Content Extraction and Interpretation of Superimposed Captions for Broadcasted Sports Videos

Huang-Chia Shih and Chung-Lin Huang

***Abstract*—This paper illustrates how to interpret the superimposed caption box (SCB) in broadcasted sports videos of which the SCB template is presumably not given as a priori. The embedded captions in sports video programs represent digested key information of the video content. Most of the previous studies assume that the SCB template and the character bitmaps are known. The major contributions of this paper are (1) caption template extraction and identification, (2) symbol extraction and modeling, and (3) semantic interpretation of the identified captions and symbols. Experimental results show that the algorithm performs the SCB contents understanding for several commercial sports video programs.**

***Index Terms*—Content-based video analysis, contextual understanding, sports video, superimposed caption box (SCB), video annotation.**

I. Introduction

NOWADAYS, sports videos have received great attention, motivated by the applications in video indexing, summarization, browsing, and retrieval. In sports videos, there is a superimposed caption box (SCB) on the screen, which provides the on-going information of the game, such as the score, the period of game, the time remaining clock, and so on. With the view to automatically understanding the semantics of a sports video, we urgently need a SCB understanding system. The MPEG-7 [1], [2] has tried to standardize the media access methods based on its contents. Many approaches have been proposed to extract the semantic concepts or abstract attributes, such as events, scene types, and objects, from the videos [3]–[5], [14]. Normally, we use the semantic descriptions to specify the contextual information embedded in the video for multimedia information retrieval. The retrieval accuracy depends on the accessible semantic descriptions of the video.

For content-based information retrieval, we may also apply the technology of the video text detection and recognition. Captions are embedded in the video as the instant information for the subscribers. The caption information is a useful clue in multimodal approach [6]–[9] for dealing with the multimedia information retrieval and browsing. Sato *et al.* [10] proposed a system for news videos caption recognition which consists of spatial filters to segment the word from image and video OCR. Accurate video OCR is a major technology for searching news video archives. Tang *et al.* [11] presented news video caption detection and recognition system based on a fuzzy-clustering neural network classifier and combined caption-transition detection scheme. The semantic analysis [12], [13] has become one of the popular research topics for the applications of sports multimedia retrieval and browsing [34], [35].

Many researchers have tried to solve the problem of SCB extraction, enhancement, and recognition. Lie *et al.* [15] illustrated a baseball event classification system based on the caption and visual features. The caption template is needed before recognizing the caption content. A general and domain-specific caption text extraction and recognition has been proposed in [16] which combines the transition model in specific domain to improve the recognition accuracy. Sung *et al.* [17] developed a knowledge-based numeric caption recognition system to recover the valuable information from an enhanced binary image by using a *Multilayer Perceptron (MLP) Network*. The results have been verified by a knowledge-based rule set designed for a reliable output and applied for live baseball programs. Lyu *et al.* [18] proposed an approach to detect and extract the text for the multilingual video. Miao *et al.* [32] proposed a real-time approach to detect score region and recognize score numbers in basketball video based on the domain-knowledge such as score number's spatial lay-out and time-varying appearance.

Text enhancement has been developed to reduce the visual noise sensitivities due to quantization blurring, complex backgrounds, and flashing interferences etc. In [19], the temporal redundancy is exploited to improve text segmentation performance. Input images and videos can be rescaled and integrated in a multi-resolution approach. Lienhart [20] illustrated the performance improvement for video OCR system with a gain of 1.5dB in PSNR at low bit rates by multiple frames instead of single one. To enhance the quality of the extracted text sub-images, multi-frame averaging for recognition has been proposed [21], [22].

Different from [23] which detects the text in natural scene, we focus on analyzing the *overlay (superimposed) caption*. The captions embedded in sports videos help the audience to keep track of the video program. The changing captions indicate the current key information of video contents. Unfortunately, none of the existing work is capable of interpreting the semantic of the captions automatically. Most of the previous studies [6],

Manuscript received May 11, 2007; revised May 12, 2008. First published April 11, 2008; last published August 20, 2008 (projected).

H.-C. Shih is with the Department of Electrical Engineering, National Tsing Hua University Hsinchu 300, Taiwan, R.O.C. (e-mail: hc.shih@ieee.org).

C.-L. Huang is with the Department of Electrical Engineering, National Tsing Hua University Hsinchu 300, Taiwan, R.O.C. He is also with the Department of Informatics, Fo-Guang University I-Lan, Yilan City 260, Taiwan, R.O.C. (e-mail: clhuang@ee.nthu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBC.2008.2001143

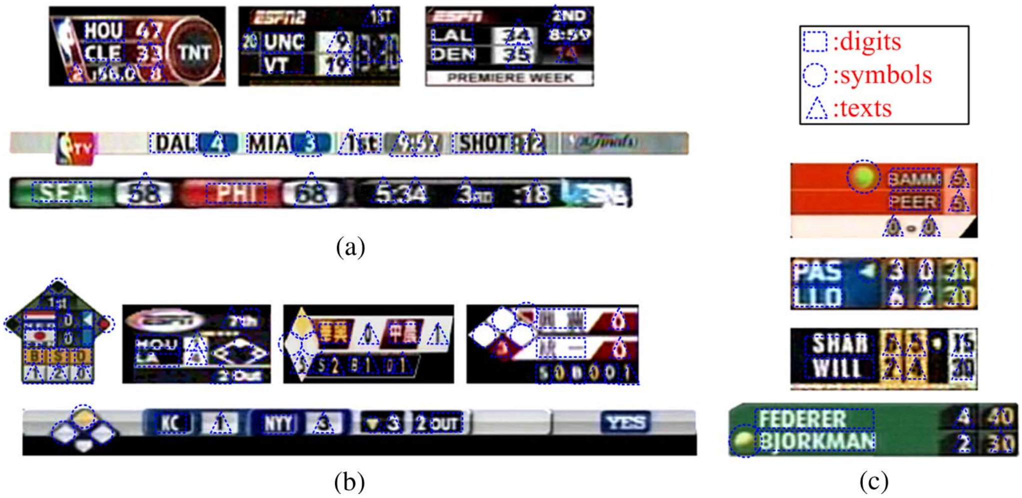


Fig. 1. SCB of sports videos: (a) basketball; (b) baseball; (c) tennis.

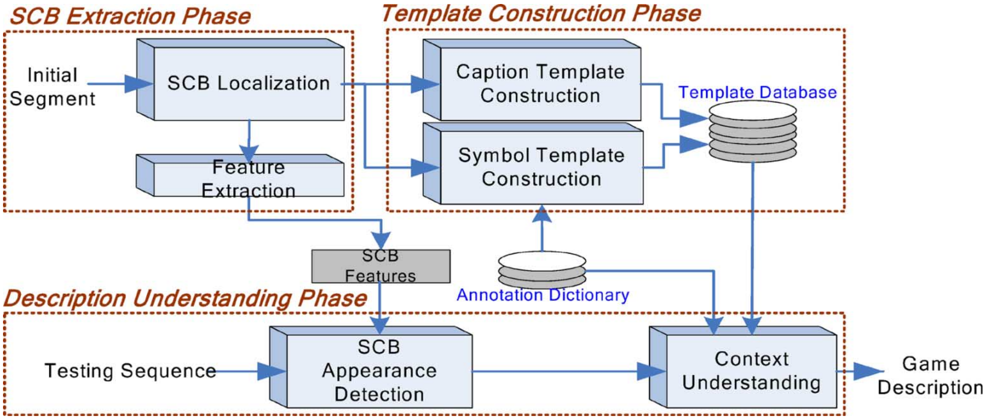


Fig. 2. The flow diagram of system overview.

[7], [33] assume that the SCB template is known, and require that the representative character bitmaps have been provided. Zhang's method [16] can extract the character bitmap and recognize the text automatically, but it avoids the problem of SCB semantic interpretation. This paper proposes a robust SCB extraction, recognition, and semantic interpretation for the sports videos understanding. The targeted SCBs are shown in Fig. 1.

The rest of the paper is organized as follows. In Section II, we show the overview of the content understanding framework. Section III presents generic SCB segmentation and modeling, we also illustrate the keyframe detection method in our application, while in Section IV and Section V the proposed caption/symbol template generation scheme are individually presented. An evaluation of system performance and experimental results are shown in Section VI.

II. System Overview

The SCB contains the symbols and the characters. The former indicates the occurrence of certain events or the current offensive team/player, and the latter can be further decomposed into texts and digits. The texts indicate the names of the team, and the period of the game, etc. Whereas the digits present the current scores of the game, or the count-down indicators. The texts are grouped as a semantic unit called the annotative object, whereas the digits are grouped as the digit object. Normally, the digit object comes with certain annotative objects. In basketball videos, the digit object indicating the current quarter is usually followed by an annotative object, i.e., "ST", "ND", "RD", or "TH". The digit object may also appear by itself, such as the countdown clock.

Our system consists of three phases: (1) SCB extraction phase; (2) template construction phase; (3) description understanding phase (shown in Fig. 2). More specifically, in the 1st phase, the system locates the SCB and generates a smooth SCB mask and SCB color histogram. In the 2nd phase, the system segments and separates the characters into texts and digits, recognizes the texts/digits using SVM, and locates and identifies the symbols. In the 3rd phase, the system understands the contextual information by using the caption and symbol templates.

The type of the sports video is presumably known and all meaningful annotations may used for the specific sport are manually pre-stored in the annotation dictionary. By referring to the dictionary, the segmented texts (or digits) are grouped as a semantic unit called the annotative (or digit) object which represents the current statistics of the on-going game. We propose a probabilistic labeling algorithm to link the digit object with an annotative object. The linked annotative and digit objects are

TABLE I
THE CHANGE FREQUENCY OF DIGIT OBJECT

Change Rate		slow→	…………………	→fast	
Baseball	#Inning	Team Score	#Out	#Strike	#Ball
Basketball	#Quarter	Team Score	Time(min.)	Time(sec.)	Shot Clock
Tennis	Inning/Score		Match Score	-	-

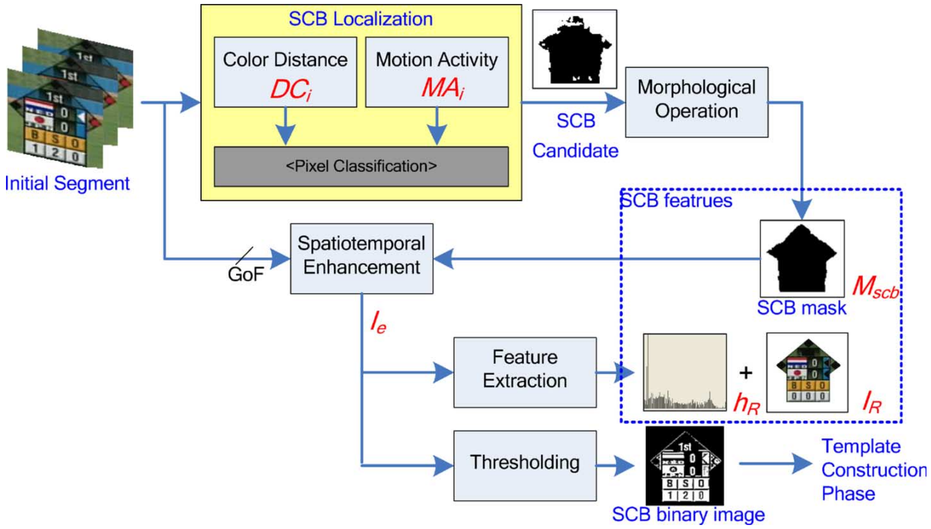


Fig. 3. Flow diagram of SCB extraction and feature extraction.

stored in the corresponding templates which is a prerequisite for understanding the semantic meaning of the SCB.

In the 2nd phase, we also locate and identify the symbol blobs in the SCB. In sports video, symbols present the specific information for different games. For instance, in a baseball game, symbols indicate the different base-taken scenarios or the current offensive team, whereas, in a tennis video, the symbol implies the current advantageous player. For different broadcast sports video, we may find different kinds of symbols in SCB. Here, we need to identify the symbols of all varieties and then generate the symbol templates.

In the 3rd phase, we interpret the semantic meanings for the succeeding video by using the generated caption/symbol templates. Various domain-specific annotations have been used for sports videos. Generally speaking, the digit object is usually associated with an annotative object to demonstrate its semantic meaning. If the digit object is not associated with any annotative object, then it is called an un-linked digit object of which the semantics will be directly interpreted based on its change rate (frequency) as shown in Table I. For instance, in basketball video, the un-linked digit objects could be either the shot clock or the quarter number. The digit object with high change rate implies a stronger indication of the shot clock. For the time remaining clock, we find two digit objects and the in-between "colon".

III. SCB EXTRACTION

In Fig. 3, we illustrate a segmentation mechanism to extract the SCB region by using color and motion information. The SCB region represents the sub-image of the embedded scoreboard template. First, we extract the SCB features from the enhanced image, and then threshold the enhanced image to obtain the representative SCB binary image which is utilized for template construction. Once the first SCB region is extracted, we may have the SCB color model which can be applied for the SCB segmentation in the succeeding video.

A. SCB Region Localization

It is obvious that the SCB region is either stationary globally or varying locally. We combine color-based local dynamic and temporal motion consistency to locate the SCB from a group of frames (GoF) due to the high color correlation and low motion activities within the SCB region. Here, we set $\mathrm{GoF} = 5$ for continuous frames to observe a SCB candidate.

Firstly, the HSI color distance [24] between two consecutive frames f_k and f_{k+1} is pixel-wisely computed. The HSI color distance for pixel i in f_k and the same pixel i in f_{k+1} is measured by

$$DC_i = \sqrt{(d_{intensity}(i))^2 + (d_{chroma}(i))^2}, \qquad (1)$$

where H_i^k, S_i^k, I_i^k are the three color components of pixel i in frame f_k, $d_{intensity}(i) = |I_i^k - I_i^{k+1}|$, $\Omega_i = |H_i^k - H_i^{k+1}|$, $d_{choma}(i) = \sqrt{(S_i^k)^2 + (S_i^{k+1})^2 - 2S_i^k S_i^{k+1}\cos(\theta_i)}$, $\theta_i = \Omega_i$ if $\Omega_i \leq 180^\circ$ else $\theta_i = 360^\circ - \Omega_i$. Following that, we define the motion activity MA_i for every pixel i of each frame f_k using standard block-based motion estimation. The value of MA_i is derived from the average magnitude of all motion displacement

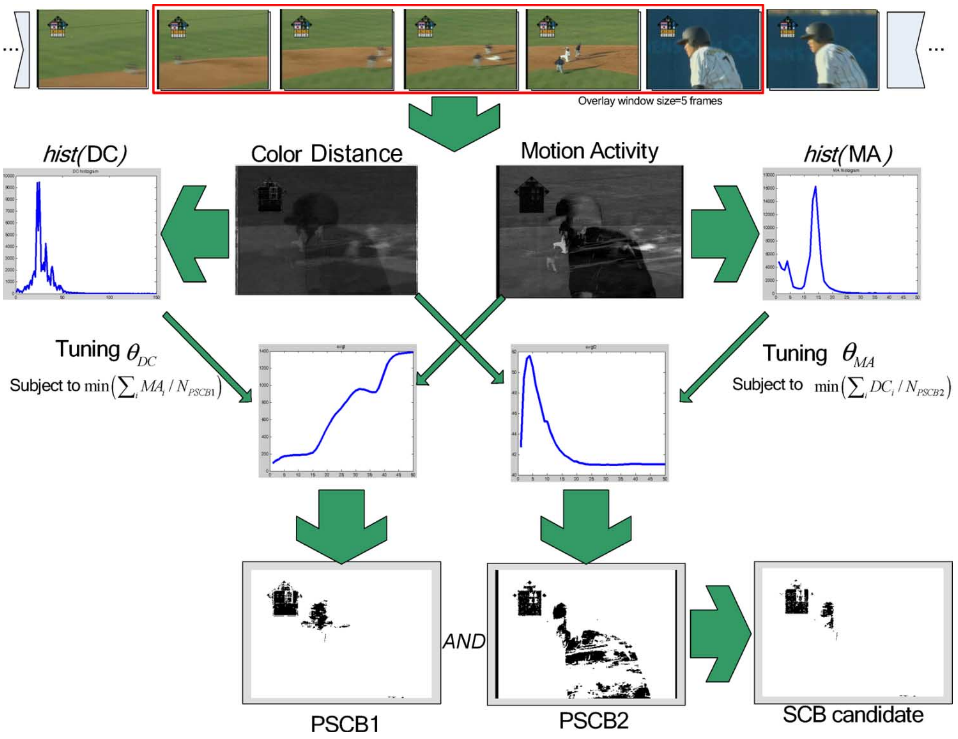

...
...
Overlay window size=5 frames
hist(DC)
Color Distance
Motion Activity
hist(MA)
Tuning θ_{DC}
Subject to $\min\left(\sum_i MA_i / N_{PSCB1}\right)$
Tuning θ_{MA}
Subject to $\min\left(\sum_i DC_i / N_{PSCB2}\right)$
AND
PSCB1
PSCB2
SCB candidate

Fig. 4. The flow chart of SCB localization.

vectors $\boldsymbol{dv}(i)$ in the macroblock centered by pixel i and normalized by the maximum displacement vector $\boldsymbol{dv}_{max}(i)$ among block B. With N_b pixels within a block B, we define the MA_i as

$$MA_i = \sum_{i=1}^{N_b} |\boldsymbol{dv}(i)| / N_b \, |\boldsymbol{dv}_{max}(i)|, \tag{2}$$

where $|\boldsymbol{dv}(i)|$ is the magnitude of displacement vector $\boldsymbol{dv}(i)$. Subsequently, we classify pixel i based on the thresholding process applied on color distances DC_i and motion activities MA_i. We generate the accumulated histogram of DC_i by adding the difference of five consecutive frames $f_k \sim f_{k+4}$. For each pixel i, we define the $DC_i(k)$ as the color distance between frame f_k and f_{k+1}, and then add the color difference of five consecutive frames to generate the accumulated color difference histogram $hist(DC)$. Similarly, we may generate the accumulate motion histogram $hist(MA)$.

To locate the SCB region, we apply two thresholdings on $hist(DC)$ and $hist(MA)$ as shown in Fig. 4. The bottom of the valley of $hist(DC)$ and $hist(MA)$ may not indicate the best thresholds. We propose a fine adjusting process to find the best thresholds θ_{DC} and θ_{MA} as follows:

1) Select the first valley of $hist(DC)$ as θ_{DC} and the first valley of $hist(MA)$ as θ_{MA}.
2) Apply thresholding process to classify pixel p_i to SCB, if $DC_i < \theta_{DC}$, then $p_i \in PSCB1$.
3) Adjust θ_{DC} subject to minimizing $\Sigma_i MA_i / N_{PSCB1}$, where $\Sigma_i MA_i$ denotes the summation of motion activities for all the pixels $p_i \in PSCB1$ and N_{PSCB1} is the number of pixels in $PSCB1$.
4) Apply thresholding process to classify pixel p_i to SCB, if $MA_i < \theta_{MA}$, then $p_i \in PSCB2$.
5) Adjust θ_{MA} subject to minimizing $\Sigma_i DC_i / N_{PSCB2}$, where $\Sigma_i DC_i$ denotes the summation of color distances for all the pixels $p_i \in PSCB2$ and N_{PSCB2} is the number of pixels in $PSCB2$.

From the above processes, we may obtain two potential SCB regions, $PSCB1$ is obtained by analyzing the motion activity with threshold θ_{DC} and $PSCB2$ is obtained by measuring the color distance with threshold θ_{MA}. The SCB candidate can be extracted by integrating $PSCB1$ and $PSCB2$. Because of the transparent effect, the extracted SCB region may not be contiguous, so we apply the morphology operation to merge all the pixels classified to SCB, and obtain a contiguous SCB mask M_{scb}.

B. SCB Feature Extraction

The SCB features consist of representative color histogram h_R, representative image I_R, and SCB mask M_{scb}. To generate the SCB color model, the three HSI color components of each pixel within SCB mask are coded by 4 bits, 2 bits and 2 bits, respectively. The color model for the SCB in frame k is

$$h^k(j) = \sum_i \delta\left(L_{HSI}\left(H_i^k, S_i^k, I_i^k\right) - j\right) \quad for\ 0 \leq j \leq 255, \tag{3}$$

where H_i^k, S_i^k, I_i^k are the three color components of pixel i in frame k, $\delta(i-j) = 1$ for $i = j$, $\delta(i-j) = 0$ otherwise. $L_{HSI}(\cdot)$ represents the color mapping function which converts a 3-D color component (H, S, I) to a 1-D color index. The representative SCB 1-D color histogram h_R is created by averaging

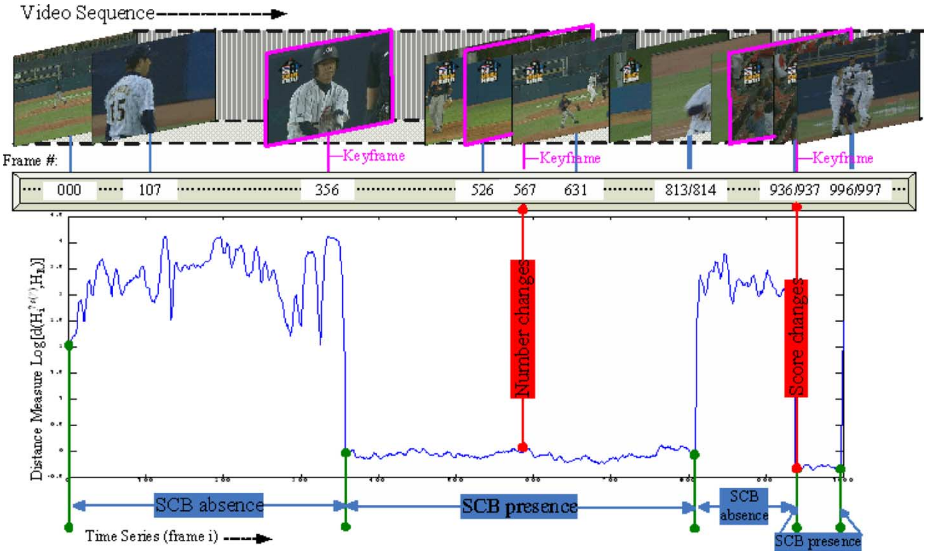


Fig. 5. SCB appearance detection.

the histograms of the segmented SCBs in a group of frames (GoF) as

$$h_R(j) = \frac{1}{N}\sum_{k=1}^{N} h^k(j), \tag{4}$$

where N is the number of frames in GoF.

C. SCB Appearance Detection

Here, we definite two types of keyframes: *show-up keyframe* and *content-change keyframe*. The former is determined when the SCB abruptly appears on the screen. The latter is defined when the on-screen SCB has changing content. The *show-up keyframe* indicates globally changing, whereas the *content-change keyframe* represents locally varying. The similarity measure is used to extract keyframe. Since the SCB template does not change much during the entire video, we may use the SCB color model to identify its presence even if the SCB may be transparent. The similarity measure between the pre-stored SCB model h_R and the potential SCB is formulated by *Mahalanobis distance* [26] $d(h_k^{Mscb}, h_R) = (h_k^{Mscb} - h_R)^{\mathrm{T}}\mathbf{C}_{\mathrm{m}}^{-1}(h_k^{Mscb} - h_R)$, where h_k^{Mscb} denotes the color histogram of the SCB mask M_{scb} in k^{th} frame and $\mathbf{C}_{\mathrm{m}}^{-1}$ is a covariance matrix. We use the distance $d(h_k^{Mscb}, h_R)$ to detect *content-change keyframe*. By comparing the color histogram of on-going video frame masked by h_k^{Mscb} with the representative histogram h_R, we may identify whether the content of the SCB has changed. In Fig. 5, the *content-change keyframe* is detected in frame #567 and frame #936. However, if $d(h_k^{Mscb}, h_R)$ is larger than certain threshold, then the SCB is not found on the screen (frame #0 $\sim$ frame #356 in Fig. 5), else the SCB appears. The *show-up keyframe* denotes the 1st frame which detected SCB presence such as frame #356 and frame #937.

D. Spatiotemporal Enhancement

To improve the image quality inside SCB, we propose a hybrid scheme by integrating the spatial and temporal enhancement schemes. Because the SCB is always overlaid with high contrast in spatial domain, we apply the adaptive unsharp masking algorithm [25] to emphasize the band-pass frequency components. First, we classify each pixel within SCB to low/medium/high contrast class based on the local variance of this pixel computed over a 3 × 3 pixel block. Second, the adaptive filter is used to emphasize the medium-contrast in the input image rather than the high-contrast details such as abrupt edges. In temporal domain, we perform multiple frame average operation over w consecutive frames as

$$I_e(i,j) = \frac{1}{w}\sum_{d=1}^{w} \overline{I}_d(i,j), \tag{5}$$

where $\overline{I}_d(i,j)$ is the intensity of the pixel at (i,j) for each color component within the SCB.

IV. Caption Template Construction

Here, we develop a method to extract and interpret the SCB as shown in Fig. 6. The Connected Component Analysis (CCA) method is employed to extract the characters and the font-independent features for the recognize phase. We divide characters into two classes: texts and digits, and then interpret them by using the object labeling method.

A. Character Extraction

Once the SCB region is identified, we may extract the character blocks in the SCB. First of all, we apply Otsu's approach [27] to segment the SCB into character and non-character regions. Subsequently, we apply the CCA to group the neighboring pixels in character regions into the blocks. Since the characters may consist of high intensity pixels with low intensity background, or low intensity pixels with high intensity background, we need to perform the same operations in the contrast reversed SCB sub-image. Finally, the blocks will be remained if the following constraints are satisfied: the range of block size (between 30$\sim$250 pixels), the range of aspect ratio (between 1$\sim$2), and the range of orientation (between 70$\sim$110

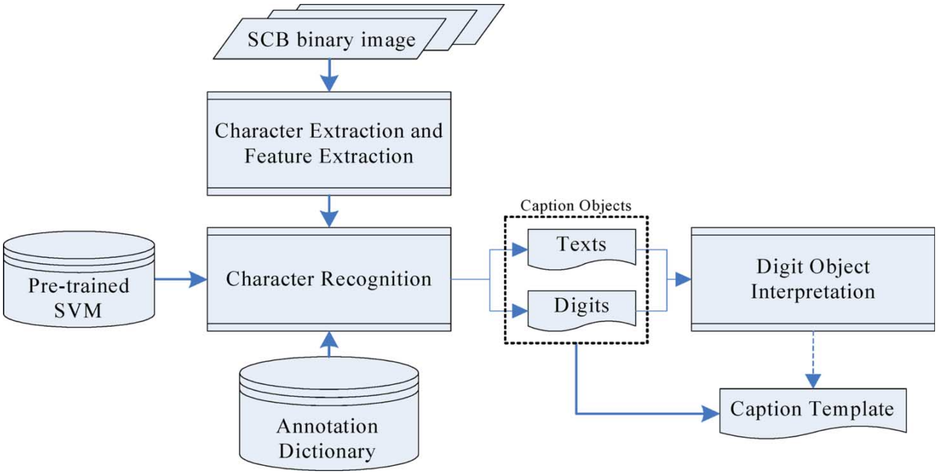
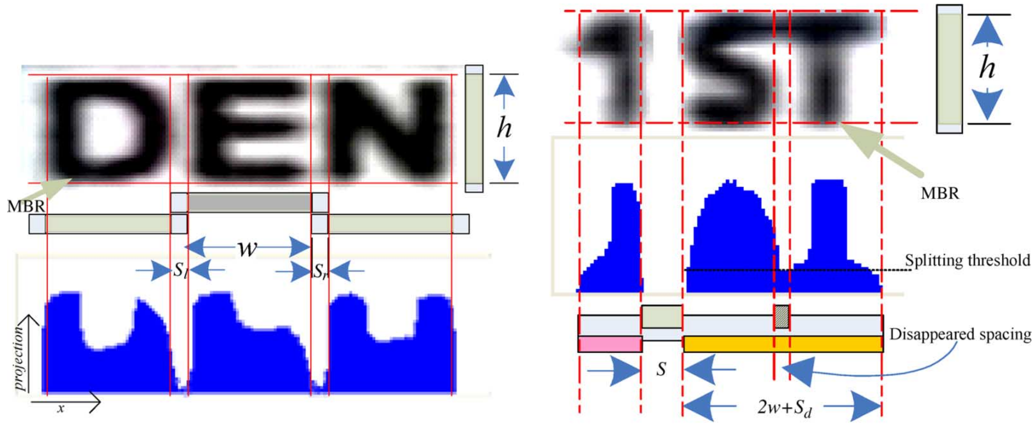


Fig. 6. The diagram of caption template extraction and interpretation.



(a) (b)

Fig. 7. Text merging and splitting for MBR.

degrees). The remained blocks are called *minimum bounding rectangle* (MBR).

Most of overlay texts within the SCB have the same orientation and spacing. Each text line is composed of characters of equal size. As shown in Fig. 7(a), every character C_i has the fixed aspect ratio w_i/h_i and with the character spacing s_l and s_r located in the text line. Sometimes the characters can be clearly separated by vertical projection profile. But due to low image resolution, the characters on the same text line will be merged, if the spacing $s < \delta(w_i/h_i)$, where $s = max(s_l, s_r)$ and the δ is the scaling factor. As shown in Fig. 7(b), the original character spacing disappears. However, the height of MBR can be used to determine the proper character width w and spacing s. If the MBR width is close to $nw + (n-1)s$, where n is the character number, then the MBR can be spitted by the segmentation method [10] which analyses the vertical projection profile. The annotation dictionary stores the domain-specific annotation such as the abbreviation of the team name (i.e., BOS or NYY), the annotation of the number of the out (i.e., "out", "O", or "OUT"). Finally, the MBR will be grouped into digit object or annotative object based on the pre-stored meaningful annotation in the dictionary. For example, the three detected MBRs includes "o", "u", and "t" can be grouped together based on the character spacing rules mentioned above.

B. Character Recognition and Caption Template

Due to the limitation of image resolution, how to achieve good character recognition rate is a nontrivial problem. Various schemes [16], [18], [22] have been developed by using the Zernike Moments [28], [29] and Support Vector Machine (SVM) [30], [31], which presented well performance. In this paper, we use a pre-trained SVM classifier as the recognition kernel to identify whether the MBR is a text or not. The font-independent features are used and the feature vector of character block consisting of *Area, Centroid, Convex Area, Eccentricity, Equivalent Diameter, Euler Number, Orientation,* and *Solidity*. The size of character block is normalized to 30 × 40. Once a new character is recognized, we may generate a specific caption template defined as

$$\boldsymbol{CT}^i = [\boldsymbol{Pos}, text/digit, \boldsymbol{SV}, \boldsymbol{CB}], \tag{6}$$

where $\boldsymbol{Pos}$ indicates the relative location in the SCB, text/digit bit indicates whether the caption is text or digit, $\boldsymbol{SV}$ indicates a set of support vectors for this character, $\boldsymbol{CB}$ is the corresponds character block. This caption template can be used to identify the characters in the succeeding videos.

C. Digit Object Interpretation

Obviously, the SCB has the following regularities: (1) the digit and annotative objects are related based on their distance; (2) the digit objects and its associated annotative objects are always on the same horizontal line or vertical line; (3) two related digit objects may sandwich the same annotative object, but two independent digit objects cannot be related to the same annotative object. In SCB region, we may identify the time remaining indicator which consists of minute digits, second digits, and an in-between "colon". This indicator follows three constraints: (i) the MBRs are parallel, (ii) the MBRs are close to one another, (iii) a "colon" or "dot" may be found between two digit objects. However, if the digit object is not associated with any annotative object, it becomes a so-called "*un-linked digit object*", of which its semantics will be interpreted based on its change frequency.

For other digit objects, we develop a method to find the correspondence between the digit object and the annotative object for the semantic interpretation of the digit objects. Here, we transform the semantic interpretation problem to the object labeling problem by using Relaxation Labeling [32] which may assign each digit object a label (or an annotative object).

Given (1) a set of objects $\{O_i, i = 1, 2, \ldots, n\}$ including digit objects Λ_{d} and annotative objects Λ_{a}, (2) a set of labels $\{\lambda_j, j = 1, 2, \ldots, m\}$ with priorities, and (3) the three regularities mentioned above, our goal is to assign one label to one digit object. Moreover, we use the relaxation approaches to label each object $\in \Lambda_{\mathrm{d}}$ by a neighboring object $\in \Lambda_{\mathrm{a}}$, and to obtain the maximal associated probabilities. We define the variable $p_i(\lambda_j)$ to represent the probability of the object $O_i \in \lambda_j$, with $0 \leq p_i(\lambda) \leq 1$ and $\Sigma_{\mathrm{j}} p_i(\lambda_j) = 1$. We define the compatibility of object i labeled by λ with object j labeled by λ' as $c(i, \lambda, j, \lambda')$ with the constrain that

$$\sum_{\lambda} c(i, \lambda, j, \lambda') = 1, \qquad for\, \forall i, j, \lambda', \tag{7}$$

The compatibility c represents the inter-dependency between object j labeled as λ, and object i labeled as λ. It is insufficient that if the compatibility is only determined by the distance between digit object and the label (or the annotative object), and it may introduce the false interpretation. Therefore, we define the compatibility as follows

$$c(i,\lambda,j,\lambda') = \begin{cases} -1 & i,j \in \Lambda_d \cap \lambda = \lambda' \\ \varepsilon \left|\cos\left\{2\left[\theta_i(\lambda) - \theta_j(\lambda')\right]\right\}\right| & \\ \quad +(1-\varepsilon)\frac{d_j(\lambda')}{d_i(\lambda)} & i,j \in \Lambda_d \cap \lambda \neq \lambda' \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

where ε is the weighting factor for the distance confidence and the orientation confidence, $\theta_i(\lambda)$ is the direction from object i to object j labeled as λ, $d_i(\lambda)$ is the Euclidean distance between object i and object j labeled as λ. If $p_j(\lambda')$ is high and $c(i, \lambda, j, \lambda')$ is positive, then $p_i(\lambda)$ is increased. This labeling algorithm is an iterative parallel procedure analogous to the label-discarding rule used in the probabilistic relaxation. The operator iteratively adjusts label weights in accordance with other weights and the compatibilities. For each object and each label, a new weight $q_i^{(r)}(\lambda)$ is computed as

$$q_i^{(r)}(\lambda) = \sum_{j, j \neq i} \sum_{\lambda'} c(i, \lambda, j, \lambda') p_j^{(r)}(\lambda'), \tag{9}$$

where the superscript r is denotes the r^{th} iteration. In Eq. (9), the product sum is the expectation that object O_i has label λ, given the evidence provided by object O_j. $q_i^{(r)}(\lambda)$ is thus a weighted sum of current assignment values $p_i^{(r)}(\lambda)$. Then, a new assignment is updated as

$$p_i^{(r+1)}(\lambda) = \frac{P_i^{(r)}(\lambda)\left[1 + q_i^{(r)}(\lambda)\right]}{\sum_{j=1}^{m} p_i^{(r)}(\lambda_j)\left[1 + q_i^{(r)}(\lambda_j)\right]} \tag{10}$$

Here, we simply pick the $p_i^{(r)}(\lambda)$ and $c(i, \lambda, j, \lambda')$ and apply Eq. (10) to recursively update the $p_i^{(r)}(\lambda)$ until they stop varying or converge to 1. Each digit object is iteratively verified for correct semantic interpretation.

Eventually, we identify the semantic meanings for the succeeding video by using the caption templates. Not every annotative object will appear within the SCB. For different sports videos, different domain-specific annotations appear. For basketball video, the digit object may be accompanied by an annotative object, or it may have no related annotative object (if $n_a \leq n_d$). The semantic meaning of each linked digit object is determined by its corresponding annotative object. For each un-linked digit object, we then check its content change frequency.

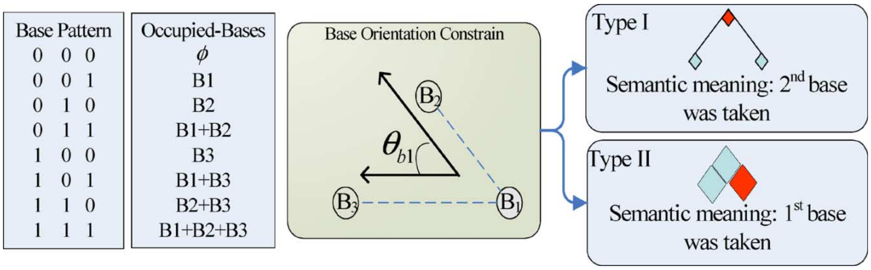
V. Symbol Template Construction

The symbols within the SCB may show-up with constant/varying shape or fixed/non-fixed location with consistent color over a period of time. We need to identify the symbols in different sports videos. In baseball sports video, there are three high-contrast color symbol blobs in the SCB, indicating different base-taken scenarios. The symbol blobs serve as indicators for different events occurring in the videos. For tennis or badminton, the symbol may indicate the current advantageous player.

The symbol blobs may have many different varieties. The shape, location, and color of symbol blobs are not a priori. Here, we use $\boldsymbol{ST^j}$ to denote the j^{th} symbol annotation as

$$\boldsymbol{ST}^j = [\boldsymbol{Pos}, SType, N_s, hu/\overline{hu}], \tag{11}$$

where $\boldsymbol{Pos} = (x, y)$ represents the centroid location of the symbol, $SType$ stands for the constant shape type of the symbols (i.e., $SType = 0$ denotes that all the SCB symbols are of constant shape, otherwise it indicates the SCB symbols have different shapes). There are p different kinds of symbols, i.e., $SType = p - 1$, N_s denotes the number of symbol blobs, and $hu/\overline{hu}$ indicates the hue of the appearing/disappearing symbol blob respectively.



Base Pattern	Occupied-Bases
0 0 0	ϕ
0 0 1	B1
0 1 0	B2
0 1 1	B1+B2
1 0 0	B3
1 0 1	B1+B3
1 1 0	B2+B3
1 1 1	B1+B2+B3

Base Orientation Constrain
B2
θ_{b1}
B3
B1
Type I
Semantic meaning: 2nd base was taken
Type II
Semantic meaning: 1st base was taken

Fig. 8. Base orientation constrain.

The symbol template generation consists of the following four steps:

1) *Symbol Blob Detection*: Once the *content-change keyframe* is detected, we may analyse the difference to see whether there is a new blob. Since the symbol blobs are stationary for a period of the time, by measuring $d(h_k^{Mscb}, h_R)$, we may identify the appearing/disappearing symbol blobs.
2) *Symbol Blobs Grouping*: The symbol blobs may indicate the same meanings if their locations are close to each other with the similar shape. We can construct the symbol template based on the relative location among candidate symbol blobs. By comparing the locations and shape of the previously and the currently extracted symbol blob, we may group the related symbol blobs as the symbol annotation.
3) *Symbol Classification*: Classifying the symbol template into four categories based on the number and the shape variation of symbol blobs as: Multiple Blob Constant Shape (MBCS), Single Blob Constant Shape (SBCS), Single Blob Dynamic Shape (SBDS), and Multiple Blob Dynamic Shape (MBDS). The last category is rarely used, and will not be discussed.

A. Multiple Blob Constant Shape (MBCS)

In baseball videos, there exist some regularities among the symbols, i.e., the three color symbol blobs are linked as a triangle. We define the angle θ_{bi} of the i^{th} symbol as the angle between the two linkages between the i^{th} symbol and the other two symbols. These three symbols have fixed relative orientation. Normally, the 3rd base symbol (B3) is located on left side of the 1st base symbol (B1), and the 2nd base symbol (B2) is placed on top of the 3rd base symbol (B3). For base template generation, we define the regularities of the three symbol blobs as: (1) color consistency: $\overline{hu_1} \approx \overline{hu_2} \approx \overline{hu_3}$ and $hu_1 \approx hu_2 \approx hu_3$; (2) shape consistency: $SType = 0$ (constant shape symbol); (3) angles relations: $\theta_{b1} \approx \theta_{b3} \leq \theta_{b2}$ and $\theta_{b1} + \theta_{b2} + \theta_{b3} = 180$; (4) in-between distance: $|\boldsymbol{Pos}_1 - \boldsymbol{Pos}_2| \approx |\boldsymbol{Pos}_2 - \boldsymbol{Pos}_3| \leq |\boldsymbol{Pos}_1 - \boldsymbol{Pos}_3|$; (5) relative location: $x_1 > x_2$ and $y_1 < y_2$; $x_2 > x_3$ and $y_2 > y_3$; $x_1 > x_3$ and $y_1 \approx y_3$.

Furthermore, we describe different base-taken scenarios by using the base pattern, i.e., (b_3, b_2, b_1), where $b_i = 1$ or 0 indicates whether the i^{th} base is taken. As shown in Fig. 8, there are eight different combinations $\{(b_3, b_2, b_1)\}$ which are called Base Patterns (BPs). Once an event occurs, the BP of the content-change key frame will change. The following rules define the variation of the BP at the two consecutive key frames as:

1) If $\mathrm{BP}(\mathrm{keyframe_k}) = \{(000)\}$ then $\mathrm{BP}(\mathrm{keyframe_{k+1}}) \neq \{(011), (101), (110), (111)\}$
2) If $\mathrm{BP}(\mathrm{keyframe_k}) = \{(001), (010), (100)\}$ then $\mathrm{BP}(\mathrm{keyframe_{k+1}}) \neq \{(111)\}$

For every *content-change keyframe*, whenever the difference in the non-character regions is detected, it indicates a new base-taken. There are may different kinds of new base-taken variations, such as $(001) \rightarrow (011) \rightarrow (111)$, or $(010) \rightarrow (101)$ etc.

B. Single Blob Constant Shape (SBCS)

Normally, the single blob is used to indicate the current offensive (or advantageous) team/player. The rules for the symbol blobs are: (1) color consistency: $\overline{hu_1} \approx \overline{hu_2}$ and $hu_1 \approx hu_2$; (2) shape consistency: $SType = 0$ (constant shape symbol), (3) relative location: $x_1 = x_2$ and $y_1 \neq y_2$ or $x_1 \neq x_2$ and $y_1 = y_2$. After character region extraction, we may move on to investigate symbol regions in SCB. In Fig. 9, the symbol blob indicates the current offensive player at different $\boldsymbol{pos}$. The symbol pattern (SP) for SBCS is defined as $\{(b_1, b_2)\}$ where $b_i = \{0, 1\}$. It will change at the time instance of *content-change keyframe*.

C. Single Blob Dynamic Shape (SBDS)

Normally, two different shape symbol blobs are used to indicate different offensive team/player. The symbol is accompanied by an annotative object presenting the names of the team/player. The spatial rules for the symbol blobs are: (1) color consistency: $\overline{hu_1} \approx \overline{hu_2}$ and $hu_1 \approx hu_2$; (2) shape consistency: $SType = 1$ (dynamic shape symbol blob), (3) relative location: $x_1 = x_2$ and $y_1 \neq y_2$ or $x_1 \neq x_2$ and $y_1 = y_2$. As Fig. 9(b), the offensive team is indicated by two different shape symbol blobs with different $\boldsymbol{pos}$. The symbol pattern (SP) can be defined as $\{(b_1, b_2)\}$ where $b_i = 1$ or 0 and different SP is accompanied with different symbol shapes. In the case of baseball game, the SP in the SCB will change at that specific time instance of *show-up keyframe* appearance.

VI. Experimental Results

Here, we illustrate four experiments. In the 1st experiment, we test the *show-up keyframe* and *content-change keyframe* detection on three types of video sequences. In the 2nd experiment, we evaluate the character extraction and recognition by using the SVM-based character recognition in two cases: $n_a = n_d$ (the number of annotative objects equals to the number of digit objects) or $n_a \neq n_d$. The 3rd experiment is the digit object interpretation, and the last experiment is the symbol understanding experiment. The inputs are one baseball video and five basketball videos from three TV channels. These videos include

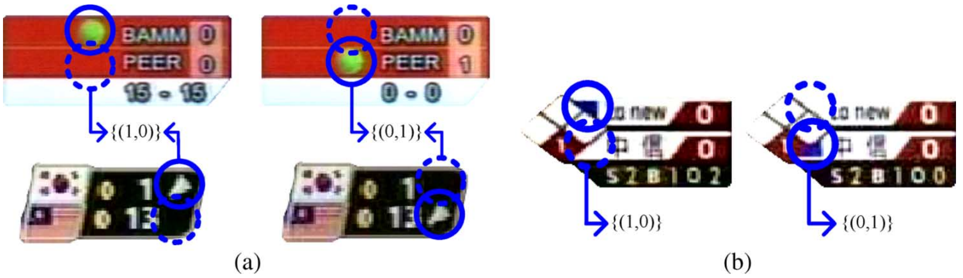


Fig. 9. Single blob constant shape: offensive transition.

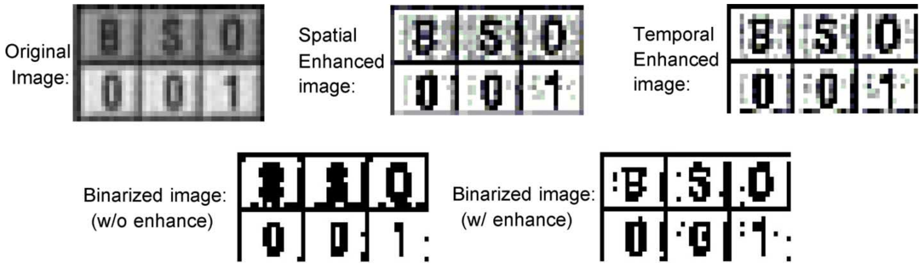


Fig. 10. Results of text enhance and binarization.

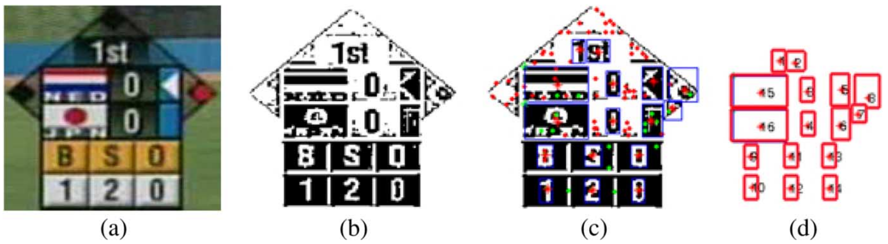


Fig. 11. The character extraction for baseball SCB: (a) a keyframe of SCB; (b) the binarized SCB; (c) the isolated regions with center points; (d) the 16 MBRs.

TABLE II
THE RESULTS OF THE KEYFRAME DETECTION

Sports type		Basketball	Soccer	Baseball
Team compete		HOU vs. CLE	MANC vs. BIR	NED vs. JPN
Clip Duration		(22:15)	(47:15)	(6:56)
Total Frame		40,067	84,912	11,822
Show-up keyframe	Ground Truth	6	1	14
	Detected	7	1	16
Content-change keyframe	Ground Truth	1,871	2,691	34
	Detected	1,724	2,438	33
Average Accuracy		92.1%	90.6%	97.1%

different SCB template styles. They have the same annotative description, but different fonts, different intensity values and contrast values from background. The format of the videos is MPEG-2 with resolution 720 × 480 pixels.

A. Keyframe Detection

We test three types of sports videos under the different broadcasting channels. The testing data are captured from (1) the 3rd quarter of basketball game of TNT channel with advertisement, (2) the 1st half of UEFA soccer game from ESPN channel without advertisement, and (3) the highlight review of 2004 Olympic baseball games. The results are shown in Table II.

For the 1st example, the *show-up keyframe* has only appeared 6 times because the highlight replays are mixed with the commercials. The SCB *content-change keyframe* appears about twice in every second. There are the time remaining clock and the shot clock running non-synchronously. For the 2nd example, the SCB appears for the entire game period, and SCB content-change is detected for every second indicating the time remaining clock digits. However, the scoring digits changes only once in a while for the entire videos. For the 3rd example, the *show-up keyframe* and the *content-change keyframe* appear more frequently than the other videos. It is due to the fact that the testing video consists mainly of highlight segments. The average accuracy of keyframe detection for baseball game is the highest among the others. The *content-change keyframes* are found only when the captions in the SCB changes.

B. Character Extraction and Recognition

Once the keyframe is detected, we perform a spatiotemporal enhancement before recognizing the characters as shown in Fig. 10. The binarization method is used to separate character regions from the background. The binarized character regions can be further processed to extract the features (i.e., area, centroid, convex area, eccentricity, diameter, Euler no., orientation, solidity) for SVM-based character recognition.

For the baseball videos of Olympic 2004, the experimental results are shown in Fig. 11. There are six annotative objects (i.e., 2, 9, 11, 13, 15, 16), six digit objects (i.e., 1, 3, 4, 10, 12, 14), and four symbol objects (i.e., 5, 6, 7, 8). We also test our text

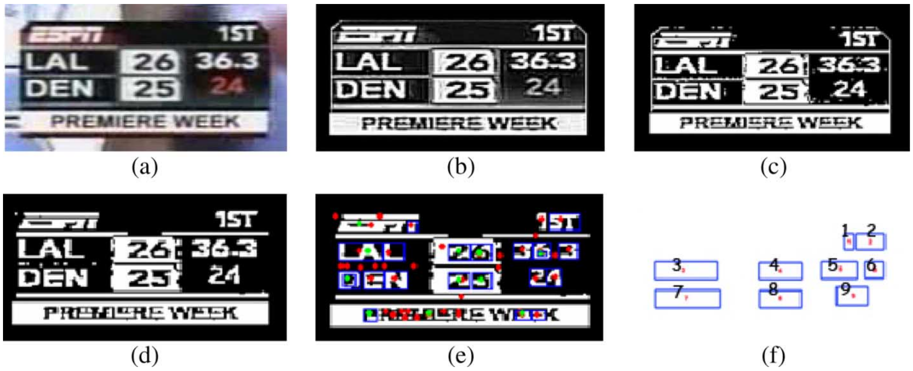


Fig. 12. The character extraction for basketball SCB: (a) original SCB image; (b) enhanced SCB image; (c) binarized SCB image; (d) morphological image; (e) the MBRs; (f) the identified text blocks.

TABLE III
CHARACTER RECOGNITION BY TWO-CLASS SVM

	Ground Truth (Text/non-Text)	FPR	FNR	Average Accuracy
Single Frame	833/337	11.5%(134)	0.4%(5)	88.12%
GoF=5	805/312	8.3%(93)	0.3%(3)	91.41%
GoF=10	759/268	6.9%(71)	0.6%(6)	92.50%
GoF=25	632/210	6.5%(55)	0.2%(2)	93.24%

extraction scheme for the sports video of NBA regular season 2005-2006 as shown in Fig. 12. There are three annotative objects is three (i.e., 2, 3, 7), and six digit objects is six (i.e., 1, 4, 5, 6, 8, 9).

Suppose we only consider the size, aspect ratio, and orientation as the features, the overall accuracy rate of the character extraction is about 64%. It is because some parts of logos look similar to the characters. To improve the accuracy we use the SVM-based character recognition approach. As a result, the average hit rate is increased as 93.2%. The performance is also related with the number of frame in GoF as shown in Table III. The results of text recognition are evaluated by False Positive Rate (FPR) and False Negative Rate (FNR) defined as

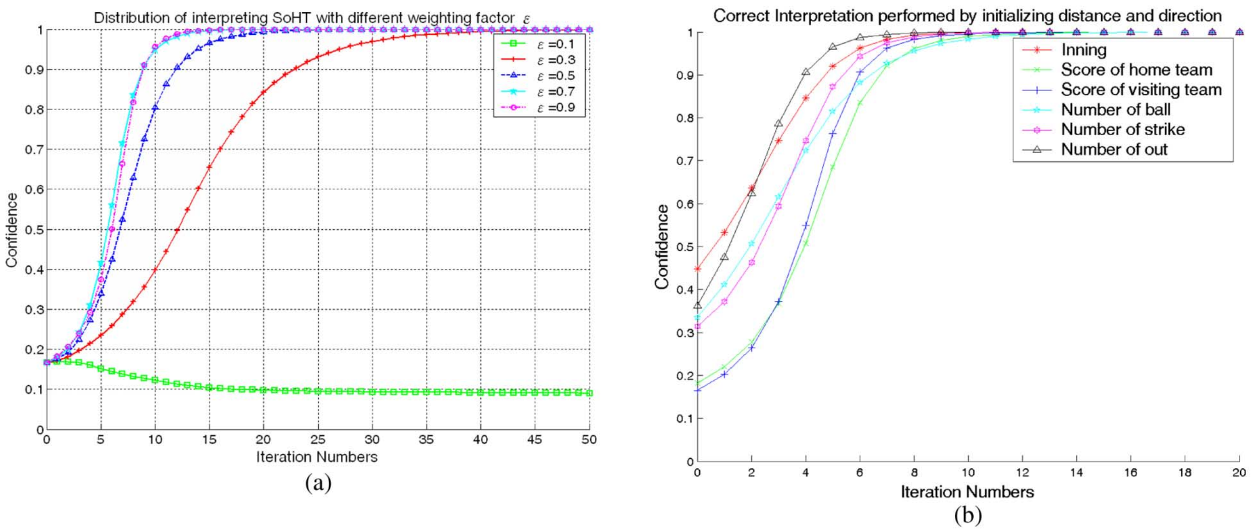
$$FPR = \frac{\#\, of\, non-text\, MBR\, falsely\, recognized\, as\, texts}{\#\, of\, non-text\, MBR}, \tag{12}$$

$$FNR = \frac{\#\, of\, text\, MBR\, miss-classified\, as\, non-texts}{\#\, of\, text\, MBR} \tag{13}$$

C. Digit Objects Interpretation

To relate the digit object with the annotative object, we apply relaxation labeling under different weighting factor ε as shown in Fig. 13. For the digit object "*score*" and the annotative object "*home team*", we try five different ε values. From the results shown in Fig.13(a), we find that the iteration converges faster for higher weighting factor ε. It indicates that the angle constraint is more important than the distance constraint. Initial guess of $p_i(\lambda_j)$ is determined either by the normalized distance between current digit object and the closest labeled object (annotative object), or a simple equal probability for each label as $p_i(\lambda_j) = 1/m$. Fig. 13(b) shows the results of the iteration of $p_i(\lambda_j)$ with initial assignments of label weights based on the normalized distance between current digit object and closest labeled object. Similar results can be obtained if each object is assigned a label with equal initial probability. Obviously, if $p_i(\lambda_j)$ increases and converges fast, it indicates a strong confidence that object $O_i \in \lambda_j$. For instance, the object i has a shortest distance with annotative object j and the angle constrain between annotative object and others digit object is weaker than object i. The iteration results of $p_i(\lambda_j)$ with mutually exclusive equal initial confidences (i.e., $p_i(\lambda_j) = 1/m$, $m = 6$) are presented in Fig. 14.

We test our system by using five basketball video programs as Fig. 15, which are captured from three different TV broadcasting channels, called *Game-A* ~ *Game-E*. In Figs. 15(a)–(e), the green dots denote the centroid of the identified annotative objects, the red dots represent the centroid of the identified digit objects, and the digits 1~10 denote the object number. These SCB templates are apparently different, but they both describe the same game status information such as the number of *quarter* (NoQ), the score of the *home team* (SoHT), and the score of the *visiting team* (SoVT). The results of the iteration of $p_i(\lambda_j)$ with initial confidences of label weights are mutually exclusive (i.e., $p_i(\lambda_j) = 1/m$, $m = 3$) as shown in Fig. 16. Each curve in Fig. 16(b) denotes the certainty of the digit object labeled to the correct annotative object. For instance, after five iterations, the digit object "4" (NoQ) will be confidently assigned to an annotative object "#Quarter". Let n_a be the number of the annotative



(a)

(b)

Fig. 13. The iteration of $p_i(\lambda_j)$ for (a) "Home-Team" with different weighting factors; (b) baseball game with different initial guess case.

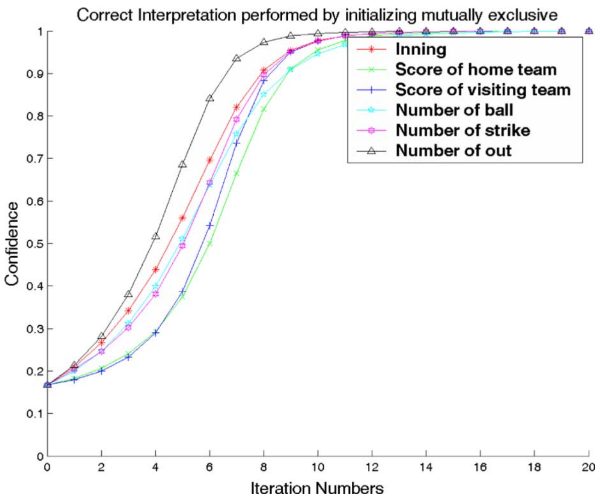
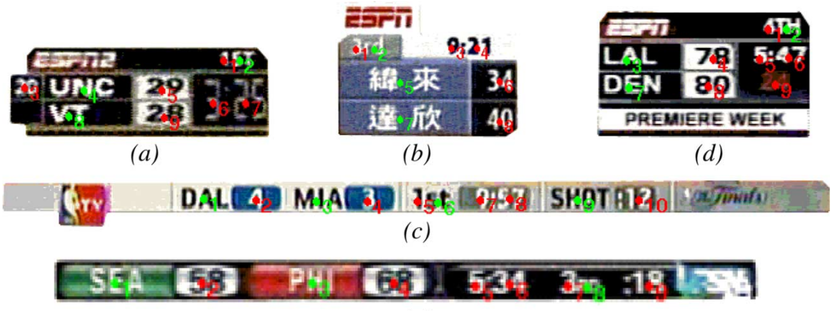


Fig. 14. The interpretation results of baseball game (mutually exclusion case).



(a)

(b)

(d)

(c)

(e)

Fig. 15. The testing SCB image (red points: digit objects, green points: annotative objects. (a) Game-A; (b) Game-B; (c) Game-C; (d) Game-D; (e) Game-E.

objects which have been identified, and n_d denotes the number of the digit objects. Intuitively, each digit object is not always accompanied with an annotative object. We consider all of the annotative objects, and for each of them, we label a digit object. Figs. 17(a)–(c) shows the labeling results of the digit objects to SoHT, SoVT, and NoQ respectively. Based on the content change frequency, system interprets whether the un-linked digit objects indicate the *shot clock* or the *quarter number*. The one with higher content-change rate is more likely to indicate the *shot clock*. To interpret the *time remaining clock*, we find the "colon" and two digit objects appearing on both sides.

For different SCB in five basketball videos, we show the result of iterative labeling. Figs. 17(a)–(c) show the convergence within 35 iterations, 10 iterations, and 5 iterations respectively.

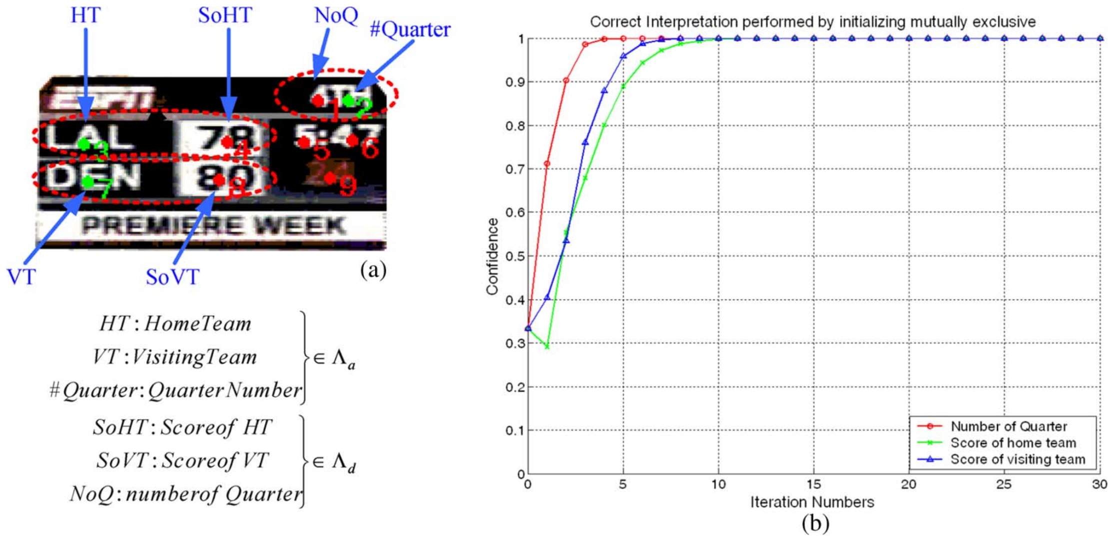


Fig. 16. The interpretation results of basketball game.

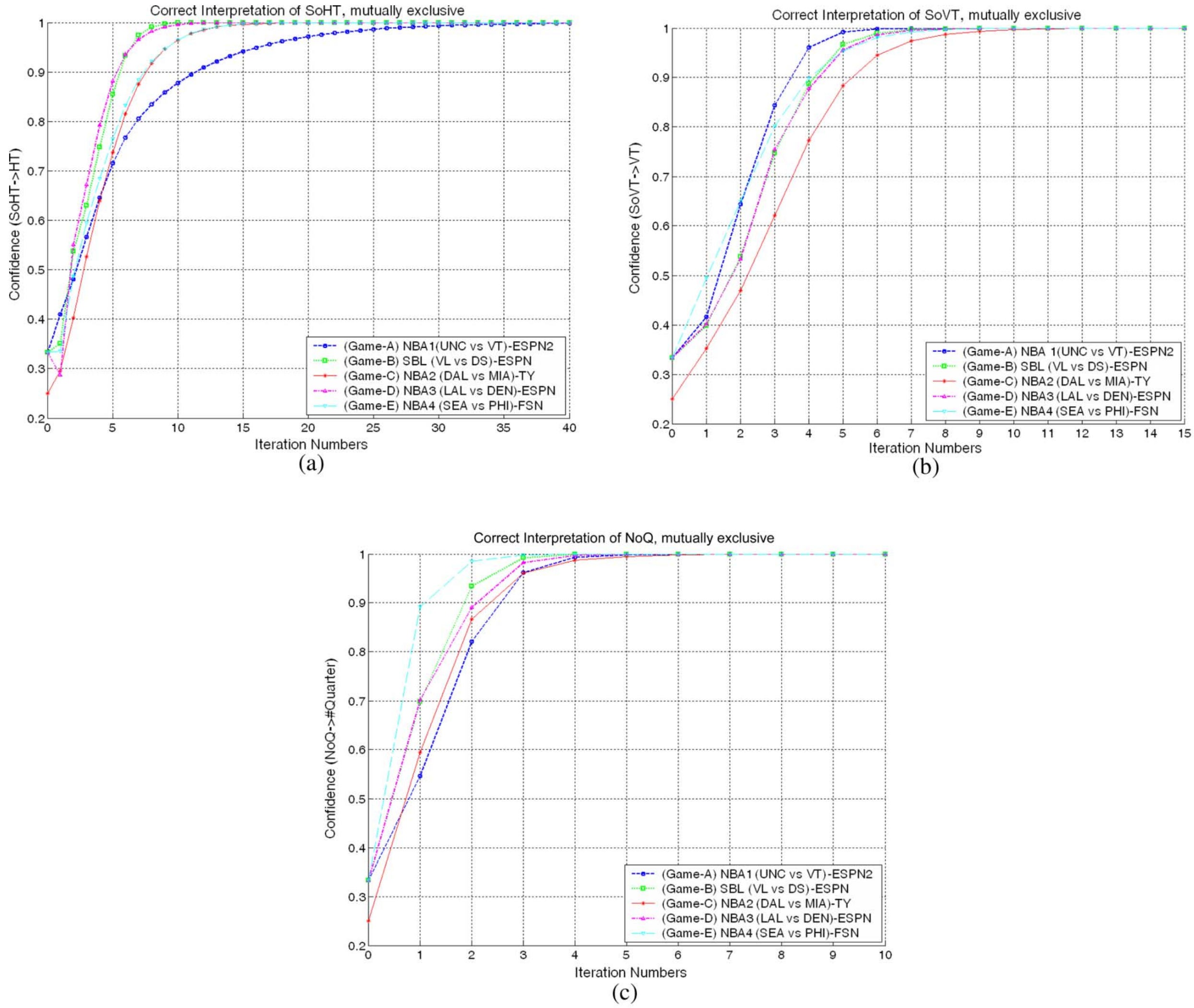
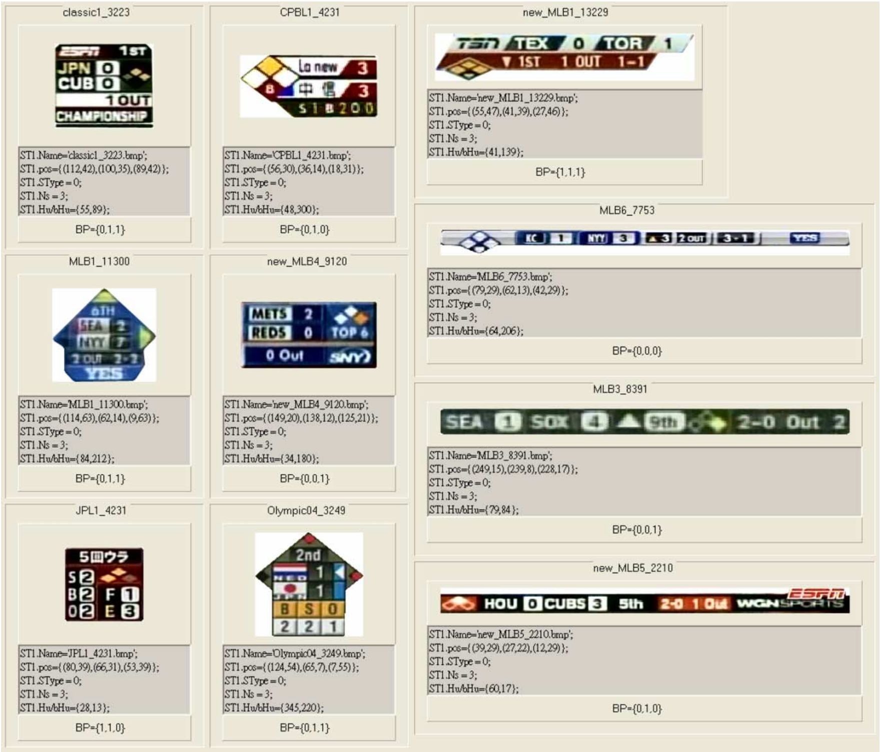


Fig. 17. The results of iterative labeling. (a) Score of home team; (b) score of visiting team; (c) quarter.



```
classic1_3223
ST1.Name='classic1_3223.bmp';
ST1.pos={(112,42),(100,35),(89,42)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={55,89};
BP={0,1,1}
```

```
CPBL1_4231
ST1.Name='CPBL1_4231.bmp';
ST1.pos={(56,30),(36,14),(18,31)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={48,300};
BP={0,1,0}
```

```
MLB1_11300
ST1.Name='MLB1_11300.bmp';
ST1.pos={(114,63),(62,14),(9,63)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={84,212};
BP={0,1,1}
```

```
new_MLB4_9120
ST1.Name='new_MLB4_9120.bmp';
ST1.pos={(149,20),(138,12),(125,21)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={34,180};
BP={0,0,1}
```

```
JPL1_4231
ST1.Name='JPL1_4231.bmp';
ST1.pos={(80,39),(66,31),(53,39)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={28,13};
BP={1,1,0}
```

```
Olympic04_3249
ST1.Name='Olympic04_3249.bmp';
ST1.pos={(124,54),(65,7),(7,55)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={345,220};
BP={0,1,1}
```

```
new_MLB1_13229
ST1.Name='new_MLB1_13229.bmp';
ST1.pos={(55,47),(41,39),(27,46)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={41,139};
BP={1,1,1}
```

```
MLB6_7753
ST1.Name='MLB6_7753.bmp';
ST1.pos={(79,29),(62,13),(42,29)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={64,206};
BP={0,0,0}
```

```
MLB3_8391
ST1.Name='MLB3_8391.bmp';
ST1.pos={(249,15),(239,8),(228,17)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={79,84};
BP={0,0,1}
```

```
new_MLB5_2210
ST1.Name='new_MLB5_2210.bmp';
ST1.pos={(39,29),(27,22),(12,29)};
ST1.SType=0;
ST1.Ns=3;
ST1.Hu/bHu={60,17};
BP={0,1,0}
```

Fig. 18. The results of base template generation.

Firstly, Fig. 17(c) shows that the NoQ is the easiest one to label, it is because the MBR of the quarter digit is much closer to the quarter annotation. Secondly, Fig. 17(a) shows that the confidence convergence of *Game-A* is the slowest one. This is primarily due to the increased influence of another digit object. Finally, Fig. 17(b) illustrates the confidence curve with a rapid increasing for *Game-A*, because the score digits of "VT" is at a distance from the other digit objects. For the SCB of *Game-C*, the labeling of the "VT" requires at least 10 iterations.

D. Symbol Understanding

The game semantic description may also be represented by domain-specific symbol annotation. The symbol blobs may be different for different sports videos. In baseball sports video, there are three high-contrast color symbol blobs in the SCB indicating different base-taken scenarios. For tennis or badminton, the symbol may indicate the current advantageous player.

Based on the constraint of MBCS, we may identify and generate the three individual base templates. Initially, we find the variation in the SCB and evaluate the hue of symbol to acquire the presence/absence situation. Then, we verify the corresponding BP as (001), (010) or (100). Once the *content-change keyframe* is detected, we check whether the BP change is $(001) \rightarrow (010)$, $(001) \rightarrow (011)$, $(010) \rightarrow (100), \ldots$,etc. We define the BP change as $(ijk) \rightarrow (lmn)$ where $|i-l| \leq 1$, $|j-m| \leq 1$, $|k-n| \leq 1$, and $0 < |i-l|+|j-m|+|k-n| \leq 3$. Besides applying the temporal rule to verify the BP variation is correct or not, we apply the spatial rule to identify the base template. For instance, if $|i-l| \leq 1$, $|j-m| \leq 1$, and $|k-n| = 0$, and then we may identify the 1st and the 2nd base templates by applying the spatial rules. Once the base templates are generated, they can be used for semantic interpretation for the base-taken events in the succeeding video. For each *content-change keyframe*, we may identify the base template in terms of its location, shape type, and $hu/\overline{hu}$. Some results of base template generation and base pattern are shown in Fig. 18.

VII. Conclusions

This paper presents a new method to automatically identify and interpret of SCB in sports videos without a priori information of SCB template. It can also construct the SCB template automatically for the context understanding in the succeeding videos. It is a robust caption interpretation method for sports videos understanding. Experimental results demonstrate that the algorithm performs well and can be applied to different sports videos. This paper has also shown good results for character/symbol interpretation. The contributions and characteristics of the proposed automated scheme can be summarized as followings:

- Effectiveness: able to locate and understand the diverse SCB templates from various sports videos
- Real-time processing: able to be implemented in real-time due to its low complexity and a property which allow frame skipping since it identifies the keyframe automatically.
- Open framework: a combination with an SCB-based application is possible, e.g., the content-oriented sport video retrieval, browsing, summarization, and real-time game status (text) broadcasting.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their insightful comments and constructive suggestions.

REFERENCES

[1] "MPEG-7: Overview (version 8)," ISO/IEC, JTC1/SC29 /WG11, Jul. 2002, N4980.
[2] S.-F. Chang, T. Sikora, and A. Puri, "Overview of the MPEG-7 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 688–695, Jun. 2001, special issue on MPEG-7.
[3] S. Dasiopoulou, V. Mezaris, I. Kompatsiaris, V.-K. Papastathis, and M. G. Strintzis, "Knowledge-assisted semantic video object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 10, pp. 1210–1224, Oct. 2005.
[4] M. R. Naphade, I. V. Kozintsev, and T. S. Huang, "A factor graph framework for semantic video indexing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 1, pp. 40–52, Jan. 2002.
[5] M. R. Naphade, S. Basu, J. R. Smith, C.-Y. Lin, and B. Tseng, "Modeling semantic concepts to support query by keywords in video," in *Proc. IEEE ICIP 2002*, Rochester, New York, Sep. 22–25, 2002.
[6] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, "Personalized abstraction of broadcasted American football video by highlight selection," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 575–586, Aug. 2004.
[7] C.-H. Liang, W.-T. Chu, J.-H. Kuo, J.-L. Wu, and W.-H. Cheng, "Baseball event detection using game-specific feature sets and rules," in *Proc. IEEE ISCAS 2005*, Kobe, Japan, May 23–26, 2005.
[8] C. G. M. Snoek and M. Worring, "Multimodal video indexing: A review of the state-of-the-art," *Multimedia Tools and Applications*, vol. 25, no. 1, pp. 5–35, 2005.
[9] D. A. Sadlier and N. E. O'Connor, "Event detection in field sports video using audio-visual features and a support vector machine," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 10, pp. 1225–1233, Oct. 2005.
[10] T. Sato, T. Kanada, E. Hughes, and M. Smith, "Video OCR for digit news archives," in *IEEE Workshop on CAIVD*, Jan. 1998, pp. 52–60.
[11] X. Tang, X. Gao, J. Liu, and H. Z. Zhang, "A spatial- temporal approach for video caption detection and recognition," *IEEE Trans Neural Networks*, vol. 13, no. 4, pp. 961–971, Jul. 2002.
[12] G. Xu, Y.-F. Ma, H.-J. Zhang, and S.-Q. Yang, "An HMM-based framework for video semantic analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 11, pp. 1422–1433, Nov. 2005.
[13] H. C. Shih and C. L. Huang, "MSN: Statistical understanding of broadcasted sports video using multilevel semantic network," *IEEE Trans. Broadcasting*, vol. 15, no. 4, pp. 449–459, Dec. 2005.
[14] C. Y. Chao, H. C. Shih, and C. L. Huang, "Semantics-based highlight extraction of soccer program using DBN," in *Proc. IEEE ICASSP 2005*, Philadelphia, PA, Mar. 18–23, 2005.
[15] W.-N. Lie and S.-H. Shia, "Combining caption and visual features for semantic event classification of baseball video," in *Proc. IEEE ICME 2005*, Jul. 6–8, 2005.
[16] D. Zhang, R. K. Rajendran, and S.-F. Chang, "General and domain-specific techniques for detecting and recognizing superimposed text in video," in *Proc. IEEE ICIP 2002*, Sep. 22–25, 2002.
[17] S.-H. Sung and W.-S. Chun, "Knowledge-based numeric open caption recognition for live sportscast," in *Proc. IEEE ICPR 2002*, Aug. 11–15, 2002.
[18] M. R. Lyu, J. Song, and M. Cai, "A comprehensive method for multilingual video text detection, localization, and extraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 243–255, Feb. 2005.
[19] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images and videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 14, pp. 256–268, Apr. 2002.
[20] R. Lienhart, "Video OCR: A survey and practitioner's guide," *The Kluwer International Series in Video Computing*, vol. 6: Video Mining, 2003.
[21] H. Li and D. Doermann, "Text enhancement in digit video using multiple frame integration," in *Proc. of 7th ACM Int. Conf. on Multimedia*, Orlando, Florida, Oct. 1999, pp. 19–22.
[22] J. Xi, X.-S. Hua, X.-R. Chen, W. Liu, and H.-J. Zhang, "A video text detection and recognition system," in *Proc. IEEE ICME 2001*, Tokyo, Japan, Aug. 22–25, 2001.
[23] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in *Proc. IEEE CVPR 2004*, Washington, DC, Jul. 2004, pp. 366–373.
[24] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Trans. Image Processing*, vol. 12, no. 7, pp. 796–807, Jul. 2003.
[25] A. Polesel, G. Ramponi, and V. J. Mathews, "Image enhancement via adaptive unsharp masking," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 505–510, Mar. 2000.
[26] K. Fukanaga, *Introduction to Statistical Pattern Recognition*. New York: Academic Press, 1972.
[27] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Mar. 1979.
[28] O. Trier, A. Jain, and T. Taxt, "Feature extraction methods for character recognition—A survey," *Pattern Recognition*, vol. 29, no. 4, pp. 641–662, 1996.
[29] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 489–497, May 1990.
[30] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.
[31] K. Crammer and Y. Singer, "On the algorithmic implementation of multi-class kernel-based machines," *J. of Machine Learning Research*, vol. 2, pp. 265–292, Dec. 2001.
[32] G. Miao, G. Zhu, S. Jiang, Q. Huang, C. Xu, and W. Gao, "A real-time score detection and recognition approach for broadcast basketball video," in *Proc. IEEE ICME 2007*, Beijing, China, Jul. 2–5, 2007, pp. 1691–1694.
[33] Y. Takahashi, N. Nitta, and N. Babaguchi, "User and device adaptation for sports video content," in *Proc. IEEE ICME 2007*, Beijing, China, Jul. 2–5, 2007, pp. 1051–1054.
[34] A. Kokaram, N. Rea, R. Dahyot, A. M. Tekalp, P. Bouthemy, P. Gros, and I. Sezan, "Browsing sports video," *IEEE Signal Processing Magazine*, pp. 46–58, March 2006.
[35] Z. Xiong, X. S. Zhou, Q. Tian, Y. Rui, and T. S. Huang, "Semantic retrieval of video," *IEEE Signal Processing Magazine*, pp. 18–27, March 2006.

Huang-Chia Shih (S'03-M'08) 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 (NTHU), Hsinchu, Taiwan, in 2002. He is currently pursuing the Ph.D. degree in electrical engineering at NTHU, 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 Sep. 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 member of IEEE.

Chung-Lin Huang received his B.S. degree in Nuclear Engineering from the National Tsing-Hua University, Hsin-Chu, Taiwan, ROC, in 1977, and M.S. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan, ROC, in 1979 respectively. He obtained his 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 Distinguish 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 form the CVGIP Society, Taiwan, ROC. In Dec. 1997, he received the best paper award from 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.