gamebook · Photos

First Quarter

Tennessee Titans at 15:00

R.Bironas kicks 71 yards from TEN 30 to BLT -1. B.Sams to BLT 24 for 25 yards (C.Roby; R.Boiman). TEN#22 V. Fuller is injured on the play.

Baltimore Ravens at 14:55

```
1-10-BAL24    (14:55) J.Lewis right guard to BLT 21 for -3 yards (P.Sirmon, B.Kassell).
2-13-BAL21    (14:18) J.Lewis left tackle to BLT 19 for -2 yards (B.Kassell, A.Haynesworth).
3-15-BAL19    (13:34) (Shotgun) A.Wright pass to C.Taylor to BLT 24 for 5 yards (A.Haynesworth).
4-10-BAL24    (13:03) (Punt formation) D.Zastudil punts 45 yards to TEN 31. A.Thurman to TEN 35 for 4 yards (A.Thomas).
```

Tennessee Titans at 12:54

```
1-10-TEN35    (12:54) S.McNair pass to C.Brown to TEN 46 for 11 yards (A.Thomas).
1-10-TEN46    (12:30) T.Henry left end to 50 for 4 yards (J.Johnson).
2-6-50        (11:51) S.McNair pass incomplete to B.Jones.
3-6-50        (11:45) S.McNair pass to T.Calico to BLT 41 for 9 yards (R.Lewis).
1-10-BAL41    (11:01) S.McNair up the middle to BLT 37 for 4 yards (K.Gregg).
2-6-BAL37     (10:22) T.Henry right tackle to BLT 35 for 2 yards (W.Demps).
3-4-BAL35     (9:40) S.McNair pass incomplete to T.Henry.
4-4-BAL35     (9:36) (Punt formation) C.Hentrich punts 35 yards to end zone, Center-K.Amato, Touchback.
```

Baltimore Ravens at 09:29

```
1-10-BAL20    (9:29) A.Wright pass to D.Dinkins to BLT 26 for 6 yards (L.Thompson).
```

Figure 3.7: Example of American football play-by-play report (Source - http://www.nfl.com).

> *Chelsea's dominance started as early as the fourth minute when Damien Duff's rasping daisy cutter was well held by Given.*
> *But the breakthrough came* ***in the 25th minute*** *from an unlikely source.*
> ***Wayne Bridge****'s left wing* ***cross*** *was missed by a host of players and the ball fell to full-back* ***Glen Johnson*** *who took his time before* ***blasting high past Given****.*
> *Fourteen minutes later the Blues doubled their lead as another attack wide on the left saw Duff whip in a cross from which Hernan Crespo tucked the ball home from six yards.*

Figure 3.8: Excerpt of a match report for soccer.

would generally be able to find the events in the proximity of the given time in the video. However, the complex structure and varying syntax of the description make extraction unreliable. Besides, documentation of events are generally incomplete. A typical soccer match report would cover some *goals*, a few remarkable *attempt-on-goals*[1] and some other events such as serious *fouls* or *substitutions*. Statistics show that about $10 \sim 20\%$ of all events are documented. Figure 3.8 shows an excerpt of a match report for soccer.

- *Detailed descriptions* are generally live textcasts on the Web. They provide realtime update for viewers to follow the match closely. Examples are game logs for soccer (Figure 3.6) and play-by-play reports for American football (Figure 3.7). Documentation of happenings are organized in time-indexed entries. The temporal granularity of entries is in seconds, which is at the same level as that of actions. Each entry would provide information on one or a few actions, including actions performed, involved players, and outcomes. A large number of significant actions are documented in detailed descriptions. However, most time stamps do not agree with those found in the video. This phenomenon is called asynchronism between the audiovisual signals and external information sources. More detail about the asynchronism will be given in Section 3.3.3.

External information sources and audiovisual signals are complementary in detecting events. On one hand, external information sources provide consistent semantic-rich information that audiovisual signals lack. They provides interac-

[1] *Attempt-on-goal* is the union of *shot-off-target* and *save*.

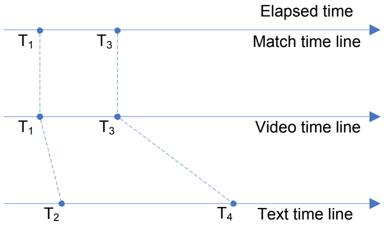


Figure 3.9: Formation of offset - continuous match

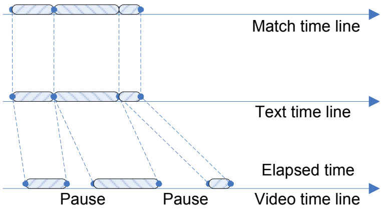


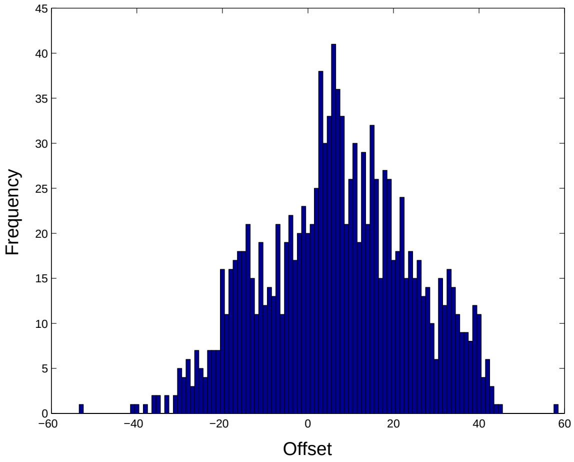
Figure 3.10: Formation of offset - intermittent match

tions between objects (players, the ball, the pitch, the goalpost, etc), which help to identify actions in the semantic composition models. It may also provide subtle semantic entities directly, such as occurrences of *foul* or *offside* in soccer. This semantic-rich information cannot be reliably detected by audiovisual analysis. On the other hand, audiovisual signals are good at pinning events' boundaries, at which external information sources are poor. Therefore, it would be desirable to perform an integrated analysis exploiting strengths of both.

3.3.3 Asynchronism between Audiovisual Signals and External Information Sources

The asynchronism refers to the phenomenon in which a time instant is recorded differently in audiovisual and text time lines. The asynchronism obscures the temporal correspondence between the two time lines and thus hinders the integration of audiovisual signals and external information sources. The asynchronism has different causes with compact and detailed descriptions. Between audiovisual signals and compact descriptions, the asynchronism results from the use of different temporal granularity in describing events. Also, the text time line may describe happenings that are irrelevant to the events in video.

Causes of the asynchronism between detailed descriptions and audiovisual signals are different. In a continuous match, e.g. soccer or hockey, human operator may need some time before he/she can tell the type and outcome of an action or he/she may anticipate a sure-fire action before it actually happens. This is illustrated



Min = -52 s
Max = 58 s
Mean = 6.97
Std = 17.31

Lilliefors test for
goodness of fit to a
normal distribution
Alpha = 0.05
p = 0.017
Decision: hypothesis
rejected

Figure 3.11: Distribution of offsets in second

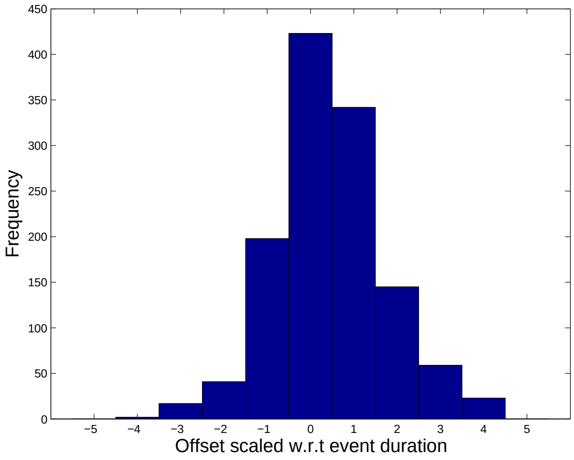


Figure 3.12: Distribution of offsets w.r.t. event durations

in figures 3.9. In an intermittent match, e.g. American football, the video is intermittent with timeouts (commercials, replays or narratives) between plays whereas text is continuous on the match time line, as illustrated in Figure 3.10. The difference in stamps of the same time instant is called *offset*.

The offset is random and could be large. Figure 3.11 shows distribution and some statistics of offsets obtained from the game logs of 5 soccer matches. It suggests that the value of offset is a random variable and does not follow a Gaussian distribution, though the distribution looks like one. The absolute value of offset could be as large as 50 seconds. The relative value (with respect to the duration of the event) could be as large as 3, although most text entries overlap or adjoin with the corresponding events. Random and non-trivial offsets make direct indexing of event segments based on time stamps given in the external information sources unreliable, especially when boundaries are considered. The problem requires more sophisticated effort.

3.4 Common Operations

There are some common steps before each framework is performed. They include parsing the video to units and timeout detection for American football. This Section describes these steps.

3.4.1 The Processing Unit

The conventional processing unit for video indexing is the shot. However, shot might not be a good choice for team sports video because the camera usually keeps tracking the activity for a long time. A shot may contain too many evolutions of actions to be called atomic. Another concern is the weak distinction between intra-shot evolutions and gradual transitions between shots, resulting in unreliable shot segmentation. Another option is a fixed length window [98]. However heterogeneousness in the video content makes it hard to find an appropriate length in a principled way. Arbitrary cut-off may result in an evolution of event being cut or unrelated content being grouped together.

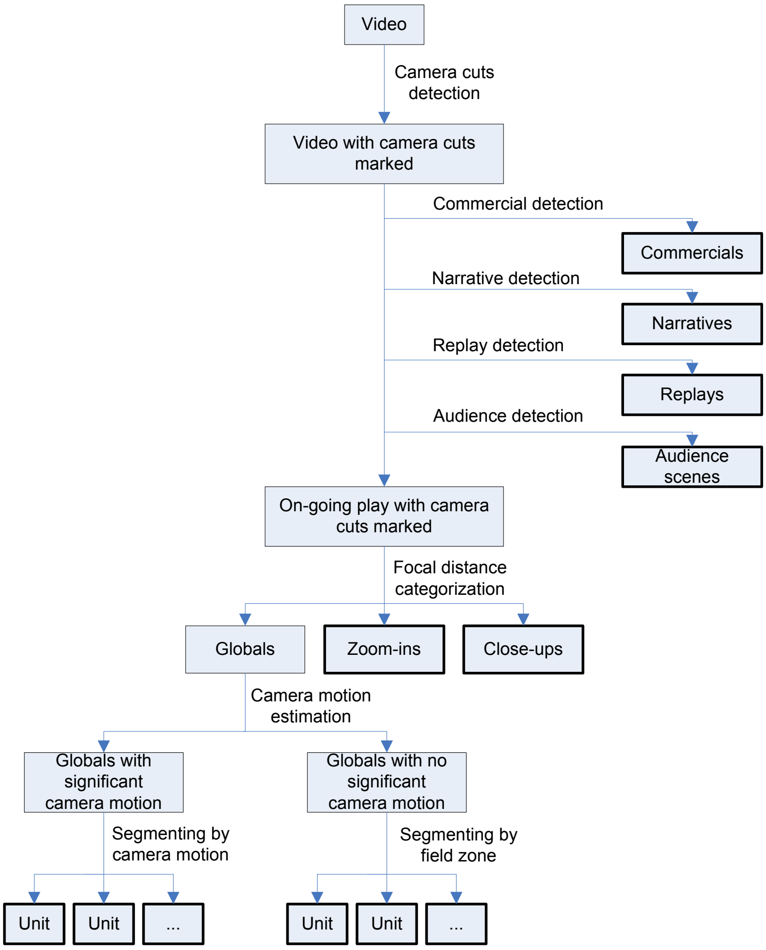


Figure 3.13: Parsing a team sports video to processing units

The chosen processing unit is the complete span of a camera motion. A more detailed description of the processing unit is as follows. A unit is a continuous video segment that is not intervened by camera cuts and conforms to one of the following criteria: 1) during on-going play, a unit is the complete span of a significant camera motion; 2) during video portions of on-going play with no significant camera motion, a unit is a continuous segment staying on the same field zone; and 3) otherwise, a unit is a complete content-homogeneous segment, e.g. a commercial or a replay. This choice ensures units having homogeneous content with natural boundaries. Figure 3.13 shows the procedure for parsing a video into units. The entities in thick-bounded boxes are generated units. Note that there are many steps involving detection of semantic entities, which will be explained in "3.4.2 Extraction of Features". Processing units of semantic content (commercials, narratives, replays, audience-scenes, zoom-ins and close-ups) are represented by their semantic labels, whereas other units are represented by audiovisual features.

3.4.2 Extraction of Features

Most features used in both frameworks are semantic entities. The benefit of this practice is to make the system more robust, to facilitate processing of high-level semantics and to save training samples. This practice was also adopted by Duan [26] and Lekha [18]. Extraction of these features and some basic processing steps are described in the following paragraphs, mostly based on standard techniques. Note that these features include those required in unit parsing steps.

- *Camera cut detection.* Change in chrominance and luminance exceeding a threshold is regarded as a cut. The threshold is adapted by statistics of this change in a fixed-length sliding window [36].

- *Commercial detection.* Commercials are detected by the presence of black frames, high cut rate, still frames, and audio silence [52].

- *Replay detection.* Replays refer to the video segments showing activities occurring a short while ago. In our experimental data, they are sandwiched by logos, we detect them by detecting logos (see narrative detection).

- *Narrative detection.* Narratives refer to those segments that provide briefing to viewers about the background information about the teams or players. They are characterized by stereotypical scenes showing stationary commentators or textual overlays. Since both logos and narratives are stereotypical scenes, they are detected by matching image sequences. Similarity between images is obtained by CCV histograms with contribution from perceptually similar colors [22]. Similarity between two image sequences is obtained by longest common subsequence algorithm [18].

- *Audience-scene detection.* Audience scenes are signified by rich edges. Richness of edge is simulated from DCT coefficients in a way similar to [53].
- *Focal distance categorization.* Cameraman use arrangement of globals, zoom-ins and close-ups to direct viewers' attention. They are differentiated by grass area ratio. An adaptive grass detector is in place by statistics of hues from random frames.
- *Camera motion estimation.* Camera motion parameters are zoom, tilt and pan factors. They build a projective transformation of which each pair of intercoded macroblocks is an noisy sample. The factors are estimated by regression [86]. After zoom, tilt and pan factors are obtained, the camera motion pattern is quantized to a small discrete label set {pan-left, pan-right, tilt-up, tilt-down, zoom-in, zoom-out, trivial}.
- *Field zone categorization.* Field zones are classified by naïve Bayes classifier based on features of field region shape, region area, region corner position, field line orientation and middle line position [6].
- *Audio genre categorization.* A series of SVM classifiers classifies each unit's audio content to one of these genres: whistle, cheering, speech, music, music+speech, excited-speech and noise. The audio features used for the classification are: zero-crossing rate (ZCR), spectral power (SP), mel-frequency cepstral coefficients (MFCC), linear prediction coefficient (LPC), short term energy (STE), and linear prediction cepstral coefficients (LPCC).

The accuracy of commercial, replay, narrative, audience-scene, zoom-in and close-up has a direct impact on the the accuracy of events. Their frame precision is in the range of $0.71 \sim 0.84$ and frame recall in around $0.65 \sim 0.85$.

3.4.3 Timeout Removal from American Football Video

Timeouts make the asynchronism between the audiovisual and text time lines become larger as the match progresses. To make the integrated analysis possible, timeouts need to be removed. Timeout and play are modeled by a HMM each and are segmented by a hierarchical HMM. A similar approach was taken to segment play and break in soccer([97]). The accuracy is satisfactory with frame-level precision of 82.6% and recall of 81.3% for play, thanks to timeout being signified by commercials, replays and narratives. The detected plays are linked and linearly scaled to the duration of a match (60 minutes). Note that the resultant video is only an approximation of the actual match and offset still exists.

3.4.4 Criteria of Evaluation

To evaluate the quality of event detection, we consider both the correctness of he event type being recognized as well as the accuracy in boundary. Hence in this research, we choose frame precision and recall in evaluation. Segment-based evaluation with tolerance like that used in [81] is not chosen because a suitable tolerance is hard to find given the varying event durations.

3.5 Training and Test Data

Both frameworks use a common set of training and test data. Sources of the experimental data are documented in Table 3.1. Some statistics are given in Tables 3.2 and 3.3. All matches are in full length. Match reports and game logs in soccer are obtained from www.soccernet.com; recaps and play-by-play reports in American football are downloaded from www.nfl.com.

		When	Where	Teams	Program	TV network
Soccer	Training	August 24, 2003	The Riverside Stadium	Middlesbrough vs. Arsenal	EPL	ESPN
		September 20, 2003	Molineux Stadium	Wolverhampton vs. Chelsea	EPL	ESPN
		September 21, 2003	Old Trafford	Man Utd vs. Arsenal	EPL	ESPN
		September 25, 2004	Villa Park	Aston Villa vs. Crystal Palace	EPL	ESPN
		September 25, 2004	White Hart Lane	Tottenham vs. Man Utd	EPL	EPSN
	Test	November 8, 2003	Highbury	Arsenal vs. Tottenham	EPL	ESPN
		November 9, 2003	Anfield	Liverpool vs. Man Utd	EPL	ESPN
		November 9, 2003	Stamford Bridge	Chelsea vs. Newcastle	EPL	ESPN
		December 7, 2003	St. Mary's Stadium	Southampton vs. Charlton	EPL	ESPN
		December 13, 2003	Old Trafford	Man Utd vs. Man City	EPL	ESPN
		December 21, 2003	White Hart Lane	Tottenham vs. Man Utd	EPL	ESPN
		September 25, 2004	City of Manchester Stadium	Man City vs. Arsenal	EPL	ESPN
		September 26, 2004	Fratton Park	Portsmouth vs. Everton	EPL	ESPN
American football	Training	October 27, 2003	Sun Devil Stadium	Miami Dolphins vs. San Diego Chargers	NFL	EPSN
		November 2, 2003	Hubert H. Humphrey Metrodome	Green Bay Packers vs. Minnesota Vikings	NFL	ESPN
		September 26, 2004	Network Associates Coliseum	Tampa Bay Buccaneers vs. Oakland Raiders	NFL	ESPN
		October 4, 2004	M&T Bank Stadium	Kansas City Chiefs vs. Baltimore Ravens	NFL	ESPN
		November 7, 2004	M&T Bank Stadium	Cleveland Browns vs. Baltimore Ravens	NFL	ESPN
	Test	November 14, 2004	Gillette Stadium	Buffalo Bills vs. New England Patriots	NFL	ESPN
		November 15, 2004	Texas Stadium	Philadelphia Eagles vs. Dallas Cowboys	NFL	ESPN
		November 21, 2004	Reliant Stadium	Green Bay Packers vs. Houston Texans	NFL	ESPN
		November 22, 2004	Arrowhead Stadium	New England Patriots vs. Kansas City Chiefs	NFL	ESPN
		November 29, 2004	Lambeau Field	St. Louis Rams vs. Green Bay Packers	NFL	ESPN

Table 3.1: Sources of the experimental data

	Training data	Test data
Number of matches	5	8
Total duration	460 mins	785 mins
Number of goals	13	34
Number of saves	28	87
Number of shot-off-targets	62	124
Number of penalties	2	2
Number of corner-kicks	28	75
Number of free-kicks	17	31
Number of offsides	23	39
Number of substitutions	19	24
Number of yellow-cards	8	13
Number of red-cards	1	2

Table 3.2: Statistics of experimental data - soccer

	Training data	Test data
Number of matches	5	5
Total duration on match time line	308 mins	317 mins
Number of touchdowns	30	19
Number of conversions	30	19
Number of field-goals	14	16
Number of safeties	1	0
Number of punts	39	44
Number of punt-returns	23	20
Number of fumble-opponents	15	12
Number of interception	9	11
Number of touchbacks	7	4
Number of kickoffs	53	45
Number of fumbles-own	12	10
Number of incomplete-passes	12	7

Table 3.3: Statistics of experimental data - American football

Chapter 4

THE LATE FUSION FRAMEWORK

4.1 The Architecture of the Framework

The late fusion framework has three major modules (see Figure 4.1): (a) audiovisual analysis, which processes the audiovisual signals, (b) text analysis, which processes the external information sources, and (c) fusion, which combines outcomes from the two analysis. The domain knowledge drives all of the three modules.

Compared to existing systems, the algorithms used in the late fusion framework has the following novelties. First, the audiovisual analysis adopts a two-step pipeline so that events can be detected locally after phases are segmented. Second, the algorithms for processing compact or detailed descriptions are domain-independent thus reusable. Third, three novel fusion schemes are proposed, which are effective in fusing asynchronous items. Fourth, by making audiovisual and text analysis independent, the system is extensible, i.e. it achieves stronger detection capability given external information of increasing detail:

- With only audiovisual signals, the system achieves comparable detection capability to state-of-the-art audiovisual-based systems.
- By incorporating compact descriptions, the system can ensure correct de-

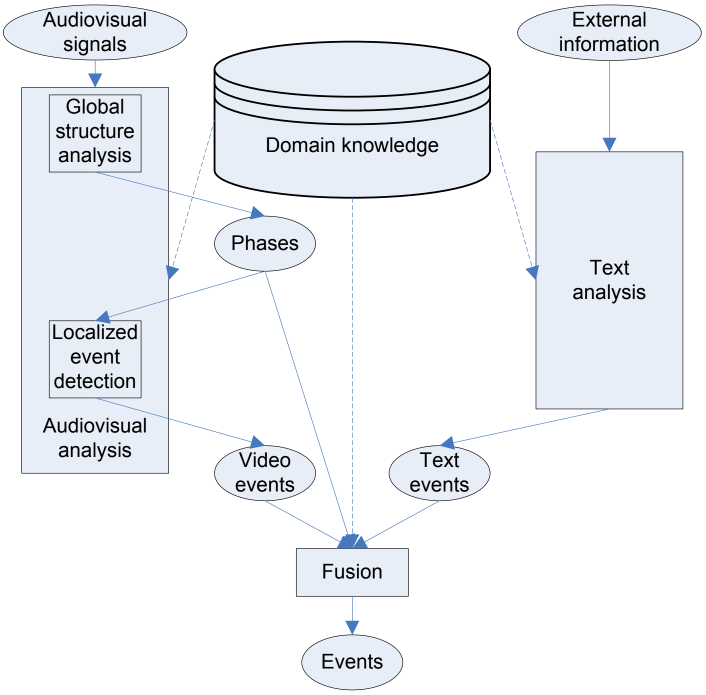


Figure 4.1: The late fusion framework.

tection of most important events, such as scorings, and those key events that do not have consistent or distinctive audiovisual patterns, such as the *yellow-cards* and *substitutions* in soccer matches.

- By utilizing detailed descriptions, the system can detect the full range of events with their boundaries.

4.2 Audiovisual Analysis

Analysis on audiovisual signals generates a list of entries called *video events*, each representing an event in terms of

$$\textsf{video event} := \langle \textsf{start time}, \textsf{end time}, \textsf{event type} \rangle \tag{4.1}$$

Note that the audiovisual analysis only aims at detecting a subset of target event types that have consistent and distinctive audiovisual patterns due to limitation in capabilities. Audiovisual analysis has two steps: (a) global structure analysis, which segments phases using a statistical learning method; and (b) localized event classification, which identifies events locally using event-specific feature set and algorithm. The use of divide-and-conquer pipeline minimizes the need for training samples and particularly alleviates the data sparseness problem. It would contribute to higher precision and recall in audiovisual-based event detection, and ensure that our technique is able to detect a wide range of event types and deal with full-length videos.

4.2.1 Global Structure Analysis

To segment the video sequence into phases, the global structure analysis uses a two-layer hierarchical HMM (HHMM). The top layer of HHMM models inter-phase transitions; and the bottom layer models the evolution in a phase. To ensure the success of a learning-based approach, the judicious choice of features is important. We select a set of features applicable to various domains, though values of some features may be domain-specifci. They are extracted using the techniques explained in "3.4.2 Extraction of Features":

- *Shot category.* It categorizes a shot into one of these categories: commercial, narrative, replay, audience-scene and on-going play.

- *Focal distance.* It categorizes a shot of on-going play into one of three types: global, zoom-in and close-up. The arrangement of shots of different focal distance conveys how the cameraman view the match's progress.

- *Canonical view.* A canonical view is taken at a certain vintage position to provide best coverage of a particular scene. It usually has strong correlation with happenings in the match, e.g. a frontal view of the goalpost signifies a *field-goal* or *conversion* in American football.

- *Field zone.* Change in field zone helps to reveal the direction of the *advance*.

- *Camera motion pattern.* Panning and tilting help reveal the direction of the *advance*. Also, zooming implies something of great interest, which may be related to scoring attempts.
- *Motion magnitude.* Motion magnitude helps in differentiating *break* from other phases. It is quantized to three levels - low, moderate, and intense.

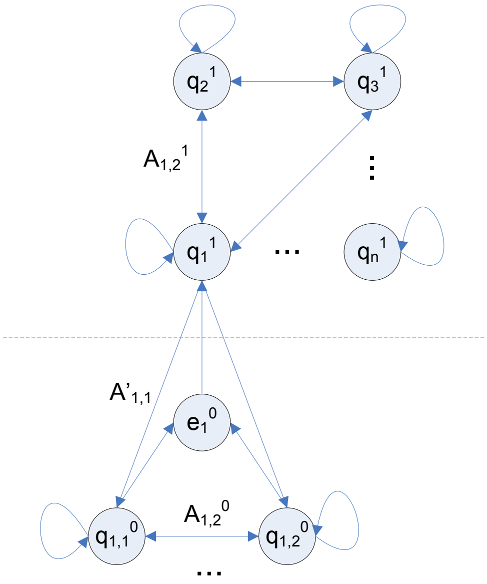


Figure 4.2: Global structure analysis using HHMM

The general topology of the two-layer HMM is given in Figure 4.2. Each node ${q_j}^1 (j = 1...n)$ at the top layer denotes a phase-level unit. It could represent all instances of a phase type, e.g. a *left-advance* in soccer, or a group of instances having a consistent audiovisual pattern, e.g. an *advance* starting with a return in American football. The top-layer topology is ergodic, since domain knowledge says in the most general case, a phase is free to transit into any other phase or remain as it is. Under each top layer node, there is a number of states ${q_{j,k}}^0 (j = 1..n, k = 1..m)$ representing evolution in this phase, plus an exit state ${e_j}^0$, through which the top layer node transits to another. The number of states m at the bottom layer is chosen from a number of candidate numbers that gives the best accuracy over a validation set. The top- and bottom-layer HMM are trained separately using Baum-Welch algorithm. The Viterbi path at the top layer indicates the phases with boundaries.

4.2.2 Localized Event Classification

Following global structure analysis, phases conforming to the same temporal location specifications are grouped. For example, a group of phases in soccer video is *advances* which are sandwiched by two *breaks*. As a specification may host more than one event type, every phase in this group need to undergo a series of classifications to determine which event type (including null) it hosts. Note that this operation implicitly models the sequential relationships among the event types in the same group as exclusive of one another. The series of classifications are based on audiovisual patterns of the possible event types, either in a heuristic or in a learning-based method. Figure 4.3 illustrates this process.

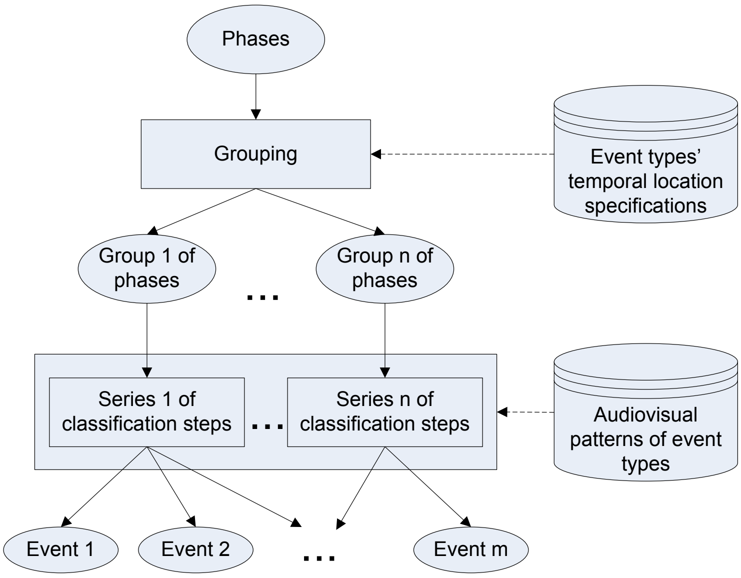

Phases
Grouping
Event types' temporal location specifications
Group 1 of phases
Group n of phases
...
Series 1 of classification steps
...
Series n of classification steps
Audiovisual patterns of event types
Event 1
Event 2
...
Event m

Figure 4.3: Localized event classification

4.3 Text Analysis

Text analysis aims to generate a list of entries called text events, each representing an event in terms of

$$\mathsf{text\ event} := \langle \mathsf{start\ time}, \mathsf{end\ time}, \mathsf{event\ type} \rangle \tag{4.2}$$

A piece of description needs to be classified as compact or detailed before it can be processed as the two forms require different techniques. A SVM classifier is used to perform the classification based on the following set of features: (a) number of paragraphs (PG), (b) number of time entities (TE), and (c) number of player names (PN), with TE and PN normalized by the length of the article. Experiments show the differentiation of compact and detailed descriptions is reliable with the accuracy of over 98%.

4.3.1 Processing of Compact Descriptions

Compact descriptions such as the match reports in soccer and recaps in American football are in free text form, and cover only important events of interest to general readers. Due to the difficulty in processing free-form text with missing information, the list of events detected would probably be error-prone and incomplete. We tackle this as an information extraction (IE) problem using rule-based IE techniques.

Before the IE process starts, some domain knowledge needs to be put in place, namely, player database and game rules. How to establish domain knowledge was explained in “3.2 Domain Knowledge Used in Both Frameworks”. The IE process has four steps:

1. We induce from the training samples syntactic rules indicative of times and events using the GRID technique [96].

2. We identify the time entities using the rules.

3. A window of terms [-x, +y] around each time entity is picked. Usually, x and y are both set to be one sentence.

4. The window of terms is analyzed against all rules to see if it satisfies any rule indicative of an event type.

When an event is found, its temporal boundaries are set to be one minute before and after the time entity, in view that the compact descriptions only give coarse timing information. Usually, one article would only mention a handful of important events; it is thus advisable to process a few ($5 \sim 7$) articles. In case of multiple events detected in the same term window, all are kept until the fusion module makes a decision with the help of audiovisual cues.

4.3.2 Processing of Detailed Descriptions

Detailed descriptions of team sport matches are usually composed of entries. Each entry roughly corresponds to an action, giving information on the time, action type, the player who performs it, and outcome of the action. Descriptions in some games may also give position information, e.g. in American football play-by-play reports. Based on information extracted from entries as well as inter-entry context, each entry is transformed into a node

$$\phi := \langle \mathsf{action\_type}, \mathsf{outcome}, \{\mathsf{pre\_tran}\}, \{\mathsf{post\_tran}\} \rangle \tag{4.3}$$

which is the composing unit of event's semantic composition model (Figure 3.2). It is thus logical to transform the whole detailed description to a sequence of $L = \phi_1\phi_2...\phi_L$ and detect events from L by model checking. Note that entries of a detailed description could be in a field-delimited format, such as game logs in soccer (Figure 3.6), or in free text, such as play-by-play reports in American football (Figure 3.7). The two formats are handled differently when extracting information from entries. For the former, it is convenient to check keywords, as fields are usually filled by standard terms; whereas for the latter, rule-based IE techniques similar to those described in 4.3.1 are employed. Having a complete sequence of $L = \phi_1\phi_2...\phi_L$, occurrences of a particular event type can be identified by checking all subsequences against the model (Figure 3.2). A more computation-efficient alternative is to perform model checking around nodes that match the key actions. From such a node, we keep crawling and analyzing preceding and following nodes as long as they satisfy the model. The last nodes leading to INIT and END respectively signal the detection of an event as well as mark the event's first and last actions. The event's starting boundary is taken to be the first action's recorded time, and the ending boundary is estimated to be the last action's recorded time plus the average duration of this action derived from training data. Note that events' boundaries obtained in this way are only approximate. This is because: (a) the time given in an entry is usually not accurate; and (b) the duration of the last action is only an estimate. Identification of accurate boundaries requires the audiovisual cues.

4.4 Fusion of Video and Text Events

Audiovisual analysis is accurate in pinning events boundaries while text analysis is accurate in identifying event type. It is expected that fusion will take advantage of both strengths. We investigate three fusion schemes that work in the late fusion framework. They differ in the way offsets are modeled. The rule-based scheme assumes that offsets cannot be numerically modeled; whereas aggregation models offsets to follow a probabilistic distribution; and Bayesian inference models them to be binary.

4.4.1 The Rule-Based Scheme

Following the guideline of "identifying the pair of items before fusing them", the rule-based scheme is accomplished in three steps.

Aligning text events and phases

Given that there is no modeling of offsets, the alignment is sought by maximizing the number of matches between text events and phases. Here a match between a text event and a phase means that the phase conforms to the temporal location specification of the text event (e.g., an *advance* followed by a *break* conforms to *goal*'s temporal location specification), are within a temporal range, and they occur in the same sequential temporal order. As text events may overlap temporally, such as a *corner-kick* and a resulting *goal* in soccer, and multiple text events may occur in the same phase, such as a *punt-return* and a *touchdown* respectively in the beginning and at the end of an *advance* in American football, a phase may match multiple text events. The maximization problem is similar to the Longest Common Subsequence (LCS) matching problem and can be solved by dynamic programming technique.

Determining event type

This step resolves the conflicts in event type based on video and text events' comparative accuracy. Pseudo code is given in procedure 1. If the fused event type is suggested only by text analysis, we proceed with additional audiovisual analysis to find the event's location and temporal boundaries. During this process, some distinctive characteristics in the events' audiovisual patterns are utilized. Note that these characteristics alone may not be discriminative enough to differentiate between event types. An example of these audiovisual characteristics is the hard-cut between an *advance* and a *break* in the audiovisual patterns of *goal*, *save*, and *shot-off-target*.

Determining event boundaries

After video and text events agree on the event type, boundaries of the fused event are determined as those of the video event.

Algorithm 1 Pseudo-code for determining event type.

```
for phase such that phase is aligned to a textevent do
    videoevent ← result of audiovisual analysis in phase
    if videoevent.type == textevent.type then
        event.type ← videoevent.type
    else if F1 of audiovisual analysis on videoevent.type > F1 of text analysis
    on textevent.type then
        event.type ← videoevent.type
    else
        event.type ← textevent.type
        find videoevent such that videoevent.type == event.type
    end if
end for
for phase such that [phase] is aligned to no textevent do
    videoevent ← result of audiovisual analysis in phase
    if videoevent.type ≠ null then
        if text analysis is on detailed description then
            event.type ← null
        else
            event.type ← videoevent.type
        end if
    else
        continue
    end if
end for
for textevent such that textevent is aligned to no phase do
    discard textevent
end for
```

4.4.2 Aggregation

As discussed earlier on, either video or text time line is intervened with offsets. For soccer, the text time line is intervened with offsets and video time line is not; but for American football, the reverse is true. For simplicity, we use *offset time line/analysis/event* to respectively refer to the time line intervened with offsets, analysis conducted on the time line, and the detected events. Similarly we use *non-offset time line/analysis/event* to respectively refer to their corresponding counterparts associated with accurate time line.

In general, the detection results of a particular event type can be depicted by a likelihood curve on a time line. The idea of aggregation is for the likelihood curve given by the offset analysis to migrate from the offset time line to the accurate time line. By doing this, the two likelihood curves given respectively by the offset and non-offset analysis are synchronized and can be combined. The whole process is carried out in three steps as explained in the following.

1. Modeling the offset distributions for start/end of events. This is similar to that of Yang et al.[107] that modeled the probability of a face occurring along the time line with respect to when the name is mentioned.

2. Computing the likelihoods of the offset event given by the two analysis on the accurate time line. Based on the two distributions showing probabilistically when the offset event starts or ends on the accurate time line, we calculate the probability of any time point t on the accurate time line being in the span of the offset event.

$$P_{in}(t) = \int_{-\infty}^{t} D_s(x)\,dx \cdot \int_{t}^{\infty} D_e(x)\,dx \tag{4.4}$$

where $D_s(x)$ and $D_e(x)$ are distributions of the offset event's start and end on the accurate time line, respectively. Suppose offset event has event type i, the likelihood of event type i at time point t seen by the offset analysis on the accurate time line is:

$$P_{i-O}(t) = C_i P_{in}(t) \tag{4.5}$$

where C_i reflects the confidence of the offset analysis on event type i. We take it to be the precision of the offset analysis on i over the training set. Suppose event type j is detected by the non-offset analysis at time point t, the likelihood of event type i at time point t seen by non-offset analysis is:

$$P_{i-N}(t) = Confusion_{i,j} \tag{4.6}$$

where $Confusion_{i,j}$ is the element of confusion matrix that indicates the percentage of type i samples out of all samples detected to be of type j. Note that the confusion matrix includes the null event type.

3. Combining the likelihoods of the offset and non-offset analysis. Let $P_{i-N}(t)$, $P_{i-O}(t)$ and $P_i(t)$ denote the likelihoods seen by non-offset analysis, offset analysis, and the fused likelihood on the accurate time line. Then $P_i(t)$ is computed by

$$P_i(t) = wP_{i-N}(t) + (1-w)P_{i-O}(t) \tag{4.7}$$

If $P_i(t)$ is greater than a threshold thr, the fused event at time point t on the accurate time line is of type i. We find the optimal parameters (w, thr) by optimizing the detection accuracy over a validation set using the gradient descent method.

4.4.3 Bayesian Inference

Different from aggregation, this scheme does not model the offsets in probabilistic distribution. Instead, it only differentiates if the offset is within a maximum allowed range. Unless specified, the following description is for a particular event type p. Regarding whether p is present at time point t on the accurate time line, there is a binary hypotheses: H_0 - not-present, and H_1 - present. The maximum likelihood hypothesis is

$$\arg\max_{i\in\{0,1\}} P(x_N, x_O \mid H_i) \cdot P(H_i) \tag{4.8}$$

where x_O and x_N are two variables derived from offset and non-offset analysis, respectively. Usually, x_O refers to whether p is detected in the maximum allowed range, and x_N refers to the event type of the detection by non-offset analysis (it could be different from p) at time t. Since x_O and x_N are outcomes of two independent analysis, we have:

$$\arg\max_{i\in\{0,1\}} P(x_N, x_O \mid H_i) \cdot P(H_i) = \arg\max_{i\in\{0,1\}} P(x_N \mid H_i) \cdot P(x_O \mid H_i) \cdot P(H_i) \tag{4.9}$$

where $P(H_i)$, $P(x_N \mid H_i)$ and $P(x_O \mid H_i)$ are obtained from the training set.

4.5 Implementation of the Late Fusion Framework on Soccer And American Football Video

We implement the framework on soccer and American football videos to test its portability to different domains.

4.5.1 Implementation on Soccer Video

Domain knowledge

Soccer's phases and events were listed Chapter 3. Event types in soccer are grouped according to temporal location specifications:

- goal/save/shot-off-target/ offside - in an *advance* which is followed by a *break*,
- penalty/corner-kick/free-kick - in an *advance*, which is sandwiched by two *breaks*, and if an *advance* precedes the first *break*, the two *advances* should be in the same direction, and
- yellow-card/red-card/substitution - in a *break* in the midst of the match.

In the semantic composition models of these event types, actions are modeled by a definite set of action types and outcomes. Action types and outcomes are enumerated as follows.

$$\begin{aligned}\textsf{action-type} \in \{ & \textsf{dribble}, \textsf{pass}, \textsf{cross}, \textsf{shoot}, \textsf{goal-kick}, \textsf{block}, \textsf{clear}, \textsf{catch}, \textsf{throw}, \\ & \textsf{substitute}, \textsf{parry}, \textsf{kick-at-penalty-spot}, \textsf{kick-at-corner}, \\ & \textsf{kick-at-other-spots}, \textsf{unsportsmanlike-conduct}\} \qquad (4.10)\end{aligned}$$

$$\begin{aligned}\mathsf{outcome} \in \{ \; & \mathsf{success\text{-}catch}, \mathsf{failed\text{-}catch}, \mathsf{success\text{-}block}, \mathsf{failed\text{-}block}, \mathsf{success\text{-}clear}, \\ & \mathsf{failed\text{-}clear}, \mathsf{offside}, \mathsf{ball\text{-}over\text{-}bar}, \mathsf{ball\text{-}out\text{-}of\text{-}goal\text{-}line}, \\ & \mathsf{ball\text{-}out\text{-}of\text{-}sideline}, \mathsf{ball\text{-}inbounds}, \mathsf{ball\text{-}on\text{-}target}, \mathsf{ball\text{-}hit\text{-}woodwork}, \\ & \mathsf{scoring}, \mathsf{players\text{-}in\text{-}attacking\text{-}third}, \mathsf{players\text{-}in\text{-}defending\text{-}third}, \\ & \mathsf{foul\text{-}declaration}, \mathsf{yellow\text{-}card\text{-}issuance}, \mathsf{red\text{-}card\text{-}issuance}, \\ & \mathsf{open\text{-}play}, \mathsf{play\text{-}stop}\} \end{aligned} \tag{4.11}$$

Soccer video has one canonical view - {behind-goal-post}, which captures activities close to the goalpost from behind the goal net.

Domain-dependent design and processing

The subset of event types detectable by audiovisual analysis is

{goal, attempt-on-goal[1], penalty, corner-kick, free-kick}

while all event types are detectable by text analysis. The top layer of the HHMM for global structure analysis has four nodes, each corresponding to a phase. After experimenting with different numbers of states ranging from 2 to 9, we found that 3 states at the bottom layer give the best accuracy. Based on temporal location specifications, events detectable by audiovisual analysis occur in two groups of phases. One is "an *advance* which is followed by a *break*", which may contain a *goal* or an *attempt-on-goal*; and the other is "an *advance* which is sandwiched by two *breaks*", which may contain a *penalty*, *corner-kick*, or *free-kick*. The series of classifications for the two groups of phases are shown in Tables 4.1 and 4.2.

4.5.2 Implementation on American Football Video

Domain knowledge

American football's phases are listed in "3.2 Domain Knowledge Used in Both Frameworks". There is a temporal unit at a level lower than phase - play. A play is made up of a continuous segment of match; it reflects the intermittence of

[1] *Attempt-on-goal* is the union of *save* and *shot-off-target*

Step	Purpose	Input	Outcome	Algorithm	Features
(a)	To differentiate *goals* from *non-scoring* (union of *attempt-on-goals* and *none-of-the-group*)	*Advances* satisfying temporal location specifications	*Goals*	By rule: No *advance* between this *advance* and subsequent presence of goal videotext.	(a)Presence of goal videotext.
(b)	To differentiate *attempts-on-goal* from *none-of-the-group*	*Advances* satisfying temporal location specifications, *goals* excluded	*Attempts-on-goals*	By rule: high excitement level in commentators speech during the *advance* with close-ups following the *advance*	(a) Excitement level; (b) Presence of subsequent close-ups.

Table 4.1: Series of classifications on group I phases (soccer)

Step	Purpose	Input	Outcome	Algorithm	Features
(a)	To differentiate *placed-kicks*[a] from *none-of-the-group*	*Advances* satisfying temporal location specifications	*Placed-kicks*	HMM	(a) Focal distance; (b) Unit duration; (c) Motion activity; (d) Camera motion pattern.
(b)	To differentiate among *penalty*, *corner-kick* and *free-kick*	*Placed-kicks*	*Penalties*, *corner-kicks* and *free-kicks*	Multi-class SVM	(a) Distance of the overall camera motion along the flying of the ball; (b) Angle of the overall camera motion along the flying of the ball; (c)-(e) Lengths of the three longest field lines before the flying of the ball; (f)-(h) Angles of the three longest field lines before the flying of the ball; (i) Duration of the zoom-in shots in the preceding *break* before the flying of the ball.

Table 4.2: Series of classifications on group II phases (soccer).

[a] *Placed-kick* is the union of *penalty*, *corner-kick* and *free-kick*.

the American football match. As events are usually bound in plays, play specifies temporal locations of event types best:

- touchdown - in the second last play of an *advance*, which is followed by a *conversion* as the last play;
- conversion - in the last play of an *advance*, which is preceded by a *touchdown* as the second last play;
- field-goal/safety/punt/ fumble-opponent/interception/touchback - in the last play of an *advance*;
- punt-return/kickoff/safety - in the first play of an *advance*;
- fumble-own/incomplete-pass - in any play of an *advance.*

Semantic composition models of event types are composed of action types and outcomes.

$$\begin{aligned}\mathsf{action\text{-}type} \in \{ \; & \mathsf{tackle, pass, recover, forward\text{-}progress,} \\ & \mathsf{backward\text{-}progress, kick, punt\text{-}kick, snap}\} \end{aligned} \tag{4.12}$$

$$\begin{aligned}\mathsf{outcome} \in \{ \; & \mathsf{ball\text{-}passed, ball\text{-}intercepted, ball\text{-}out\text{-}of\text{-}end\text{-}line, return,} \\ & \mathsf{ball\text{-}dropped, ball\text{-}recovered, success\text{-}tackle, ball\text{-}out\text{-}of\text{-}bounds,} \\ & \mathsf{failed\text{-}tackle, scoring\text{-}touchdown, scoring\text{-}field\text{-}goal}\} \end{aligned} \tag{4.13}$$

American football has three canonical views: {facing-goal-post, snap-play, kick-and-dash}. Facing-goal-post captures a player from behind in the background of goalpost; snap-play shows an on-going play started from a snap scene; and kick-and-run depicts a global view of the field when a long kick is made and both teams start to run towards each other.

Step	Purpose	Input	Outcome	Algorithm	Features
(a)	To differentiate among *conversion*, *field-goal* , *safety* and *non-scoring* (union of *punt* and *none-of-the-group*)	Last plays of all *advances*	*conversion*, *field-goal*, *safety* and *non-scoring*	By rule: particular score indicates particular event type	(a) Score update by videotext.
(b)	To differentiate between *punt* and *none-of-the-group*	*Non-scorings*	*Punt* and *none-of-the-group*	SVM	(a) Distance of the overall camera motion during the play; (b) Angle of the overall camera motion during the play; (c) Presence of canonical view of kick-and-run.

Table 4.3: Series of classifications on group I plays (American football).

Step	Purpose	Input	Outcome	Algorithm	Features
(a)	To differentiate between *turnover* and *non-of-the-group*	First plays of all *advances*	*return* and *non-of-the-group*	SVM	(a)-(b) Number of left/right pans during the play; (c)-(d) Overall distance of the camera motion in the left/right direction.

Table 4.4: Series of classifications on group II plays (American football).

Domain-dependent design and processing

The subset of event types detectable by audiovisual analysis is

$$\{\mathsf{touchdown}, \mathsf{conversion}, \mathsf{field\text{-}goal}, \mathsf{safety}, \mathsf{punt}, \mathsf{turnover}^2\}$$

while all event types are detectable by text analysis. In global structure analysis, the top layer HMM has four nodes, with two representing *advances* in the same direction starting with a return and without a return, respectively. Experiments with different numbers of states ranging from 2 to 9 show that 3 states at the bottom layer give the best accuracy. In audiovisual analysis, events are bound in plays. Play grouping is similar to phase grouping for soccer video. Two groups of

[2] *Turnover* is the union of *punt-return* and *kickoff*.

plays are formed:

- conversion/field-goal/punt/safety - the last play of an *advance*;
- turnover - the first play of an *advance*.

Each group undergoes a series of classifications as Tables 4.3 and 4.4 show. Note that as *touchdown* and *conversion* always come together, they are detected at one go; and they are identified by particular bounding plays.

4.6 Evaluation of the Late Fusion Framework

We evaluate the global structure analysis, audiovisual and text analysis, and event detection after fusion. Also we compare performance of various fusion schemes under different conditions in order to find the best fusion scheme.

4.6.1 Evaluation of Phase Segmentation

Tables 4.5, 4.6, 4.7 and 4.8 show accuracy of detected phases of soccer and American football videos. Phases are sequences of frames bearing a label and thus we evaluate both phase-level and frame-level accuracy. Phase-level accuracy measures how many phases are missing or falsely detected giving a sketchy picture without considering boundaries. Frame-level accuracy measures how close detected boundaries are to the ground truth as well as enables us to look into misclassifications.

Tables 4.5 and 4.7 show that misses and false positives are satisfactory for both games. The error rates are around $0.05 \sim 0.09$ for *advance*, $0.06 \sim 0.08$ for *break*, and $0.10 \sim 0.11$ for *draw* of both games. From Tables 4.6 and 4.8, we notice that misclassification between any pair of phase types is not trivial, which suggests that inaccurate boundaries are common. We also notice that: (a) frames belonging to *advances* are more likely to be misclassified as *draw* than as *break*; (b) frames

	Ground truth	Misses	False positives
Left-advance	976	84	69
Right-advance	943	79	52
Draw	465	47	49
Break	311	24	18

Table 4.5: Misses and false positives of soccer phases by the late fusion framework.

	Ground truth	Detected				Recall	F1
		(a)	(b)	(c)	(d)		
Left-advance (a)	266530	217755	9316	21707	17752	.817	.813
Right-advance (b)	246028	6768	201742	20756	16762	.820	.802
Draw (c)	255119	19865	16254	201278	17722	.789	.781
Break (d)	409484	24498	29944	16880	338162	.826	.846
Detected total	1177161	268886	257256	260621	390398	-	-
Precision	-	.810	.784	.772	.866	-	-
Weighted F1[a]	.815						

Table 4.6: Frame-level accuracy of soccer phases by the late fusion framework.

[a] Based on items whose F1 values are available. Weighted according to frame frequency in ground truth. This applies to all similar scenarios wherein weighted F1 is calculated.

	Ground truth	Misses	False positives
Left-advance	63	3	4
Right-advance	63	5	5

Table 4.7: Misses and false positives of American football phases by the late fusion framework.

	Total in ground truth	Left-advance	Right-advance	Other	Recall	F1
Left-advance	201935	174535	8205	19195	.864	.852
Right-advance	273279	12081	233039	28159	.853	.852
Other	-	21257	32794	-	-	-
Detected	-	207873	274038	-	-	-
Precision	-	.840	.850	-	-	-
Weighted F1	.852					

Table 4.8: Frame-level accuracy of American football phases by the late fusion framework

belonging to *break* frames are more likely to be misclassified as *advance* than as *draw*; and (c) the misclassification between the opposite *advances* is significantly lower than the average.

The results suggest that:

- The good accuracy may be attributed to three factors. First, the patterns of alternating *advances* in opposite directions are distinct and are suitable to use as a basis to analyze the structure of team sports videos. Second, the camera motion pattern, motion magnitude and field zone are effective in differentiating in-play from out-of-play and *advances* in different directions. Third, the hierarchical HMM is effective in segmenting concatenated sequences characterized by different audiovisual evolution.

- Though audiovisual analysis is in general effective for phase segmentation, the accuracy varies and depends on the distinctiveness and consistency of audiovisual appearance of the phase type. The *advance* has consistent camera motions while the *break* has replay logos. They are more distinct than *draw* and therefore they achieve better accuracy.

- Most errors in phase segmentation come from errors in feature extraction or inability of the audiovisual analysis to handle subtle differences between phase types. The features refer to the shooting or editing artifacts, such as camera motion, replay logos and close-ups (see unit parsing steps in 3.4.2). False extraction of them would result in missing or runaway *breaks* or falsely detected *advances*. Subtle difference between *advance* and *draw* in motion magnitude and field zone account for most of the misclassifications between them.

- Generally speaking, phase boundaries are inaccurate with a probabilistic algorithm such as hierarchical HMM. This is particularly true with gradual transitions such as between *advance* and *draw*. Fortunately, phase boundaries are not crucial for the purposes of localizing video events and aligning phases to text events.

As we know from the domain knowledge, the *advance* and *break* are more important than the *draw* because events' temporal locations are mostly specified with them. Therefore, we should minimize missing or falsely detected *advances* or *breaks*.

4.6.2 Evaluation of Event Detection By Separate Audiovisual/Text Analysis

Evaluation of event detection by audiovisual analysis

Tables 4.9 and 4.10 give the confusion matrices of soccer and American football events, respectively, by audiovisual analysis only. They show that in general the accuracy is not satisfactory. A small number of event types have relatively high precision or recall values - the *goal* and *penalty* in soccer have recall of above 0.8, and *touchdown* in American football has precision of above 0.8, too. However, the majority of event types have recall and precision in the range of $0.5 \sim 0.7$. Note that that misclassified frames are all associated with missing or falsely detected events because video events have no boundary issue.

Looking into the error cases, we make the following observations.

- A significant number of errors are related to erroneous feature extraction. Errors may exist in videotext, audio genre and camera motion pattern. Errors in videotext would result in misclassification of *goal*, *touchdown/conversion*, *field-goal* and *punt*; errors in audio genre may cause errors in *attempt-on-goals*; and errors in camera motion pattern may result in misclassification among *corner-kick*, *free-kick* and *penalty* as well as errors in *turnover*.
- When features are correctly extracted, the accuracy of an event type depends mainly on how distinct its audiovisual patterns are and how well the classifier captures this distinction. Some event types may have distinctive audiovisual patterns, such as goals with scoring videotext and turnovers with a shift in camera motion. However, many event types only have subtle differences, e.g.

	Ground	Detected						Recall	F1
	Truth	(a)	(b)	(c)	(d)	(e)	(f)		
Goal (a)	17440	14726	853	0	0	0	1861	.844	.776
Attempt-on-goal (b)	65978	0	42303	0	5344	1242	17089	.641	.610
Penalty (c)	1532	0	0	1247	0	0	285	.814	.461
Corner-kick (d)	28627	0	1760	449	18040	5434	2944	.630	.636
Free-kick (e)	21640	0	975	2182	2704	11803	3976	.545	.573
Other (f)	-	5767	26776	0	1989	1067	-	-	-
Detected total	-	20493	72667	3878	28077	19546	-	-	-
Precision	-	.719	.582	.322	.643	.604	-	-	-
Weighted F1	.630								

Table 4.9: Accuracy of soccer events by audiovisual analysis only.

	Ground	Detected							Recall	F1
	truth	(a)	(b)	(c)	(d)	(e)	(f)	(g)		
Touchdown (a)	10426	7228	336	0	863	542	0	1457	.693	.745
Conversion (b)	3094	289	1673	711	0	175	0	246	.541	.537
Field-goal (c)	2937	0	314	1721	0	317	0	585	.586	.569
Safety (d)	0	-	-	-	-	-	-	-	-	-
Punt (e)	14629	0	382	238	0	7807	4601	1601	.534	.563
Turnover (f)	36832	0	139	0	0	2691	28425	5577	.772	.719
Other (g)	-	1456	294	439	546	1594	9215	-	-	-
Detected total	67918	8973	3138	3109	1409	13126	42241	-	-	-
Precision	-	.806	.533	554	-	.595	.673	-	-	-
Weighted F1	.675									

Table 4.10: Accuracy of American football events by audiovisual analysis only.

free-kick, *corner-kick* and *penalty*. It is hard for audiovisual-based classifiers to grasp such subtle differences.

- Audiovisual analysis may effectively capture distinct audiovisual patterns but may not distinguish different semantics associated to the same patterns. For example, audiovisual analysis cannot distinguish *punt-return* and *kickoff*, but can detect them as a whole (turnover). Besides, some event types have no distinct or consistent audiovisual patterns and is beyond the capability of audiovisual analysis, such as *substitution*.

These above discussion suggests that despite constraint from event localization, audiovisual analysis cannot achieve satisfactory accuracy because of its limited capabilities in capturing semantics.

Evaluation of event detection by text analysis

Tables 4.11 and 4.12 show how many events are correctly detected by text analysis based on detailed descriptions. An event is considered correct if it appears in the video within a temporal range (1 minute for soccer and 30 seconds for American football) around the time given by the text. The Tables show that text analysis can detect the full range of event types and is accurate in terms of misses and false positives. For soccer, fewer than half of the event types (4 out of 10) have misses or false positives, and the ratios of misses and false positives are all under or around 10% except that of the false positives of *free-kick*. Misses and false positives across all event types amount to 22 as compared to a total of 431 instances in the ground truth. For American football, a marginal number of event types (1 out of 12) have misses or false positives. There is only 1 miss and 0 false positive for a total of 207 events in the ground truth. Although text analysis correctly indicates most events, the frame-level accuracy based on the time stamps given in the text is poor, with the recall and precision in the range of $0.3 \sim 0.5$.

We identify the error causes of text events as follows.

	Ground truth	Misses	False positives
Goal	34	0	0
Save	87	2	1
Shot-off-target	124	5	1
Penalty	2	0	0
Corner-kick	75	0	0
Free-kick	31	0	9
Offside	39	0	4
Substitution	24	0	0
Yellow-card	13	0	0
Red-card	2	0	0

Table 4.11: Misses and false positives of soccer events by text analysis.

	Ground truth	Misses	False positives
Touchdown	19	0	0
Conversion	19	0	0
Field-goal	16	0	0
Punt	44	0	0
Punt-return	20	0	0
Fumble-opponent	12	0	0
Interception	11	0	0
Touchback	4	0	0
Kickoff	45	0	0
Safety	0	0	0
Fumble-own	10	0	0
Incomplete-pass	7	1	0

Table 4.12: Misses and false positives of American football events by text analysis.

- Occasional human errors in the logging of events in detailed descriptions would result in misses or falsely detected events. Such logging errors include both missing and false entries. Such errors account for all missing and falsely detected events of types *save* and *shot-off-target*.
- Inconsistency between the ground truth and the game rule results in other falsely detected events. The ground truth and the game rule may have different definitions on some event types. A typical example is *free-kick* in soccer. Free-kicks made in the middle or defending side of the field are considered as free-kicks according to the game rule. However, they are not regarded as positive instances in the ground truth. This accounts for all the falsely detected free-kicks. Another scenario of inconsistency between the ground truth and the game rule is when the camera fails to follow the match closely. For example, the camera may fail to capture offsides whereas they are logged.
- Very rarely, the text analysis fails to identify the event because of occasional inconsistency in phrasing. This accounts for the missing *incomplete-pass*.

From the results we derive these observations.

- Text is rather complete, accurate and consistent and therefore is a reliable source of information that indicates events.
- As textual description of each event type is rather consistent in the use of terms/phrases and they match to the unique semantic composition well, thus they tend to be correctly classified, even though different semantic compositions may share phrases.
- Despite its effectiveness in indicating events, text analysis does a poor job in pinning event boundaries. Most detected events would be approximately half the duration off the ground truth. And a significant number (around 10%) of detections are totally disjoint from the ground truth. The poor frame-level accuracy is caused by the large offsets in soccer and imperfect

	Rule-based			Aggregation			Bayesian inference		
	Recall	Precision	F1	Recall	Precision	F1	Recall	Precision	F1
Goal	.923	.913	.918	.756	.821	.787	.884	.876	.880
Save	.834	.828	.831	.598	.794	.682	.777	.757	.767
Shot-off-target	.831	.821	.826	.573	.659	.613	.752	.734	.743
Penalty	.914	.922	.918	.862	.768	.812	.820	.853	.836
Corner-kick	.817	.805	.811	.439	.679	.533	.756	.708	.731
Free-kick	.958	.640	.767	.698	.625	.659	.743	.735	.739

Table 4.13: Comparing accuracy of soccer events by various fusion schemes.

timeout removal in American football. The underlying fact is that text is clueless about audiovisual evolution of events.

4.6.3 Comparison among Fusion Schemes of Audiovisual and Detailed Text Analysis

We would like to compare the fusion schemes to find the most effective one in tackling the asynchronism problem. We will first look at the accuracy of different fusion schemes side by side in order to acquire a general picture, and then we will look into individual schemes regarding their particular strengths or weaknesses.

Table 4.13 compares the performance of each fusion schemes on soccer with detailed text. As aggregation and Bayesian inference only support fusion of audiovisual-detectable event types, the comparison is restricted to this subset. We observe that the rule-based scheme consistently achieves the best performance, with aggregation being the worst, while Bayesian inference comes in between.

Aggregation has lower accuracy than the rule-based scheme probably because it is sensitive to the diversity of offsets. When the diversity is large, there is a large overlap between the distributions describing when the event starts and ends. In this case, the probability $P_{in}(t)$ (see Equation 4.4) is small and keeps the combined probability $P_i(t)$ small (see Equation 4.7). Consequently, it becomes difficult to distinguish positive and negative instances.

To study the impact of diversity on the performance of different schemes, we conduct experiments with varying diversity. The study is between aggregation and Bayesian inference because the rule-based fusion scheme is independent from magnitude of offsets. To measure diversity, we define $\theta = R/S$ for each event type, where R is the range of offset and S is the average temporal span of the event type. We can vary the diversity by varying θ and keep text events proportionally positioned. The experimental set-up is described as follows. We manipulate the range of offsets by putting the text events at various temporal distances away from the actual occurrence while keeping them proportionally positioned and the span of events unchanged. For Bayesian inference, the maximum allowable range of offsets is kept updated. We apply this manipulation to all occurrences of all event types. In the aggregation scheme, w is kept constant, and there is an optimal threshold thr for each θ. Figure 4.4 depicts how the optimal threshold, precision/recall rates of aggregation, and precision/recall rates of Bayesian inference, change in response to θ.

We can see from Figure 4.4 that as θ grows, the diversity of offset becomes larger, both the accuracy of aggregation and Bayesian inference declines and that of aggregation declines at a faster rate. The rapidly falling optimal threshold suggests that positive and negative instances become increasingly difficult to distinguish when the distribution of offsets get flatter (more diverse) and the accuracy deteriorates rapidly. On the other hand, although the accuracy of Bayesian inference also declines, the rate is mild; and for the range of $\theta \leq 3$ which most offsets fall under, the accuracy maintains well. This suggests that aggregation is more sensitive to the diversity of offsets than Bayesian inference is.

Next, we focus on Bayesian inference. Table 4.13 shows that generally precision and recall of Bayesian inference are both poorer than those of the rule-based

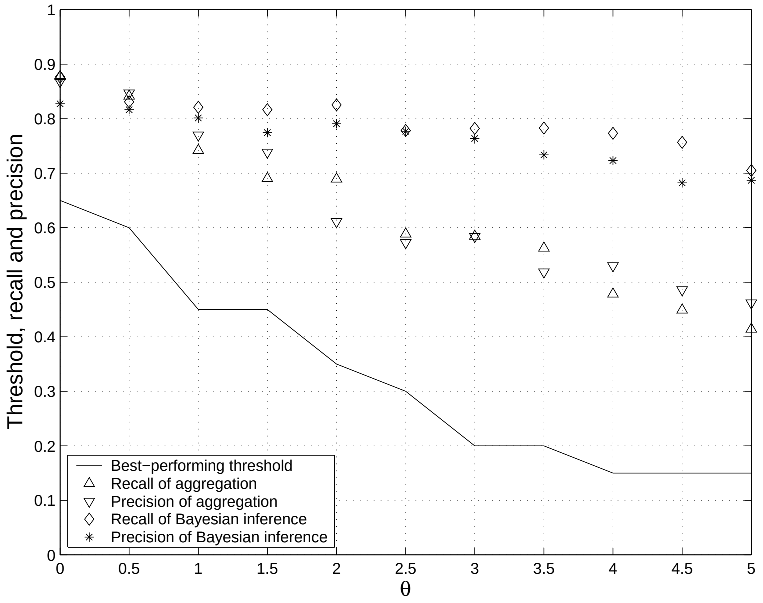


Figure 4.4: Sensitivity of performance of aggregation and Bayesian inference to θ

scheme. The difference is significant in precision of *corner-kick* and *shot-off-target* and in recall of *save.* Poorer recall and precision may both be related to the maximum allowable ranges of offsets. As for precision, when the maximum allowable ranges associated with multiple different text events overlap and a video event falls into this overlap, the video event could lead to detection of multiple events, and some of them could be false positives. This explains why event types with large maximum allowable ranges such as *corner-kick* and *shot-off-target* suffer greater loss in precision than other types. Large maximum allowable range may also lead to loss in recall, especially when the event duration is short. The smaller is the ratio of event duration over the maximum allowable range, the stronger is $P(x_T = 1|H_0)$, and the weaker is the evidence from the text event. When the ratio gets smaller than a critical point (determined by ratio of $P(x_{AV}|H_1)$ and $P(x_{AV}|H_0)$), the positive hypothesis (H_1) will lose to the negative hypothesis (H_0) and the event would be missing. When the maximum allowable range is set universal for all event types, the event types with shorter durations would suffer more, such as *save.* The negative effect of large maximum allowable range explains the decline of precision and recall of Bayesian inference in Figure 4.4.

The comparison of the fusion schemes may suggest the followings. First, aggregation is sensitive to high diversity in offsets and thus is not a reliable choice. Bayesian inference has comparable accuracy to rule-based schemes if most offsets are in the normal range ($\theta \leq 3$). However, the accuracy of Bayesian inference will suffer when the maximum allowable range becomes large. Bayesian inference has another major drawback, that it is restricted to audiovisual-detectable event types. Note that this drawback is shared by aggregation, too. Second, from the domain modeling point of view, Bayesian inference has poorer performance than the rule-based scheme because Bayesian inference is less specific in the modeling of domain knowledge than the rule-based scheme is: temporal correspondence between video and text events is weaker; evidence from text event is weaker; and knowledge about phase is not used. Third, the rule-based scheme is the best-performing scheme among the three. Its good performance relies on two factors: accurate and complete detection of text events and good structural analysis. The results of phases and text events turn out to meet the requirements. These two factors ensure quality alignment of phases and text events, which in turn ensures accuracy of the fused events.

4.6.4 Evaluation of the Overall Framework

We test the performance of overall framework on soccer and American football and compare the use of compact and detailed descriptions, respectively. We choose the rule-based scheme as it performs the best and supports a wider range of event types.

For integrated analysis of audiovisual signals and compact descriptions (see Tables 4.14), we derive the following observations:

- Incorporation of compact descriptions results in marginal improvement in accuracy. This may be because the compact descriptions mentions only a few events, and does not provide substantial help in recovering events missing

	Audiovisual only			Audiovisual + match reports			Audiovisual + game log		
	Recall	Precision	F1	Recall	Precision	F1	Recall	Precision	F1
Goal	.795	.592	.679	.894	.892	.893	.923	.913	.918
Save	.487	.498	.492	.525	.508	.516	.834	.828	.831
Shot-off-target							.831	.821	.826
Penalty	.911	.332	.487	.892	.929	.910	.914	.922	.918
Corner-kick	.569	.686	.622	.539	.698	.608	.817	.805	.811
Free-kick	.475	.382	.423	.531	.416	.467	.958	.640	.767
Offside	0	-	-	0	-	-	.842	.795	.818
Substitution	0	-	-	.207	.913	.337	.831	.926	.876
Yellow-card	0	-	-	.354	.519	.421	.904	.926	.915
Red-card	0	-	-	.901	.919	.910	.899	.915	.907
Weighted F1	.533			.532			.842		

Table 4.14: Comparing accuracy of soccer events by rule-based fusion with different textual inputs.

from audiovisual analysis. Furthermore, unreliable analysis of free form text may bring in false positive events.

- Compact descriptions helps in detecting some subtle events which cannot be detected by audiovisual analysis, such as *yellow-card*/*red-card*, though the detection of these event types is often incomplete. When these events are detected, their boundaries are usually dependable, that is why frame-level precision of *substitution*, *yellow-card* and *red-card* is high.

- Compact descriptions helps in ensuring detection of the most important events. For example on *goal*, compact descriptions help achieve an accuracy comparable to that by the detailed description.

For integrated analysis of audiovisual signals and detailed descriptions, we observe that:

- Precision and recall are generally above 80% for all event types of soccer and American football. The high accuracy is attributed to the fusion taking

	Audiovisual only			Audiovisual + play-by-play report		
	Recall	Precision	F1	Recall	Precision	F1
Touchdown	.693	.806	.745	.831	.834	.832
Conversion	.541	.533	.537	.855	.860	.857
Field-goal	.586	.554	.569	.923	.912	.917
Punt	.534	.595	.563	.803	.819	.811
Punt-return	.772	.673	.719	.843	.857	.850
Fumble-opponent				.865	.837	.851
Interception				.883	.869	.876
Kickoff				.842	.851	.846
Safety	0	-	-	0	-	-
Touchback	-	-	-	.899	.882	.890
Fumble-own	-	-	-	.812	.871	.840
Incomplete-pass	-	-	-	.822	.841	.831
Weighted F1	.675			.843		

Table 4.15: Frame-level accuracy of American football events by the rule-based fusion.

advantage of the strengths of both audiovisual and text analysis. Text analysis is accurate and complete in indicating events, but audiovisual analysis is good at pinning boundaries.

- Besides, fusion of the two analysis can detect the full-range of event types, including those that are undetectable by audiovisual analysis, i.e., *offside*, *substitution*, *yellow-card* and *red-card*. Moreover, with the help of detailed text, events that are not differentiable by audiovisual analysis can now be differentiated, i.e. *save* vs. *shot-off-target* in soccer, and *punt-return* vs. *kickoff* in American football.

- Although accuracy is high, there are still errors. Errors in text events are present because text events are more trusted in the fusion process. Free-kicks and offsides, that tend to be falsely detected in text events, are low in precision; and missing saves and shot-off-targets are not recovered. Although most text events are aligned to correct phases, some to the wrong phases as in the case of *corner-kick*.

The results obtained with the help of compact and detailed descriptions affirm the conjecture that the framework is extensible: (a) given no external information, the framework analyzes only audiovisual signals and is able to identify the overall structure of video in terms of phases as well as scoring-related events; (b) when given compact text, the framework can detect a wider range of event types and ensure detection of the most important events; and (c) when given detailed text, the fusion can handle the full-range of event types, and achieves high accuracy.

Typical error causes

Table 4.16 lists the typical errors incurred in the late fusion framework and their percentage of occurrences. The percentages are calculated based on events that are wrong either in terms of event type or in boundary. The error cases are collected from soccer and American football videos combined, based on the rule-based fusion scheme as it is the best-performing scheme. Table 4.16 suggests the following sources of errors:

Error cause	Percentage
(a) Missing documentation in detailed descriptions	7.6%
(b) False documentation in detailed descriptions	16.3%
(c) Erroneous model checking	1.1%
(d) Erroneous unit parsing or timeout detection	10.9%
(e) Erroneous phase segmentation	23.9%
(f) Erroneous alignment of text events to phases	6.5%
(g) Erroneous video events	0
(h) Erroneous boundary identification	31.5%
(i) Miscellaneous	2.2%

Table 4.16: Typical error causes in the late fusion framework

- Errors brought by text events (the sum of (a),(b) and (c)) take up a significant portion of approximately 25% in all errors. Most of them are due to inaccurate documentation in the detailed descriptions. If audiovisual signals could come into play in filtering the inaccurate documentation, the system would be less vulnerable to such errors. However, in practice, the text analysis is performed separately from audiovisual signals. In other words, the integration of audiovisual signals and external information sources is only partial - the system entirely relies on the text to identify the event types even though the audiovisual signals may be helpful in some occasions. It is conjectured that having audiovisual signals interfere with the text features during analysis might help.

- Though phase segmentation and alignment of text events to phases are reliable in general, they are still a noticeable source of errors, accounting for approximately 30% of all errors. More importantly, these errors will be irreversible and will limit the overall accuracy under an upper bound because the late fusion framework utilizes serial processing architecture. In view of this, it is conjectured that a consolidated architecture might outperform a serial one. Besides, it may also be desirable to utilize the external information sources in phase segmentation.

- A large portion of errors (over 40%) are boundary-related. They may be caused by errors during unit parsing or timeout detection. They may also be erroneous boundaries of video events that pass on. Event boundaries not conforming to the ground truth are acceptable as long as they give complete and natural video segments.

Chapter 5

THE EARLY FUSION FRAMEWORK

Though the late fusion framework has achieved good accuracy in detecting the full-range of event types, it has the following drawbacks: (a) late fusion prevents full integration of audiovisual signals and text features to improve event detection accuracy; and (b) in its serial processing architecture, errors made in phase segmentation or alignment are irreversible and will limit the overall accuracy under an upper bound. These two problems motivate us to explore a framework that implements a closer integration and consolidated processing.

A Dynamic Bayesian Network-based framework described in this Chapter is such a framework. It characterizes a consolidated probabilistic approach and early fusion of audiovisual signals and external information sources. Compared to existing DBN-based systems, our early fusion framework has the following novelties. First, standard DBNs only handles fixed-rate inputs and enforces rigid correspondence between them, but our early fusion framework accommodates loose correspondence between external information source and audiovisual signals by treating them differently. Second, the topology is specially designed (i.e., the diagonal arcs between phase and event nodes) to model constraints derived from temporal location specifications. We would like to investigate how the early fusion framework compares to the late fusion. The content of this Chapter covers network design, implementation of the two targeted team sports, experimental results and discussions.

5.1 The Architecture of the Framework

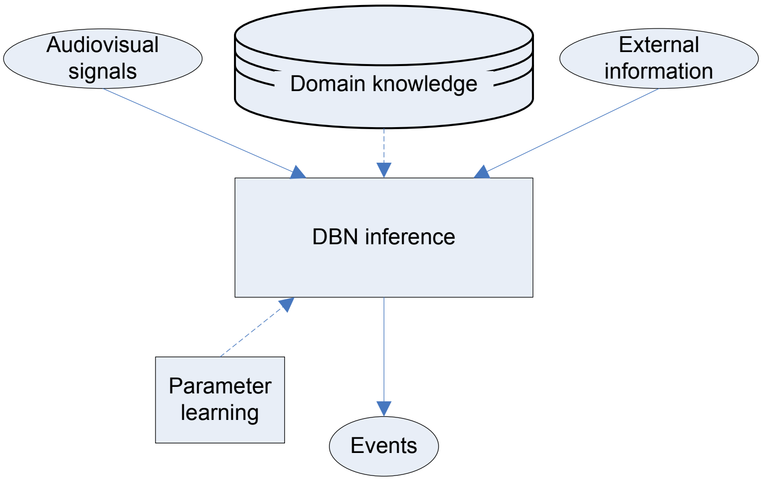


Figure 5.1: The early fusion framework.

As shown in Figure 5.1, the framework has a single step: the inference of events (phases as by-products) by a Dynamic Bayesian Network (DBN) given observations and learned parameters. The domain knowledge influences the design of DBN topology, choice of learning and inference algorithms.

We explain why we choose DBN to build the framework. As a unified algorithm, the early fusion framework is meant to directly leverage on the characteristics of both audiovisual signals and text. Text correlates events with associated phrases; while audiovisual signals contain spatiotemporal characteristics. Dynamic Bayesian Network (DBN) is capable of modeling correlation and temporal characteristics as it is Bayesian network spanning over time. Furthermore, DBN has the capability to model hierarchical stochastic processes, as it is a generalization to hierarchical HMM (HHMM). This capability is very relevant as sports videos are hierarchically stochastic inherently, e.g. the phase - event hierarchy. Therefore, it becomes a natural choice to develop the early fusion framework.

5.2 General Description about DBN

This Section gives a general description of DBN and the next Section is dedicated to the design of our particular system. Dynamic Bayesian Networks are directed graphical models of stochastic processes. They are hybrid products inheriting from both Bayesian Networks (BNs) and state-space models [64]. DBNs inherit from BNs in the sense that DBNs are directed acyclic graphs (DAGs) and any variable is independent from any other variables given its parents or its distribution is specified by its prior in the case of no parents. They inherit from state-space models in the sense that the models are characterized by some underlying hidden state of the world that generates the observations, and that this hidden state evolves over time. DBNs are an generalization of Hidden Markov Models (HMMs) in that they allow more than one state variable and hence it is a more general graph topology. In addition, DBNs allow state variables to be visible aside from observation variables. They are also an generalization of Bayesian Networks (BNs) in the sense that they model a Bayesian Network spanning over time. That is, a DBN not only models the interdependencies between variables at each time slice like what a Bayesian Network does, but also models the interdependencies across time slices.

Though DBNs may have generalizations in various aspects, the most commonly used ones are based on the following set of assumptions. (a) The stochastic process is on discrete time, and the index to the time slice increments by one every time a new observation arrives. (b) The state at time t depend on the state at the preceding time slice $t-1$ and not any earlier time slices, i.e. the first-order

Markov property. (c) The network topology and parameters do not change over time despite the name "dynamic", i.e. the model is time-invariant. In order to tailor to our problem of video analysis, we also assume that states are discrete and finite. This is because semantic labels (phases and events), which will be treated as states, are all discrete. (e) The model has no input variables[1]. (f) Observation only depends on the state in the same time slice. The following paragraphs give a brief description on notation of DBNs.

A DBN is a directed acyclic graph with variables $Z_t = (X_t, Y_t)$,
where

t is the index to the time slice,

X_t is the set of state variables at time slice t. X_t are discrete,

Y_t is the set of observation variables at time slice t. Y_t can be continuous or discrete.

From the topology point of view, a DBN is a pair - $(B_1|B_{\rightarrow})$, where B_1 is a BN which defines the prior $P(Z_1)$, and $B_{\rightarrow}$ is a two-slice temporal Bayesian Network (2TBN) which defines $P(Z_t|Z_{t-1})$ by means of a DAG as follows:

$$P(Z_t|Z_{t-1}) = \prod_{i=1}^{N} P(Z_t^i|Pa(Z_t^i)) \tag{5.1}$$

where

- N is the total number of variables (state and observation variables combined) in one time slice,
- Z_t^i is the i'th variable at time t, which is an element of X_t or Y_t,
- $Pa(Z_t^i)$ are the parents of Z_t^i in the graph. They can either be in the same time slice or in the previous time slice. The arcs between slices must be from older ones to newer ones, reflecting the flow of time. The arcs within a slice are arbitrary, so long as the overall DBN is a DAG.

[1] Input variables reflect control from external forces. Anyway, this is irrelevant to video.

From the parameter point of view, the variables in the first slice of a 2TBN do not have any associated parameters, but each variable in the second slice of the 2TBN has an associated conditional probability distribution (CPD), which defines $P(Z_t^i|Pa(Z_t^i))$ for all $t > 1$. Table 5.1 gives the most common priors and CPDs for variables with discrete parents. Among $P(Z_t^i|Pa(Z_t^i))$, $P(X_t^i|Pa(X_t^i))$ defines a state transition function, and $P(Y_t^i|X_t^i)$ defines an observation function. The whole parameter set Θ of a DBN consists of the priors, the state transition functions and the observation functions.

$$\begin{aligned}\Theta &= \{P(Z_1^i), i = 1 \ldots N\} \\ &\cup \{P(X^i|Pa(X^i)), i = 1 \ldots M\} \\ &\cup \{P(Y^i|Pa(Y^i)), i = 1 \ldots L\}\end{aligned} \tag{5.2}$$

where

M and L are the numbers of state and observation variables, respectively.

Discrete parents	Name	CPD
None	Multinomial	$P(Y = j) = \pi(j)$
i	Conditional multinomial	$P(Y = j\|X = i) = A(i, j)$
None	Gaussian	$P(Y = y) = \mathcal{N}(y; \mu, \Sigma)$
i	Conditional Gaussian	$P(Y = y\|X = i) = \mathcal{N}(y; \mu_i, \Sigma_i)$

Table 5.1: Most common priors and CPDs for variables with discrete parents.

5.3 Our Early Fusion Framework

This Section explains our early fusion framework, including network structure, choice of the observations, learning and inference algorithms, complexity issues and how domain knowledge is incorporated to the framework.

5.3.1 Network Structure

In designing the DBN to perform integrated audiovisual and text analysis, there are some guidelines to be kept in mind.

- The framework should make the best effort in incorporating domain knowledge. Domain knowledge provides constraints in addition to training data. DBN may be more apt than other statistical learning algorithms in incorporating domain knowledge as its DAG topology may be expressive in sketching semantic ontology.
- We should particularly exploit DBN's capability in modeling hierarchical stochastic processes. In our particular task, this hierarchy refers to the one composed by phase - event as described in Section 3.1.
- The DBN should grasp different mechanisms - spatiotemporal evolution associated with audiovisual signals and correlation associated with text.
- Temporal correspondence between cues from the two forms of information should be reasonably slackened to accommodate asynchronism.
- The time and space complexity should be contained to make the system tractable. This involves constraining the number of layers, number of state variables, sizes of CPDs and algorithms for learning and inference.

A 2TBN representation of the network structure is shown in Figure 5.2. The network borrows the notation of hierarchy from HHMM, which dictates that the Markov chains at adjacent levels are bonded by alignment - one step in the Markov chain of the higher-level is aligned to the complete Markov chain at one level lower. The network has three levels aside from the observation. From the top downward, they represent phase, event, and hidden states of two independent HMMs, respectively. There is not a level representing actions, because actions are not well represented by audiovisual characteristics or completely covered by text. There is an exit variable attached to each level but level 3 that signals the

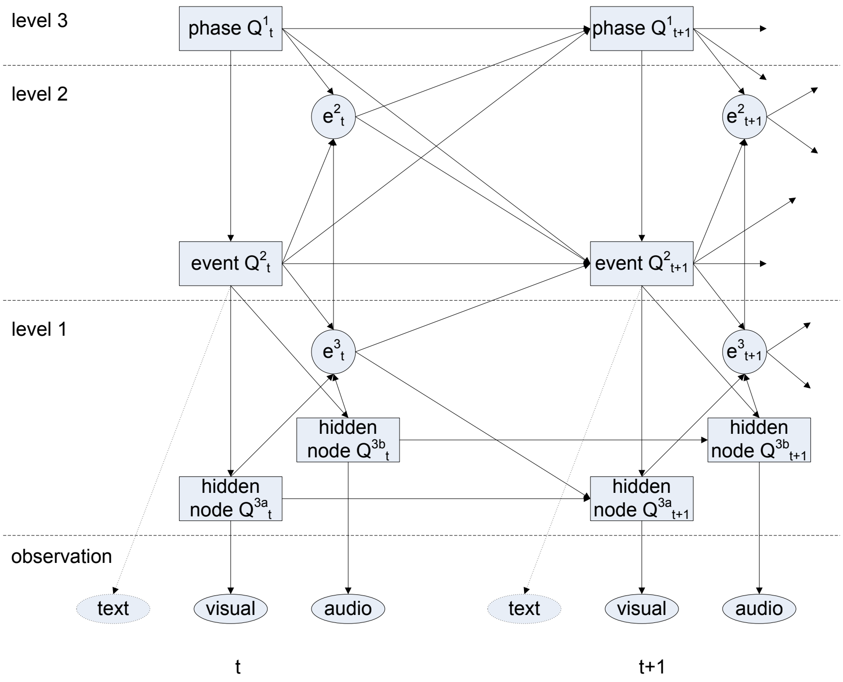


Figure 5.2: Network structure of the early fusion framework.

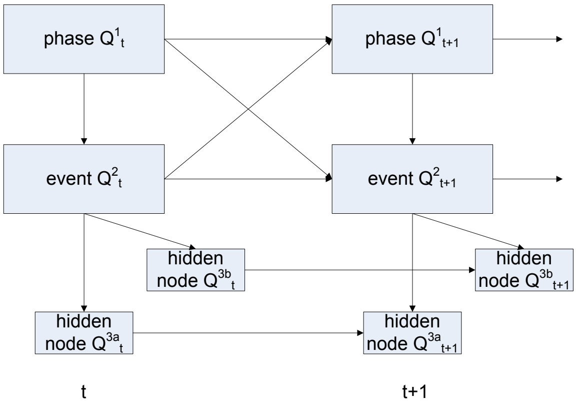


Figure 5.3: The backbone of the network.

start-over of the Markov chain at the level. Level 3 has no exit variable because it is not allowed to start over. To better explain the whole network structure, we will look at one part at one time. Note that we will be using HMM notation (A, B, π) in explaining Markov characteristics ($A := \{a_{i,j}\}$: state transition probabilities, $B := \{b_j(k)\}$: emission probabilities, and $\pi := \{\pi_j\}$): priors).

The backbone

Levels 2 and 3 form the backbone of the network. Horizontal arcs between phases or between events mean phase and event form Markov chains, respectively. Horizontal arcs associated with the two hidden node Q^{3a} and Q^{3b} represent two HMMs, accounting for the visual and audio evolution, respectively. Vertical arcs mean the Markov characteristics are subject to certain conditions. Specifically, Markov characteristics of event is subject to the current phase. The HMMs of visual and audio evolution are subject to the current event. The arc pointing from phase to event across slice means that event's conditional probability distribution may be differentiated with regards to the preceding phase as well. So may the phase variable. How the diagonal arcs interfere with Markov characteristics is explained in the part on “The exit variables”.

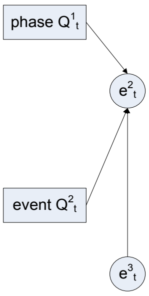


Figure 5.4: Exit variables (a)

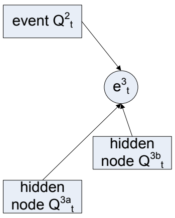


Figure 5.5: Exit variables (b)

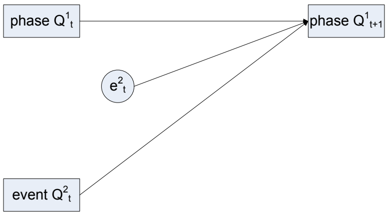


Figure 5.6: Exit variables (c)

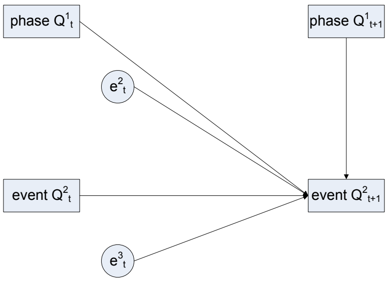


Figure 5.7: Exit variables (d)

The exit variables

The exit variable is a special state variable; it signals the start-over of Markov chains at its own level and disables/enables the transition of the higher-level variable. It has binary states: on (1) and off (0). Exit variable turned on means that the current Markov chain has come to an end, and a new Markov chain is about to start. As dictated by the alignment bond, this implies that the variable at the higher level may transit and all exit variables at lower levels are on.

Figure 5.4 shows the level 2 exit variable with incoming arcs. Its CPD is interpreted as follows.

$$P(e^2_t = 1 | e^3_t = e, Q^1_t = k, Q^2_t = i) = \begin{cases} 0, & \text{if } e = 0 \\ a^2_k(i, \mathsf{end}), & \text{if } e = 1 \end{cases} \tag{5.3}$$

Figure 5.5 shows the level 1 exit variable with incoming arcs. This exit variable signals that the two independent HMMs will come to an end simultaneously. Its

CPD is interpreted as follows.

$$P(e_t^3 = 1 | Q_t^2 = i, Q_t^{3a} = p, Q_t^{3b} = q) = a_i^{3a}(p, \mathsf{end}) a_i^{3b}(q, \mathsf{end}) \tag{5.4}$$

Figure 5.6 shows how the level 2 exit variable interferes with the phase variable. When e_t^2 is on, certain values (meaning certain events) of Q_t^2 may specify the phase's state transition probability in spite of normal Markov characteristics.

$$\begin{aligned} &P(Q_{t+1}^1 = j | e_t^2 = e, Q_t^1 = k, Q_t^2 = i) \qquad (5.5)\\ &= \begin{cases} 1, & \text{if } e = 0, j = k \\ 0, & \text{if } e = 0, j \neq k \\ a_i^1(k,j), & \text{if } e = 1, i \in \{\text{certain events}\} \\ a_{normal}^1(k,j), & \text{if } e = 1, i \notin \{\text{certain events}\} \end{cases} \end{aligned}$$

Figure 5.7 shows how the exit variables interfere with the event variable. Only when e_t^3 is on can the event variable advance a step along its Markov chain. In normal case, the event variable's Markov characteristics is only dependent on the covering phase; however, under certain circumstances the first event of a phase may be forced to obey a different Markov characteristics transiently. This is made possible by e_t^2 being turned on.

$$\begin{aligned} &P(Q_{t+1}^2 = m | e_t^2 = e, e_t^3 = f, Q_t^1 = k, Q_t^2 = i, Q_{t+1}^1 = j) \qquad (5.6)\\ &= \begin{cases} 1, & \text{if } f = 0, m = i \\ 0, & \text{if } f = 0, m \neq i \\ a_j^2(i,m), & \text{if } f = 1, e = 0 \\ \pi_j^2|_k, & \text{if } f = 1, e = 1, k \in \{\text{certain preceding phases}\} \end{cases} \end{aligned}$$

The observations

The observations F comprise those derived from three modalities.

$$F := F^T \cup F^V \cup F^A \tag{5.7}$$

where

F^T, F^V and F^A denote observations derived from text, visual and audio signals, respectively.

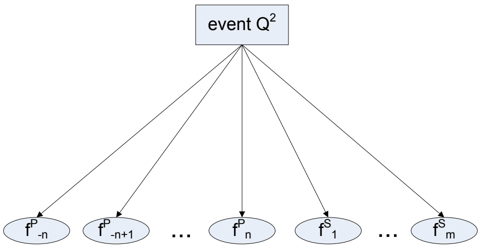


Figure 5.8: Textual observations

F^T are derived from detailed text. we only expect the early fusion framework to work with detailed text as compact text provides too sparse recordings to establish sound conditional probabilities. There are a number of separate observations derived from text, although there is only one oval in Figure 5.2. The expanded illustration is shown in Figure 5.8. There are two types of textual observations, aiming at capturing patterns based on slot-specific phrase similarity and sequence similarity, respectively.

$$F^T := \left\{f^P_{-n:n}\right\} \cup \left\{f^S_{1:m}\right\} \tag{5.8}$$

where

f^P are unigrams and f^S are bigrams. The unigram lexicon is formed after stemming and synonym consolidation. The bigram lexicon is formed based on the unigram lexicon and is restricted to m most meaningful bigrams for complexity concern. m is domain-dependent and is set empirically. n is half width of a symmetric slot window positioned at the center of the current processing unit (for explanation of the processing unit, please see "3.4.1 The processing unit"). Textual observations associated to a processing unit are derived from terms in this window. n may be dependent on the game and format of the textual description.

$f^P_i, i = -n : n$ is the phrase at the ith slot

$f^S_j = \{\mathsf{true}, \mathsf{false}\}, j = 1 : m$ refers to whether bigram_j is included in the window.

Some units of the video are represented by categorical labels (e.g. commercials, narratives and replays) while others by numeric audiovisual features. To unify them, a discrete feature representation is adopted. The visual observation is

$$f^V = \begin{cases} \in \{\textsf{commercial}, \textsf{narrative}, \textsf{replay}, \textsf{audience-scene}, \textsf{canonical-scene}\} \\ \text{or} \\ \text{camera motion pattern } AND \text{ motion magnitude } AND \text{ field-zone} \end{cases}$$

For audio features, we choose only those that provide a general idea of the match progress and the level of excitement. The values of the audio observation are common to various games.

$$f^A = \begin{cases} \in \{\textsf{commercial}, \textsf{narrative}, \textsf{replay}, \textsf{audience-scene}, \textsf{canonical-scene}\} \\ \text{or} \\ \in \{\textsf{whistle}, \textsf{cheering}, \textsf{speech}, \textsf{music}, \textsf{music+speech}, \textsf{excited-speech}, \\ \quad \textsf{noise}\} \end{cases}$$

To avoid zero conditional probabilities due to incomplete coverage by the training data, zero conditional probabilities are replaced by a very small float number (10^{-9}).

5.3.2 Learning and Inference Algorithms

The learning task of the whole DBN is split to several subtasks to exploit the fact that a part of DBN can be estimated independently from the rest if all participating variables are visible at all slices. In the case when this separable part of DBN is a single variable and this variable is visible at all slices, we obtain its CPD by counting the co-occurrences

$$CPD^{Z_i} := \{P(Z_i = c)|Pa(Z_i) = p), \text{ for all } c \text{ and all } p\} \tag{5.9}$$

$$P(Z_i = c)|Pa(Z_i = p)) = \frac{\text{number of } (Z_i = c, Pa(Z_i) = p)}{\text{number of } (Pa(Z_i) = p)} \tag{5.10}$$

This method applies to parameters associated with the phase and event variables as well as the textual observations. In the case when this separable part contains hidden variables, as in the case of visual and audio HMMs, the parameters can be estimated by the Baum-Welch algorithm on a smaller scale. We use all observation sequences that are associated to the same event (visually and aurally separately) as training sequences to estimate the corresponding (A, B, π) the way a regular HMM does. Separation into independent subtasks reduces the time and space complexity (see Table 5.2).

		Time complexity	Space complexity
Learning	Standard EM	$O(T^{a}K^{b2}), T \sim (300, 1000), K \sim (500, 1000)$	$O(S^{c}TK), T \sim (300, 1000), K \sim (500, 1000)$
	Splitting adopted	$O(TK^2), T \sim (5, 15), K = 5$	$O(STK), T \sim (5, 15), K = 5$
Inference	Offline	$O(TK^2), T \sim (300, 1000), K \sim (500, 1000)$	$O(STK), T \sim (300, 1000), K \sim (500, 1000)$
	Online	$O(L^{d}K^2), L = 5, K \sim (500, 1000)$	$O(SLK), L = 5, K \sim (500, 1000)$

Table 5.2: Complexity control on the DBN.

[a] T: length of the sequence
[b] K: number of the states
[c] S: size of a forward/backward message
[d] L: lenght of the fixed lag

Because test sequences may be as long as having thousands of slices, we need an online inference algorithm to achieve space tractability. We choose a fixed-lag smoothing algorithm presented by Murphy [64] and will briefly explain it here. The algorithm is based on a pair of forward-backward messages. Note that the following formulas are based on HMM notation. A DBN made up of all discrete nodes such as the one used in the early fusion framework is equivalent to a HMM.

The forward message $\alpha_t(i)$ and backward message $\beta_t(i)$ are defined as

$$\alpha_t(i) \overset{\text{def}}{=} P(X_t = i | y_{1:t}) \tag{5.11}$$

$$\beta_t(i) \overset{\text{def}}{=} P(y_{t+1:T} | X_t = i) \tag{5.12}$$

where the subscript t refers to the slice; $1 : t$ refers to all slices from 1 to t; i refers to the i-th state X can take on. Inference $\gamma_t(i) \overset{\text{def}}{=} P(X_t = i | y_{1:T})$ is obtained by combining the two messages

$$\begin{aligned} P(X_t = i|y_{1:T}) &= \frac{1}{P(y_{1:T})} P(y_{t+1:T}|X_t = i, y_{1:t}) P(X_t = i, y_{1:t}) \\ &= \frac{1}{P(y_{1:T})} P(y_{t+1:T}|X_t = i) P(X_t = i|y_{1:t}) \end{aligned}$$

or

$$\gamma_t \propto \alpha_t . * \beta_t$$

where $.*$ denotes elementwise product, i.e., $\gamma_t(i) \propto \alpha_t(i)\beta_t(i)$.

The forward message

α can be recursively calculated as follows.

$$\alpha_t(j) = P(X_t = j|y_{1:t}) = \frac{1}{c_t} P(X_t = j, y_t|y_{1:t-1})$$

where

$$P(X_t = j, y_t|y_{1:t-1}) = \left[\sum_i P(X_t = j|X_{t-1} = i) P(X_{t-1} = i|y_{1:t-1}) \right] P(y_t|X_t = j) \tag{5.13}$$

and

$$c_t = P(y_t|y_{1:t-1}) = \sum_j P(X_t = j, y_t|y_{1:t-1}) \tag{5.14}$$

In vector-matrix notation, this becomes

$$\alpha_t \propto O_t A' \alpha_{t-1} \tag{5.15}$$

where A' denotes the transpose of A and $O_t(i,i) \overset{\text{def}}{=} P(y_t|X_t = i)$ is a diagonal matrix containing the conditional likelihood of the evidence at time t.
The base case is

$$\alpha_1(j) = P(X_1 = j|y_1) = \frac{1}{c_1} P(X_1 = j) P(y_1|X_1 = j)$$

or

$$\alpha_1 \propto O_1 \pi$$

The backward message
Because $P(y_{T+1:T}|X_T = i) = P(\emptyset|X_T = i) = 1$, the base case is

$$\beta_T(i) = 1$$

The recursive step is

$$P(y_{t+1:T}|X_t = i) = \sum_j P(y_{t+2:T}|X_{t+1} = j)P(y_{t+1}|X_{t+1} = j)P(X_{t+1} = j|X_t = i) \tag{5.16}$$

or

$$\beta_t = AO_{t+1}\beta_{t+1}$$

Combination of α_T and β_T

$$P(X_t = i|y_{1:T}) \propto P(X_t = i|y_{1:t})P(y_{t+1:T}|X_t = i)$$

Since $P(X_t = i)$ is a probability, it is determined by normalization subject to $\sum_i P(X_t = i) = 1$.

The fixed-lag smoothing
Fixed-lag smoothing estimates $P(X_{t-L}|y_{1:t})$, where $L > 0$ is the lag. If the delay can be tolerated, then this is clearly a more accurate estimate than the filtered quantity $P(X_{t-L}|y_{1:t-L})$, which does not take “future” evidence into account. On the other hand, as an online algorithm it does not store all messages. The procedure of fixed-lag smoothing is given in 5.9. The first part is the main routine, and the second part is the the recursive part.

The notations are explained here. f is a forward message. As this is an online algorithm, we cannot store all messages, instead we keep a wrap-around buffer of length $L + 1$ - $f[1 : L]$. $f[i]$ is the i-th entry in this buffer; k is a pointer to the position in the buffer that contains the most recent forward message. The algorithm assumes one-based indexing, and use the notation $t \oplus 1$ and $t \ominus 1$ to

```
$f[1] = \text{Fwd1}(y_1)$
for $t = 2 : L$
    $f[t] = \text{Fwd}(f[t-1], y_t)$
$k = L + 1$
for $t = L + 1 : \infty$
    $(b_{t-L|t}, f[1:L], k) = FLS(y_t, f[1:L], k)$

function $(b, f[1:L], k) = FLS(y_t, f[1:L], k)$
$L = \text{length}(f)$
$k' = k \ominus 1$
$f[k] = \text{Fwd}(f[k'], y_t)$
$b = \text{BackT}(f[k])$
for $\tau = 1 : L$
    $b = \text{Back}(b, f[k'])$
    $k' = k' \ominus 1$
$k = k \oplus 1$
```

Figure 5.9: Pseudo-code for fixed-lag smoothing.

represent addition/subtraction modulo L. Fwd and Back are the forward and backward passes; Fwd1 and BackT are their base cases. Definition and call to function $(b, f[1:L], k) = FLS(y_t, f[1:L], k)$ are represented in MATLAB syntax.

5.3.3 Incorporating Domain Knowledge

Recall that domain knowledge associated with events of team sports generally has four parts: (a) temporal location specification, (b) sequential relations between events, (c) semantic composition and (d) audiovisual patterns.

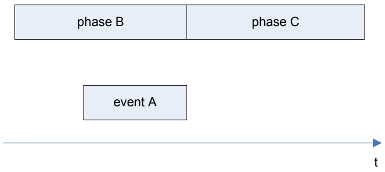


Figure 5.10: Constraint of event A followed by phase C.

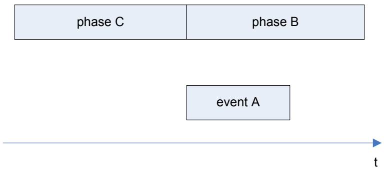


Figure 5.11: Constraint of event A preceded by phase C.

Preceding phase	Preceding event	This phase	Conditional probability
left-advance	goal	break	1
	goal	not break	0
	...	...	...
right-advance	corner-kick	break	1
	corner-kick	not break	0
	...	...	...

Table 5.3: Illustrative CPD of the phase variable in Figure 5.10 with diagonal arc from event to phase across slice.

Preceding phase	Preceding event	This phase	Conditional probability
left-advance	any	break	(0,1)
		draw	(0,1)
		left-advance	(0,1)
		...	...
right-advance	any	break	(0,1)
		draw	(0,1)
		left-advance	(0,1)
		...	...

Table 5.4: Illustrative CPD of the phase variable in Figure 5.10 with no diagonal arc across slice.

Modeling temporal location specifications

The temporal location specifications point out hosting phases' types and conditions. Hosting phases' types are modeled by conditional probabilities $P(Q^2|Q^1)$. Modeling hosting phases' conditions is attempted by diagonal arcs. They help impose constraints concerning events and adjacent phases derived from domain knowledge (see Chapter3.1). Figures 5.10 and 5.11 constrain the phase in terms of what follows or precedes, respectively. Take the *corner-kick* for example (in Figure 5.10 scenario), the effect of diagonal arc on the conditional probability distribution (CPD) of the phase node is illustrated in Table 5.3 and 5.4. The diagonal arcs not only improve accuracy of detected events, but that of phases since the event and phase nodes are inferred simultaneously by a unified algorithm. Experiments show that 1/3 of missing *breaks*, 1/3 of false positives of *left advances* and

1/4 of false positives of *right advances* are reduced; and 13 misplaced events get corrected with the help of diagonal arcs. However, arcs across slices as a means of modeling constraints are not handy. They provide limited modeling capability at the cost of more expensive computation and scattering of training samples. In our case, diagonal arcs can only model the conditions in terms of adjacency of phases to events at one (either the starting or the ending) boundary. Modeling conditions on both boundaries or involving phases further away (e.g. those of *corner-kick*, *free-kick* or *penalty*) would entail more complex model structure. Considering their cost-effectiveness, such models are opted out.

Modeling sequential relations between events

As a stochastic algorithm, DBN effectively models sequential relations between events in transition probabilities. By contrast, the late fusion framework failed to provide this modeling.

Modeling semantic composition

Detailed text descriptions generally use a limited vocabulary and exhibit strong syntactic patterns. Therefore pattern-based matching may be a solution to modeling semantic composition. Sequential patterns are desirable as an event instantiates a path of the semantic composition model. To achieve good generality and to keep CPDs small, bigrams are employed for inducing sequential patterns. Unigrams play a supplementary role in the circumstances where descriptions are sparser and consistent bigrams are not available for every event type. Table 5.5 shows that neither unigrams nor bigrams alone would be sufficient to tell positives from negatives. However, the combination of them would. The idea of grasping patterns by unigrams and bigrams is similar to that described in [25]. Wrapping of unigrams and bigrams is important because the system needs to differentiate positive and negative instances well in the asynchronous circumstance and in the meantime keeps the CPD small. Table 5.6 compares frame-level accuracy of various wrappings of unigrams or bigrams in a Bayesian network. It suggests the wrapping of slot-based unigrams and presence-based bigrams performs the best. Positives have disparate composite conditional probabilities from negatives as they

	Unigram(u)	$P(u = 1 \mid e = 1)$	$P(u = 1 \mid e = 0)$	Bigram(b)	$P(b = 1 \mid e = 1)$	$P(b = 1 \mid e = 0)$
Goal	score	1	.0013	shot - score	.9412	0
Save	save	.9770	.0021	shot - save	.7356	.0007
Shot-off-target	shot	.9597	.0501	assist - shot	.4758	.0029
Penalty	penalty	1	0	-	-	-
Corner-kick	corner-kick	1	.0035	corner-kick - shot	.1333	.0014
Free-kick	free-kick	1	.0168	foul - free-kick	.8065	.0060
Offside	offside	1	.0027	assist - offside	.1795	.0020
Substi-tution	substitute	1	0	assist - substitute	.3333	-
Yellow-card	yellow-card	1	0	foul - yellow-card	1	0
Red-card	red-card	1	0	foul - red-card	1	0

Table 5.5: Strength of best unigrams and bigrams

	Recall	Precision	F1
Unigrams only (a)	.608	.629	.618
Bigrams only (b)	.239	.574	.337
Presence-based unigrams + bigrams (c)	.670	.541	.599
Slot-based unigrams + bigrams (d)	.637	.703	.668
Time slice-based unigrams + bigrams (e)	.359	.427	.390

Table 5.6: Frame-level accuracy of various textual observation schemes

match multiple slots and match them at correct positions. Bigrams also help in filtering out negatives. Offsets are learned by slot-based statistics. Slot accommodates asynchronism better than slice, i.e. evenly paced temporal unit. CPD size is kept the smallest by having the observations independent. Despite the effectiveness of the textual observations in general, they are still subject to noise brought by irrelevant unigrams and randomness in offsets.

Modeling audiovisual patterns

Modeling of audio and visual evolution is attempted by a dedicated hidden node in the DBN each. However the effectiveness of the modeling is limited, as the evolution of different event types is heterogenous and cannot be effectively captured by a fixed number of states. Some event types don't even have consistent stochastic patterns, e.g. *shot-off-target* and *save.*

5.4 Implementation of the Early Fusion Framework on Soccer and American Football Video

Domain-dependent design or processing in the early fusion framework is mainly on definition of the phase and event variables, the single-phrase and bigram lexicons for deriving textual observations, and domain-dependent visual features for deriving visual observations.

5.4.1 Implementation on Soccer Video

Phase types are defined in "3.2 Domain Knowledge for Both Frameworks", that is, {left-advance, right-advance, draw, break}.

For the values of the event variable, we define a superset of the target events listed in "3.2 Domain Knowledge for Both Frameworks". This is because the whole duration of the video should be covered exhaustively by the DBN, which is not possible with the target events alone. For instance, there are *advances* that do not end with an attempt at the goal; there are passes at the beginning of *advances* that do not pose any pressure on the opponents. In view of these facts, we define some "padding" events in addition to the target events, and the complete value set of the event variable is

$$\begin{aligned}\mathsf{event} \in \{ & \mathsf{goal}, \mathsf{save}, \mathsf{shot\text{-}off\text{-}target}, \mathsf{penalty}, \mathsf{corner\text{-}kick}, \mathsf{free\text{-}kick}, \\ & \mathsf{offside}, \mathsf{initial\text{-}pass}, \mathsf{premature\text{-}offense}, \mathsf{attack}, \\ & \mathsf{midfield\text{-}competition}, \mathsf{yellow\text{-}card}, \mathsf{red\text{-}card}, \mathsf{substitution}, \\ & \mathsf{not\text{-}in\text{-}play}, \mathsf{editing\text{-}artifact}\} \end{aligned} \tag{5.17}$$

where
initial-pass - the part when a goal keeper and defensive players dribble or pass the ball slowly in the defending side.
premature-offense - the part when the *advance* is futile before the offensive team can pose any significant pressure on the defenders.
midfield-competition - the part when two teams are competing for control of the ball in the middle of the field. This is the only event that may take place during a *draw* phase. The purpose of using two synonyms at the phase and event levels is to keep the hierarchy clear-cut.
attack - the part when the offensive team poses significant pressure on the defenders, however, there are no attempts on goals.
not-in-play - the part when the play is not going on nevertheless the image is still on the pitch.
editing-artifact - the part when the image shows commercials, replays or narratives resulting from editing.

The two lexicons of unigrams and bigrams are given in the appendix. The size of bigram lexicon is empirically set as $m = 10$. Soccer has one domain-dependent visual features for deriving visual observation - player-density.

$$\mathsf{player\text{-}density} \in \{ \mathsf{low}, \mathsf{high}\} \tag{5.18}$$

Player density can indicate intenseness of the ongoing play, thus it is helpful in differentiating stages within an *advance*, e.g., to recognize *initial-pass* from *attack*. It is obtained from the number of players visible. A unit having visible players less than a threshold ($n = 6$) is said to be low in density and high otherwise. Players are detected in the way described in [6]. First, blobs of individual players or groups of players are segmented out from the pitch by color differencing. Players are then identified by adaptive template matching, using an elliptical template that has the height/width ratio equal to the median of the height/width ratios of the blobs extracted (outliers excluded). To remedy missing detections resulting from occlusion, numbers of players obtained from individual frames are smoothed over the unit before the average is taken as the unit's number of players.

5.4.2 Implementation on American Football Video

Phase types are defined in "3.2 Domain Knowledge for Both Frameworks", that is, $\{\textsf{left-advance}, \textsf{right-advance}\}$.

The value set of the event variable is a union of the target events and padding events, namely

$$\begin{aligned} \textsf{event} \in \{ & \textsf{touchdown}, \textsf{conversion}, \textsf{field-goal}, \textsf{punt}, \textsf{punt-return}, \textsf{fumble-opponent}, \\ & \textsf{interception}, \textsf{touchback}, \textsf{kickoff}, \textsf{safety}, \textsf{fumble-own}, \\ & \textsf{incomplete-pass}, \textsf{pass}, \textsf{rush}\} \end{aligned} \tag{5.19}$$

where
rush - a play wherein the offensive team tries to move the ball forward by running with it.
pass - a play wherein a forward pass is made, whether it is successfully received or it is incomplete.

The two lexicons of unigrams and bigrams are given in the appendix. The size of bigram lexicon is empirically set as $m = 5$. American football has no domain-dependent visual features for deriving visual observation.

5.5 Evaluation of the Early Fusion Framework

5.5.1 Evaluation of Phase Segmentation

Although phases are only the by-products of DBN, we still evaluate their accuracy for it may be related to overall events' accuracy. We would also like to use phases as a benchmark to compare the two frameworks. Tables 5.7 and 5.8 show the accuracy of soccer phases, while Tables 5.9 and 5.10 show that of American football phases.

In general, phase accuracy in terms of misses and false positives by the early fusion framework is satisfactory. All phase types have misses or false positives close to or fewer than 10% of their instances in the ground truth. The worst-performing phase type is *draw*; however it has the least impact on event detection. The best-performing phase types are the two *advances* in American football, which achieved zero misses or false positives. Compare to the results by the late fusion framework, the early fusion framework has: (a) significant improvement in reducing misses and false positives of *advances* (both in soccer and in American football) and in misses of *break*; (b) slight improvement in false positives of *draw* and *break*; and (c) a few more misses in *draw*. Generally speaking, the accuracy of phases in term of misses and false positives produced by the early fusion framework is better than that of the late fusion framework. As for the frame-level accuracy, however, the improvement is not as significant. Recall rates are in the range of $0.75 \sim 0.88$, and precision rates in $0.79 \sim 0.87$ as compared to the ranges of $0.79 \sim 0.86$ for recall and $0.77 \sim 0.87$ for precision by the late fusion framework.

In the perspective of phase segmentation, the early fusion framework is similar to the hierarchical HMM used in the late fusion framework except for the additions of text cue and of the event node. We will discuss how text cue and the event node affect the results as follows.

	Ground truth	Misses	False positives
Left-advance	976	54	46
Right-advance	943	41	36
Draw	465	51	42
Break	311	15	15

Table 5.7: Misses and false positives of soccer phases by the early fusion framework.

	Ground	Detected				Recall	F1
	truth	(a)	(b)	(c)	(d)		
Left-advance (a)	266530	221825	9243	19042	16420	.832	.825
Right-advance (b)	246028	8479	203475	18296	15778	.827	.821
Draw (c)	255119	22139	20436	191694	20850	.751	.770
Break (d)	409484	18905	16765	14067	359747	.879	.875
Detected total	1177161	271348	249919	243099	412795	-	-
Precision	-	.817	.814	.789	.871	-	-
Weighted F1	.829						

Table 5.8: Accuracy of soccer phases by the early fusion framework.

	Ground truth	Misses	False positives
Left-advance	63	1	0
Right-advance	63	2	0

Table 5.9: Misses and false positives of American football phases by the early fusion framework.

	Ground	Detected		Recall	F1
	truth	(a)	(b)		
Left-advance (a)	201935	176824	6442	.876	.855
Right-advance (b)	273279	11435	234415	.858	.861
Other	-	23584	30068	-	-
Detected	-	211843	270925	-	-
Precision	-	.835	.865	-	-
Weighted F1	.859				

Table 5.10: Accuracy of American football phases by the early fusion framework.

Text is accurate but incomplete in the coverage of phases. It covers all American football *advances*, approximately 90% of soccer *advances*, 10% of *breaks* and no *draws*. Though little text on *break* or *draw* is available, text still helps in these two phase types. To be more specific, text has these effects: a) it ensures detection of all documented *advances* on top of those having consistent audiovisual appearances; (b) it reduces missing *breaks* by ensuring detection of score-attempting events (such as *goal* and *corner-kick*) and through diagonal arcs; (c) it helps to reduce false positives of *break* and *draw* by reducing missing *advances* and *breaks*; and (d) in a stochastic algorithm such as DBN, text plays an anchoring role by ensuring detection of *advances* at some points and taking advantage of propagation. However, text provides little clue to pinning boundaries.

Addition of the event node helps in phase-level accuracy but has negative effects in frame-level accuracy. It helps in the sense that it serves as the intermediary for text to function; and text is shown to be effective in indicating phases. The event node has negative effects as it replaces the original mapping of audiovisual evolution to phase with three mappings, namely audiovisual evolution to event, text to event and sequential pattern of events to phase. As a result, the system becomes more complex and less robust. Moreover, the mapping of audiovisual evolution to phase was consistent based on a small set of relevant features - camera motion pattern, motion magnitude and field zone. Yet the three new mappings are not as consistent: (a) mapping of audiovisual patterns to event may not effectively model all event types by a Markov process and a universal set of features. Therefore, event boundaries are usually inaccurate, even though the events may be indicated by the text. Poor event boundaries would implicate phase boundaries; (b) mapping of text to event would not work when text is unavailable; and (c) mapping of sequential pattern of events to phase may not capture all variations by one Markov process. Because of the negative effect of the event node, the improvement in frame-level accuracy is far less significant than reduction in misses and falsely detected phases.

5.5.2 Evaluation of Event Detection

Tables 5.11 and 5.12 give the frame-level accuracy of soccer and American football events by the early fusion framework, respectively. Confusion matrices help us to conduct error analysis.

The accuracy is acceptable - with F1 values of most event types in both games in the range of 0.70 $\sim$ 0.85; weighted F1 is 0.738 for soccer and 0.804 for American football. Nevertheless, the accuracy by the early fusion framework is lower as compared to that by the late fusion framework. For most soccer event types, F1 value is lower by a magnitude in the range of 0.04 $\sim$ 0.16, and the weighted F1 is lower by 0.11; for American football the corresponding range is 0 $\sim$ .15 for F1 value, and lower by 0.04 in weighted F1. Comparing Tables 5.11 and 5.12 we find that the confusion between event types in American football is significantly less than that in soccer because events are well separated by timeout in American football and confusions are unlikely to occur. In general, the early fusion framework is poorer than late fusion framework in identifying boundaries, and a continuous game suffers more than an intermittent game. A continuous game relies on the early fusion framework's capability of modeling boundaries, whereas an intermittent game mainly relies on the quality of timeout detection. Because of this distinction, we discuss the two kinds of games separately.

Most soccer event types have similar accuracy (F1 in the range 0.70 $\sim$ 0.78) as text plays a pre-dominant role and boundary modeling of different event types does not vary much. The best-performing event types are *red-card* and *penalty.* Their textual cue is reliable and their boundaries are clear-cut and well captured. *Shot-off-target*, *free-kick*, *offside* are the worst-performing event types. Their textual cue is less reliable, resulting in more confusion between them and other event types. This is also evidenced by more misses or false positives of these event types. The accuracy of different American football event types is also similar but this is due to timeouts. Accuracy is mainly determined by the quality of timeout detection rather than confusion between event types. And this quality does not

	Ground truth	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
Goal (a)	9054	6235	0	0	0	429	291	0	158	0
Save (b)	26851	0	21889	245	54	402	317	0	429	0
Shot-off-target (c)	39127	0	1914	27687	0	372	356	376	529	443
Penalty (d)	1456	0	0	0	1103	0	0	0	0	0
Corner-kick (e)	27482	209	527	227	0	21433	0	113	0	0
Free-kick (f)	21082	151	392	460	0	0	16597	0	178	0
Offside (g)	9239	0	334	253	0	389	0	5015	639	0
Initial-pass (h)	89723	0	0	2069	0	0	0	192	62731	3722
Premature-offense (i)	55117	0	0	338	0	0	1291	0	4465	37421
Attack (j)	233427	496	3756	6141	0	4287	4372	1438	21714	9453
Midfield-competition (k)	255119	121	1147	1292	0	0	594	166	8723	3012
Yellow-card (l)	14166	0	0	0	0	0	0	0	0	1486
Red-card (m)	2234	0	0	0	0	0	0	0	0	0
Substitution (n)	29827	0	0	0	0	339	0	0	0	532
Not-in-play (o)	166361	142	129	0	0	1314	1026	711	1450	2797
Editing-artifact (p)	196896	0	239	392	0	370	0	272	385	1412
Detected	1177161	7354	30327	39104	1157	29335	24844	8283	101401	60278
Precision	-	.848	.722	.708	.953	.731	.668	.605	.619	.621
Weighted F1	.738									

	(j)	(k)	(l)	(m)	(n)	(o)	(p)	Rec.	F1
Goal (a)	1683	258	0	0	0	0	0	.689	.760
Save (b)	2402	641	0	0	0	472	0	.815	.766
Shot-off-target (c)	4350	1421	0	0	0	623	1056	.708	.708
Penalty (d)	0	0	0	0	0	353	0	0.758	.844
Corner-kick (e)	3827	0	162	0	0	591	393	.780	.754
Free-kick (f)	1003	0	0	0	0	2301	0	.787	.723
Offside (g)	1493	574	0	0	0	308	234	.543	.572
Initial-pass (h)	10181	5135	0	0	0	2766	2927	.699	.656
Premature-offense (i)	4967	3416	0	0	0	878	2341	.679	.649
Attack (j)	139130	25377	0	0	572	12558	4133	.596	.617
Midfield-competition (k)	25820	193394	1063	0	2578	10822	6387	.758	.774
Yellow-card (l)	0	0	10911	0	832	937	0	.770	.779
Red-card (m)	0	0	0	1786	0	448	0	.799	.889
Substitution (n)	320	2136	0	0	22009	4346	145	.738	.722
Not-in-play (o)	11195	11838	1710	0	4689	123923	5437	.745	.747
Editing-artifact (p)	10930	693	0	0	425	4091	177687	.902	.894
Detected	217301	244883	13846	1786	31105	165417	200740	-	-
Precision	.640	.790	.788	1	.708	.749	.885	-	-
Weighted F1									

Table 5.11: Accuracy of soccer events by the early fusion framework

	Ground truth	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
Touchdown (a)	10426	8516	0	0	0	0	0	0	0
Conversion (b)	3094	0	2462	0	0	0	0	0	0
Field-goal (c)	2937	0	0	2368	0	0	0	0	0
Punt (d)	14629	0	0	0	11087	1863	0	0	0
Punt-return (e)	10397	0	0	0	0	8230	0	0	0
Fumble-opponent (f)	2851	0	0	0	0	0	2281	0	0
Interception (g)	2438	0	0	0	0	0	0	2049	0
Touchback (h)	1193	0	0	0	0	0	0	0	790
Kickoff (i)	22646	0	0	0	0	0	0	0	0
Safety (j)	0	0	0	0	0	0	0	0	0
Fumble-own (k)	2036	0	0	0	0	0	0	0	0
Incomplete-pass (l)	1186	0	0	0	0	0	0	0	0
Rush (m)	271849	0	0	0	437	583	0	0	0
Pass (n)	129532	0	0	0	0	416	0	0	0
Other	-	1751	657	890	2084	897	444	476	105
Detected	-	10267	3119	3258	13608	11989	2725	2525	895
Precision	-	.829	.789	.727	.815	.686	.837	.811	.883
Weighted F1	0.804								

	(i)	(j)	(k)	(l)	(m)	(n)	Other	Recall	F1
Touchdown (a)	0	0	0	0	0	0	1910	.817	.823
Conversion (b)	0	0	0	0	0	0	632	.796	.793
Field-goal (c)	0	0	0	0	0	0	569	.806	.764
Punt (d)	0	0	0	0	0	0	1679	.758	.785
Punt-return (e)	0	0	0	0	712	291	1164	.792	.735
Fumble-opponent (f)	0	0	0	0	267	142	161	.800	.818
Interception (g)	0	0	0	0	0	184	205	.840	.826
Touchback (h)	290	0	0	0	0	0	113	.662	.757
Kickoff (i)	19145	0	0	0	465	0	3036	.845	.843
Safety (j)	0	0	0	0	0	0	0	-	-
Fumble-own (k)	0	0	1597	0	283	0	156	.784	.834
Incomplete-pass (l)	0	0	0	875	0	172	139	.738	.766
Rush (m)	521	0	0	0	236231	1756	32321	-	-
Pass (n)	0	0	0	0	1891	110196	17029	-	-
Other	2798	0	196	224	27492	17224	-	-	-
Detected	22754	0	1793	1099	267341	129965	-	-	-
Precision	.841	-	.891	.796	-	-	-	-	-
Weighted F1									

Table 5.12: Accuracy of American football events by the early fusion framework

vary much across event types or across frameworks.

Typical error causes

Table 5.13 lists the typical causes of errors of the early fusion framework except erroneous modeling of audiovisual patterns. Erroneous modeling of audiovisual patterns is put aside because it accounts for over 80% of errors and makes the other causes of errors look trivial. The Table suggests that:

Error cause	Percentage
(a) Obscured conditional probability	9.2%
(b) Erroneous unit parsing or timeout detection	11.5%
(c) Bias towards longer event types	9.2%
(d) Bias towards event types having large priors	21.8%
(e) Irrelevant unigrams or bigrams in the textual observations	28.7%
(f) Insufficient training data	1.1%
(g) Miscellaneous	18.4%

Table 5.13: Typical error causes in the early fusion framework

- The typical causes of errors in the two frameworks are different except for erroneous parsing of units and detection of timeouts. In the late fusion framework, the errors are mostly related to the quality of individual subtasks, whereas those in the early fusion framework are factors that affect the descriptiveness of probabilities.

- Among the listed error types, irrelevant unigrams or bigrams in the textual observations accounts for the most errors. Because the textual observations are derived from a phrase window with a fixed number of slots, but the number of phrases belonging to the event types varies, hence some irrelevant phrases may be included and incur non-trivial probabilities. The event types having shorter descriptions suffer more seriously, such as *offside* and *substitution*. Generally this type of errors only results in misclassification in the boundaries of events.

- The early fusion framework is found to be biased towards event types with larger priors. The mechanism of DBN is such that all event types compete for generating the observations and the winner is recognized as the event type. The prior of each event type plays a part in the competition serving as a multiplier. Therefore an event type with a larger prior is more favored. This bias would generally not result in misses or false positives of events, as text cue is strong. However, it would have a negative effect on identifying the boundary between two adjacent events. For example, this bias partially accounts for the misclassifications of *goal*, *save*, *shot-off-target* and *corner-kick* frames as *attack*. Another effect of the bias is that event types with small priors generally have low recall and high precision, as evidenced by *penalty*, *red-card* and *offside*.

- The text phrases used in some events may be a subset of that used in other event types. For example, phrases used in *shot-off-target* may be a subset of those used in *save*, and phrases of *punt* is a subset of those of *punt-return*. This poses difficulties in the early fusion framework to differentiate the pair. Instances of the shorter event type are likely to be misclassified as the longer one because the common phrases would induce a non-trivial probability for the longer event type. However it is unlikely to be the other way round, as the terms unique to the longer event type would keep the probability low for the shorter one. Such errors cause mostly instances of *shot-off-target* being misclassified as *save* or instances of *punt* as *punt-return*.

- Conditional probabilities of phrases are obscured by missing or false documentation. As a result, the cues of some phrases are undermined, e.g. "shot" and "free-kick". This may lead to the misclassifications of some documented events.

- Though insufficient training data is only a marginal source of error, it deserves notice. It highlights the fact that as a probabilistic approach, the performance of the early fusion framework is subject to the availability of sufficient training data. Although training data provided by 5 full matches (for soccer and American football each) are sufficient to induce common transition patterns, they are insufficient to induce rare ones, such as those related to *penalty*, *red-card*, or *safety*, etc. In fact, this deficit results in missing of *save* following a *penalty* as the case is unseen in the training data.

- The DBN is poor at capturing audiovisual patterns of individual event types. The audiovisual patterns of different event types are heterogenous. It is hard to capture them using a uniform stochastic algorithm and a fixed feature set. Worse still, some event types do not have consistent audiovisual patterns, especially the padding event types, such as *initial-pass*. Their inaccurate boundaries would interfere with boundaries of neighboring events.

Comparison of the early and late fusion frameworks

We also notice the following differences of the early fusion framework from the late fusion framework.

- An advantage the early fusion framework has over the late fusion framework is that it leverages text and audiovisual cue simultaneously. Although text cue is generally reliable, it may not be so on particular event types. *free-kick* is such an even type. Free-kicks made in mid-field or the defending third of the field are documented but are recognized as negative instances in the ground truth; and the number of this case is significant. Depending largely on the text cue, the late fusion framework is hit severely with a result of 9 false positives. As audiovisual cue is leveraged in the early fusion framework to filter out free-kicks made in mid-field or the defending third, the situation is significantly mitigated - the number of false positives drops to 3.

- The early fusion framework utilizes a consolidated pipeline in which phases are detected simultaneously with events. Phase segmentation and event detection are performed at the best effort with the help of external information. Results show that phases are more accurate by the early fusion framework than by the late fusion framework. However, due to other factors, this advantage is not translated to higher accuracy of detected events.

- A weakness of DBN is its complexity. First, a complex model may be subject to noise brought by irrelevant features, such as irrelevant textual observations in the textual window. Second, errors propagate more easily in a complex model, e.g. errors in earlier slices may bring about errors in subsequent slices.

- The poorer performance in frame-level accuracy of the early fusion framework as compared to the late fusion framework results from poorer representation of the domain knowledge. It is poorer in three aspects. First, the early fusion framework can rule out an event's impossible hosting phases, however it cannot constrain the phase on both the start and the end. For example, temporal location specification of *corner-kick* has rather complex constraint - a *corner-kick* takes place "in an *advance*, which is sandwiched by two *breaks*, and if an *advance* precedes the first *break*, the two *advances* should be in the same direction". Second, modeling of semantic composition by linguistic statistics may introduce noise from irrelevant phrases. Third, audiovisual patterns are poorly captured by a complex unified stochastic model. Theoretically, the early fusion framework is superior to the late fusion framework in modeling sequential relationship among event types. However, this is not demonstrated in our experiments, probably because the text cue is reliable enough and relationship modeling provides little extra help. In the situations where external information is not reliable or complete, the capability of modeling sequential relations between events may be helpful.

Chapter 6

CONCLUSIONS AND FUTURE WORK

This thesis proposes integrated analysis of audiovisual signals and external information sources for detecting events in team sports video. Two frameworks are developed, namely the late fusion and the early fusion frameworks. Asynchronism between the audiovisual signals and the external information sources is the key issue in designing them. The late fusion framework has two modules to process audiovisual signals and external information source separately, with a third module to fuse their outcomes. In the early fusion framework, audiovisual signals and external information source are processed together by a Dynamic Bayesian Network.

Key findings are:

- External information sources are helpful in detecting events from team sports video. The help varies as the level of detail of the description varies. The compact descriptions ensure the detection of the most important events, while the detailed descriptions enable the detection of the full range of events. Providing complete and accurate cue about events, detailed descriptions play a crucial role in both frameworks.
- In the task of detecting events with boundaries, integrated analysis of au-

diovisual signals and external information sources outperforms analysis of a single source of information, thanks to exploitation of their complementary strengths. In terms of weighted F1, the audiovisual analysis achieves $0.63 \sim 0.68$ and the text analysis $0.3 \sim 0.5$, whereas the integrated analysis could achieve 0.84 by the late fusion framework and $0.74 \sim 0.80$ by the early fusion framework.

- The event model, which comprises temporal location specification, sequential relationship between events, semantic composition and audiovisual patterns, well captures the domain knowledge of team sports video and is effective in making both frameworks work.

- Both frameworks are effective in tackling asynchronism and give acceptable results.

- The late and early fusion framework each has strengths and weaknesses. The strengths of the late fusion framework are: (a) the incorporation of the domain knowledge is more complete and effective; and (b) it is extensible in the sense that it achieves different detection capabilities given external information at different levels of detail. Its weaknesses are: (a) the integration of the audiovisual signals and the externa information is only partial - errors entailed by the external information can hardly be corrected by the audiovisual signals; and (b) the representation of domain knowledge in the system is not automated.

- The strengths of the early fusion framework are: (a) it enables closer integration of the audiovisual signals and the external information so that the audiovisual signals can correct errors in the external information; (b) it has a higher level of automation in representing domain knowledge; and (c) it has the capability to model rich sequential relations between events. Its weaknesses are: (a) the incorporation of the domain knowledge is less complete and less effective, in particular the heterogenous audiovisual patterns are poorly captured; (b) it has a complex structure and a large number of parameters, which may partially account for the poorer accuracy; (c) the fixed feature set it uses is vulnerable to noise, in particular, irrelevant phrases in the textual observations; and (d) it is biased towards event types with larger priors or with longer strings of phrases. For our test games - soccer and American football with detailed descriptions available, all these strengths and weaknesses put together, the late fusion framework performs better than the early fusion framework both in terms of the number of detected events and frame-level accuracy.

The main contributions of this thesis are:

- We proposed integrated analysis of audiovisual signals and external information. We developed two frameworks to perform the integrated analysis. Both frameworks were demonstrated to outperform analysis of single source of information in terms of detection accuracy and the range of event types detected.
- We proposed a domain model common to the team sports, on which both frameworks were based. By instantiating this model with specific domain knowledge, the system can adapt to a new game.
- We investigated the strengths and weaknesses of each framework and suggested that the late fusion framework probably performs better because it represents the domain knowledge more completely and effectively.

The research work can be strengthened or extended in the following areas.

- First, parsing steps and timeout detection need to be more accurate and this would significantly improve the final accuracy of event detection.
- Second, we are interested in applying the proposed system to a wider range of team sports, such as basketball, rugby league, rugby union, hockey, ice hockey and Australian rules football. We expect the system to work on these games with minor adaptation because these games have consistent structure

composed of alternating advances. Many techniques described in the thesis can be reused, namely the hierarchical HMM for phase segmentation, techniques for processing compact and detailed descriptions, algorithms used in the late fusion framework, the DBN and algorithms to determine unigrams or bigrams. Effort will mainly be required in loading domain specifics, particularly in building event models and identifying domain-specific visual effects. To discover audiovisual patterns of event types, maximum entropy and decision tree methods may be helpful in selecting features and in forming threshold-based rules, respectively. They are domain-independent techniques, though not detailed in the thesis. Temporal locations, semantic composition and domain-specific visual effects may have to rely on domain experts to build. Fortunately this effort is in a manageable scale. In addition, domain-specific mid-level semantic entities may be involved in event models, e.g. score board and screen clock. They can be detected using techniques reported in the literature.

- Third, the current system is not real time because the global structure analysis in late fusion framework or the whole of early fusion framework is done offline. It would be desirable to make the system real time or semi-real time so that it can support diverse application needs. Two efforts may be required. The first addresses the modification to the algorithms. An online inference algorithm need to be in place for HHMM or DBN. The second is to reduce the overall computation load. Possibly the system needs to rely more on inexpensive text analysis to save expensive audiovisual computation.

- Fourth, as a generalization of the two sources of information (audiovisual signals and textual descriptions), the system may incorporate even more audiovisual channels to achieve better accuracy. World Cup 2006 was aired with additional cameras on top of each end besides the main camera. Multiple audiovisual channels would help in disambiguating video events.

- Fifth, it may be desirable to support personalized question answering based on the events detected. The questions could be asked on plays, events, tactics or combinations. With rich textual description available, indexing of this information would be realistic. Efforts may be mainly on modeling of personal profiles.

Bibliography

[1] Espn soccernet. http://soccernet.espn.go.com/.

[2] Trecvid nist trec video retrieval evaluation. http://www-nlpir.nist.gov/projects/trecvid/.

[3] Arnon Amir, Marco Berg, Shih-Fu Chang, Giridharan Iyengar, Ching-Yung Lin, Apostol Natsev, Chalapathy Neti, Harriet Nock, Milind Naphade, Winston Hsu, John R. Smith, Belle Tseng, Yi Wu, and Dongqing Zhang. Ibm research trecvid-2003 video retrieval system. In *TRECVID 2003 Workshop*, Gaithersburg, MD, November 2003.

[4] Arnon Amir, Janne O Argillander, Marco Berg, Shih-Fu Chang, Martin Franz, Winston Hsu, Giridharan Iyengar, John R Kender, Lyndon Kennedy, Ching-Yung Lin, Milind Naphade, Apostol (Paul) Natsev, John R. Smith, Jelena Tesic, Gang Wu, Rong Yan, and Donqing Zhang. Ibm research trecvid-2004 video retrieval system. In *NIST TRECVID 2004 Workshop*, Gaithersburg, MD, 2004.

[5] Yasuo Ariki, Masahito Kumano, and Kiyoshi Tsukada. Highlights scene extraction in real time from baseball live video. In *Multimedia Information Retrieval*, pages 209–214, 2003.

[6] Jürgen Assfalg, Marco Bertini, Carlo Colombo, Alberto Del Bimbo, and Walter Nunziati. Semantic annotation of soccer videos: automatic highlights identification. *Comput. Vis. Image Underst.*, 92(2-3):285–305, November-December 2003.

[7] Jürgen Assfalg, Marco Bertini, Alberto Del Bimbo, W. Nunziati, and Pietro Pala. Soccer highlights detection and recognition using hmms. In *Proceedings of IEEE International Conference on Multimedia and Expo*, pages 825–828, 2002.

[8] Noboru Babaguchi. Towards abstracting sports video by highlights. In *IEEE International Conference on Multimedia and Expo (III)*, pages 1519–1522, 2000.

[9] Noboru Babaguchi, Yoshihiko Kawai, and Tadahiro Kitahashi. Generation of personalized abstract of sports video. In *ICME*, 2001.

[10] Noboru Babaguchi, Yoshihiko Kawai, and Tadahiro Kitahashi. Event based indexing of broadcasted sports video by intermodal collaboration. *IEEE Transactions on Multimedia*, 4(1):68 – 75, March 2002.

[11] Noboru Babaguchi, Yoshihiko Kawai, T. Ogura, and Tadahiro Kitahashi. Personalized abstraction of broadcasted american football video by highlight selection. *IEEE Transactions on Multimedia*, 6(4):575 – 586, August 2004.

[12] Noboru Babaguchi, Yoshihiko Kawai, Yukinobu Yasugi, and Tadahiro Kitahashi. Linking live and replay scenes in broadcasted sports video. In *ACM Multimedia Workshops*, pages 205–208, 2000.

[13] Noboru Babaguchi and Naoko Nitta. Intermodal collaboration: a strategy for semantic content analysis for broadcasted sports video. In *ICIP (1)*, pages 13–16, 2003.

[14] Noboru Babaguchi, Shigekazu Sasamori, Tadahiro Kitahashi, and Ramesh Jain. Detecting events from continuous media by intermodal collaboration and knowledge use. In *ICMCS, Vol. 1*, pages 782–786, 1999.

[15] Gabriele Baldi, Carlo Colombo, and Alberto Del Bimbo. A compact and retrieval-oriented video representation using mosaics. In *VISUAL '99: Proceedings of the Third International Conference on Visual Information and Information Systems*, pages 171–178, London, UK, 1999. Springer-Verlag.

[16] Doug Beeferman, Adam Berger, and John Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210, 1999.

[17] Marco Bertini, Alberto Del Bimbo, and Walter Nunziati. Model checking for detection of sport highlights. In *Multimedia Information Retrieval*, pages 215–222, 2003.

[18] Lekha Chaisorn. *A Hierarchical Multi-modal Approach to Story Segmentation in News Video*. Phd thesis, School of Computing, National University of Singapore, 2005.

[19] Shih-Fu Chang, William Chen, Horace J. Meng, Hari Sundaram, and Di Zhong. Videoq: An automated content based video search system using visual cues. In *ACM Multimedia*, pages 313–324, 1997.

[20] Shih-Fu Chang, R. Manmatha, and Tat-Seng Chua. Combining text and audio-visual features in video indexing. In *IEEE ICASSP 2005*, Philadelphia, PA, March 2005.

[21] Tat-Seng Chua, Shih-Fu Chang, Lekha Chaisorn, and Winston Hsu. Story boundary detection in large broadcast news video archives techniques, experience and trends. In *ACM Multimedia*, New York, October 2004.

[22] Tat-Seng Chua and Chun-Xin Chu. Color-based pseudo object model for image retrieval with relevance feedback. In *Proceedings of International Conference on Advanced Multimedia Content Processing*, pages 145–160, 1998.

[23] Tat-Seng Chua, Shi-Yong Neo, Hai-Kiat Goh, Ming Zhao, Yang Xiao, Gang Wang, Sheng Gao, Kai Chen, Qibin Sun, and Tian Qi. Trecvid 2005 by nus pris. In *NIST TRECVID 2005 Workshop*, Gaithersburg, MD, 2005.

[24] Tat-Seng Chua, Shi-Yong Neo, K. Li, Gang Wang, Rui Shi, Ming Zhao, Huaxin Xu, Sheng Gao, and T. L. Nwe. Trecvid2004 search and feature extraction task by nus pris. In *NIST TRECVID 2004 Workshop*, Gaithersburg, MD, 2004.

[25] Hang Cui, Min-Yen Kan, and Tat-Seng Chua. Unsupervised learning of soft patterns for generating definitions from online news. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 90–99, New York, NY, USA, 2004. ACM Press.

[26] Ling-Yu Duan, Min Xu, Tat-Seng Chua, Qi Tian, and Chang-Sheng Xu. A mid-level representation framework for semantic sports video analysis. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 33–44, New York, NY, USA, 2003. ACM Press.

[27] F. Dufaux and F.Moscheni. Motion estimation techniques for digital tv: A review and a new contribution. *Proceedings of the IEEE*, 83(6):877 – 891, June 1995.

[28] Pinar Duygulu, Kobus Barnard, João F. G. de Freitas, and David A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV (4)*, pages 97–112, 2002.

[29] A. Ekin and A.M. Tekalp. Generic play-break event detection for summarization and hierarchical sports video analysis. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume 1, pages 169–172. IEEE, July 2003.

[30] A. Ekin, A.M. Tekalp, and R. Mehrotra. Automatic soccer video analysis and summarization. *IEEE Transactions on Image Processing*, 12(7):796–807, July 2003.

[31] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Compututer and System Sciences*, 55(1):119–139, 1997.

[32] Michel Galley, Kathleen McKeown, Eric Fosler-Lussier, and Hongyan Jing. Discourse segmentation of multi-party conversation. In *ACL*, pages 562–569, 2003.

[33] Yihong Gong, L. T. Sin, Chua Hock Chuan, HongJiang Zhang, and Masao Sakauchi. Automatic parsing of tv soccer programs. In *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, pages 167–174, 1995.

[34] Bilge Günsel and A. Murat Tekalp. Content-based video abstraction. In *ICIP (3)*, pages 128–132, 1998.

[35] Mei Han, Wei Hua, Wei Xu, and Yihong Gong. An integrated baseball digest system using maximum entropy method. In *ACM Multimedia*, pages 347–350, 2002.

[36] A. Hanjalic, M. Ceccarelli, R.L. Lagendijk, and J.Biemond. Automation of systems enabling search on stored video data. In *I.K. Sethi, R.C. Jain (eds.); Vol. 3022. SPIE - The Int. Society for Optical Engineering*, volume 12 of *ISBN 0277-786X*, pages 427–438, San Jose, California, 1996.

[37] Alan Hanjalic and HongJiang Zhang. An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(8):1280–1289, December 1999.

[38] Wei Hao and Jiebo Luo. Generalized multiclass adaboost and its applications to multimedia classification. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 113, Washington, DC, USA, 2006. IEEE Computer Society.

[39] Alexander G. Haupmann and Michael J. Witbrock. Story segmentation and detection of commercials in broadcast news video. In *ADL '98: Proceedings of the Advances in Digital Libraries Conference*, page 168, Washington, DC, USA, 1998. IEEE Computer Society.

[40] Alex Hauptmann, M.-Y. Chen, Mike Christel, C. Huang, W.-H. Lin, T. Ng, Norman Papernick, A. Velivelli, Jie Yang, Rong Yan, Hui Yang, and Howard

Wactlar. Confounded expectations: Informedia at trecvid 2004. In *NIST TRECVID 2004 Workshop*, Gaithersburg, MD, 2004.

[41] Alex Hauptmann, Dorbin Ng, Robert Baron, M-Y. Chen, Mike Christel, Pinar Duygulu, C. Huang, W-H. Lin, Howard Wactlar, N. Moraveji, Norman Papernick, C.G.M. Snoek, G. Tzanetakis, Jie Yang, R. Yan, and R. Jin. Informedia at trecvid 2003: Analyzing and searching broadcast news video. In *Proceedings of (VIDEO) TREC 2003 (Twelfth Text Retrieval Conference)*, November 2003.

[42] Marti Hearst. Multi-paragraph segmentation of expository text. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 9–16, New Mexico State University, Las Cruces, New Mexico, 1994.

[43] Winston Hsu and Shih-Fu Chang. A statistical framework for fusing mid-level perceptual features in news story segmentation. In *IEEE International Conference on Multimedia and Expo (ICME)*, Baltimore, MD, July 2003.

[44] Winston Hsu and Shih-Fu Chang. Generative, discriminative, and ensemble learning on multi-modal perceptual fusion toward news video story segmentation. In *IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, June 2004.

[45] Winston Hsu, Shih-Fu Chang, Chih-Wei Huang, Lyndon Kennedy, Ching-Yung Lin, and Giridharan Iyengar. Discovery and fusion of salient multimodal features towards news story segmentation. In *IS&T/SPIE Symposium on Electronic Imaging: Science and Technology - SPIE Storage and Retrieval of Image/Video Database*, San Jose, CA, January 2004.

[46] Ichiro Ide, Norio Katayama, and Shin'ichi Satoh. Visualizing the structure of a large scale news video corpus based on topic segmentation and tracking. In *International Workshop on Multimedia Information Retrieval (MIR2002)*, 2002.

[47] S.S. Intille and A.F. Bobick. Recognizing planned, multi-person action. *Comput. Vis. Image Underst.*, 81(3):414–445, March 2001.

[48] G. Iyengar, P. Duygulu, S. Feng, P. Ircing, S. P. Khudanpur, D. Klakow, M. R. Krause, R. Manmatha, H. J. Nock, D. Petkova, B. Pytlik, and P. Virga. Joint visual-text modeling for automatic retrieval of multimedia documents. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 21–30, New York, NY, USA, 2005. ACM Press.

[49] Jiwoon Jeon, Victor Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR*, pages 119–126, 2003.

[50] Haitao Jiang, Abdelsalam Helal, Ahmed K. Elmagarmid, and Anupam Joshi. Scene change detection techniques for video database systems. *Multimedia Syst.*, 6(3):186–195, 1998.

[51] Ewa Kijak, Lionel Oisel, and Patrick Gros. Hierarchical structure analysis of sport videos using hmms. In *ICIP (2)*, pages 1025–1028, 2003.

[52] Chun-Keat Koh and Tat-Seng Chua. Detection and segmentation of commercials in news video. Technical report, The School of Computing, National University of Singapore, 2000.

[53] Michael Lee, Surya Nepal, and Uma Srinivasan. Edge-based semantic classification of sports video sequences. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume 1, pages 57–60. IEEE, July 2003.

[54] Riccardo Leonardi, Pierangelo Migliorati, and Maria Prandini. Semantic indexing of sports program sequences by audio-visual analysis. In *ICIP (1)*, pages 9–12, 2003.

[55] Baoxin Li and Ibrahim Sezan. Semantic sports video analysis: approaches and new applications. In *ICIP (1)*, pages 17–20, 2003.

[56] Baoxin Li and M. Ibrahim Sezan. Event detection and summarization in sports video. In *CBAIVL '01: Proceedings of the IEEE Workshop on*

Content-based Access of Image and Video Libraries (CBAIVL'01), page 132, Washington, DC, USA, 2001. IEEE Computer Society.

[57] Yang Li. Multi-resolution analysis on text segmentation, 2001.

[58] Yi Lin, Mohan S. Kankanhalli, and Tat-Seng Chua. Temporal multi-resolution analysis for video segmentation. In *Proc. of Int'l Conference on Storage and Retrieval for Media Databases (SPIE)*, pages 494–505, 2000.

[59] Benoit Maison, Chalapathy Neti, and Andrew Senior. Audio-visual speaker recognition for video broadcast news: some fusion techniques. In *Multimedia Signal Processing, 1999 IEEE 3rd Workshop on*, pages 161 – 167, 1999.

[60] C. Meesookho, S. Narayanan, and C. Raghavendra. Collaborative classification applications in sensor networks, 2002.

[61] Andrew Merlino, Daryl Morey, and Mark Maybury. Broadcast news navigation using story segmentation. In *MULTIMEDIA '97: Proceedings of the fifth ACM international conference on Multimedia*, pages 381–391, New York, NY, USA, 1997. ACM Press.

[62] Shingo Miyauchi, Akira Hirano, Noboru Babaguchi, and Tadahiro Kitahashi. Collaborative multimedia analysis for detecting semantical events from broadcasted sports video. In *ICPR (2)*, pages 1009–1012, 2002.

[63] Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words, 1999.

[64] Kevin Patrick Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning.* Phd, UNIVERSITY OF CALIFORNIA, BERKELEY, Fall 2002.

[65] Frank Nack and Alan P. Parkes. Toward the automated editing of theme oriented video sequences. *Applied Artificial Intelligence*, 11(4):331–366, 1997.

[66] Surya Nepal, Uma Srinivasan, and Graham Reynolds. Automatic detection of 'goal' segments in basketball videos. In *MULTIMEDIA '01: Proceedings*

of the ninth ACM international conference on Multimedia, pages 261–269, New York, NY, USA, 2001. ACM Press.

[67] Naoko Nitta, Noboru Babaguchi, and Tadahiro Kitahashi. Story based representation for broadcasted sports video and automatic story segmentation. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, pages 813–816, 2002.

[68] Naoko Nitta, Noboru Babaguchi, and Tadahiro Kitahashi. Generating semantic descriptions of broadcasted sports videos based on structures of sports games and tv programs. *Multimedia Tools and Applications*, 25(1):59–83, 2005.

[69] H. Pan, P. Van Beek, and M. Sezan. Detection of slow-motion replay segments in sports video for highlights generation. In *IEEE International Conference on Acoustic, Speech and Signal Processing*, 2001. citeseer.ist.psu.edu/pan01detection.html.

[70] Hao Pan, Baoxin Li, and M. Ibrahim Sezan. Automatic detection of replay segments in broadcast sports programs by detection of logos in scene transitions. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 3385–3388, May 2002.

[71] Rebecca J. Passonneau and Diane J. Litman. Discourse segmentation by human and automated means. *Comput. Linguist.*, 23(1):103–139, 1997.

[72] Dinh Q. Phung, S. Venkatesh, and C. Dorai. On extraction of thematic and dramatic functions in educational films. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume 3, pages 449–452. IEEE, July 2003.

[73] Dinh Q. Phung, Svetha Venkatesh, and Chitra Dorai. High level segmentation of instructional videos based on content density. In *ACM Multimedia*, pages 295–298, 2002.

[74] Dinh Q. Phung, Svetha Venkatesh, and Chitra Dorai. Hierarchical topical segmentation in instructional films based on cinematic expressive functions. In *ACM Multimedia*, pages 287–290, 2003.

[75] Yong Rui, Anoop Gupta, and Alex Acero. Automatically extracting highlights for tv baseball programs. In *ACM Multimedia*, pages 105–115, 2000.

[76] David A. Sadlier, Sean Marlow, Noel Oconnor, and Noel Murphy. MPEG audio bitstream processing towards the automatic generation of sports programme summaries. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, August 21 2002.

[77] David A. Sadlier and Noel E. O'Connor. Event detection in field sports video using audio-visual features and a support vector machine. *IEEE Trans. Circuits Syst. Video Techn.*, 15(10):1225–1233, 2005.

[78] Frederick Shook. Sports photography and reporting. In *Television field production and reporting*, chapter 12. Longman Publisher USA, 2nd edition, 1995.

[79] Cees Snoek and Marcel Worring. Multimedia event-based video indexing using time intervals. *IEEE Transactions on Multimedia*, 7(4):638–647, 2005.

[80] Cees Snoek, Marcel Worring, and Arnold W. M. Smeulders. Early versus late fusion in semantic video analysis. In *ACM Multimedia*, pages 399–402, 2005.

[81] Cees G.M. Snoek and Marcel Worring. Time interval maximum entropy based event indexing in soccer video. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume 3, pages 481–484. IEEE, July 2003.

[82] C.G.M. Snoek, M. Worring, J.M. Geusebroek, D.C. Koelma, and F.J. Seinstra. The mediamill trecvid 2004 semantic viedo search engine. In *NIST TRECVID 2004 Workshop*, Gaithersburg, MD, 2004.

[83] G. Sudhir, John Chung-Mong Lee, and Anil K. Jain. Automatic classification of tennis video for high-level content-based retrieval. http://hdl.handle.net/1783.1/89, August 1997.

[84] G. Sudhir, John Chung-Mong Lee, and Anil K. Jain. Automatic classification of tennis video for high-level content-based retrieval. In *IEEE Workshop on Content-based Access of Image and Video Database*, pages 81–90, 1998.

[85] Tanveer Fathima Syeda-Mahmood and Savitha Srinivasan. Detecting topical events in digital video. In *ACM Multimedia*, pages 85–94, 2000.

[86] Yap-Peng Tan, Drew D. Saur, Sanjeev R. Kulkarni, and Peter J. Ramadge. Rapid estimation of camera motion from compressed video with application to video annotation. *IEEE Trans. Circuits Syst. Video Techn.*, 10(1):133–146, 2000.

[87] Kinh Tieu and Paul A. Viola. Boosting image retrieval. In *CVPR*, pages 1228–1235, 2000.

[88] Dian Tjondronegoro, Yi-Ping Phoebe Chen, and Binh Pham. Sports video summarization using highlights and play-breaks. In *Multimedia Information Retrieval*, pages 201–208, 2003.

[89] Kongwah Wan, Xin Yan, Xinguo Yu, and Changsheng Xu. Real-time goal-mouth detection in mpeg soccer video. In *ACM Multimedia*, pages 311–314, 2003.

[90] Jihua Wang and Tat-Seng Chua. A cinematic-based framework for scene boundary detection in video. *The Visual Computer*, 19(5):329–341, 2003.

[91] Jinjun Wang, Changsheng Xu, Engsiong Chng, Kongwah Wah, and Qi Tian. Automatic replay generation for soccer video broadcasting. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 32–39, New York, NY, USA, 2004. ACM Press.

[92] Jinjun Wang, Changsheng Xu, Chng Eng Siong, Ling-Yu Duan, Kongwah Wan, and Qi Tian. Automatic generation of personalized music sports video. In *ACM Multimedia*, pages 735–744, 2005.

[93] Yi Wu, Edward Y. Chang, Kevin Chen-Chuan Chang, and John R. Smith. Optimal multimodal fusion for multimedia data analysis. In *ACM Multimedia*, pages 572–579, 2004.

[94] Yi Wu, Edward Y. Chang, and Belle L. Tseng. Multimodal metadata fusion using causal strength. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 872–881, New York, NY, USA, 2005. ACM Press.

[95] Yi Wu, Ching-Yung Lin, Edward Y. Chang, and John R. Smith. Multimodal information fusion for video concept detection. In *ICIP*, pages 2391–2394, 2004.

[96] Jing Xiao, Tat-Seng Chua, and Jimin Liu. Global rule induction for information extraction. *International Journal on Artificial Intelligence Tools*, 13(4):813–828, 2004.

[97] Lexing Xie, Shih-Fu Chang, A. Divakaran, and Huifang Sun. Unsupervised discovery of multilevel statistical video structures using hierarchical hidden markov models. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume 3, pages 29–32. IEEE, July 2003.

[98] Lexing Xie, Shih-Fu Chang, Ajay Divakaran, and Huifang Sun. Structure analysis of soccer video with hidden markov models. In *IEEE Interational Conference on Acoustic, Speech and Signal Processing (ICASSP-2002)*, Orlando, FL, May 2002.

[99] Lexing Xie, Shih-Fu Chang, Ajay Divakaran, and Huifang Sun. Feature selection for unsupervised discovery of statistical temporal structures in video. In *ICIP (1)*, pages 29–32, 2003.

[100] Lexing Xie, Lyndon Kennedy, Shih-Fu Chang, Ajay Divakaran, Huifang Sun, and Ching-Yung Lin. Layered dynamic mixture model for pattern discovery in asynchronous multi-modal streams. In *Interational Conference on Acoustic, Speech and Signal Processing (ICASSP)*, Philadelphia, PA, March 2005.

[101] Lexing Xie, Peng Xu, Shih-Fu Chang, Ajay Divakaran, and Huifang Sun. Structure analysis of soccer video with domain knowledge and hidden markov models. *Pattern Recognition Letters*, 25(7):767–775, 2004.

[102] Gu Xu, Yu-Fei Ma, HongJiang Zhang, and Shiqiang Yang. A hmm based semantic analysis framework for sports game event detection. In *ICIP (1)*, pages 25–28, 2003.

[103] Min Xu. Content-based sports video analysis using multiple modalities, 2003.

[104] Peng Xu, Lexing Xie, Shih-Fu Chang, Ajay Divakaran, Anthony Vetro, and Huifang Sun. Algorithms and system for segmentation and structure analysis in soccer video. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, 2001.

[105] Rong Yan, Jun Yang, and Alexander G. Hauptmann. Learning query-class dependent weights in automatic video retrieval. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 548–555, New York, NY, USA, 2004. ACM Press.

[106] Hui Yang and Tat-Seng Chua. Fada: find all distinct answers. In *WWW (Alternate Track Papers & Posters)*, pages 304–305, 2004.

[107] J. Yang, M. Y. Chen, and A. Hauptmann. Finding person X: Correlating names with visual appearances. In *International Conference on Image and Video Retrieval*, pages 270–278, 2004.

[108] Minerva M. Yeung, Boon-Lock Yeo, and Bede Liu. Extracting story units from long programs for video browsing and navigation. In *ICMCS*, pages 296–305, 1996.

[109] Xinguo Yu, Qi Tian, and Kongwah Wan. A novel ball detection framework for real soccer video. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume II, pages 265–268. IEEE, July 2003.

[110] Xinguo Yu, Changsheng Xu, Hon-Wai Leong, Qi Tian, Qing Tang, and Kongwah Wan. Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video. In *ACM Multimedia*, pages 11–20, 2003.

[111] Xinguo Yu, Changsheng Xu, Qi Tian, and Hon-Wai Leong. A ball tracking framework for broadcast soccer video. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, volume II, pages 273–276. IEEE, July 2003.

[112] DongQing Zhang and Shih-Fu Chang. Event detection in baseball video using superimposed caption recognition. In *ACM Multimedia*, pages 315–318, 2002.

[113] Dongqing Zhang, Rajendran Kumar Rajendran, and Shih-Fu Chang. General and domain-specific techniques for detecting and recognizing superimposed text in video. In *IEEE International Conference on Image Processing (ICIP)*, Rochester, New York, September 2002.

[114] Hongjiang Zhang, Chien Yong Low, and Stephen W. Smoliar. Video parsing and browsing using compressed data. *Multimedia Tools and Applications*, 1(1):89–111, March 1995. DOI 10.1007/BF01261227.

[115] Yi Zhang and Tat-Seng Chua. Detection of text captions in compressed domain video. In *ACM Multimedia Workshops*, pages 201–204, 2000.

[116] Di Zhong and Shih-Fu Chang. Structure analysis of sports video using domain models. In *Proceedings of Internatiaonl Conference on Multimedia and Expo*, 2001.

[117] Wensheng Zhou, Asha Vellaikal, and C. C. Jay Kuo. Rule-based video classification system for basketball video indexing. In *ACM Multimedia Workshops*, pages 213–216, New York, NY, USA, 2000. ACM Press.

Appendix

Unigram lexicon of soccer

kickoff, *goal-kick*, *free-kick*, *corner-kick*, *penalty*, *score*, *shot*, *save*, *offside*, *assist*, *pass*, *block*, *clear*, *miss*, *catch*, *parry*, *tip-over*, *throw-in*, *open-play*, *attack*, *defend*, *foul*, *yellow-card*, *red-card*, *substitution*, *out-of-play*

Bigram lexicon of soccer

shot - score, *shot - save*, *shot - clear*, *shot - miss*, *assist - shot*, *corner-kick - shot*, *foul - free-kick*, *assist - offside*, *assist - clear*, *corner-kick - clear*

Unigram lexicon of American football

pass, *no-gain*, *punt*, *catch*, *penalty*, *incomplete*, *field-goal*, *tackle*, *touchback*, *intercept*, *kick*, *recover*, *touchdown*, *conversion*, *fumble*, *safety*

Bigram lexicon of American football

pass - incomplete, *pass - intercept*, *punt - catch*, *kick - touchback*, *fumble - recover*

Publications

1. Huaxin Xu and Tat-Seng Chua, *Fusion of AV Features and External Information Sources for Event Detection in Team Sports Video*, ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 2, Issue 1, February 2006.

2. Huaxin Xu, Tat-Hoe Fong and Tat-Seng Chua, *Fusion of Multiple Asynchronous Information Sources for Event Detection in Soccer Video*, IEEE International Conference on Multimedia & Expo, July 6-8, 2005, Amsterdam, The Netherlands.

3. Huaxin Xu and Tat-Seng Chua, *Detecting Events in Teams Sports Video*, the 8th International Workshop on Advanced Image Technology, Jeju Island, Korea, January 2005.

4. Huaxin Xu and Tat-Seng Chua, *The Fusion of Audio-Visual Features and External Knowledge for Event Detection in Team Sports Video*, the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval, New York, October 2004.

5. Young-Tae Kim, Huaxin Xu and Tat-Seng Chua, *Video Retrieval using Visual Sequence Matching*, Asia Information Retrieval Symposium 2004, Beijing, China, October 2004.