MSN: Statistical Understanding of Broadcasted Baseball Video Using Multi-Level Semantic Network

Huang-Chia Shih and Chung-Lin Huang, *Senior Member, IEEE*

***Abstract*—The information processing of sports video yields valuable semantics for content delivery over narrowband networks. Traditional image/video processing is formulated in terms of low-level features describing image/video structure and intensity, while the high-level knowledge such as common sense and human perceptual knowledge are encoded in abstract and nongeometric representations. The management of semantic information in video becomes more and more difficult because of the large difference in representations, levels of knowledge, and abstract episodes. This paper proposes a semantic highlight detection scheme using a Multi-level Semantic Network (MSN) for baseball video interpretation. The probabilistic structure can be applied for highlight detection and shot classification. Satisfactory results will be shown to illustrate better performance compared with the traditional ones.**

***Index Terms*—Baseball video, Bayesian belief network, multi-level semantic network, spatio-temporal analysis, sport video, statistical modeling.**

I. Introduction

VARIOUS sports video programs have been broadcasted over different networks to large audiences. However, most of the sport programs are not consistently interesting, there are many stops and goes. Compared with other videos such as news and movies, sports videos have a well-defined content structure and domain rules. The valuable plays in sport videos generally occupy only a small portion of the whole content. For instance, in a baseball video, the highlights usually comprise less than 20% of entire program. It is necessary to mark the video program for ordinary or exciting paragraphs. Automatically extracting the content highlights is a challenging research topic, because it requires more high-level content analysis [1], [2].

Many multimedia communication mechanisms via internet have been proposed, most of them using the architecture proposed by the MPEG-7 [3], [4], i.e., a set of independent or related description. The MPEG-7 has tried to standardize the media access methods, which are automatic techniques to access the video based on its content. For example, the video indexing and retrieval [5] is a useful query tool to access the media, which includes automatic classification [6], summarization [7]–[9] and video understanding [10]. MPEG-7 has defined DS (description scheme) to describe the relationships between different variations of video sources to provide adaptive selection under different delivery platform and user preference conditions.

The video understanding research begins with the video indexing and retrieval techniques which focused on the paradigm of query-by-example (QBE) [11], [12]. However, the difficulty in such a system that supports semantic retrieval using keyword lies in the gap between low-level media features and high-level concepts. Recently, there have been some efforts to bridge the gap. Naphade *et al.* [13] proposed a novel probabilistic framework for semantic indexing and retrieval in digital video. They apply the factor graphical model to provide an efficient framework for video inference.

Vasconcelos and Lippman [14] introduced a pioneer work using the Bayesian architecture for video content characterization and analysis. It provides a potential tool for accessing and browsing video database on a semantic basis. The Bayesian Belief Network (BBN) is a directed acyclic graph, which is an effective knowledge representation and inference engine in artificial intelligence and expert system [15]. Domain-specific knowledge is characterized by the network structure in terms of the causal/relevant relationships between multiple variables, which are represented by the complicated joint probability distributions that can be simplified as a set of conditionally independent relationships.

Ferman *et al.* [16] employed Hidden Markov Model (HMM) and Bayesian Belief Networks (BBNs) at various stages to characterize the content domain and extract the relevant semantic information. Chang *et al.* [17] developed a classification scheme based on BBNs, which models the interaction of multiple classes at different levels of multi-media. A Bayesian network based method for semantic object extraction in images has been proposed for object detection and tracking [18] and semantic interpretation [19].

Chang *et al.* [20] described an approach for highlights extraction directly from four statistical models based on HMM. But the accuracy of highlight detection is very sensitive to the precision of shot classification phase. Xu *et al.* [21] also proposed a semantic analysis framework based on HMMs which is applied for sports game event detection. Amhet *et al.* [22] introduced a shot identification and goal detection framework for soccer game, and video shot segmentation into plays and breaks for a basketball game. Li *et al.* [23] provided an approach to model sports video using events. Their method detects some crucial scene cut points such as setup action leading to live action, threshold of close-up view, and replay. Subsequently, based

Manuscript received September 22, 2004; revised April 27, 2005.

The authors are with the Institute of Electrical Engineering, National Tsing Hua University, HsinChu, Taiwan, R.O.C. (e-mail: clhuang@ee.nthu.edu.tw).

Digital Object Identifier 10.1109/TBC.2005.854169

on the temporally relationship of scene shots, their method can fetch out the location of key events.

Taking advantage of prior implicit knowledge about sports videos, we develop a statistically linguistic modeling and precise video analyzers to interpret the sports videos. Our method not only extracts the semantic meanings but also detects the highlights. The so-called highlights in sports video are any kind of significant offending or defending events that amuses or draw the attention of the audience. Similar to [14], [17]–[19], we propose a multi-level Semantic Network (MSN) for highlight detection in the baseball video. Based on the low-level information and the inferring processes, the MSN will infer the high-level semantic of the video. In sports, the events are governed by the rules, they contain a recurring temporal structure. The rules of editing of sports videos have also been standardized. For example, in baseball videos, there are only a few recurrent views, such as pitching, close-up, home plate, battering, crowd etc. Our system is designated for understanding the baseball video, which can be easily modified for other sports videos. Different from the previous researches, this paper proposes a MSN that can be used to bridge this gap between the low-level videos and high-level semantic meaning through BBN inference engine.

We compare our MSN with [9] which discriminate different types of soccer video, using a state transition reasoning based on the causal relationship that is deterministic and presumed by predefined threshold. The deterministic threshold is noise-sensitive and low-level-feature-dependent, so that it suffers the false alarms. To avoid this problem, we adopt the probabilistic framework to conquer the uncertainty of source quality. Based on the Bayesian Belief Network (BBN), the low-level features and high-level features are more faithfully related which are compatible for different properties of video data. We can also identify various video semantics since MSN is used to extract and characterize the video events and the content significance instead of detecting specific scenes only.

The organization of this paper is as follows: Section II discusses the Statistical Bayesian network modeling. In Section III, the system framework is described in detail. Section IV describes the results of baseball simulations and examines the effectiveness. Finally, our conclusion is stated in Section V.

II. Statistical Modeling

Bayesian Belief Network (BBN) [25] has been proved to be an effective statistical model for knowledge representation and inference. It has been widely used in artificial intelligent and expert systems. BBN is a direct acyclic graph representing the causal/relevance dependencies between variables, which are represented with the conditional probabilities. In BBNs, variables are used to represent events and/or objects in the world. We may integrate prior information about dependencies between variable and propagate the impact of evidence on the probabilities of uncertain outcomes. There are three prototypes of connections between a random variable B and its two immediate neighbors in the paths, A and C. The three possibilities are shown in Fig. 1.

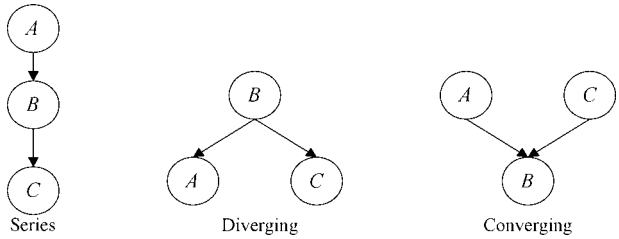


Fig. 1. The three connected prototypes.

The domain knowledge is used to construct the network. One of the major problems is how to treat the uncertainty in the specific knowledge domain. BBN process has been proved to be a powerful mechanism to model the uncertainty and the reasoning in terms of some quantity measurements, therefore we apply BBN to model our network. In BBN, direct arcs between variables represent conditional dependencies.

There are two types of computations performed with BBNs: *belief updating* and *belief revision*. Belief updating concerns the computation of probabilities over variables, while belief revision concerns finding the maximally probable global assignment. More formally, we assume W is the set of all variables in BBN, and e is the given evidence which represents a set of instantiations made on a subset of W. Any complete instantiations to all the variables in W which is consistent with e will be called an explanation or interpretation of e. Our goal is to find an explanation w^* such that $P(w^*|e) = \max\ P(w|e)$ where w^* is called the "most-probable explanation."

Belief updating on the other hand is interesting only in the marginal probabilities of a subset of variable given the evidence. It is applied to determine the best instantiation of a single variable given the evidence. According to *Bayes' rule*, the posterior probability can be expressed by the joint probability, which can be further expressed by conditional probability and prior probability as

$$P(S|E) = \frac{P(S,E)}{P(E)} = \frac{P(E|S)P(S)}{P(E)}$$

where S denotes semantic concept and E denotes evidence $P(E|S)P(S) = P(S,E)$, and where $P(S,E)$ is the probability of the joint event $S \cap E$. However, the probabilities should always be taken account of the conditioned antecedent node C, i.e., the node of category layer. The formula should be written as: $P(C|S,E)P(S|E) = P(C,S|E)$. Assume a Bayesian network for a set of variables $X = \{x_1, x_2, \ldots, x_n\}$, a set of P denotes local probability distributions associated with each variable. The network structure S is a directed acyclic graph. The nodes in S are in one-to-one correspondence with the variables X, x_i denotes both the variable and its corresponding node, and Pa_i denotes the parents of node x_i in S as well as the variables corresponding to those parents. Using the *chain rule*, we may express the joint probability distribution for $\mathbf{X}$ as $P(X) = P(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} P(x_i|\mathrm{Pa}_i)$. Therefore, a complicated joint probability distribution can be reduced to a set of conditional probability and a prior probability.

In the learning phase, to evaluate the class conditional density function (i.e., conditional probabilities, posterior probabilities), EM (Expectation-Maximization) algorithm [24] is used

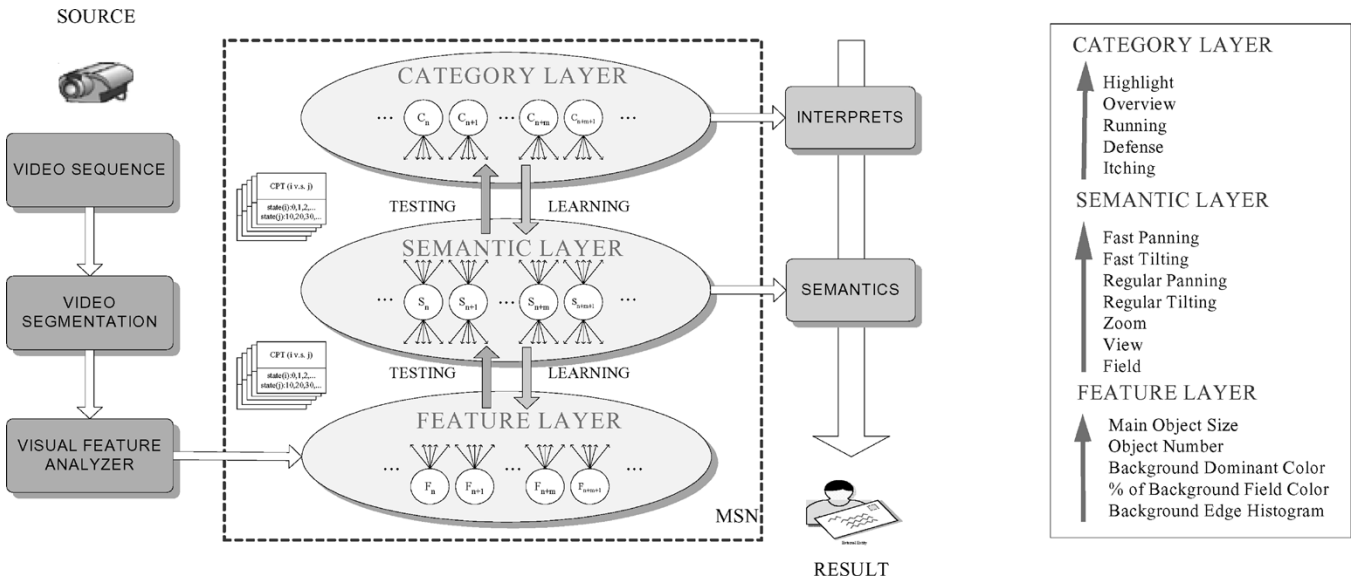


Fig. 2. Conceptual diagram of proposed understanding framework.

in training phase for batch learning [25]. Batch learning is an "off-line" learning, and it is used to generate the conditional probability tables in the knowledge database. Based on different training data, the EM-algorithm computes the conditional distribution probability for each node based on the Maximum Likelihood (ML) learning algorithm.

Having constructed and trained a BBN, we need to determine various probabilities of interest from the model by running inference procedure, which gives the observations and evidences (i.e., low-level media feature). Based on different applications, we can model a Bayesian Network to infer the semantic information of the test video sequence. It is important to determine the causal dependency relationships between the variables.

III. Proposed Sports Videos Understanding Framework

A. System Framework

Here, our system may detect five different events in baseball videos based on the inference of video descriptors and their inter-relevance which results in higher semantics meaning. First, the low-level video descriptors associated with the motion, texture, color, and object information of the video are provided. Second, with the video descriptors, MSN may infer certain high-level semantics. Similar to BBN, MSN is developed based on the causality of the low-level and mid-level descriptors in a specific domain, which contains rich spatial and temporal transitional structures of the video. MSN is a direct acyclic graph representing the causal/relevance dependencies between two descriptors represented by the conditional probabilities.

In training phase of MSN, given the join probability and priori probability of the descriptors, the system generates the posteriori probability associated to the linkage of the two descriptors in the network. Based on MSN, the system may interpret the semantic meanings of different events in the video. In this paper, we propose a multi-level semantic network (MSN) for semantic interpretation of the baseball game as shown in Fig. 2. MSN is a three level BBN structure which can be applied for the action categorization and highlight detection. Top-down learning process will assign the conditional probability table (CPTs) for the linkage between each node pair, whereas the bottom-up testing process will inference the video action based on the low-level features and semantic features.

In Fig. 2, the captured video sequences are segmented into shots and the appropriate coding units are analyzed by video analyzers to generate the low-level descriptor, such as *motion descriptors, object descriptors, texture descriptors, and color descriptors*. From MPEG video standard, we define the minimum coding unit as a Group of Pictures (GoP) which consists of I, B and P frames. These low-level descriptors can be treated as the input evidence to the MSN. The motion descriptors include vertical intercept *(VI)*, horizontal intercept *(HI)*, vertical sum of displacement vector $(VSDV)$, horizontal sum of displacement vector $(HSDV)$, vertical slope *(VS)* and horizontal slope *(HS)*, the object descriptors comprise object number *(ON)* and main object size *(MOS)*, the texture descriptors contain the background edge histogram *(BEH)*, and the color descriptors consist of background dominant color *(BDC)* and background field color percentage *(BFCP)*.

The low-level descriptor will be used to infer the existence of the mid-level descriptors such as *fast tilting, regular tilting, fast panning, regular panning, zooming, view* and *field*. The MSN is the statistical model-based classifier based on the BBN which infers certain higher-level descriptors from the lower-level descriptors. The MSN consists of three layers: category layer, semantic layer and feature layer. The category layer is related to the video types of the input sport video. The semantic layer supports few unobservable mid-level descriptors based on their relevance with the observable low-level descriptors in feature layer. The mid-level descriptors can further be used to infer the high-level descriptors which indicates the video category. The learning procedure is a top-down approach to evaluate the causality between two descriptors and the probability density of each descriptor. Thus, each descriptor is formulated as a random

1	-1
1	-1

1	1
-1	-1

$\sqrt{2}$	0
0	$-\sqrt{2}$

0	$\sqrt{2}$
$-\sqrt{2}$	0

2	-2
-2	2

Fig. 3. Filters for edge detection.

variable which may exist in certain states. Each state of the designated descriptor may have different priori probability. The number of states for each descriptor depends on its semantics definition. Responsibly, we can utilize the bottom-up understanding procedure to compute the results by using the evidence propagation procedure from the low-level descriptor in feature layer to the high-level descriptor in the semantic layer, indicating the video category of testing sequence.

B. Video Processing

The video processing first segments the video into various temporal "shots" which consist of a sequence of frames captured from a single camera that represents an event or continuous sequence of actions of the same objects, scene, and people. Each shot will be described in terms of a sequence of low-level and mid-level descriptors. There are many literatures regarding to the video shot segmentation [20]. The main problem of segmenting a video sequence into shots is the ability to distinguish between the scene breaks and the normal changes. These changes may be due to the motion of large objects or the motion of the camera (e.g., zooming and panning). Once the video shot is provided, the video processing may continue analyzing the video shot and generating a sequence of descriptor as follows.

1) Texture Descriptors: The texture information is described by the special correlation and local properties of a region. The texture descriptors are based on the analysis of the edge histogram. However, since the texture descriptor is used to describe the background rather than the foreground player, the texture descriptor is analyzing the Background Edge Histogram *(BEH)*. *BEH* captures the spatial distribution of background edges, which is also a good texture signature [29]. *BEH* is a random variable which has five different states corresponding to the four directions (e.g., vertical, horizontal, 45° diagonal, 135° diagonal), and isotropic (nonorientation specific) as shown in Fig. 3.

2) Color Descriptors: Color descriptors consist of Background Dominant Color *(BDC)* and Background Field Color Percentage *(BFCP)* which are important low-level descriptors for color image indexing. Here, analyze the color information in CIE-YUV color space. A dominant color and its percentage value will be calculated. Before analyzing the color feature, the moving object region must be segmented. To generate the color descriptors of the foreground region and background region, we analyze the foreground region to determine the color of moving objects, and then we explore the background region to estimate the *BDC* and *BFCP* of the video shots.

The dominant color is described by the peak value of each color component. The computation involves the determination of the peak index, i_{peak}, for each histogram. Then, we find an interval $[i_{\min}, i_{\max}]$ with $i_{\min} \leq i_{\text{peak}} \leq i_{\max}$, where $i_{\max}$ and $i_{\min}$ satisfy the conditions: $\mathrm{H}[i_{\min}] \geq k\mathrm{H}[i_{\text{peak}}]$, $\mathrm{H}[i_{\min}-1] \geq k\mathrm{H}[i_{\text{peak}}]$, $\mathrm{H}[i_{\max}] < k\mathrm{H}[i_{\text{peak}}]$, $\mathrm{H}[i_{\max}+1] < k\mathrm{H}[i_{\text{peak}}]$, with $0 \leq k \leq 1$, and H refers to the color histogram. These constrains define the minimum (maximum) index as the smallest (largest) index to the left (right), including the peak that has a predefined number of pixels. In our implementation, we set $k = 0.2$.

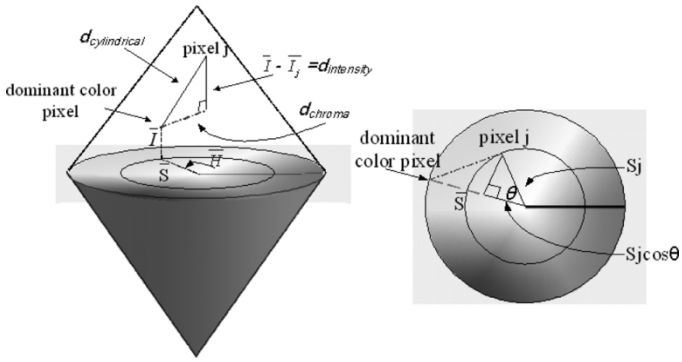


Fig. 4. The dominant color and the color of pixel j in HSI space.

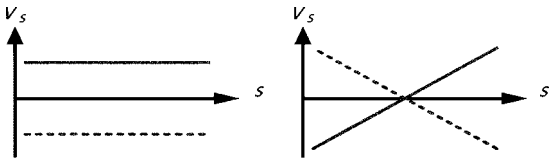
We convert the peak of each color component in RGB to HSI as $\overline{H}$, $\overline{S}$, $\overline{I}$. Field color pixels in each frame are detected by finding the distance of each pixel to the peak color (i.e., $d_{\text{cylindrical}}$) by the robust cylindrical metric. We define dintensity $d_{\text{intensity}} = |I_j - \overline{I}|$ $d_{\text{chroma}}(j) = \sqrt{(S_j)^2 + (\overline{S})^2 - 2S_j\overline{S}\cos(\theta(j))}$, $d_{\text{cylindrical}}(j) = \sqrt{(d_{\text{intensity}}(j))^2 + (d_{\text{chroma}}(j))^2}$, $\Omega(j) = |\overline{H} - H_j|$, and $\theta(\mathrm{j}) = \Omega(j)$ if $\Omega(j) \leq 180°$ else $\theta(j) = 360° - \Omega(j)$. As shown in Fig. 4, we assign the pixel to the dominant color region if it satisfies the constraint $d_{\text{cylindrical}} < T_{\text{color}}$ where T_{color} is a pre-defined threshold which is video dependent. For example, We may find the dominant color (i.e., green) in the baseball video. Fig. 5 illustrates the results of dominant color region detection for two views. There are different black pixels (dominant color) percentages in Fig. 5(a) and Fig. 5(b).

3) Motion Descriptors: The motion descriptors contain vertical intercept *(VI)*, horizontal intercept *(HI)*, vertical sum of displacement vector $(VSDV)$, horizontal sum of displacement vector $(HSDV)$, vertical slope *(VS)* and horizontal slope *(HS)* which can be used to determine three kinds of camera motion information: (1) zooming, (2) panning, and (3) tilting. To find the camera motion for video sequence, we replace the time consuming calculation of 2-dimensional $m \times n$ picture elements with that of two one-dimensional vectors by projecting the luminance in vertical and horizontal direction to generate two projection files respectively. There are two steps for motion analysis: (1) finding the displacement vector that makes the minimum *sum of absolute difference* (SAD) values from one-dimension intensity projection profile; (2) using linear regression to obtain the *Displacement Characteristic (DC)* curve as shown in Fig. 6.

Assume image frame is $W \times H$ pixels. Let f_x be a 1-D horizontal intensity projection profile and f_y denote the projection profile in vertical direction. Similar to [26], we divide the projection profile into small slices, each one with width N. Assume two consecutive frames a and b, we take a slice of the projection profile of frame a and sliding it over the projection profile

(a) (b)

Fig. 5. Results of dominant color region detection.



(a) (b)

Fig. 6. The displacement characteristic curve. (a) Panning/tilting to the left/down (solid line) and to the right/up (dotted line). (b) Zoom in (solid line) and zoom out (dotted line).

of frame b, and calculate the SAD value with a shifted value v as follows

$$v_s = \arg \min_{v=-\frac{N}{2}\sim\frac{N}{2}} \{SAD(c_0, v)\}$$

$$SAD(c_0, v) = \sum_{i=1}^{N} |f_a(i) - f_b(i+v)|$$

where c_o is the center position index of the slice taken from the projection profile of frame a, v is displacement value. At different location x, we have a corresponding slice $f(x)$, $x = 1, 2, \ldots W$ of f_a and f_b as follows

$$f(x) = f_x(i + sN) \quad \text{with } i = 1, 2, \ldots, N$$

where s denoted the slice number, $s = 0, 1, \ldots, S-1$, and $S = W/N$.

Here, we show four scenarios for the four kinds global motion in Fig. 7(a)–(d), each one includes two consecutive frames, the other two figures corresponding to two horizontal projection profiles, and the displacement characteristic curve. The left one denotes the previous frame, and the right one represents the current frame.

Fig. 7(a) are rightward panning figures of which the projection profile moves to left side along x direction. Thus, the displacement characteristic curve will similar to the dotted line of Fig. 6(a). Fig. 7(b) are leftward panning figures of which the horizontal projection profile will be smoothly shifted to right and the one-dimensional characteristic curves will be denoted as the solid line of Fig. 6(a). Fig. 7(c) are zoom-in figures of which provides the highlighted scenes to viewers. At location further away from the frame center, we find there is a larger displacement. Moreover, the left-half projection profile moves leftward indicating left panning, and the other half moves rightward indicating right panning. Hence, the displacement characteristic curve is a solid line as shown in Fig. 6(b). Fig. 7(d) are the zoom-out figures of which the left-half projection profile moves rightward showing right panning and the other half moves leftward indicating left panning. The displacement curve is a dotted cross line as shown in Fig. 6(b).

To generate the camera motion descriptor, we need to analyze several consecutive frames. The video processing is applied for a sequence of frame called a Group of Pictures (GoP). Here, the motion descriptor can be obtained by averaging *HS, VS, HI, VI*, $HSDV$, and $VSDV$ in one GoP (with 9–15 consecutive frames). Next we define $HSDV$ and $VSDV$ of each frame as

$$HSDV = \sum_{p=1}^{\frac{(W-I)}{n_0}} DV_p^{\text{horizontal}} \quad VSDV = \sum_{p=1}^{\frac{(H-I)}{n_0}} DV_p^{\text{vertical}}$$

where I is the shifting index; n_0 indicates the slice windows size, the frame size is $H \times W$.

4) Moving Object Descriptors: The moving object descriptors consist of the Object Number *(ON)* and Main Object Size *(MOS)* in the video sequence. First of all, we need to separate the moving object region and background region to generate the motion object descriptor. Here, we assume that the moving objects in complex background are somehow identifiable by their edge boundaries. Usually, the edge information is not applicable for our system because most of the edge information is redundant.

We assume that the objects are moving, and the background is complex but stationary. Using the frame difference, we can capture the motion information. By accumulating the motion information of the moving objects in several consecutive frames, we can locate the moving pixels more accurately. Similar to [28], we apply the edge detection and thresholding method to extract the edge features in the scene. Then we find the motion information by using the motion accumulator. Having extracted the edge (spatial) and the motion (temporal) information, we can combine these two kinds of information to locate the object by using "AND" operation. We use the morphology filtering techniques, such as dilation and erosion, and the region growing to remove the noise in the binary image. The step-by-step results of our segmentation algorithm are shown in Figs. 8–10.

C. Semantic Information Found in a GoP

The descriptors generated by the video analyzers (except the motion analyzer) for each coding unit (i.e. GoP) are defined as follows:

$$z_v^k = \text{medium}\,\{v(k+i) | 0 \le i \le N-1\}$$

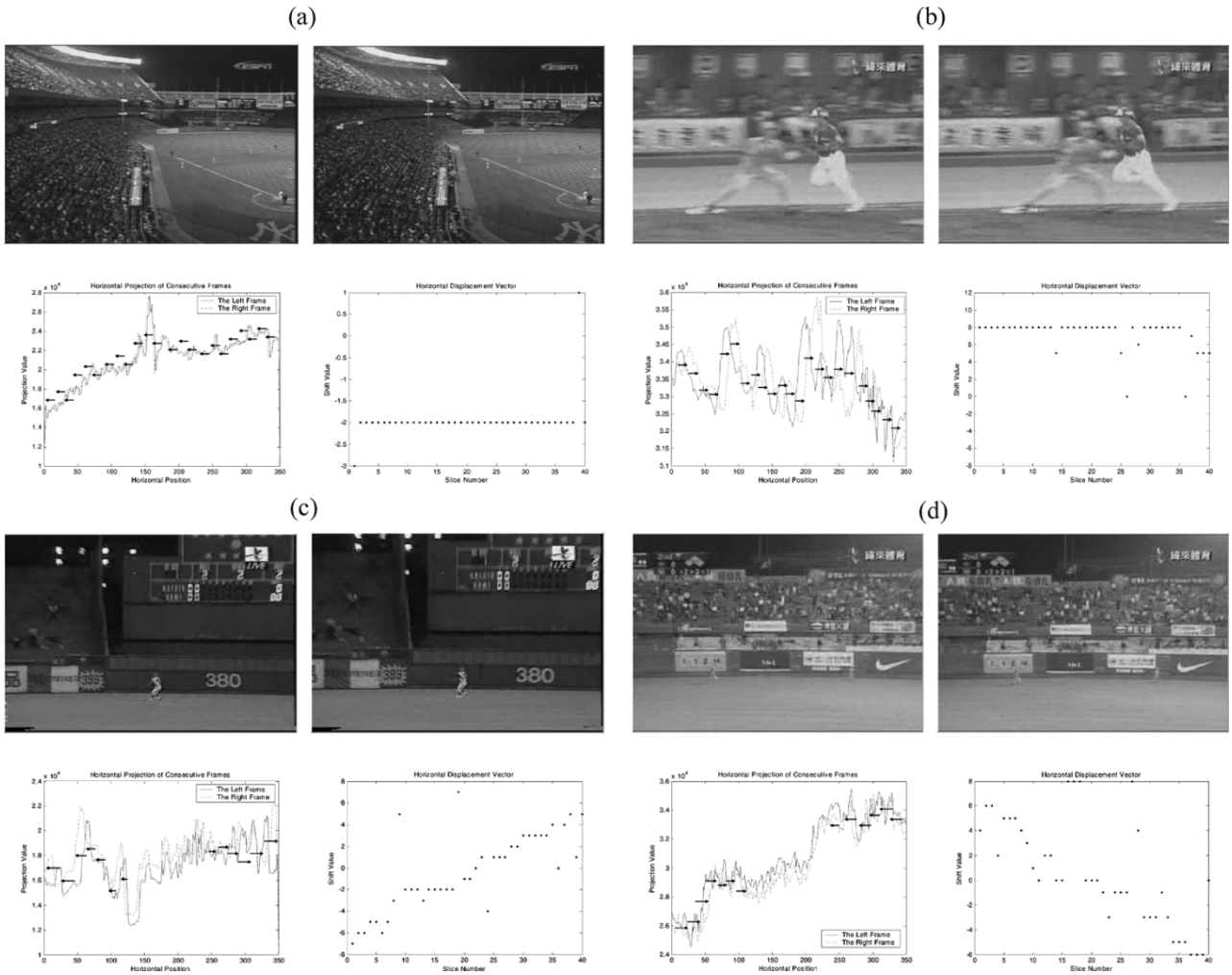


Fig. 7. Four scenarios of global motion. (a) Scenario 1: right planning; (b) scenario 2: left planning; (c) scenario 3: zoom in; (d) scenario 4: zoom out.

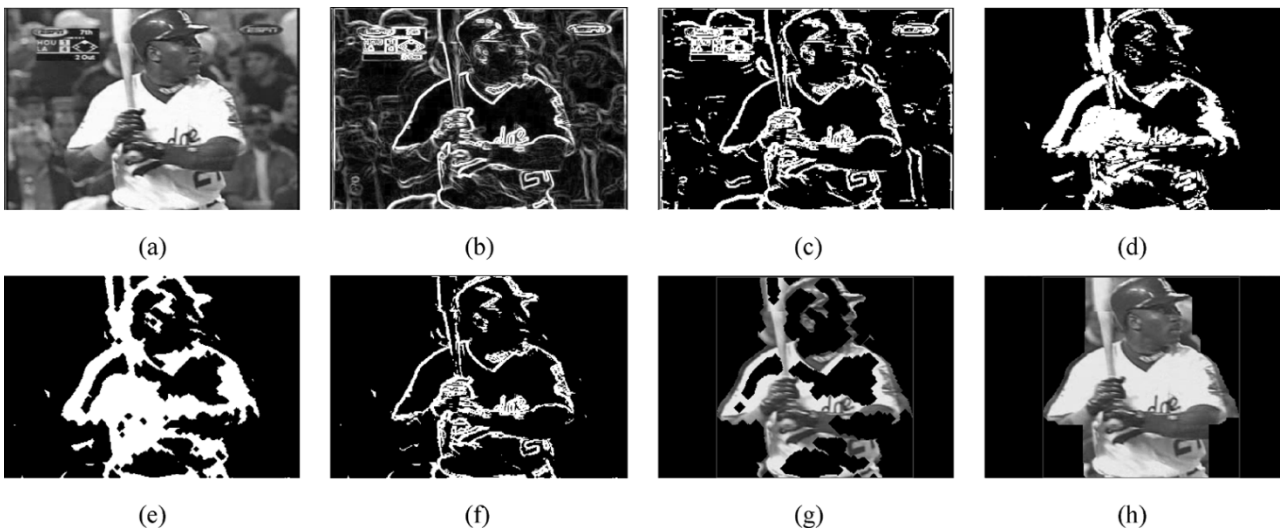


Fig. 8. A cluttered background *Batter* shot. (a) An original video shot; (b) edge image; (c) result after the Otsu thresholding; (d) frame of accumulated image; (e) result of the closing operation on (d); (f) result of the AND operator for (c) and (e); (g) result of the noise removal and closing operator on (f); (h) after region glowing image.

where $\upsilon(k)$ is *ON, MOS, BEH, BDC* or *BFCP* obtained from the k_{th} frame, N is number of frames in a GoP, and *medium* operation takes the middle value of a set of $\{\upsilon(k)\}$ generated in a GoP.

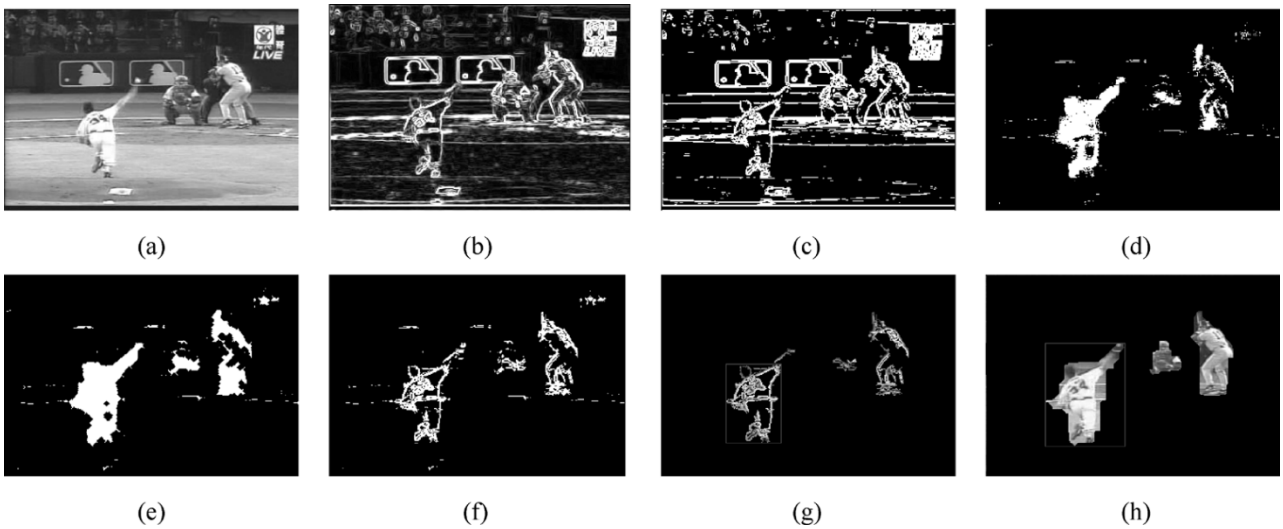


Fig. 9. A cluttered background *Pitching* shot. (a) An original video shot; (b) edge image; (c) result after the Otsu thresholding; (d) frame of accumulated image; (e) result of the closing operation on (d); (f) result of the AND operator for (c) and (e); (g) result of the noise removal and closing operator on (f); (h) after region glowing image.

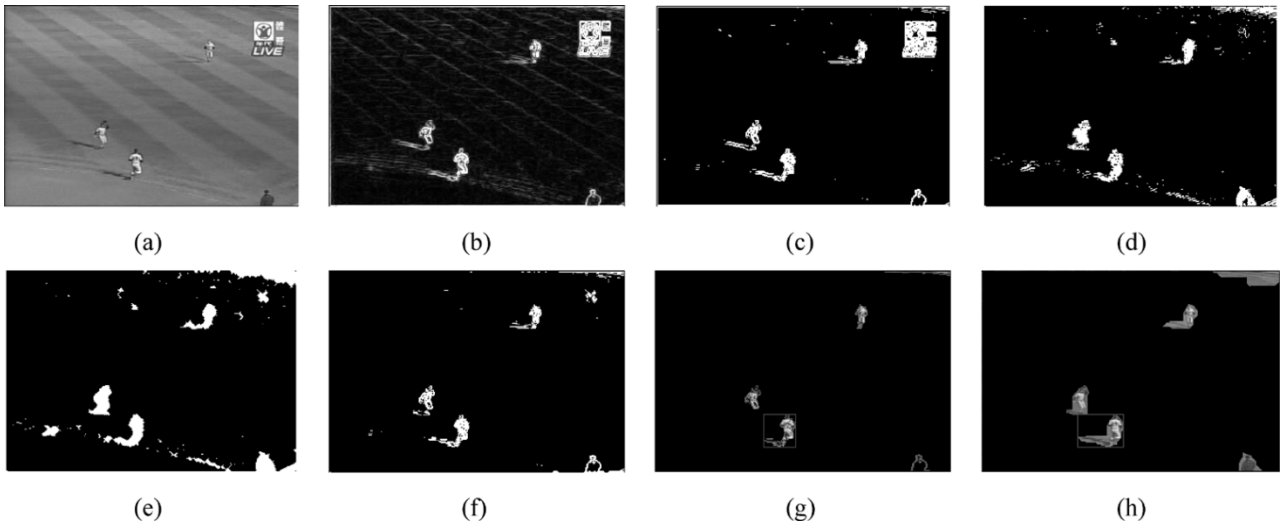


Fig. 10. A zooming long-view *Defense* shot. (a) An original video shot; (b) edge image; (c) result after the Otsu thresholding; (d) frame of accumulated image; (e) result of the closing operation on (d); (f) result of the AND operator for (c) and (e); (g) result of the noise removal and closing operator on (f); (h) after region glowing image.

IV. Experimental Results

In the experiments, the semantics are generated for a GoP (adopts 9∼15 consecutive frames) rather than a video shot. The baseball video is selected from six different baseball TV programs. Each video shot consists of a sequence of image frames indicating different semantic contents, moreover, each shot may consist of different number of image frames. The size of each full color image frame is 352 × 240, and its frame rate is 30 frames per second.

The image pre-processing, low-level feature extraction procedure and moving object segmentation are based on the decompressed MPEG video. The blocking artifacts of the input video may reduce the efficiency of the low-level information extraction. Here, we applied our method to interpret the baseball video. In part A, we will introduce the results of semantic extraction for baseball video. In part B, we will show the performance of highlight detection. Finally, the implementation and results of video interpretation will be represented in part C. In parts A and B, to evaluate our results, we define precision and recall [27] as precision $= (\mathrm{R} \cap \mathrm{C})/\mathrm{R}$ and recall $= (\mathrm{R} \cap \mathrm{C})/\mathrm{C}$, where R denotes a set of coding units recognized as highlight or corresponding mid-level semantic units, and C represents the relevant set of correct units. In part C, we use accuracy rate and false alarm to evaluate our results.

In our MSN, the states of the nodes were specified differently. Every descriptor consists of several discrete states. The video analyzers generate various descriptors and then assign each descriptor a certain state. Some mid-level descriptors and high-level descriptors in the category layer are assigned one of the two states, such as "yes" or "no." The semantic *View* descriptor, may be assigned "close-up," "mid," or "distant, "and the *Field* descriptor is assigned "in," "out," or "SZ" which represent "infield," "outfield," and "strike-zone" respectively. The details of node properties are shown in Table I.

TABLE I
DESCRIPTOR PROPERTIES

Descriptor	Type	States	# of States
MOS	number	(0~9)	10
ON	number	(0~4)	5
BDC	number	(0~255)	256
BFCP	number	(0~9)	10
BEH	number	(0~4)	5
Highlight	label	("yes" , "no")	2
Overview	label	("yes" , "no")	2
Running	label	("yes" , "no")	2
Defense	label	("yes" , "no")	2
Pitching	label	("yes" , "no")	2
Player	label	("yes" , "no")	2
F tilt	label	("yes" , "no")	2
R tilt	label	("yes" , "no")	2
F pan	label	("yes" , "no")	2
R pan	label	("yes" , "no")	2
Zoom	label	("yes" , "no")	2
View	label	("close-up", "mid", "distant")	3
Field	label	("SZ", "in", "out")	3
HS, VS	number	(-20°~0°~20°)	41
HI, VI	number	(0~16)	17
HSDV, VSDV	number	(0~20)	21

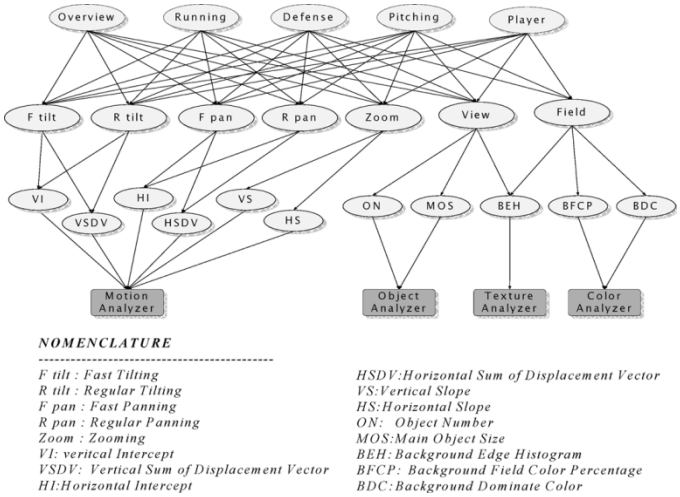


Fig. 11. The MSN of the event interpretation.

A. Semantic Extraction

In Fig. 11, the root nodes represent the certainty of the five different categories: *Overview, Running, Defense, Pitching* and *Player* which infer the high-level concept of the input video sequence. Each input video may activate more than one root node (with high certainty after BBN inference). Each root node is connected to several mid-level nodes representing the semantic concepts.

There are seven mid-level semantic concepts such as *fast tilting, regular tilting, fast panning, regular panning, zooming, view* and *field* (shows in Fig. 11). Descriptor *View* is modeled by connecting several low-level descriptors such as *Object-Number (ON), Main-Object-Size (MOS)* and *Background Edge Histogram (BEH)*. The node "*view*" consists of three states: *distant view, medium view* or *close-up view*, whereas the node "*Field*" is also composed of three states: *infield, outfield* or *striking zone*. In motion analyzer, the states of other mid-level semantics such as *Zooming, Fast panning, Regular panning, Fast tilting, Regular tilting* are either *True* or *False*. A large *ON* strongly supports the possibility that descriptor *View* indicates the distant-view rather than the close-up view. Similarly, a large *MOS* strongly suggests a close-up view rather than a distant-view.

The basic level of the framework consists of several different low-level image analyzers. Based on the extracted low-level information, the MSN may infer the highest-level concept embedded in the root node. Each input video may activate more than one root node (with high certainty after BBN inference). Each root node is connected to several mid-level nodes representing the semantic concepts.

Here we assume that each coding unit is a GoP (9∼15 consecutive frames). In the experiments, we have selected about 1100 video shots from six baseball TV programs of Major League Baseball (MLB), three baseball TV programs of Chinese Professional Baseball League (CPBL), three baseball TV programs of Japan Professional Baseball League (JPBL) and four baseball games of World Cup 2002. Totally, we have 3200 GoPs for training and 1006 GoPs for testing.

In the beginning, the states of each descriptor of the MSN were specified manually. The results of the mid-level semantic descriptors are shows in Table II.

TABLE II
THE RESULTS OF THE MID-LEVEL SEMANTICS

	Field	View	Zooming	Fast panning	Regular panning	Fast tilting	Regular tilting
Precision	79.6%	76.7%	86.6%	89.4%	75.8%	99.4%	87.0%
Recall	78.8%	76.7%	86.2%	89.4%	75.8%	99.0%	86.5%

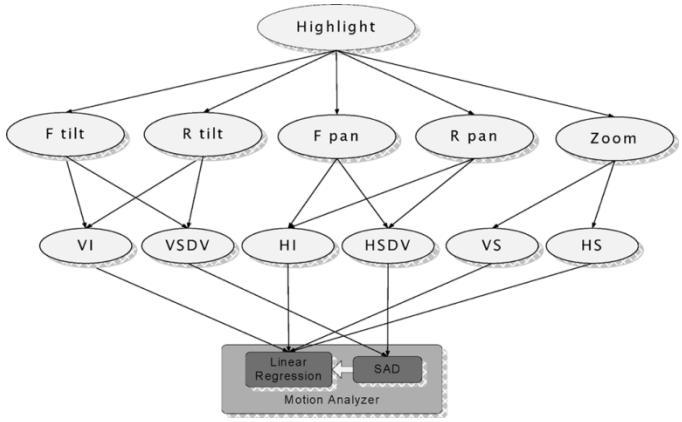


Fig. 12. The MSN for highlight detection.

TABLE III
THE RESULTS OF HIGHLIGHT DETECTION

	Highlights	Non-highlights
# of Training Unit / # of Testing Unit	523/389	477/430
# of Correct/Reject/Mismatch	337/0/52	379/2/49
Precision	86.6%	88.6%
Recall	86.6%	88.1%

B. Highlight Detection

The highlights indicate certain kinds of significant offending or defending events that may indicate the viewer's preference in the sports video program. A baseball program usually comprises almost 80% nonhighlight scenes. We need to develop a mechanism that may automatically detect the highlight events. We develop an MSN for highlight detection as illustrated in Fig. 12. The highlight will be found in the scenes of defense, hits, grounder line

(a) (b) (c) (d) (e)

Fig. 13. Five video shots with different semantics. (a) Overview; (b) runner snapshot; (c) defending view; (d) pitching view; (e) batter (player) snapshot.

TABLE IV
THE RESULTS OF EVENT INTERPRETATION

Video Category	Overview	Running	Defense	Pitching	Player
Accuracy	96.23%	97.20%	96.55%	98.70%	98.70%
False Alarm	26.79%	41.49%	28.07%	38.92%	43.60%

and cloud buster, but not in the scenes of pitching, player close-up and overview. Using MSN, we may easily detect the highlights as shown in Table III. We develop the MSN for highlight detection based on the motion information. In consumer video broadcasting, if an extraordinary event of particular occasion or specific circumstance occurs, the camera panning will move fast and the scene will change rapidly to bring the audience attention.

C. Video Interpretation

The top-level of MSN consists of the root nodes representing the certainty of the five different categories: *Overview, Running, Defense, Pitching* and *Player* (see Fig. 13). The upper level of the multi-level network may infer the highest-level concept of the input video sequence. Each input video may be interpreted by more than one root node (with high certainty after BBN inference). The linkage characteristics of the MSN are also manually determined based-on their relationships, and the probabilities of these links can be obtained by the BBN training procedure [10], [20].

The mid-level semantic, which is considered as indirect aggregations of lower level information, may also be represented by the MSN provided that they can be inferred to interpret the semantic of input video. Fig. 11 illustrates the MSN for video event interpretation of the baseball video. To test our system, all different type video clips are randomly mixed for testing. A video clip is detected as the right category if the probability of the state "yes" of the corresponding node is larger than 0.5. However, we may also find that the probability of state "yes" of the other nodes which may also be larger than 0.5, it indicates a false alarm. The testing results of detection accuracy and false alarm rate of each category are shown in Table IV. Fig. 14 illustrates the interface of video understanding results for the scene of defense. The green circle denotes that the recognized category is in the first place, red circle indicates that it is in the second place. The confidence score is measured by normalizing the post-probability of corresponding node.

The performance of our system has been shown in Table IV. There are five video categories, but it is easy to extend the system to detect more classes based on these mid-level semantic features. We may demonstrate the performance of event interpretation by the confusing matrix as shown in Table V. The false alarm occurs due to the following reasons:

1) Defense scenes are often misinterpreted as the cases of *Running* and *Overview*. Both of them are sub-network of *Defense*. For *Running* video, false alarm occurs if the *Zooming* semantic is mis-identified or the *Field* is found as a outfield. For *Overview* shot, the false alarm happens when *Zooming* and *Fast panning* feature is not accurate.
2) Player scenes are often misidentified as the cases of *Pitching* and *Running*. Because the main feature to be used to discriminate *Player* from *Pitching* is the *View* semantic. Sometimes, the pitcher is too close to the batter and the catcher, they will be merged together, hence the false alarm occurs. For *Running* shot, the false alarm will occur when *Fast panning* and *Regular panning* are misidentified.
3) Pitching scenes are often misidentified as the *Overview* scenes. Because *Overview* is a sub-network of *Pitching* case. When the *Regular panning* is not accurate or *Field* is mistaken as infield, the false alarm will occur.
4) Overview scenes are often misinterpreted as the *Running* scenes. They have similar semantic variables. False alarm occurs if *View* and *Fast panning* are miss-interpreted.

Fig. 14. The interface of video interpretation.

TABLE V
THE CONFUSING MATRIX OF EVENT INTERPRETATION

Recognized category (GoP) / Input category	# testing data/ # training data	Player	Defense	Overview	Pitching	Running
Player	77/1000	76	22	27	75	47
Defense	116/700	5	112	10	6	43
Overview	106/200	19	54	102	11	64
Pitching	77/1000	66	6	34	76	2
Running	107/300	87	21	30	66	104

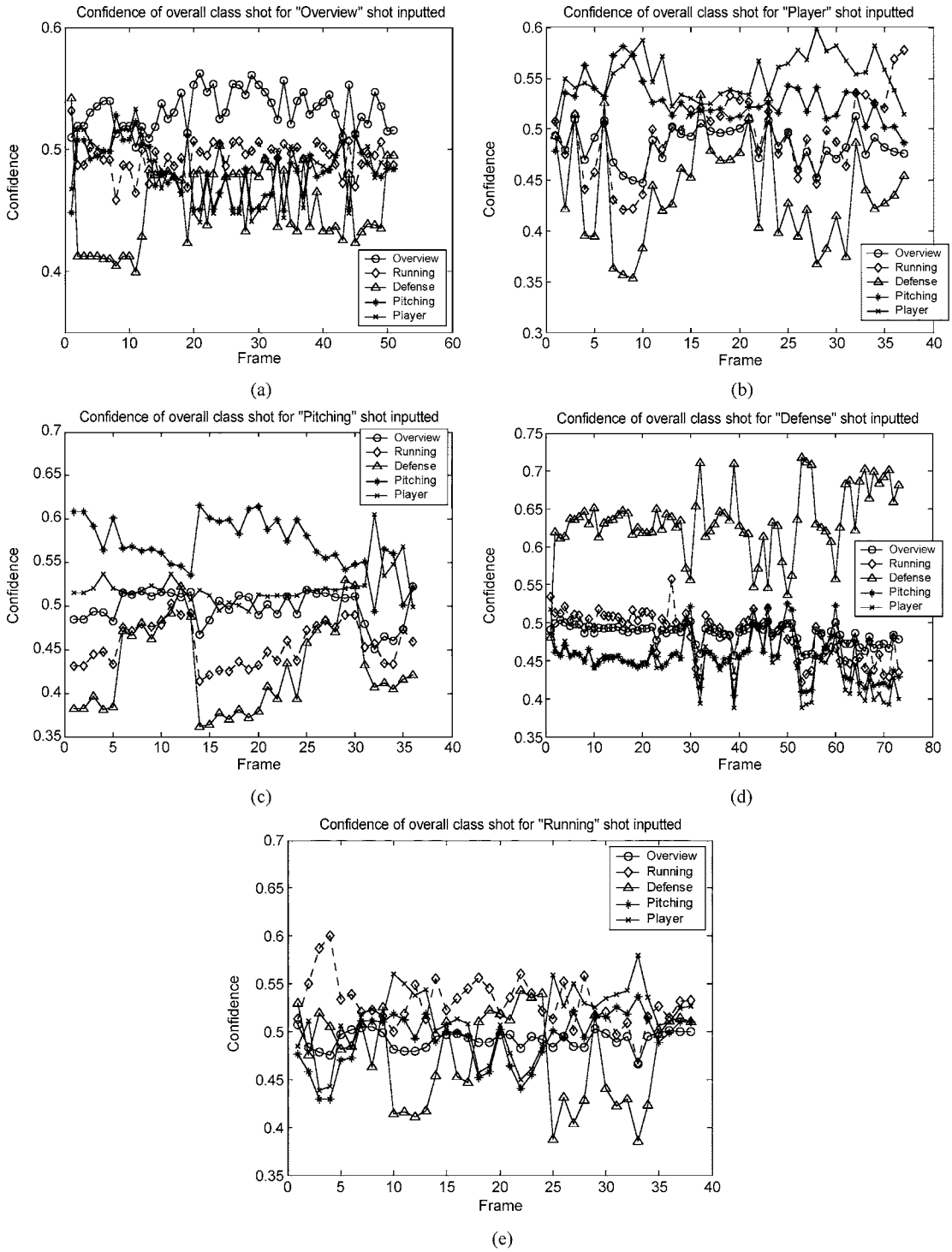


Fig. 15. The confidence score generated by proposed MSN for different categories video shot. (a) Overview, (b) player, (c) pitching, (d) defense, (e) running.

5) Running scenes are often misidentified as the cases of *Overview, Defense* and *Player*. The reason for the false alarm of *Overview* and *Defense* is similar to the above reasons. In *Player* video shot, sometimes, the batter performs fast actions such as waving his bat, however it is detected as a panning motion.

Finally, Fig. 15 shows the confidence score generated by proposed MSN when the input shot (randomly select) is "overview," "player," "pitching," "defense," or "running" shot individually. The correct BBN will generate the highest confidence score. After BBN inference, the system will inspect the confidence score of root node. If the confidence score of the other root node is larger than 0.5, the false alarm occurs. If we classify the video clip into the most likely category due to the corresponding root node has the highest confidence score, and the result is acceptable. For instance, in Fig. 15(d), the defense video shot is analyzed, and it is classified into the right category based on the most likely categorization scheme.

V. Conclusion

This paper proposes a video understanding scheme for baseball program using MSN. This proposed technique could be extended to more general interesting sports programs. Experiments show that the results of semantic extraction and highlight detection are satisfactory. Actually, The highlight detection based on low-level features has three limitations: (1) Memory requirements: Because the large amounts of video information needs to be processed simultaneously. (2) Short latency time: To execute the evidence propagation procedure, it needs to be real-time process. (3) High-level knowledge such as common sense and human perceptual information cannot be obtained based on the low-level features only. Our future studies will focus on producing a dynamic modeling to handle temporal time-slice transition.

References

[1] N. Dimitrova, H. J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of video-content analysis and retrieval," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 42–55, Jul.–Sep. 2002.

[2] U. Hahn and I. Mani, "The challenges of automatic summarization," *Computer*, vol. 33, no. 11, pp. 29–36, Nov. 2000.

[3] *MPEG-7: Overview (version 8)*, ISO/IEC, JTC1/SC29/WG11, Jul. 2002. N4980.

[4] S.-F. Chang, T. Sikora, and A. Puri, "Overview of the MPEG-7 standard," *IEEE Trans. Circuits Syst. Video Technol.*, Jun. 2001. special issue on MPEG-7.

[5] M. Abdel-Mottaleb and S. Krishnamachari, "Multimedia descriptions based on MPEG-7 extraction and applications," *IEEE Trans. Multimedia*, vol. 6, no. 3, pp. 459–468, Jun. 2004.

[6] J. Fan, A. K. Elmagarmid, X. Zhu, W. G. Aref, and L. Wu, "ClassView: hierarchical video shot classification, indexing, and accessing," *IEEE Trans. Multimedia*, vol. 6, no. 1, pp. 70–86, Feb. 2004.

[7] B. L. Tseng, C. Y. Lin, and J. R. Smith, "Video personalization and summarization system," in *IEEE Workshop on Multimedia Signal Processing*, Dec. 9–11, 2002, pp. 424–427.

[8] J. H. Lee, G. G. Lee, and W. Y. Kim, "Automatic video summarizing tool using MPEG-7 descriptors for personal video recorder," *IEEE Trans. Consum. Electron.*, vol. 49, no. 3, pp. 742–749, Aug. 2003.

[9] E. Ahmet, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 796–807, Jul. 2003.

[10] H. C. Shih and C. L. Huang, "A semantic network modeling for understanding baseball video," in *Proc. IEEE-ICASSP 2003*, Hong-Kong, Apr. 2003.

[11] D. Zhong and S. F. Chang, "Spatio-temporal video search using the object-based video representation," in *Proc. IEEE-ICIP*, Oct. 1997.

[12] H. Zhang, A. Wang, and Y. Altunbasak, "Content-based video retrieval and compression," in *Proc. IEEE-ICIP*, Oct. 1997.

[13] M. R. Naphade, I. Kozintsev, and T. S. Huang, "A factor graph frame work for semantic indexing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 1, pp. 40–52, Jan. 2002.

[14] N. Vasconcelos and A. Lippman, "Bayesian modeling of video editing and structure: semantic features for video summarization and browsing," in *Proc. IEEE-ICIP*, Chicago, 1998.

[15] F. V. Jensen, K. G. Olesen, and S. K. Anderson, "An algebra of bayesian belief universes for knowledge-based systems," *Networks*, vol. 20, pp. 637–659, 1990.

[16] A. M. Ferman and A. M. Tekalp, "Probabilistic analysis and extraction of video content," in *Proc. IEEE-ICIP*, Tokyo, Japan, Oct. 1999.

[17] S. F. Chang and H. Sundaram, "Structural and semantic analysis of video," in *Proc. IEEE-ICIP*, Vancouver, Sep. 2000.

[18] J. Luo, A. E. Savakis, S. P. Etz, and A. Singhal, "On the application of bayes Networks to semantic understanding of consumer photographs," in *Proc. IEEE-ICIP*, Vancouver, Sep. 2000.

[19] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Extraction of semantic description of event using bayseian network," in *Proc. IEEE-ICIP*, 2001.

[20] P. Chang, M. Han, and Y. Gong, "Extract highlights from baseball game video with hidden markov models," in *Proc. IEEE-ICIP*, 2002.

[21] G. Xu, Y. F. Ma, H. J. Zhang, and S. Yang, "A HMM based semantic analysis framework for sports game event detection," in *Proc. IEEE-ICIP*, 2003.

[22] E. Ahmet and A. M. Tekalp, "Shot type classification by dominant color for sports video segmentation and summarization," in *Proc. IEEE-ICASSP*, Hong-Kong, Apr. 2003.

[23] B. Li, H. Pan, and I. Sezan, "A general framework for sports video summarization with its application to soccer," in *Proc. IEEE-ICASSP*, Hong-Kong, Apr. 2003.

[24] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, Nov. 1996.

[25] F. V. Jensen, *An Introduction to Bayesian Networks*: Springer-Verlag, 1996.

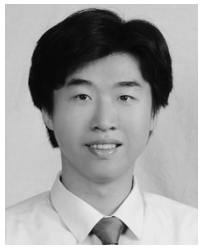
[26] M. K. Kim, E. Kim, D. Shim, S. L. Jang, and G. Kim, "An efficient global motion characterization methods for image processing application," *IEEE Trans. Consum. Electron.*, vol. 43, no. 4, pp. 1010–1018, Nov. 1997.

[27] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, 1st ed: Addison-Wesley, 1999.

[28] C. L. Huang and S. H. Jeng, "A model-based hand gesture recognition system," *Machine Vision and Appl.*, vol. 12, pp. 243–258, 2001.

[29] B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, June 2001.

[30] Y. Li, S. Narayanan, and C.-C. Jay Kuo, "Content-based movie analysis and indexing based on audio visual cues," *IEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 8, Aug. 2004.

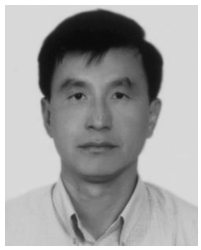


Huang-Chia Shih was born in Changhua, Taiwan, in 1978. He received the B.Sc. degree with the highest honors in electronic engineering from the National Taipei University of Technology, Taipei, Taiwan, in 2000 and the M.S. degree in electrical engineering from the National Tsing Hua University, Hsinchu, Taiwan, in 2002. He is currently pursuing the Ph.D. degree in electrical engineering at the University of Tsing Hua, Hsinchu, Taiwan.

His research interests are content-based video summarization, video indexing and retrieval, object-based video representations, applications of statistical models in multimedia processing, and model based human motion capturing and recognition. During Summer 2002, he was a Summer Intern at Computer & Communications Research Labs, Industrial Technology Research Institute, Taiwan.

Mr. Shih has received several awards and prizes, including the Excellent Student in the field of engineering on the national level from The Chinese Institute of Engineers, in 2000. Awards of the superiority young on college level from China Youth Corps, in 2000. He also election as the unique Taiwan delegate of the Dragon 100 Young Chinese Leaders Forum held in September 2004 in Hong Kong and Beijing. From 1995 to 2005, he obtained several prizes from the NTUT, the scholarship from the Chung Hwa Rotary Educational Foundation, the scholarship from ASUS, the scholarship from Taiwan Power Company, and so on. He has also served on the program committee of several international conferences and workshops.

Mr. Shih is a student member of the IEEE Signal Processing Society and the IEEE Broadcast Technology Society.



Chung-Lin Huang received the B.S. degree in nuclear engineering from the National Tsing-Hua University, Hsin-Chu, Taiwan, ROC, in 1977, and the MS. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, ROC, in 1979 respectively. He obtained the Ph.D. degree in electrical engineering from the University of Florida, Gainesville, FL, USA, in 1987. From 1987 to 1988, he worked for the Unisys Co., Orange County, CA, USA, as a project engineer. Since August 1988 he has been with the Electrical Engineering Department, National Tsing-Hua University, Hsin-Chu, Taiwan, ROC. Currently, he is a professor in the same department. In 1993 and 1994, he had received the Distinguished Research Awards from the National Science Council, Taiwan, ROC. In Nov. 1993, he received the Best Paper Award from the ACCV, Osaka, Japan, and in Aug. 1996, he received the Best Paper Award from the CVGIP Society, Taiwan, ROC. In Dec. 1997, he received the Best Paper Award from the IEEE ISMIP Conference held Academia Sinica, Taipei. In 2002, he received the Best Paper Annual Award from the Journal of Information Science and Engineering, Academia Sinica, Taiwan. His research interests are in the area of image processing, computer vision, and visual communication. Dr. Huang is a senior member of IEEE.