修 士 論 文 の 和 文 要 旨

研究科・専攻	大学院　情報システム学研究科　情報ネットワークシステム学専攻 博士前期課程		
氏　　名	蒋飛虎	学籍番号	1252021
論 文 題 目	ベイジアンネットワークを用いたMPEG-2高画質サッカー映像のセマンティック解析		

要　旨

近年，インターネットのブロードバンド化に伴い，映像配信が普及し，また，地上デジタル放送や，BS・CSデジタル放送などの衛星放送により，ユーザが試聴できる番組の数が急増してきている．パソコンやレコーダのハードディスクの容量も増え，大量の番組（コンテンツ）を保存することが可能となったが，その反面，膨大な映像データの中から，視聴者の求めるシーンを素早く検索する技術の必要性がこれまでにも増して高まって来ている．本研究はサッカー映像のリプレーシーンとゴール付近のハイライトシーンの検出方法を提案する．シーンの検出には，MPEG-2エンコーダによって圧縮されたハイビジョンサッカー映像から抽出した特徴量とハイライトシーンとの間の因果関係をベイジアンネットワークで記述する手法を用いる．

ベイジアンネットワークを用いることにより，抽出された特徴量からハイライトシーンの発生を確率的に推論することが可能になる．

すでにベイジアンネットワークを用いたサッカー映像のハイライトシーンの検出法は提案されているが，それらの方法では，フレーム毎に画素単位でさまざまな画像処理を映像に施すことによって求めた特徴量を利用している．そのため，画面が大きくなると計算コストも大きくなるので，リアルタイム処理には専用の処理装置が必要になる．本研究で提案する方法はMPEG-2圧縮データに含まれている符号化パラメータから特徴量を計算するので，従来法に比べて計算量が少なく，ハイビジョンなどの高解像度映像であっても，通常のPCを用いてリアルタイム処理が可能である．
また，従来法では各種シーンに対してベイジアンネットワークが提案されているが，いずれも，ネットワークモデル中のシーンに関わるイベントがすべてフレーム単位で定義されている．例えば，従来法のゴールシーンに関わる，ゴールゲートの出現，観客の声，リプレーの発生等のイベントは全てフレーム単位で数えている．しかし，各イベントの開始・終了フレームを明確に判定する手法が明らかにされておらず，場合によっては人の手で行わなう必要がある．そのため，ベイジアンネットワークを学習する時に、各種イベントの時間帯の与え方に誤差が含まれる可能性がある．さらに、テストビデオから，シーン検出する時，シーンの始終時間帯の検出も困難である．

本研究の提案手法では，まず，MPEG-2圧縮データから直接抽出した符号化パラメータの特徴的な変化から，カメラの切り換えに伴う画面の切り替るカット点を検出し，隣接する二つのカット点間をショットとして定義する．さらに各ショットの特徴量を調べることにより，ショットをいくつかのイベントクラスに分類する．さらに，シーンをある特徴的なイベントの発生として捉えることにより，シーンの検出を行う．本手法では，各イベントの開始・終了時刻をショットのカット点によって明確に与えることができることができ，しかもMPEG-2圧縮データから自動的に求めることが可能である．

提案方式の性能評価のために，実際のビデオデータを使用した検出実験を行ったところ，ゴール付近で起こるイベントシーンの再現率が86.17%，適合率90.76%，またリプレーシーンの再現率が81.00%，適合率92.57%という検出結果が得られた．一方，従来法の検出結果では，同一のビデオデータではないが，ゴール付近で起こるイベントシーンの再現率71.1%，適合率89.8%であり，提案方式のほうが従来法に比べ，再現率，適合率ともに上回り，とくに再現率の向上が顕著である．以上のことより，提案法の有効性が確認された．

平成２５年度

Master Thesis

Semantic Analysis of High-definition MPEG-2 Soccer Video Using Bayesian Network

Department of Information Network Systems
Graduate School of Information Systems

Student ID: 1252021
Jiang Feihu

Chair : Professor Hiroyoshi MORITA
Member : Assoc. Professor Hiroyuki KASAI
Member : Assoc. Professor Hidetsugu IRIE

2014.1.27

Contents

Chapter 1

Introduction

1.1 Introduction

In recent years, with the development of multi-media technology and computer network technology, the technologies of digital video storage and transmission have made significant developments. Abundant videos make people more and more frequently use them. To facilitate the users to quickly find the interesting video clips, we need efficient methods to manage database of video. In recent years, driven by the requirement, the techniques of content-based video analysis and retrieval have got remarkable development, and become a hotspot in field of information. A flexible and scalable way to manage the sports video is demanded; for instance, automatic and real time sports video summarization. Obviously, the main gap between low-level media features and high-level concepts needs to be bridged.

Soccer games have a wide raging audience, therefore, the analysis and retrieval of soccer video are important branches of content-based video analysis and retrieval. In this thesis, we present new algorithms for soccer video structure analysis. By our structure, we are primarily concerned with a temporal sequence of the high-level concepts, namely two kinds of events: replay scene and goal area scene. In the middle-level layer, six kinds of meaningful content shots are classified. In the low-level layer, some effective features are arranged. Given a video in a specific domain, we aim to extract the low-level features and interpret the input video in terms of high-level concepts. Our final goal is to extract and present the meaningful information for viewers.

1.2 Organization of the Thesis

The organization of this thesis is as the following.

Chapter 2 describes the structure of soccer video and video coding concept. The structure layer of soccer video is described, including the scene layer, the shot layer and the frame layer. The video coding concepts including (1) Prediction from the Previous Video Frame (2) Block-based motion estimation and compensation; and (3) The special transmission sequence in MPEG, called GOP.

Chapter 3 reviews the reference work of highlight scene detection by using Bayesian network for highlight scene analysis in soccer video. And the differences between our system with the reference work will be compared.

Chapter 4 introduces some basic concepts of Bayesian network.

Chapter 5 reviews the existing shot change detection method by using MPEG-2 codec information directly, and describes some main features extraction for shot classification and BN building.

Chapter 6 presents the proposed system by using Bayesian Network (BN) to detect the highlight scene of high-definition MPEG-2 soccer video. It describes two components of our system; the training stage and testing stage.

Chapter 7 describes the proposed system for highlight scene analysis. The implementation results are also presented. The pre-experiment of shot change detection and experiment of scene detection will be described.

Chapter 8 summarizes the results in this thesis and discusses directions for future works.

Chapter 2

Layer Structure of Soccer Video and Video Coding Concept

2.1 Layer structure of soccer video

Usually, video programs need some kinds of ways to be provided for the users, such as: a television broadcast wave, CATV and other networks. These video programs are made by some editing process to join the plurality of images together taken by the camera. Therefore, the video program is consist of a series of the structured image contents.

In soccer video, we can define three kinds of layer structures: Scene Layer, Shot Layer and Frame Layer. The layer structure of soccer video is shown in Fig 2.1. A scene a period which the video holds the same semantic meanings, such as goal scene. A shot depicts a period that the camera is shooting in a consecutive interval time. A cut point describes a point at which the image content is completely changed from one shot to another shot.

A video captured by the camera is made up from a plurality of images. Each image is called frame or picture. The standard frame rate for motion pictures is 24 frames per second and, for television, 29.97 frames per second.

Usually, the video program is composed of a plurality of scenes, each scene is composed of a plurality of shots, and each shot is taking a hierarchical structure including some frames.

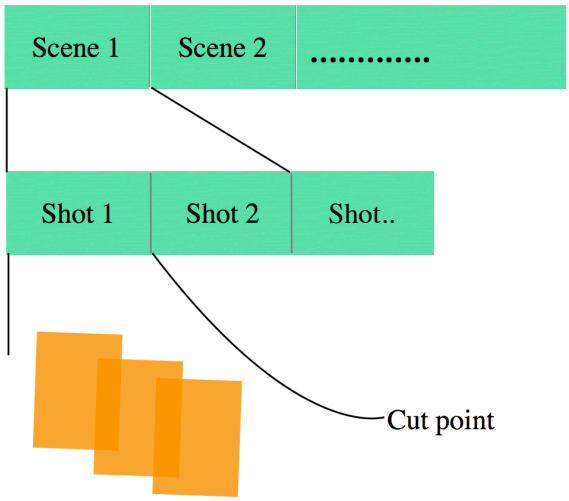


Figure 2.1: Construction layer of soccer game

2.2 MPEG Parameter

Soccer video program is broadcasted either in terrestrial digital signal or in BS digital signal, encoded by the MPEG-2 coding scheme in advance.

Our research is based on MPEG-2 soccer video analysis. For building the Bayesian Network, the shot change detection, shot classification and some features extraction are using various MPEG parameters. Here we will explain some parameters of MPEG-2 briefly.

2.2.1 Prediction from the Previous Video Frame

The bidirectional temporal prediction is the key feature of MPEG-2 video. Pictures coded with Bidirectional prediction uses two reference pictures; one in the past and one in the future. A Target Macroblock in bidirectionally coded pictures can be predicted by a prediction macroblock from the past reference picture (Forward Prediction), or one from the future Reference picture (Backward Prediction), or by an average of two prediction macroblocks, or one from each reference picture (Interpolation). The motion-compensated interpolation for a macroblock in a bidirectionally predicted picture is illustrated in Fig 2.2.

Pictures coded by means of bidirectional prediction are called B-pictures. Pictures that are bidirectionally predicted are never themselves used as reference pictures, that is, reference pictures for B-pictures must be either P-pictures or I-

pictures. Similarly, reference pictures for P-pictures must also be either P-pictures or I-pictures. In a video stream, three kinds of coding are defined in MPEG-2 encoder for temporal pictures prediction as follows:

Forward predictive coding: Taking the past pictures as reference

Backward predictive coding: Taking the future pictures as reference

Bi-direction predictive coding: Taking the past and future pictures as reference

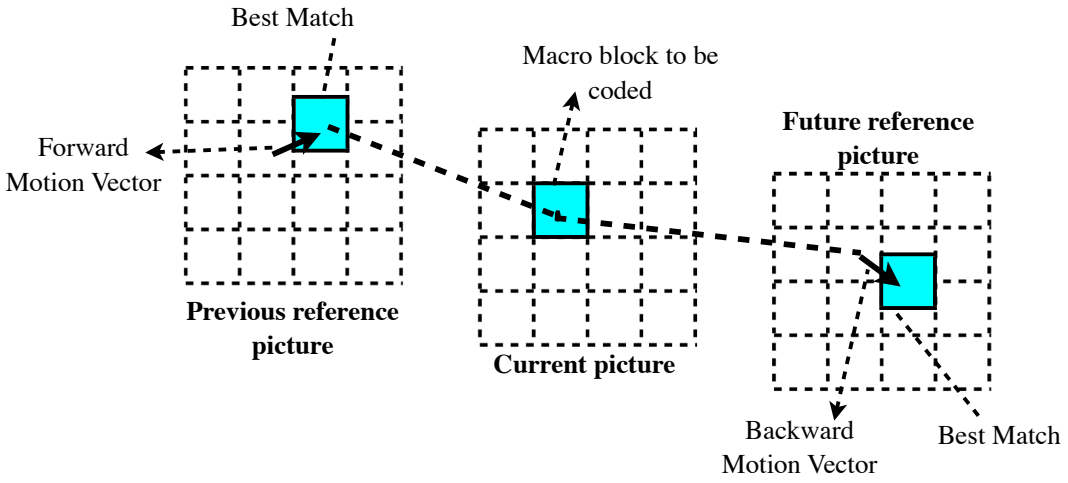


Figure 2.2: Motion-Compensated Interpolation using Bidirectional prediction

2.2.2 Block-based Motion Estimation and Compensation

A practical and widely-used method of motion compensation is to compensate for movement of rectangular sections or 'blocks' of the current frame. The following procedure is carried out for each block of $M \times N$ samples in the current frame:

1. Search an area in the reference frame (past of future frame) to find a 'matching' $M \times N$-samples region. This is carried out by comparing the $M \times N$ block in the current frame with some of all of the possible $M \times N$ regions in the search area (usually a region in the current block position) and finding the region that gives the 'best' match. This process of finding the best match is known as motion estimation (see Fig 2.2).

2. The chosen candidate region becomes the predictor for the current $M \times N$ block and is subtracted from the current block to form a residual $M \times N$ block (motion compensation).

3. The residual block is encoded and transmitted and the offset between the current block and the position of the candidate region (motion vector) is also transmitted.

Motion Estimation: Motion estimation of a macroblock involves finding a region in a reference frame that closely matches the current macroblock. The reference frame is a previously encoded frame from the sequence and may be before or after the current frame in display order.

Motion Compensation: The selected 'best' matching region in the reference frame is subtracted from the current macroblock to produce a residual macroblock (luminance and chrominance) that is encoded and transmitted together with a motion vector describing the position of the best matching region (relative to the current macroblock position).

2.2.3 Group of Pictures (GOP)

In MPEG, a sequence of transmitted video pictures is typically divided into a series of GOPs, where each GOP begins with an Intra-coded picture (I-picture) followed by an arrangement of Forward Predictive-coded pictures (P-pictures) and Bidirectionally Predicted pictures (B-pictures). We apply standard GOP structure to this research. One GOP consists of I frame, four P frames, and ten B frames. The order of these frames is : I B B P B B P B B P B B P B B I. Fig 2.3. shows an example of MPEG GOPs.

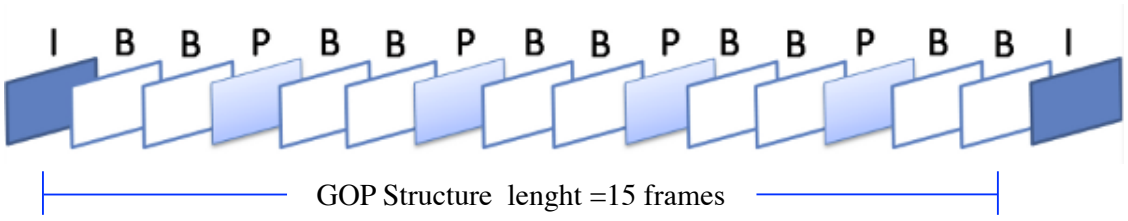


Figure 2.3: GOP Structure

Chapter 3

Related Work and Proposed Semantic Analysis System

Sports videos have been broadcasted to large audiences for their daily life entertainment. A flexible and scalable way to manage the sports video is demanded; for instance, automatic and real time sports video summarization. Obviously, the main gap between low-level media features and high-level concepts needs to be bridged.

There exist a number of related works in this research area. Related works mainly lie in sports video analysis including soccer and various other games, and general video segmentation. For soccer videos, prior works have focused on shot classification [5], video reconstruction [7] and rule-based semantic classification [9]. Some other methods are based on Bayesian networks (BNs).Work [10] have been also applied to semantic analysis. In [10] Sun et al. used BNs for scoring event detection in soccer videos based on six different low-level features including gate, face, audio, texture, caption and text. Shih et al. [11] developed so-called multi-level semantic network (MSN) to interpret the highlights in baseball game video. Another highlight detection method [12] exploits visual cues estimated from video streams, the currently framed play-field zone, player's positions, and the colors of players'uniforms. Low-level features are used for semantic analysis to identify highlights , i.e., object, color and texture features are employed to represent highlights. Xuet al. [13] proposed an effective algorithm for soccer videos, which detects the plays and breaks in soccer games by motion and color features.

Unfortunately, previous scene analysis system [4] cannot meet the need of real-time processing for HD video since most of their feature extraction approaches are based on complex image processing. In this research, we present a new algorithm

for soccer video structure analysis. We are primarily concerned with a temporal sequence of the high-level concepts, namely two kinds of events: replay scene and goal area scene. In the middle-level layer, seven kinds of meaningful content shots are classified. In the low-level layer, some effective features are arranged. Given a video in specific domain, we aim to extract the low-level features from the input video and interpret them in terms of high-level concepts. Our final goal is to provide the meaningful information to viewers. The main differences between our system with the existing system are shown in Fig 3.1. Some key points are as follows:

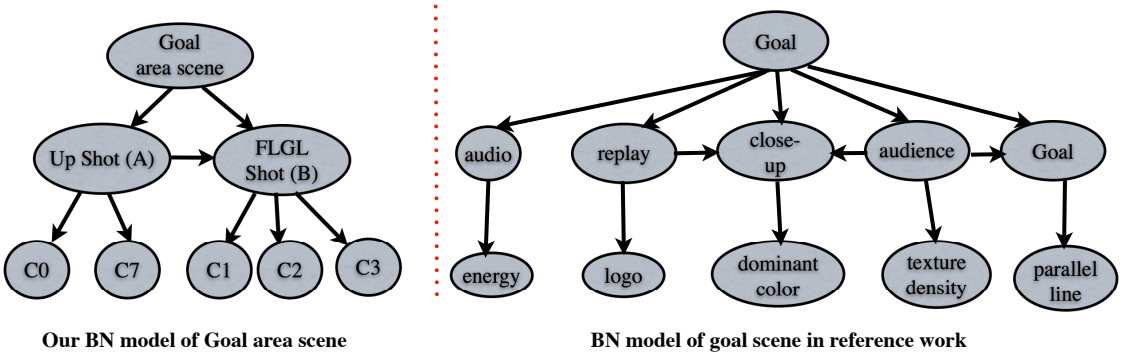


Figure 3.1: A comparison of our BNs and BNs from reference work [4]

- Hidden-node component (Shot vs. Frame)
  The hidden-node components in the existing system [4] is based on the results of frame unit analysis. Considering that it is hard to define the precise boundary in each kind of highlight scene, the shot based scene analysis is utilized in our system. Besides, the shot change detection in our system is carried out by using the parameters embedded in MPEG2 directly, and the high speed shot change detection contributes to the high speed scene analysis with BN.

- Evidence-node component (MPEG domain vs. Pixel domain)
  The evidence-node component in the reference work [4] are some image features. These features are almost based on pixel domain image processing. Due to the complex image process of features extraction, their systems are limited to low-definition video games as a direct consequence of slow computing speed. The evidences extraction in our work is using the parameters embedded in MPEG-2 codec directly which reduce the computing time greatly, and can be used in hi-definition video analysis.

- Method for building links between hidden nodes and evidence nodes.
  Since each kind of the scene types is related with a certain kind of shot patterns, the edgs between hidden nodes and evidence nodes in our BN model assigned by ad-hoc.

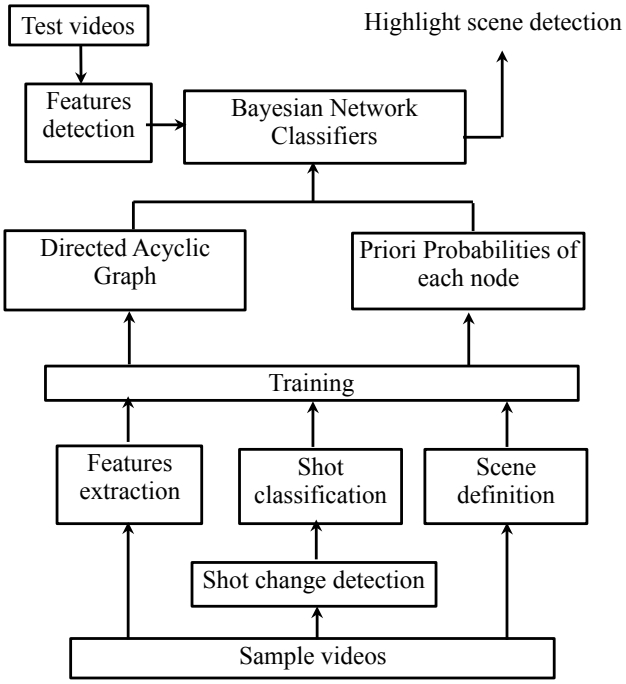


Figure 3.2: System Flowchart

As shown in Fig 3.2, our system consists of two components; the training stage and the testing stage. In the training stage, to aim at video structure analysis, the shot change detection was carried out, and then seven kinds of shots are classified by extracting some features embedded in MPEG video codec. At the same time, three kinds of scenes are defined: replay scene and goal area scene. Our method exposes hierarchical structure of soccer videos, and soccer video was abstracted into three levels, from high-level to low-level: the Scene layer, the Shot layer and the Evidence layer. Main gaps between low-level features and high-level concepts are bridged by BNs.

Based on the posteriori probabilities of all the concepts, given that some evidences detected out, a particular highlight scene can be estimated. Given a video in MPEG-2 domain, we aim to extract the low-level features and interpret the input video in terms of high-level concept. This approach is different from existing

works, most of the works focus on the feature detection in pixel domain by purely image processing method and result in inaccurate boundary of detected scenes. The advantages of applying MPEG-2 codec data to event detection are as follows;

(1) The amount of computing time for extracting and managing necessary information from MPEG codec directly is significantly less than that for extracting features from each frame, and thus, our scheme makes it possible to analyze high-definition (1440×1080 resolution) soccer video in real-time.

(2) A certain shot classification is carried out after accurate shot change detection, which is powerful for the semantic analysis of highlight scene.

Chapter 4

Bayesian Network

4.1 Introduction

Bayesian networks are powerful tools for modeling causes and effects in a wide variety of domains. They are compact networks of probabilities that capture the probabilistic relationship between variables, as well as historical information about their relationships.

Bayesian networks are very effective for modeling situations where some information is already known and incoming data is uncertain or partially unavailable. These networks also offer consistent semantics for representing causes and effects (and likelihoods) via an intuitive graphical representation. Because of all of these capabilities, Bayesian networks are being increasingly used in a wide variety of domains where automated reasoning is needed.

In simpler terms, a Bayesian network is a model. It can be a model of anything: the weather, a disease and its symptoms, a military battalion, even a garbage disposal. Bayesian networks are especially useful when the information about the past and/or the current situation is vague, incomplete, conflicting, and uncertain.

Because Bayesian networks offer consistent semantics for representing uncertainty and an intuitive graphical representation of the interactions between various causes and effects, they are a very effective method of modeling uncertain situations that depend on causes and effects.

4.2 Definition

A Bayesian network (BN) is a graphical structure that allows us to represent and reason about an uncertain domain. The nodes in a Bayesian network represent a set of random variables, X = X_1,..X_i,...X_n, each of which takes a value in the domain with a probability. Directed arcs in the BN connect pairs of nodes, $X_i \rightarrow X_j$, representing the direct dependencies between variables. Assuming that random variables are discrete, the strength of the relationship between variables is quantified by conditional probability distributions associated with each node. The only constraint for BNs is that there must not be any directed cycles. Such networks are called directed acyclic graphs.

For all the following, let $G = (V, E)$ be a directed acyclic graph (or DAG), and let $X = (X_v)_{v \in V}$ be a set of random variables indexed by V.

X is a Bayesian network with respect to G if its joint probability density function (with respect to a product measure) can be written as a product of the individual density functions, conditional on their parent variables, where pa(v) is the set of parents of v.

$$p(x) = \prod_{\nu \in V} p(x_\nu | x_{pa(v)})$$

For any set of random variables, the probability of any member of a joint distribution can be calculated from conditional probabilities using the chain rule as:

$$P(X_1 = x_1, ..., X_n = x_n) = \prod_{\nu=1}^{n} P(X_\nu = x_\nu | X_{\nu+1} = x_{\nu+1}, ..., X_n = x_n)$$

4.3 Structure of a Bayesian network

Graphically, Bayesian networks are models in which each variable is represented by a node, and causal relationships are denoted by an arrow, called directed arcs.

Nodes: A node represents a variable in the situation being modeled. A node is often represented graphically by a labeled oval. A simple example in Fig 4.1 shows two nodes, 'Precipitation' and 'Road Conditions'.

Edges: An edge represents a causal relationship between two nodes. It is represented graphically by an arrow between nodes; the direction of the arrow indicates the direction of causality. The intuitive meaning of an edge drawn from node X to node Y is that node X has a direct influence on node Y. For example, in Fig

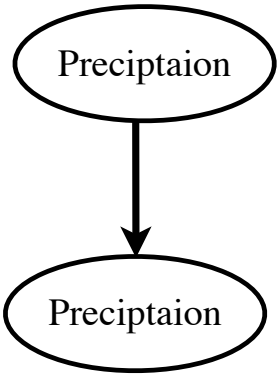


Figure 4.1: Two nodes and an edge in a very simple Bayesian belief network

4.2, the edge shows that the 'Precipitation' directly influences 'Road Conditions'. How much one node influences another is defined by the conditional probability tables associated with the nodes. Edges also determine some qualifying terms for nodes. When two nodes are joined by an edge, the causal node is called the parent of the other node. In this example, 'Precipitation' is a parent of 'Road Conditions', and 'Road Conditions' is a child of 'Precipitation'. Child nodes are conditionally dependent upon their parent nodes.

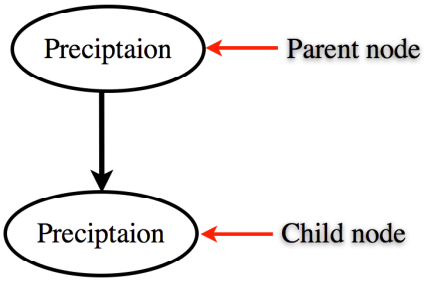


Figure 4.2: An edge indicates causality and conditional dependence parent node

States: The values taken on by a variable (represented by a node) are referred to as states. For example, the important states of 'Precipitation' are 'None' , 'Light' , and 'Heavy' . We know that precipitation causes a road to be passable or impassible. Those become the states of 'Road Conditions' , as shown in Fig 4.3.

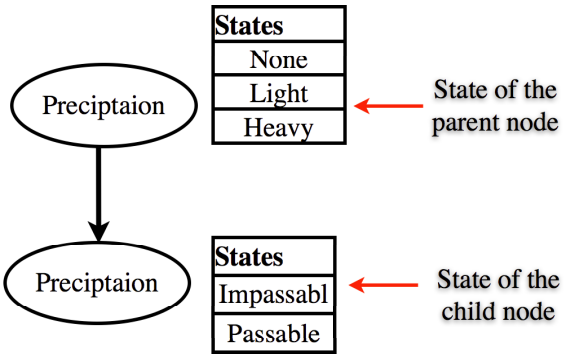


Figure 4.3: States are values that can be taken on by a node

4.4 Conditional probabilities

Once the topology of the BN is specified, the next step is to quantify the relationships between connected nodes. This is done by specifying a conditional probability distribution for each node. As we are only considering discrete variables at this stage, this takes the form of a conditional probability table (CPT).

First, for each node, we need to look at all the possible combinations of values of those parent nodes. Such a combination is called an instantiation of the parent set. For each distinct instantiation of parent node values, we need to specify the probability that the child will take each of its values.

For example, consider the Cancer node of Fig 4.4. We will begin with the restricted set of nodes and values shown in Table 4.1. Its parents are Pollution and Smoking and take the possible joint values [H, T], [H, F], [L, T], [L, F]. The conditional probability table specifies in order the probabilities of cancer for each of these cases to be: [0.05, 0.02, 0.03, 0.001]. Since these are probabilities, and must sum to one over all possible states of the Cancer variable, the probability of no cancer is given as one minus the above probabilities in each case; i.e., the probabilites of no cancer in the four possible parent instantiations are [0.95, 0.98, 0.97, 0.999].

Clearly, if a node has many parents or if the parents can take a large number of values, the CPT can get very large. The size of the CPT is, in fact, exponential in to number of parents. Thus, for Bayesian networks a variable with n parents requires a CPT with 2^{n+1} probabilities.

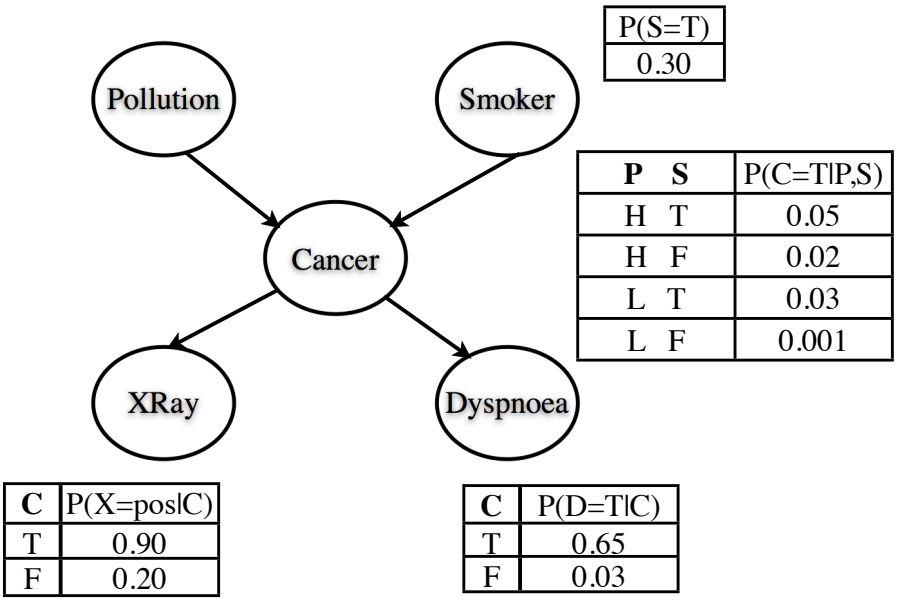


Figure 4.4: A BN for the lung cancer problem

Table 4.1: Choices of nodes and values for lung cancer example

Node name	Type	Values
Pollution	Binary	{low,high}
Smoker	Boolean	{T,F}
Cancer	Boolean	{T,F}
Dyspnoea	Boolean	{T,F}
X-ray	Binary	{pos,neg}

4.5 Inference and learning

One of the main usages of Bayesian networks is, based on a newly introduced evidence, to update the probability that a hypothesis may be true.

There are three main inference tasks for Bayesian networks:

4.5.1 Inferring unobserved variables

Because a Bayesian network is a complete model for the variables and their relationships, it can be used to answer probabilistic queries about them. For example, the network can be used to find out updated knowledge of the state of a subset of variables when other variables (the evidence variables) are observed. This pro-

cess of computing the posterior distribution of variables given evidences is called probabilistic inference.

The most common exact inference methods are:

(1) Variable elimination, which eliminates (by integration or summation) the non-observed non-query variables one by one by distributing the sum over the product.

(2) Clique tree propagation, which caches the computation so that many variables can be queried at one time and new evidence can be propagated quickly.

(3) Recursive conditioning, which allows for a space time tradeoff and matches the efficiency of variable elimination when enough space is used. All of these methods have complexity that is exponential in the network's tree width. The most common approximate inference algorithms are stochastic MCMC (Markov Chain Monte Carlo) simulation, mini-bucket elimination that generalizes loopy belief propagation, and variation methods.

4.5.2 Parameter learning

In order to fully specify the Bayesian network and thus fully represent the joint probability distribution, it is necessary to specify for each node X the probability distribution for X conditional upon X 's parents. The distribution of X upon its parents may have any form. It is common to work with discrete or Gaussian distributions since that simplifies calculations. Sometimes only constraints on a distribution are known; one can then use the principle of maximum entropy to determine a single distribution, the one with the greatest entropy given the constraints.

These conditional distributions often include parameters that are unknown and must be estimated from data, sometimes using the maximum likelihood approach. Direct maximization of the likelihood (or of the posterior probability) is often complex when there are unobserved variables. A classical approach to this problem is the expectation maximization algorithm which alternates computing expected values of the unobserved variables conditional on observed data, with maximizing the complete likelihood (or posterior) assuming that previously computed expected values are correct. Under mild regularity conditions this process converges to maximum likelihood (or maximum posterior) values for parameters.

A more fully Bayesian approach to parameters is to treat parameters as addi-

tional unobserved variables , compute a full posterior distribution over all nodes on the basis of observed data, and then to integrate out the results. This approach can be expensive and lead to large dimension models, so in practice classical parameter setting approaches are more common.

4.5.3 Structure learning

In the simplest case, a Bayesian network is specified by an expert and then used to perform inference. In other applications the task of defining the network is too complex for humans. In this case the network structure and the parameters of the local distributions must be learned from data.

Automatically learning the graph structure of a Bayesian network is a challenge pursued within machine learning. The basic idea goes back to a recovery algorithm developed by Rebane and Pearl [20]. The distinction between the three possible types of adjacent triplets allowed in a directed acyclic graph (DAG) are as follows:

$$1. X \rightarrow Y \rightarrow Z$$

$$2. X \leftarrow Y \rightarrow Z$$

$$3. X \rightarrow Y \leftarrow Z$$

Type 1 and type 2 represent the same dependencies (X and Z are independent given Y) and indistinguishable. Type 3, however, can be uniquely identified, since X and Z are marginally independent and all other pairs are dependent. Thus, while the skeletons (the graphs stripped of arrows) of these three triplets are identical, the direction of the arrows is partially identifiable.

The same argument is valid when X and Z have common parents, except that they must first be conditioned to their parents. Algorithms have been developed to systematically determine the skeleton of the underlying graph and, then, make up all arrows whose directions are determined by the observed conditional independencies.

4.5.4 Building Bayesian Networks

The construction of a Bayesian network involves three major steps:

First, we need decide on the set of relevant variables and their possible values.

Next, we need build the network structure by connecting the variables into a DAG.

Finally, we need define the CPT for each network variable.

The last step is the quantitative part of this construction process and can be the most involved in certain situations.

For the BN building for each kind of scene of soccer video in our research, because each kind of highlight scene is strongly related with certain kind of shot type, and each shot boast some key features, so the DAG of each BNs can be determined by human observation in training stage. Then the CPT will be defined by calculating the conditional probability between the nodes pairs.

Chapter 5

Shot Change Detection and Shot Classification

To realize highlight scene detection in soccer games, the precise boundary of one scene should be identified in advance. Some main features are extracted by using the information embedded in MPEG codec directly. These main features are the basic factors for the shot classification and BNs construction. Under the precise shot change detection and shot classification, video structure can be precisely classified into three layers: scene layer, shot layer and evidence layer.

5.1 Shot Change Detection

The shot change detection is the basic step for high-level concept recognition, such as goal detection. And, the shot change detection is the fundamental task in content-based analysis and indexing of videos, as it helps us to provide a hierarchical structure of video and enables the extraction of meaningful highlights from such a structure.

In general, a shot change is defined to be an image content change between two consecutive frames. In the standard GOP structure [2], a GOP consists of three kinds of frame types: I frame, B frame and P frame. Their order is given as follows;

$$IBBPBBPBBPBBPBB$$

In I frame, there are only I mode MB. In P frame, there are F mode and I mode (not compulsory) MBs. In B frame, four MB types exist. Moreover, MB (Macro Block) [3] can be divided into four types; Intra prediction (I mode) MB, Forward prediction (F mode) MB, Backward prediction (B mode) MB, and Bidirectional

prediction (BI mode) MB. These prediction methods are introduced in [3]. For the sake of high computing speed of further video structure, MBT method [18] is adapted for shot change detection in our research.

5.1.1 Patterns of Macro-block (MB) types in abrupt transition

In this thesis, we classify abrupt transition into five situations (shown in Fig 5.1), and take MB type information in two consecutive B frames (B_i, B_{i+1}) into consideration. We mainly focus on abrupt transitions detections. Five types of abrupt transition are classified as: (a) scene change occurs before B_i frame, (b) scene change occurs between B_i and B_{i+1} frame, (c) scene change occurs after B_{i+1} frame.

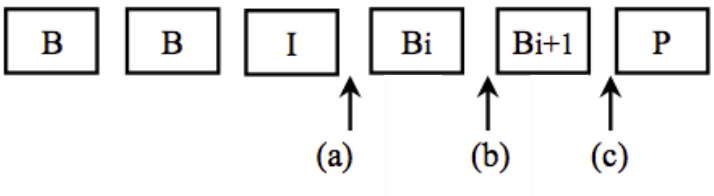


Figure 5.1: Shot change position

5.1.2 Relationship between MB types and each kind of abrupt transition

Detail of five kinds of shot change are described as follows:

In the case (a), shown in Fig 5.2, plenty of MBs in P frame are I mode since the P frame is the first reference frame of new shot. Simultaneously, most MBs of the two consecutive B frames are B mode because image change exists between I frame and the following two B frames.

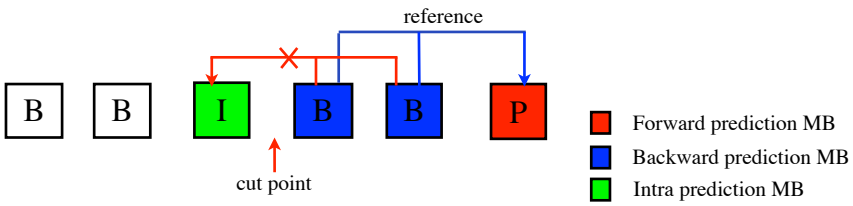


Figure 5.2: Shot change occurs before B_i frame

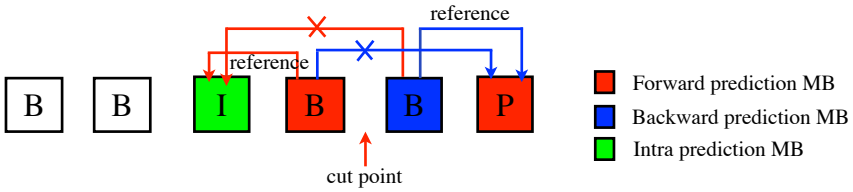


Figure 5.3: Shot change occurs between B_i andB_{i+1} frame

In the case (b), shown as Fig 5.3, many I mode MBs exist in P frame. And most MBs in B_i frame are F modes, while most of MBs in B_{i+1} frame are B modes.

In the case (c), shown as Fig 5.4, Most MBs of the two consecutive B frames are F mode because they have more similarities to I frame than P frame.

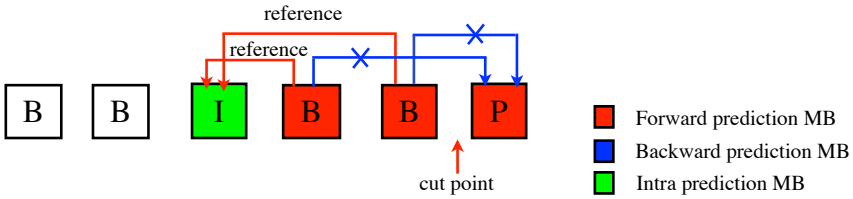


Figure 5.4: Shot change occurs after B_{i+1} frame

5.1.3 Processing method

Due to occurrence of abrupt transition at each kind of positions discussed above, the MB prediction directions are varied. Therefore, the MB types and their proportions in two consecutive B frames are distinct. Our processing method for detectiing abrupt transition is as follows:

First, according to distribution of macro-block types in P frame or B frame, we classify the frames into seven types, and label them from number 0 to 6 (as shown in Table 5.1).

Then, based on patterns of MB types in abrupt transitions, we make a decision. For two consecutive B frames B_i and B_{i+1}, if their frame types belong to a certain type, a shot change occurs.

Table 5.1: FrameType Decision Rule

FrameType	Conditions
0	Number of F mode MB is the largest in B frame and $450 > (mbBack + mbInter)$
1	Number of F mode MB is the largest in B frame and $1150 > (mbBack + mbInter) \geq 450$
2	Number of F mode MB is the largest in B frame and $450 > (mbBack + mbInter) \geq 1150$
3	Number of B mode MB is the largest in B frame and $450 > (mbForward + mbInter)$
4	Number of B mode MB is the largest in B frame and $1150 > (mbForward + mbInter) \geq 450$
5	Number of B mode MB is the largest in B frame and $(mbForward + mbInter) \geq 1150$
6	Number of I mode MB is the largest in B frame

5.2 Main features extraction

Most of the semantic analysis methods rely on the low-level evidences in the scene. Here, we briefly describe the methods for calculating the probability of the existence of the low-level evidence including 16*8 size prediction MB, field line slope and score board. The feature extraction process provides low-level evidences based on different media components of soccer game videos. These low-level evidences are essential for the BN. Here some main features extracting are introduced as follows:

A. 16*8 size Macro-block(reference work [18])

To avoid the time-consuming overhead of frame-by-frame image processing, we utilize MPEG-2 codec data directly for detecting Up/Long shots. In MPEG video codec, a practical and widely used method of motion compensation is to compensate for the movement of rectangular sections or 'blocks' of the current frame. In MPEG-2/4 video compression standards, the 16*8MBs frequently occur especially when the camera captures a player in close-up and an active motion occurs.

Fig 5.5 shows the 16*8 size MBs occur in Long Shot (the left) and Up Shot (the right), where the red blocks in Figi 5.5) represent 16*8 MBs. We can clearly see that, in the right picture, a large number of 16*8 size MBs occur because of the sharp movements of players. On the other hand, only a few 16*8 size MBs appear in Long Shot (shown in the left picture of Fig 5.5) because most of the motions in Long Shots are caused the camerawork movement. By using the information of 16*8 size MBs embedded in MPEG-2 coded, the Up/Long shot can be detected rapidly from HD video without any image processing approach.

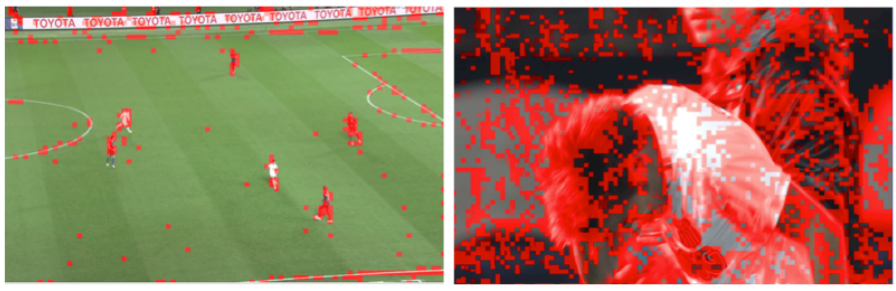


Figure 5.5: The 16*8 MB in Up/Long Shot

B. Field Line Slope(reference work [18])

The appearance of field lines in a Long shot view can be used to indicate the existence of the goal gate. In other words, the appearance of the gate and the appearance of field lines are highly correlated. The gate is visible when the players appear near or within one of the penalty boxes. The information of parallel lines indicating the penalty box is very useful for the gate detection. Indeed, the information of the field line is more reliable than the information of the goal post from the video scene, since detection of the goal post may be affected by the cluttered background noise.

We extract the luminance (Y) and chrominance (Cb,Cr) from MBs of each I frame to identify the location of field lines. Moreover, we utilize the number of AC components ($ACqt$) of the luminance (Y) contained in an MB if an MB has (Y), (Cb,Cr), and $ACqt$ satisfying the following inequalities we judge that there is a field line in it.

$$65 \leq Y \leq 235 \quad and \quad 90 \leq Cb \leq 129$$

$$105 \leq Cr \leq 130 \quad and \quad 4 \leq ACqt \leq 20$$

For the Long Shot, in most situations , a camera is shooting from the stand area. As a result, the touch-line or half-way line which are only in vertical direction (Shown in Fig 5.6). On the other hand, the Goal line or Penalty line appears in the scene when the camera is shooting the corner or near the goal area. During these situations discussed above, these lines are either vertical or oblique. By means of a method proposed in [18], binary images are transformed directly from the MPEG-2 compressed images, and then field line is extracted by using the Hough transform. Fig 5.7 shows an example of the binary image and detected field line.

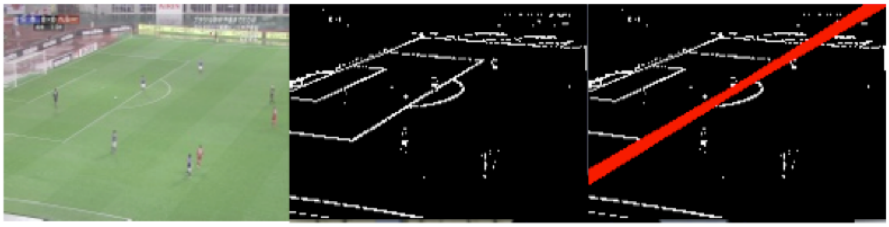


Figure 5.6: Extracted touch-line and half-way line

By extracting the field line of the ground, the slope can be computed. During one shot, the slopes of the field line in the first GOP and last 3 GOPs are used to classify the long shot into four kinds: first goal long shot, last goal long shot, first and last goal long shot ,and center long shot.

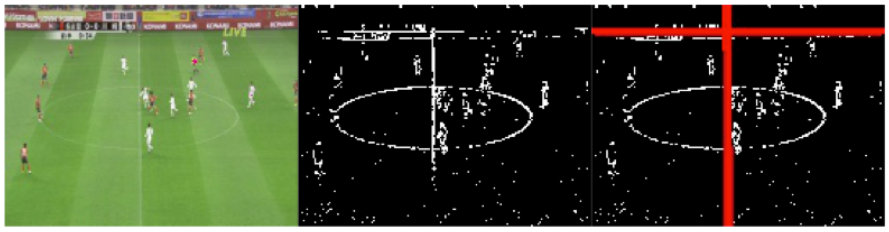


Figure 5.7: Extracted field-line

C. Score Board(reference work [18])

In broadcast sports videos, a scoreboard is very often superimposed in a fixed position on display to indicate status such as team names, the game score, etc. for viewers . During the progression of soccer games, the time and the score board always appear, but according to their observation they disappear during a Replay scene (shown in Fig 5.8). Therefore, by detecting the score board location and the duration of its disappearing period, it is possible to detect replay scene.

They utilize the DCT coefficients and motion vector extracted from MPEG2 codec. If a frame contains an area where no motion vectors occur, and the sum of the absolute value of DCT coefficients in that area exeed a thresholds, which is recognized as a candidate of score board area. Then, a candidate area for the score board will be landed to the candidate list when the situation set above is satisfied over several consecutive frames. Finally, the candidate list is labeled with color as detected score board area (shown in Fig 5.9).

Figure 5.8: Score board disappeared during Replay Scene

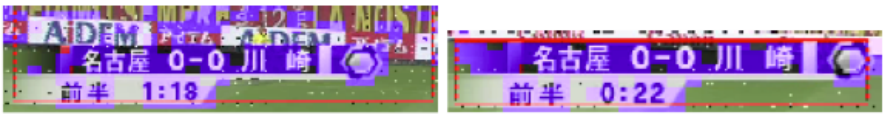


Figure 5.9: Detected out score board

D. Average Ground Color MB number(reference work [18])

For extracting the ground region, some features are considered by extracting the code data from MPEG-2, the DC component of chrominance (Cb,Cr) and AC component of Cr are utilized. By setting a proper color threshold, the ground area can be extracted out (shows in Fig 5.10, where the white blocks are the ground color MB).

In the up shot, especially when camera shoots to the spectors, the camera work motion is very small. As a result, the 16*8 MB information is invalid. Therefore, we

use the number of average ground color MB, since it is effective only for up shots.

Figure 5.10: Extracted ground region

5.3 Shot Classification

After the shot change detection, we use some new parameters for shot classification as follows:

- $\overline{M}$

  The average number of 16×8 macro blocks of GOPs in a shot

- A

  The slope of field line near goal area in an I frame of the first GOP of a shot.

- $\overline{B}$

  The slope of field line near goal area in an I frame of the last GOP of a shot.

- $\overline{C}$

  The cut point count in a non-score board scene.

- $\overline{G}$

  The average number of ground area macro blocks of I frames of last 3 GOP in a shot

- $nGOP$

  The total number of GOPs in one shot

Based on the shot change detection and main features extraction, six kinds of shot are defined as follows: (1) center Long Shot, (2) first goal long shot, (3) last goal long shot, (4) first and last goal long shot, (5) up shot, and (6) non-score board shot. Their characteristics are simply described below.

Center Long shot: A center long shot displays the global view of the game middle field as shown in Fig 5.11. It is captured by a camera at a long distance. The center line is one of the key features in the center long shot. The field and the scale of players is small in a center long view. A center long shot is useful for accurate localization of events in the middle field.

First Goal Long shot: A first goal long shot contains the view of the goal area as shown in Fig 5.12. It is captured by a camera at a long distance. The goal line is one of the key features in first goal long shot. The goal line slope under the

Figure 5.11: Center Long shot

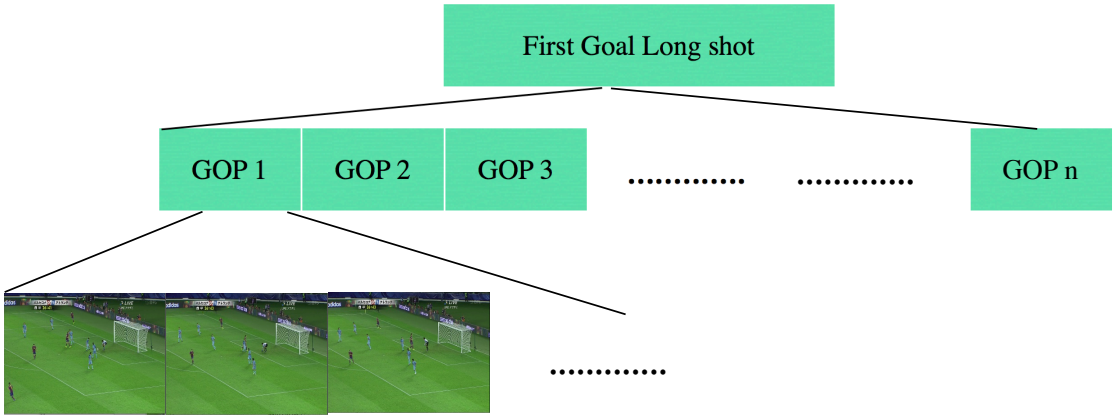


Figure 5.12: First Goal Long shot

condition: $0.2 \leq A \leq 0.6$ appears in the first GOP of a shot. A First Goal Long shot is useful for accurate localization of events around goal field.

Last Goal Long shot: A last goal long shot displays the view of the goal area as shown in Fig 5.13. It is captured by a camera at a long distance. The goal line is one of the key features in last goal long shot. The goal line slope under the condition: $0.2 \leq \left|\overline{B}\right| \leq 0.6$ appears in the first GOP of a shot. A First Goal Long shot is useful for accurate localization of events around the goal field.

First and Last Goal Long shot: A first and last goal long shot displays the view of the goal area as shown in Fig 5.14. It is captured by a camera at a long distance. The goal line is one of the key features in the first and last goal long shot. The goal line slope under the condition: $0.2 \leq |A| \leq 0.6 \quad \textit{and} \quad 0.2 \leq \left|\overline{B}\right| \leq 0.6$ appears in the first and last GOP of one shot. A first and last goal long shot is useful for accurate localization of events around the goal field.

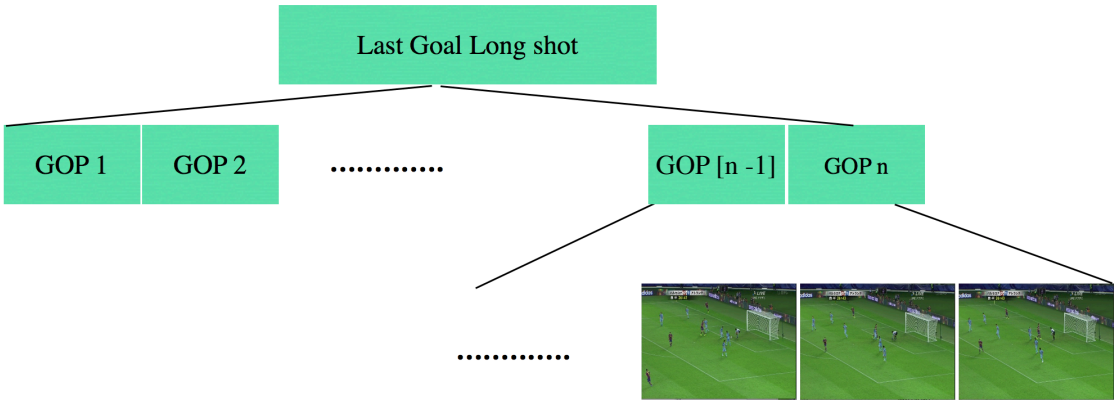


Figure 5.13: Last Goal Long shot

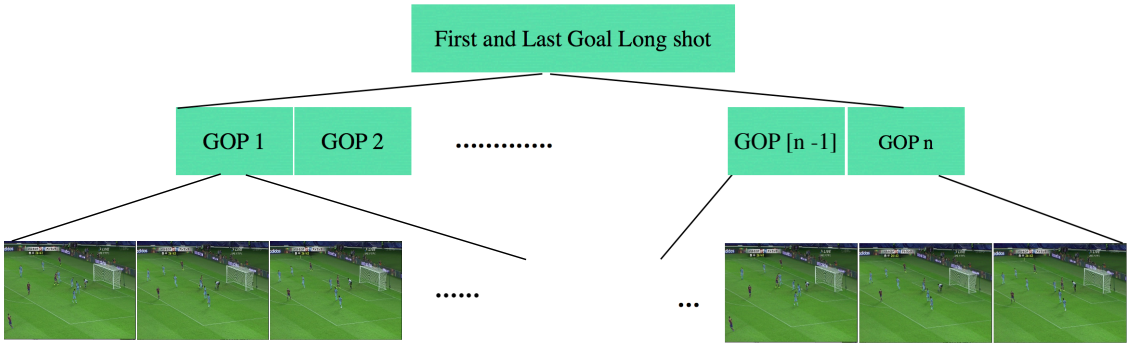


Figure 5.14: Last Goal Long shot

Figure 5.15: Up shot

Up shot: An up shot usually shows an upper half of the body of one person who is captured by a close-up camera as shown in Fig 5.15. An up shot often contains leading players.

Non-Score Board Shot: A replay scene is usually utilized to play back an interesting or important segment with a slow-motion pattern in broadcast video. It is often taken as a significant cue for semantic event detection or summarization. One important feature is that when the replay appears, the score board disappears

gradually. Fig 5.16 shows the score board fades out gradually in Non-Score Board Scene.

Figure 5.16: Non-Score Board Shot

Chapter 6

Semantic analysis using BN

Bayian Network (BN) is a powerful semantic analysis tool which has been applied to stochastic modeling the high-level semantic information embedded in the video data. In various kinds of sports, the high-level semantics is the occurrence of the highlight events containing specific temporal structures that appear repeatedly. Here, we use BNs to model the semantic highlights of soccer games such as replay scene and goal area scenes.

6.1 Three semantic layers of BN of soccer video

To realize highlight detections in soccer games, high-level semantic information embedded in the video is needed. Therefore, three kinds of semantic layers are defined as follows:

High-level layer: replay scene and goal area scene.

Middle-level layer: which consists of the center long shot, first goal long shot (FGL Shot), last goal long shot (LGL Shot), first and last goal long shot (FLGL Shot), up shot and non-score board shot (NSB Shot).

Low-level layer: average number of 16*8MBs, the average number ground region MBs, field line slope, score board information, the number of GOPs in a shot, etc.

Eight kinds of low-level features, are labeled from C_0 to C_7 for simplify. The meaning of each featur is described as follows:

C_0: The average number of ground area MB of I frames of last 3 GOP in a shot is more than 3,000.

C_1: There is a field line of the goal area in an I frame of the first GOP of a shot.

C_2: There are field lines of the goal area in at least two I frames of the last three GOPs including the last one of a shot.

C_3: C_1 and C_2 occur simultaneously in a shot.

C_4: There is no score board in more than three consecutive shots but it appears again in the next one.

C_5: The score board disappeared consecutively no less than 100 frames in one shot.

C_6: The total number of GOPs in a shot is no less than 30.

C_7: : The average number of 16×8 macro blocks of GOPs in a shot is more than 1400.

The structure of BN layer is shown in Fig6.1.

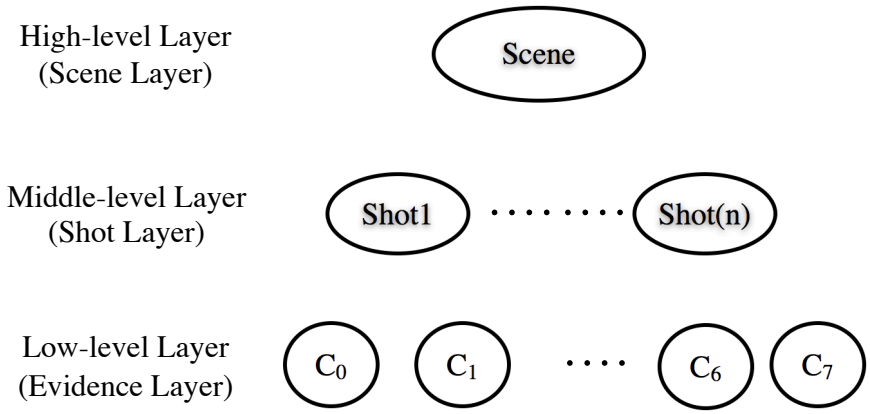


Figure 6.1: Three layers of BN

6.2 Training Phase

A training can be categorized into two classes: the qualitative (structural) training and the quantitative (parameter) training. The qualitative training concerns the network structure of the model and the quantitative training determines the specific conditional probabilities.

(1) The qualitative training: Previous shot change detection and shot classification were prepared for the structure training. One kind of highlight scene consists of two or three kinds of consecutive shots (shown in Fig 6.2). These shot patterns were observed in this step; for example, the shot pattern Up Shot $\rightarrow$ NSB Shot means that of two consecutive shots, where the former is up shot, the latter is non-score board shot. These two kinds of shots were observed for the replay scene. Each kind of shots has certain main feature. For example, in the up shot: C_0 and C_7 often appear. The NSB Shot (Non-Score Board Shot) is related to both of: C_4 and C_6.

(2) The quantitative training: In the quantitative training, some dependence between nodes and the occurrence possibility of each node in the network are calculated. The training procedure can be divided into two steps as follows:

Step 1:

In this step, we compute all the conditional probabilities associated with nodes in the high-level nodes and the middle-level, shown in Fig 6.2. If a joint event occurs in the two levels, only an appearance of the high-level node should be counted. For example, the conditional probability of an up shot, under the condition that a replay scene occurs, can be calculated as fllows:

$$P(Up\ Shot = True | Replay\ Scene = True)$$

$$= P(Replay\ Scene = True, Up\ Shot = True)/P(Replay\ Scene = True)$$

Step 2:

In this step, the conditional probability of the existing edges between the feature and shots will be calculated. It is carried out for each pair of nodes between the middle-level node and a low-level, shown in Fig 6.2. If the event associated with these two nodes jointly occur, only an appearance of the middle-level node should be counted. For example, the conditional probability of feature C_7, under the condition that the up shot happens, can be calculated as:

$$P(C_7 = True | Up\ Shot = True)$$

$$= P(Up\ Shot = True, C_7 = True) / P(Up\ Shot = True)$$

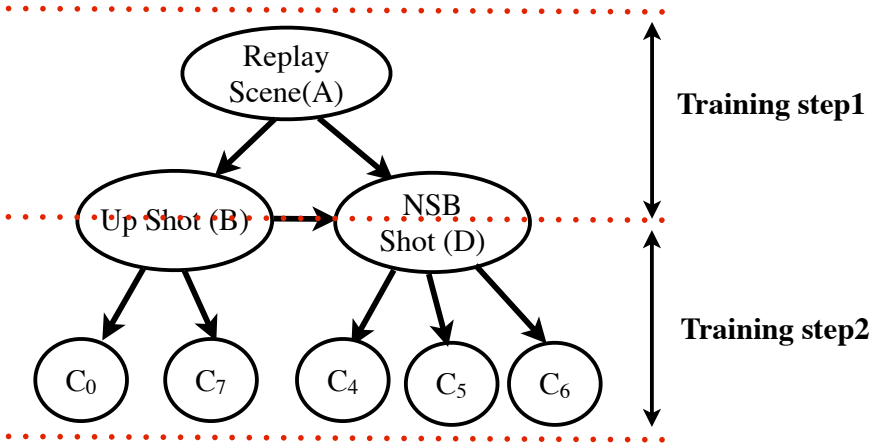


Figure 6.2: The BN of replay scene

6.3 Experiment for training

In the training step, we have used 10 hours of live-recordings of soccer videos for training. The purpose of the training are: (1) find the shot patterns correspond to each kind of highlight scenes. (2) find the feature related to each kind of shot, and (3) learn the CPT of each kind of BNs. Training video data is utilized to construct DAGs and compute prior probabilities of each highlight scene BN.

6.3.1 The qualitative training:

Shot pattern training for one kind of scene

Previous shot change detection and shot classification are contribute to the shot pattern training. From 10 hours video training data, five kinds of shot patterns are observed as follows:

(1) The shot pattern of Replay Scene :

Shot pattern : $Up\ Shot \rightarrow NSB\ Shot$, correspond to the Replay Scene

It indicates that when a replay scene happen, an Up Shot happens at first, and an NSB Shot will happen consecutively.

(2) The shot pattern of Goal Area Scene:

Shot pattern 3_5 : $Up\ Shot \rightarrow FGL\ Shot$, correspond to the $Goal\ area\ Scene3\_5$

Shot pattern 5_3 : $FGL\ Shot \rightarrow Up\ Shot$, correspond to the $Goal\ area\ Scene5\_3$

Shot pattern 5_4: $Up\ Shot \rightarrow FLGL\ Shot$, correspond to the $Goal\ area\ Scene5\_4$

Shot pattern 4_5: $FLGL\ Shot \rightarrow Up\ Shot$, correspond to the $Goal\ area\ Scene4\_5$

Four shot patterns above are different, and each of shot pattern correspond to one kind of goal area scene. Taking the shot pattern: $Up\ Shot \rightarrow FGL\ Shot$ for example, it means that when an Up Shot happen , and then a First Goal Long Shot (FGL Shot) happen consecutively, this kind of shot pet tern is defined as $Goal\ area\ Scene3\_5$. The difference between $Up\ Shot \rightarrow FGL\ Shot$ and $FGL\ Shot \rightarrow Up\ Shot$ is that which shot occur at first.

In training data, the count of each kind of highlight scene happen and their proportions are shown in the Table 6.1.

Table 6.1: Five kinds of scene and their proportions

Scene Type	Replay Scene	Goal area Scene5_4	Goal area Scene4_5	Goal area Scene5_3	Goal area Scene3_5	Total
Total count	86	56	28	12	7	189
Probability	45.50%	29.63%	14.81%	6.35%	3.70%	1

Because each kind of shots has special features, the shot and it related features are list as following Table 6.2.

Table 6.2: The features related with each kind of shot judge

Shot Type	Center Long Shot	FGL Shot	LGL Shot	FLGL Shot	Up Shot	NSB Shot
Shot label	1	2	3	4	5	6
Related Features	$C_0\ C_7$	$C_1\ C_2\ C_3$	$C_1\ C_2\ C_3$	$C_1\ C_2\ C_3$	$C_1\ C_2\ C_3$	$C_4\ C_5\ C_6$

6.3.2 The quantitative training:

In quantitative training, the dependence between the nodes and the occurrence possibility of each node in the network will be determined. Nodes are the graphical representation of the evidence of the events in the video which are usually termed as variables or states. For example, the CPT of the *Goal area Scene*5_4 are learned from training data as shown in Fig 6.3.

Some kinds of BNs for each highlight scene are shown in Fig 6.4.

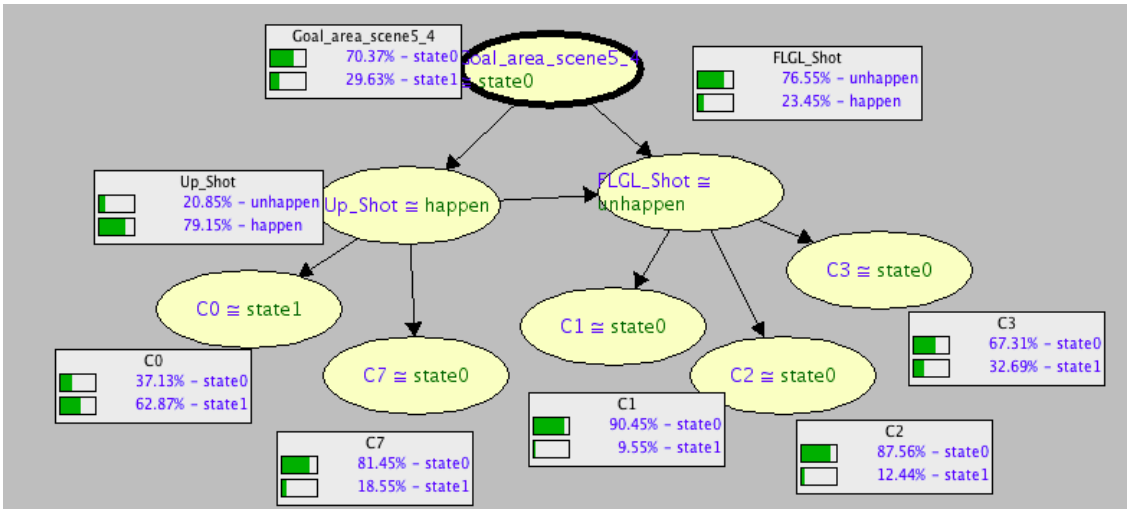


Figure 6.3: CPT of *Goal area Scene*5_4

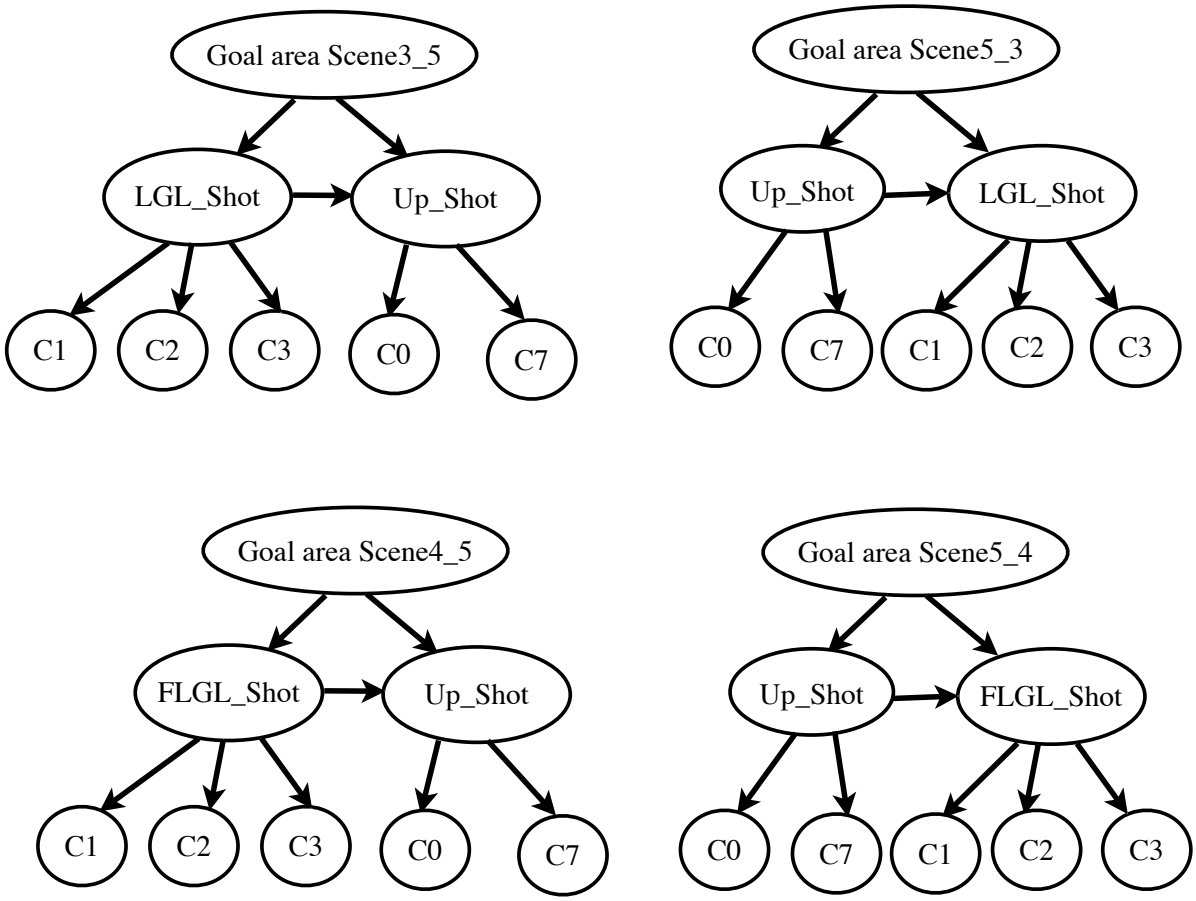


Figure 6.4: Four kinds of BNs associated with the goal area scene

6.4 Computing posterior probability

The inference can be performed using various algorithms such as expectation maximization variational algorithms [14], belief propagation [17], Markov Chain Monte Carlo [14], particle filter [16], etc. In our research, we have customized the application for soccer video sequences and used Variable Elimination algorithm [14] for inference. We present here one of the simplest inference algorithms for our research, which is based on the principle of variable elimination. Our goal here is restricted to computing the probability of marginal distributions under the assumption that some evidences are given in advance.

6.4.1 Factors

Before we discuss the method of variable elimination in our BN, we first need to discuss its central component: a factor. A factor f is a function over a set of variables, that maps each instantiation x of the random variables X to a non-negative number, can be written as $f(x)$ [14]. For example, the probabilities in Fig 6.2,

$$f(A) = P(A), f(A, B, D) = P(B|A)P(D|A)P(A)$$

To calculate probabilities in a BN, usually start with factors that represent conditional probabilities and end up with factors that represent marginal probabilities.

6.4.2 Elimination as a basis of inference

The Variable Elimination algorithm provides pseudo a code for computing the marginal probability over some variables Q in a Bayesian network based on the previous elimination method. The algorithm takes a value of Q, instantiations e and an elimination order π of remaining variables as input of a Bayesian network N. $\pi(1)$ is the first variable, $\pi(2)$ is the second variable, and so on.

The algorithm runs reputedly for each variable $\pi(i)$ in the order, to pick up all factors that contain variable $\pi(i)$, and to multiplies them to yield factor f, in which variable $\pi(i)$ is summed out, and finally replacing factors by factor $\Sigma_{\pi(i)}f$. When all variables in the order π are eliminated, we end up with a set of factors over

variables Q. Multiplying these factors gives the answer to our query; that is, the joint marginal $Pr(Q, e)$.

The method of variable elimination [14] can be extended to compute a jointly marginal probability if we start by zeroing out those rows in the joint probability distribution that are inconsistent with instantiations e.

Algorithm 1 Variable Elimination N, Q, e, π

Input:

N: Bayesian network
Q: variables in network N
e: instantiation of some variables in network N
π: an ordering of network variables not in Q

Output:

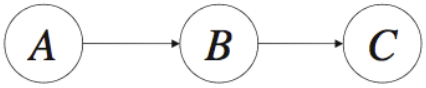
The joint marginal $\Pr(Q, e)$;

Main

1: $S \leftarrow f^e$: is CPT of network N
2: for $i = 1$ to length of order π do
3: $S \leftarrow \prod_k f_k$, where f_k belong to S and mentions variable $\pi(i)$
4: $f_k \leftarrow \sum_{\pi(i)} f$
5: replace all factors f_k in S by factor f_i
6: **end for**
7: **return** $\prod_{f \in S} f$;

6.4.3 An example of Elimination Variable for inference

One simple example of the Variable Elimination works as follow:



A	Θ_A
true	.6
false	.4

A	B	$\Theta_{B\|A}$
true	true	.9
true	false	.1
false	true	.2
false	false	.8

B	C	$\Theta_{C\|B}$
true	true	.3
true	false	.7
false	true	.5
false	false	.5

Figure 6.5: A Bayesian Network

Let us consider an example with respect to the Bayesian network in Fig 6.5. Our goal here is to compute the prior marginal on variable C, $\Pr(C)$, by first eliminating variable A and then B. There are two factors that mention variable A, Θ_A and $\Theta_{B|A}$. We must multiply these factors first and then sum out variable A from the resulting factor. Multiplying Θ_A and $\Theta_{B|A}$, shown in Table 6.3,

Table 6.3: The CPT of $\Theta_A\Theta_{B|A}$

A	B	$\Theta_A\Theta_{B\|A}$
ture	ture	0.54
ture	false	0.6
false	ture	0.6
false	false	0.6

Summing out variable A, shown in Table 6.4,

Table 6.4: The CPT of $\Sigma_A\Theta_A\Theta_{B|A}$

B	$\Sigma_A\Theta_A\Theta_{B\|A}$
ture	0.62 = 0.54 + 0.08
false	0.38 = 0.06 +0.32

We now have two factors, $\Sigma_A\Theta_A\Theta_{B|A}$ and $\Theta_{C|B}$, and we want to eliminate variable B. Since B appears in both factors, we must multiply them first and then sum out B from the result, shown in Table Table 6.5,

Table 6.5: The CPT of $\Theta_{C|B}\Sigma_A\Theta_A\Theta_{B|A}$

B	C	$\Theta_{C\|B}\Sigma_A\Theta_A\Theta_{B\|A}$
ture	ture	0.186
ture	false	0.434
false	ture	0.190
false	false	0.190

Summing out B and C respectively from Table 6.5, we get the final result shown in Table 6.6,

Table 6.6: The CPT of $\Sigma_B\Theta_{C|B}\Sigma_A\Theta_A\Theta_{B|A}$

C	$\Sigma_B\Theta_{C\|B}\Sigma_A\Theta_A\Theta_{B\|A}$
ture	0.376
false	0.624

This factor is then the prior marginal for variable C, $\Pr(C)$. Therefore, according to the Bayesian network in Fig 6.5, the probability of $C = true$ is 0.376 and the probability of $C = false$ is 0.624.

Chapter 7

Experiments

In the training phase, four video games (360min) are used to generate BNs for highlight scenes. In the testing experiments, we have tested 360 min video sequences from four soccer programs. The proposed algorithm is evaluated by using four MPEG-2 soccer videos. The video resolution is 1440 × 1080 resolution and played back in 29.97 frames per second for each detection evaluation.

7.1 The pre-experiment : Shot Change Detection

The Shot change detection is the fundamental task in content-based analysis and indexing of videos, as it helps us to provide a hierarchical structure of video and enables the extraction of meaningful highlights from such a structure.

Two metrics are defined to describe the shot change detection accuracy; the recall and precision. The recall is defined to be the ratio of the number of correct detections and missed detections, and the precision is defined to be the ratio of the number of correct detections to both correct detection and false detection. That is, the recall and the precision are given by:

$$Recall = \frac{CDC}{CC} \times 100$$
$$Precision = \frac{CDC}{ADC} \times 100$$

Here, CDC denotes the number of correct detections, CC denotes the number of existing shot changes, ADC denotes the number of all detections.

The experiment result, Table 7.1 shows that it is suitable to use MB type information to detect shot change, which is a benefit for the shot classification and video structure further head, and strongly contribute to our BN construction.

Table 7.1: Shot change detection result accuracy

	CC	CDC	ADC	NDC	FDC	Recall	Precision
Soccer 1	427	411	445	16	34	96.3%	92.4%
Soccer 2	300	294	306	6	12	98.0%	96.1%
Soccer 3	676	602	620	74	18	89.1%	97.1%
Soccer 4	208	182	206	26	24	87.5%	88.3%
Total of four soccer	1611	1489	1577	122	88	92.4%	94.4%

7.2 Scene detection accuracy

Two kinds of highlight scene detection are carried out in scene detection stage. The replay scene and goal area scene. The BN of the replay scene shown in Fig 6.2. Four kinds of BN correspond to the goal area scene are shown in Fig 6.4.

Some other metrics are defined to describe the scene detection accuracy. They are correct scene (CS), correct detected scene (CDS), all detected scene (ADS), not detected scene (NDS), and false detected scene (FDS). For simplicity, we denote CS the number of already known scenes, CDS as the number of all correct detection, and ADS as the number of all detections, NDS as the number of miss detections and FDS as the number of false detections. Table 7.2 and Table 7.3 show the result of recall and precision of replay scenes and goal area scenes. The evaluation of recall and the precision are given by:

$$Recall = \frac{CDS}{CS} \times 100$$

$$Precision = \frac{CDS}{ADS} \times 100$$

Table 7.2: Result of replay scene detection

	CS	CDS	ADS	NDS	FDS	Recall	Precision
Soccer 1	52	45	54	7	9	83.33%	86.54%
Soccer 2	40	39	48	1	6	86.67%	97.50%
Soccer 3	41	38	49	3	11	75.55%	92.68%
Soccer 4	42	40	52	2	12	76.92%	95.24%
Total	175	162	200	13	38	81.00%	92.57%

The performances of scene detection are shown in Table 7.2 and Table 7.3. We observed 81.00% recall and high 92.57% precision for replay scene; high 90.76% recall and 86.17% precision for goal area scene.

Table 7.3: Result of goal area scene detection

	CS	CDS	ADS	NDS	FDS	Recall	Precision
Soccer 1	83	69	76	14	7	83.13%	90.79%
Soccer 2	102	89	95	13	6	87.25%	93.68%
Soccer 3	108	92	104	16	12	85.19%	88.46%
Soccer 4	83	74	82	9	8	89.16%	90.24%
Total	376	324	357	52	33	86.17%	90.76%

We analyze the reason thy the low performance of precision are obtained in replay scene. From the false detection result, we find some shot patterns associate with the replay scene are abnormal, these shot patterns are listed as follows:

In the false detection shot pattern, neither of the shots matches the key shot type (Up Shot or NSB Shot), these shot patterns include:

Shot pattern 3_1: First, the LGL Shot occur, then the Center Long Shot occur

Shot pattern 4_2: First, the FLGL Shot occur, then the FGL Shot occur

Shot pattern 3_4: First, the LGL Shot occur, then the FLGL Shot occur

Shot pattern 4_4: First, the FLGL Shot occur, then the FLGL Shot occur

According to our investigation on the features of each shot type in these shot patterns when false detection happen, we note that the second shot in the shot patterns above has the same characteristic: the state of feature C_4 equals 1. It means that the score board in these shot disappears for quite a long period. Then by checking the video of the false detection period, we found that although the score board disappear, the replay shot did not happen. We think this is one of the reason of the false detection for replay scene.

7.3 Scene detection result comparison

Here, we compare our experimental results with reference work [4] and [18] to illustrate our system performance.

1. Result comparison with reference work [4]

In the work [4], their system is frame-based events detection. They have tested their algorithms based on a data set of seven soccer video games for more than 11 hours. The format of their video source is MPEG-1 clips in 320 × 240 resolution at 30 frames/s. Two kinds of events are detected in their system, the goal events and corner events. However, in our system, we define these two kinds of highlight events as goal area scene since the goal events and the corner events must happen near the goal area.

Table 7.4 and 7.5 show the statistics of the experiment results of goal event detection and corner event detection.

Table 7.4: Result of Goal Event Detection

	Video1	Video2	Video3	Video4	Video5	Video6	Video7	All
Detected	31	40	36	34	28	28	39	236
False	5	8	9	5	8	5	11	51
Missed	1	0	1	0	0	0	0	2
Recall	96.9%	100%	97.3%	100%	100%	100%	100%	99.2%
Precision	86.1%	83.3%	80.0%	87.2%	77.8%	84.8%	78.0%	82.2%

Table 7.5: Result of Corner Event Detection

	Video1	Video2	Video3	Video4	Video5	Video6	Video7	All
Detected	23	18	20	21	18	21	24	145
False	9	7	7	8	9	7	12	59
Missed	1	2	3	1	1	1	1	10
Recall	95.8%	90.0%	87.0%	95.5%	94.7%	95.5%	96.0%	93.5%
Precision	71.9%	72.0%	74.1%	72.4%	66.7%	75.0%	66.7%	71.1%

From the result, we can see that they achieved average recall rate 99.2% and average precision rate 82.2% of goal event detection; average recall rate 93.5% and average precision rate 71.1% of corner event detection. Compared with our goal area scene detection result shown in Table 7.3, although the average recall rate 86.17%

of our result is not better than them, our average precision rate 90.76% is better than them. Besides, their experiment video source is MPEG-1 clips in 320 × 240 resolution, while our system is carried on high-definition(1440 × 1080 resolution) video.

2. Result comparison with reference work [18]

In the work [18], their system is a shot-based event detection. They tested their algorithms based on a data set of four soccer video games for more than 5 hours. The video source is MPEG-2 clips in 1440 × 1080 resolution at 30 frames/s. They utilize the pattern matching method to detect highlight scene. They classify the shot into seven kinds, and label the video sequences with each kind of shot type. Shot patterns are detected from the video sequence, and the highlight scene will be detected. The goal area scene detection result are shown in Table 7.6.

Table 7.6: Result of goal area scene detection by Pattern Matching method

Video	CS	CDS	ADS	NDS	FDS	Recall	Precision
Soccer 1	89	67	81	22	14	77.2%	82.7%
Soccer 2	57	45	48	12	3	78.9%	93.8%
Soccer 3	120	78	84	42	6	65.0%	92.9%
Soccer 4	42	29	31	13	3	69.0%	93.5%
Total	308	219	244	89	26	71.1%	89.8%

From the result, we can see that they achieved average recall rate 71.1% and average precision rate 89.8% of goal area scene detection. Compared with our goal area scene detection result shown in Table 7.3, both of the average recall rate 86.17% and average precision rate 90.76% is better than them, especially we increase about 15% in average recall rate.

Compared with reference work [4], both of us utilize the BN to soccer video semantic analysis, we achieve better average precision rate but lower average recall rate. The reason of the recall rate is not better than reference work [4] is because (1) fewer key features are extracted for BN building in our research, except for the image features, some important identical features are extracted in reference work [4] , for example, the audio energy which is strongly related with the goal scene. (2) compared with complex image processing in pixel domain of reference work [4], feature extraction in our approach is based on MPEG domain, therefore some features detection may fail. We achieve better average precision rate because several

features are used for hidden-node which is more reliable and precise than use a single feature in reference work [4] .

Compared with reference work [18], both of us utilize the shot based soccer video semantic analysis, we increase both of the average recall rate and average precision rate, especially about 15% up in average recall rate , this is because unlike detect some shot patterns in their approach, we compute the probability of each shot pattern occur, this can reduce the miss detect of some shot patterns which are rarely occur in a low probability.

Chapter 8

Conclusion

We have proposed a video program understanding system. By our structure, we are primarily concerned with a temporal sequence of the high-level concepts, namely two kinds of events: replay scene and goal area scene. In the middle-level layer, six kinds of meaningful content shots are classified. In the low-level layer, some effective features are arranged. Given a video in a specific domain, we aim to extract the low-level features and interpret the input video in terms of high-level concepts.

The contribution of this research are:

(1) A certain shot classification is carried out after accurate shot change detection, which is powerful for the semantic analysis of highlight scene.

(2) The amount of computing time for extracting and managing necessary information from MPEG codec directly is significantly less than that for extracting features from each frame, and thus, our scheme makes it possible to analyze high-definition (1440×1080 resolution) soccer video in real-time.

The algorithms leaves much room for improvement and extension:

(1) There are other relevant low-level features that might provide complementary information and may help improve performance, such as camera motion, audio. etc;

(2) Other kinds of highlight scenes, such as object detectors, goal and whistle detection, can be integrated;

(3) Models that are more general and more capable for capturing interactions and temporal evolvement of features and scenes.

Thanks

First of all, I would like to express my gratitude to my supervisor, Prof. Hiroyoshi Morita for giving me the continuous guidance and invaluable encouragement throughout the whole work.

I am deeply grateful to Assist. Prof. Akiko Manada for giving invaluable comments and help for ISICO 2013 conference paper submission and presentation in Bali, Indonesia.

Finally, I wish to thank all the members of Morita Laboratory and Kasai Laboratory for their cooperation.

Refenrence

[1] Jiang Feihu, H.Morita, A.Manada " Semantic Analysis of Structured High-definition MPEG-2 Soccer Video Using Bayesian Network " ISICO 2013, Bali, Indonesia, pp.483-490, Dec. 2013

[2] Barry G.Haskell, Atul Puri, and N.Netravali, " Digital video: an introduction to MPEG-2 ".

[3] ISO/IEC 13818MPEG-2." Information Technology-Generic Coding of Moving Pictures and Associated Audio Information ".

[4] C.-L.Hung, H.-C.Shih " Semantic Analysis of soccer video Using Dynamic Bayesian Network ", IEEE, Trans. Multimedia,vol. 8, no. 4, pp. 749-76, August 2006.

[5] Y. Gong, L.T. Sin, C. Chuan, H. Zhang, and M. Sakauchi " Automatic parsing of TV soccer programs ", Proc. ICMCS '95, pp.167-174, May 1995.

[6] S.Aoki, H.Morita, Y.Aramata " Cut Detection in MPEG2 Compressed Data Using Macro Block Types ", CVIM, Japan, Vol. 46, May 1995.

[7] Ming Luo, Yu-Fei Ma, and Hong-Jiang Zhang, " Pyramidwise Structuring for Soccer Highlight Extraction ", IEEE, Trans. Multimedia vol. 2, pp. 945-949, Dec. 2003.

[8] M. H. Kolekar, K. Palaniappan, and S. Sengupta " A novel framework for semantic annotation of soccer sports video sequences ", IET Int. Conf. on Visual Media Production, pp.1-9, 2008.

[9] M. H. Kolekar, K. Palaniappan, and S. Sengupta. " Semantic event detection and classification in cricket video sequences ", IEEE Indian Conf. Computer Vision, Graphics and Image Processing, pp.382-389, 2008

[10] X. Sun, G. Jin, M. Huang, and G. Xu " Bayesian network based soccer video event detection and retrieval ", Multispectral Image Processing and Pattern Recognition, vol. 5286 pp.71-76, 2003.

[11] H. C. Shih and C. L. Huang, " MSN: Statistical understanding of broadcasted sports video using multilevel semantic network ", IEEE Trans. Broadcast, vol. 51, no. 4, pp. 449-459, 2005.

[12] J. Assfalg and M. Bertini, " Semantic annotation of soccer videos: automatic highlights identification ", Comput.Vis. Image Understand. vol. 91, pp.285-305, 2003.

[13] P. Xu, L. Xie, and S.-F. Chang, " Algorithms and system for segmentation and structure analysis in soccer video ", Proc. IEEE ICME, Tokyo, Japan, 2001, pp. 721-724.

[14] Adnan Darwiche, “ Modeling and reasoning with Bayesian network ”, 2009.

[15] Almond,R. “ Graphical Belief Modelling ”,London: Chapman and Hall, 1995.

[16] M.Bolic, “ Theory and Implementation of Particle Filters ”,University of Ottawa, Nov. 2004

[17] M.Bishop. “ Pttern Recognition and Machine Learning ”,Springer. 2006

[18] M.Hosaka, “ Detection of highlight scenes from MPEG2 compressed hi-ghdefinition soccer video ”, Master thesis,University of Electro-Communications (in Japanese), Mar. 2010.

[19] Bayy G.Haskell, AtulPuri, and ArunN,Nerravali, “ Digitalvideo: an introduction to MPEG-2 ”, 1996.

[20] G.Rebane, and J.Pearl,“ The recovery of causal poly-trees from statistical data ”,1987