Rushes Video Summarization by Audio-Filtering Visual Features

Yin-Fu Huang and Wei-Chung Chen

***Abstract*—In this paper, we propose a video summarization system for analyzing basketball game videos. In contrast to previous video analysis technologies employing only visual and motion features to do video filtering, we add audio features to do video summarization in the system. First, we extract replay highlights by special effect detection. Next, we filter landscape shots using color range pixel and fast motion activity. Then, the corresponding audio features extracted from these landscape shots are used to identify landscape shot highlights by an SVM. Finally, we integrate the replay highlights and landscape shot highlights to complete the video summarization. From the experimental results, we find that the accuracy on the special effect detection, landscape shot extraction, and landscape shot highlight detection is very high. Thus, the final video summarization has high recall values on highlight extraction.**

***Index Terms*—Basketball video, landscape shot highlight, replay highlight, video summarization.**

I. Introduction

The multimedia community needs an effective management scheme for handling the increasing amount of sports videos from TV broadcasts. Researchers have proposed many techniques to take full advantage of the characteristics that sports videos have, such as temporal structures, recurrent events, consistent features, and a fixed number of camera views. In this paper, we propose a video summarization system for analyzing basketball game videos as a basketball game is very popular in North American. With the huge collections of these video data, how to manage them becomes an important issue. In addition to visual features employed in the previous video analysis technologies, we would add audio features to do video summarization in our system.

Video analysis technologies have been widely investigated for providing users with more rapid and convenient access to the interesting or important part of videos. Especially, accompanying with the increasing amount of the videos in TV, Internet, et al., the demand to automatically analyze and summarize videos becomes more urgent. With these technologies, users can save time by only watching the summarization of concerned programs. Besides, Internet or wireless communication networks can also be benefited from these technologies because their limited bandwidth is not suitable for transmitting a large amount of data. So far, some applications of video summarization have been investigated. For sports games, Wang *et al.* proposed the replay detection on soccer sports [1]. For news, TRECVID is the most popular provider from which Cunha *et al*. compared their work on the datasets [2]; they focused on automatic rushes video summarization with BBC Rushes datasets, and compared with the methods CityU [3], Lip6 [4], and Nii [5]. Besides, some researchers focused on special effect detection [6], [7], and used special effects to extract highlights. Then, Luo *et al.* [8] and Zhao *et al.* [9] summarized sports videos by detecting slow-motion replays to extract highlights. Nevertheless, analyzing general rushes videos is still an open problem because of the variance and diversity of videos in different domains.

In this paper, we retrieve the videos from NBA, and segment these videos into a series of consecutive pictures first. Then, the shot boundaries between pictures are detected. After the shot detection, we exclude bright pictures (or noises) using color histogram. Next, we detect special effects and landscape shots. Special effects are made by human, within which the replay highlights can be extracted. A landscape is the scene that captures the full dynamic basketball game in playing. First, landscape shots are found, from which the corresponding audio features including "MFCC", "LPC", and "ZCR" can be extracted. These feature vectors can be used to identify landscape shot highlights by an SVM. Finally, we integrate the replay highlights and landscape shot highlights to complete the video summarization.

The remainder of this paper is organized as follows. Section II presents the basic concepts of video summarization systems. Section III introduces the system architecture and the features used in the system. Section IV describes the approaches used in the video summarization system. Section V discusses experimental results. Finally, Section VI draws some conclusions and discusses future work.

II. Basic Concepts

Video summarization providing concise and informative video summaries for people to browse and manage video files effectively has received more and more attention in recent years. Basically, there are two kinds of video summarization: static video summary and video compression.

A. Static Video Summary

A static video summary is composed of a set of salient images (or key frames) extracted or synthesized from an original video. Based on the way a key frame is extracted, the existing work in this area can be categorized into three classes:

Manuscript received February 13, 2014; revised May 10, 2014. This work was supported by National Science Council of R.O.C under Grant NSC100-2218-E-224-011-MY3.

Yin-Fu Huang is with National Yunlin University of Science and Technology, Yunlin, Taiwan 640 (e-mail: huangyf@yuntech.edu.tw).

Wei-Chung Chen is with Chung Shan Institute of Science and Technology, Taoyuan, Taiwan 325 (e-mail: evolution.bu@gmail.com).

shot-based, sampling-based, and segment-based.

For shot-based approaches, a shot is defined as a video segment taken from a continuous period. A natural and straightforward way is to extract one or more key frames from each shot using low-level features such as color or motion. A typical approach was proposed in [10] where key frames are extracted in a sequential fashion via thresholds. The first frame in each shot is always chosen, and then a new key frame is extracted when the color-histogram difference from the preceding frame exceeds a certain threshold. However, one drawback of the shot-based approach is that it does not scale up well for a long video.

For sampling-based approaches, key frames are either randomly chosen or uniformly sampled from an original video. The video Magnifier [11] and the Mini Video [12] systems are two examples. This approach is the simplest way to extract key frames, but such an arrangement may fail to capture the real video content, especially when a video is highly dynamic.

For segment-based approaches, efforts are made in extracting key frames at a higher level, referred to as the segment level. Various clustering-based extraction schemes have been proposed. In these schemes, segments are first generated from frame clustering, and then the frames closest to the centroid of each qualified segment are chosen as key frames [13], [14].

B. Video Compression

A video is composed of images, audio, and texts. It can be converted into frames and the video play rate is at least 25 to 30 frames per second. A video can also be compressed. Video compression refers to reducing the quantity of data used to represent video images and is a straightforward combination of image compression and motion compensation. There are four video compression methods: discrete cosine transform (DCT), vector quantization (VQ), fractal compression, and discrete wavelet transform (DWT).

Discrete cosine transform is a lossy compression algorithm that samples an image at regular intervals, analyzes the frequency components present in the sample, and discards those frequencies which do not affect the image as human eyes perceive it. DCT is the basis of standards such as JPEG, MPEG, H.261, and H.263.

Vector quantization is also a lossy compression that looks at an array of data, instead of individual values. It generalizes what it sees, and compresses redundant data, while at the same time retaining the desired object or the original intent of a data stream.

Fractal compression is a form of VQ. The compression is performed by locating self-similar sections of an image, using a fractal algorithm.

Discrete wavelet transform, like DCT, mathematically transforms an image into frequency components. The process is performed on an entire image, which differs from DCT working on smaller pieces of the desired data. The result is the hierarchical representation of an image where each layer represents a frequency band.

III. System Overviews

In this section, we introduce the system architecture and the features used in the video summarization system. As illustrated in Fig. 1, the system architecture consists of three major parts: 1) replay highlight extraction, 2) landscape shot extraction, and 3) landscape shot highlight detection. First, replay highlights are usually enclosed by a pair of special effects, so special effects must be detected in advance. Next, we filter landscape shots using color range pixel and fast motion activity. Then, the corresponding audio features including "MFCC", "LPC", and "ZCR" can be extracted from these landscape shots. These feature vectors can be used to identify landscape shot highlights by an SVM. Finally, we integrate the replay highlights and landscape shot highlights to complete the video summarization.

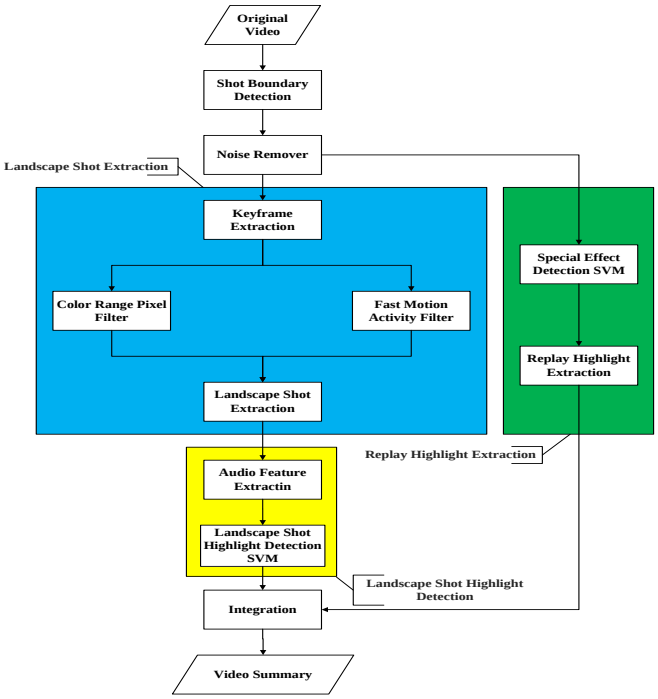

Fig. 1. System architecture.

A. Visual Features

The visual features used to train the SVM for special effect detection include Color Layout Descriptors (CLD) and Color Structure Descriptors (CSD) defined in the MPEG-7 Specification [15]-[18]. They are described as follows.

1) Color layout

The color layout descriptor (CLD) specifies the spatial distribution of colors for high-speed retrieval and browsing. It can be also applied both to a whole image and to any connected or unconnected parts of an image with arbitrary shapes. The color space is YCbCr with quantization to 8 bits, and the feature extraction process consists of two parts: grid-based representative color selection and a DCT used to represent features in a frequency domain. An input picture is divided into 64 (8×8) blocks and their average colors are derived. The derived average colors are transformed into a series of coefficients by performing 8×8 DCT. The quantized coefficients are scanned in a zigzag manner to obtain the descriptor values. Here, 12 coefficients of color layout, 6 coefficients for luminance (Y) and 3 coefficients for each chrominance (Cb and Cr), are used to train the SVM for special effect detection.

2) Color structure

The color structure descriptor (CSD) [19] is another color descriptor selected in our work. It captures both color contents (similar to a color histogram) and information about the content structure. The main function is image-to-image matching for still-image retrieval, where an image may consist of either arbitrarily shaped, possibly disconnected, regions or a single rectangular frame.

Color structure information is embedded into the descriptor by taking into account all colors in a structuring window of 8×8 pixels while the extraction method slides over an image, instead of considering each pixel individually. Although the number of samples in the structuring window is always 64, the spatial extent of the structuring window can scale with the image size.

The color structure descriptor containing a 256-bin histogram is extracted directly from the image based on a 256-cell quantization of the hue-min-max-difference (HMMD) color space. Actually, 128, 64, or 32 bins can be computed based on the unification of the bins of the same 256-bin descriptor. The HMMD color space is non-uniformly quantized into 5 subspaces, and each color subspace is uniformly quantized along the Hue and Sum axes. Here, the 32-bin histogram of color structure is used to train the SVM for special effect detection.

B. Audio Features

Audio features are extracted usually faster than visual features. In the system, we use "MFCC", "LPC", and "ZCR" to train the SVM for landscape shot highlight detection. They are useful features for speech recognition, and were very popular for the audio researches in recent years. They are described as follows.

1) MFCC (Mel-frequency cepstrum coefficients)

MFCCs are known and popular in the short-term power spectrum of a sound. They are based on the known variation of human ear's critical bandwidths with frequency. The MFCC technique makes use of two types of filters, namely, linearly spaced filters and logarithmically important characteristics of speeches; a signal is expressed in the Mel-frequency scale. MFCCs are commonly used as features in speech recognition systems, such as the systems which can automatically recognize numbers spoken into a telephone. They are also common in speaker recognition, which is the task of recognizing people from their voices. MFCCs are also increasingly finding uses in music information retrieval applications such as genre classification, audio similarity measures, etc. Here, 13 MFCCs (or features) are used to train the SVM for landscape shot highlight detection.

2) LPC (Linear predictive coding)

Linear predictive coding (LPC) is a tool used mostly in audio signal processing and speech processing for representing the spectral envelope of a digital signal of speech in compressed form, using the information of a linear predictive model. It is one of the most powerful speech analysis techniques, and one of the most useful methods for encoding good quality speech at a low bit rate and provides extremely accurate estimates of speech parameters. LPC is generally used for speech analysis and resynthesis. Here, 10 LPC coefficients (or features) are used to train the SVM for landscape shot highlight detection.

3) ZCR (Zero-crossing rate)

ZCR is the number of zero-crossing of the waveform within a given frame. In general, ZCR is large if an environment is full of noises. Here, we use this characteristic to detect audience acclaim. It can be expressed as follows.

$$ZCR_t = \frac{1}{2}\sum_{n=1}^{N}|sgn(x[n])\text{-}sgn(x[n\text{-}1])| \quad (1)$$

IV. Video Summarization Approaches

In this section, we describe the characteristics of basketball videos and the approaches used in the video summarization system, as mentioned in Section III.

A. Characteristics of Basketball Videos

A basketball video is presented with consecutive scenes (or shots). These shots could be overlooking, long distance, replay, interviews *et al.* Among them, the highlights in a basketball video are the most ones to draw the attention of the audience. However, these highlights are not consecutive and appear at different time. Through further observation, some common features of the highlights can be found, such as noise around the highlights and the highlights enclosed by special effects. In addition, the highlights are usually located in landscape shots. Therefore, a specific algorithm is used to capture landscape shots.

B. Shot Boundary Detection

In order to detect special effects and landscape shots, the first step is to identify precise shot boundaries in a video. In general, there are two types of shot boundaries: abrupt also referred to as cuts, and gradual such as fades and dissolves. Here, we only focus on the abrupt shot boundary detection. In the past, many shot boundary detection algorithms using color histogram have been proposed and they achieved satisfied accuracy. For our method, we transfer the RGB color space to HSV color space first. The triple-color component (H, S, V) is uniformly quantized with 16 bins in H, and 4 bins in each S and V (i.e., 256 bins in total). In other words, the H is divided equally into 16 parts, the S into 4 parts, and the V into 4 parts as well. Then, we compare the color/intensity histograms of two consecutive frames, and identify a shot boundary if their difference exceeds a certain threshold.

C. Noise Remover

After the shot detection, we remove bright shots (or noises) as shown in Fig. 2. In the video summarization researches, removing noises is an essential step. Usually, the videos coming from peer-to-peer shared resources on the Internet go through compression processing, so they definitely have noises. In order not to reduce the accuracy of subsequent steps, it is necessary to remove these noises. In our basketball video, we observe that most noise shots have only one frame. If a shot with only one frame is found, we transform the frame into a gray level and then extract the HSV color histogram. We only take the V component to determine whether it is a noise frame since over-brightness represents a noise. Finally, after

replacing the noise frame with the last frame in the preceding shot, we merge the preceding shot, the noise shot, and the following shot into a complete shot, in order to reduce the total number of shots.

Fig. 2. Bright shots.

D. Special Effect Detection

Special effects refer to the artificial effects that the post-processing is added in films to emphasize highlights. So far, some methods integrated highlights for more complete sports video summarization [20]. In most sport films, if players have wonderful performances, the replay fragments are usually produced in films. In general, these replay fragments (or replay highlights) are enclosed by a pair of special effects. Thus, before extracting replay highlights, we must identify special effects first, as shown in Fig. 3. Since a special effect is added artificially and accompanied by rapid changes in color, a series of regular shots would be produced. Here, we found that a special effect is always continuous for five shots. Therefore, we use a sliding window <$w1$, $w2$, $w3$, $w4$, $w5$> to detect special effects, which is a series of shots. Then, we extract the color layout and color structure from each frame of shot <$w3$>, and take their averages as the features of shot <$w3$>. Besides, we also extract the number of frames for each shot <$w1$>, <$w2$>, …, <$w5$>, respectively. Thus, we have totally 49 features (5 features for the number of frames, 12 features for color layout, and 32 features for color structure) for the SVM to detect special effects.

Fig. 3. Special effects.

E. Key-Frame Extraction

Selecting a key-frame to represent its shot for further processing has been an important issue addressed in the previous work [21]-[23]. In general, the simplest way in the key-frame extraction is to select the first frame of a shot as the key-frame for the shot. However, since a shot is composed of a sequence of frames with multiple continuous camera operations, this simple strategy may not obtain good results. Here, we select a frame as the key-frame such that its color is the most similar to the average color histogram of a shot. The reason of using color to select key-frames is that the feature "Color Range Pixel" in key-frames would be used to detect landscape shots.

F. Landscape Shot Extraction

Since most important events are presented in landscape shots, extracting relevant landscape shots is the key point of video summarization. Landscape shots as shown in Fig. 4 are shots with full views in a basketball video. Here, we filter landscape shots using color range pixel and fast motion activity. We observe most basketball films still have the same floor color although they play in different basketball courts. Here, we use many fragmented pictures of basketball courts to estimate the color histogram of the pixels in courts (i.e., the color range pixel we call) as thresholds. Thus, we can calculate the color histogram of key-frames to filter landscape shots.

Some researchers ever used motion vectors to detect similarity shots [24]. Here, at the same time, we also calculate the motion vectors of each shot, called the fast motion activity. The fast motion activity is one of the motion activity defined in the MPEG7 specification [25]. In the MPEG7, the motion activity [26] consists of four definitions; i.e., intensity of activity, direction of activity, spatial distribution of activity, and temporal distribution of activity. Here, we adopt intensity of activity as the fast motion activity used in our algorithm. The fast motion activity of a shot can be regarded as the average motion value between adjacent frames in the shot by macro block-based computing.

Fig. 4. Landscape shots.

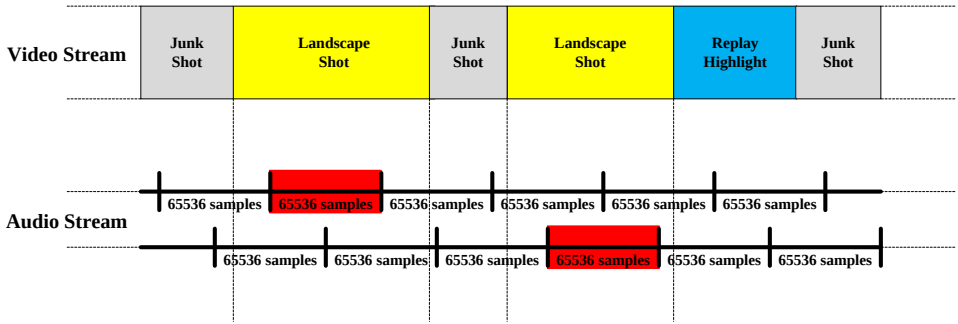

Fig. 5. Corresponding audio from landscape shots.

G. Landscape Shot Highlight Detection

First, we map the landscape shots back to the original videos, and extract the corresponding audio, respectively. We represent the audio clips with <$c1$, $c2$, …, cn> where each clip with 65536 samples is about 4.2 sec. and overlaps its adjacent clips, as shown in Fig. 5. Then, the audio features including "MFCC", "LPC", and "ZCR" are extracted from these clips, which can be used to identify landscape shot highlights by an SVM.

H. Integration

Finally, we integrate the replay highlights and landscape shot highlights to complete the video summarization. In order to ensure the time consistency of these highlights, we put them one-by-one according to their frame numbers.

V. Experiments

The training/test videos used in the experiments are downloaded from the website with NBA bit-stream spaces. All the videos are compressed in the MPEG-4 format with frame size 480×270 and frame rate 30 fps. We select three videos from the NBA games randomly, which are Kings vs. Pistons on February 17, 2012, Kings vs. Lakers on March 2, 2012, and Suns vs. Hawks on February 6, 2012. These videos are divided into the first half and second half, and the length of each half is about 40 to 50 minutes. Three performance measures including precision, recall, and accuracy are applied to analyze the special effect detection, landscape shot extraction, and landscape shot highlight detection. Finally, we also present the compression rates and recall values of video summarization.

A. Special Effect Detection

In our proposed method, a correct replay highlight extraction means a pair of correctly detected special effects. From the experimental results as shown in Table I, we find that two features such as color layout and color structure indeed correct the deficiencies of only using the window to detect special effects. For example, some special effects are advertising effects, but not ball game effects. However, after we consider the visual features, the precision reaches 100% and the accuracy is greatly improved (i.e., more than 99%), as shown in Table I. The reason with high accuracy is that we have many non-special-effect shots in the videos, and they can be easily detected.

TABLE I: Special Effect Detection Results

Date	Game	Half	Precision	Recall	Accuracy
2/17/2012	Kings vs. Pistons	first	1.000	0.824	0.997
		second	1.000	0.846	0.995
3/2/2012	Kings vs. Lakers	first	1.000	0.833	0.997
		second	1.000	0.889	0.995
2/6/2012	Suns vs. Hawks	first	1.000	0.842	0.994
		second	1.000	0.889	0.995

B. Landscape Shot Extraction

After removing noises, the video recording the game Kings vs. Pistons on February 17, 2012 includes 1647 shots, of which only 269 ones are landscape shots. After the filtering (i.e., using the color range pixel and the fast motion activity), we find that the number of correctly filtered landscape shots is 249, the number of falsely filtered landscape shots is 56, and the number of missed landscape shots is 20. The other video recording the game Kings vs. Lakers (or Suns vs. Hawks) includes 1442 (or 1361) shots, of which only 137 (or 103) ones are landscape shots. After the filtering, we find that the number of correctly filtered landscape shots is 134 (or 91), the number of falsely filtered landscape shots is 41 (or 32), and the number of missed landscape shots is 3 (or 12). All the filtered results as shown in Table II have very high accuracy (i.e., more than 94%). As similar to the mention in Section V, we have many non-landscape shots in the videos, and they can be easily filtered out.

TABLE II: Landscape Shot Extraction Results

Date	Game	Half	Precision	Recall	Accuracy
2/17/2012	Kings vs. Pistons	first	0.820	0.953	0.962
		second	0.813	0.900	0.945
3/2/2012	Kings vs. Lakers	first	0.750	0.969	0.971
		second	0.780	0.986	0.968
2/6/2012	Suns vs. Hawks	first	0.620	1.000	0.963
		second	0.893	0.806	0.951

C. Landscape Shot Highlight Detection

To identify landscape shot highlights by an SVM, the corresponding audio features are extracted from these landscape shots. To verify the detected landscape shot highlights, we retrieve the truth information recorded in the NBA official website. As a result, as shown in Table III, more than 65% of the ground truth appears in the detected landscape shot highlights. Besides, the results also present very high accuracy (i.e., more than 82%). In other words, many landscape shots without highlights can be easily discriminated.

TABLE III: Landscape Shot Highlight Detection Results

Date	Game	Half	Precision	Recall	Accuracy
2/17/2012	Kings vs. Pistons	first	0.706	0.698	0.868
		second	0.549	0.770	0.899
3/2/2012	Kings vs. Lakers	first	0.600	0.778	0.874
		second	0.621	0.701	0.823
2/6/2012	Suns vs. Hawks	first	0.596	0.779	0.885
		second	0.566	0.663	0.843

D. Results of Video Summarization

In the video summarization, we integrate the replay highlights and landscape shot highlights by a time sequence. Based on the audio clips used in the landscape shot highlight SVM, we extend these clips for more five seconds on the front and rear of landscape shots, respectively. The extended shots can be mapped back to the original videos for retrieving the corresponding frames. Extending the shot length is to ensure that the viewing quality of the summarization results is smooth and flowing. By the way, it also increases user awareness and video integrity. We calculate the compression

rates and recall of the summarized videos over the original videos, as shown in Table IV. Here, we find that the final video summarization has high recall values on highlight extraction, and the compression rates could be extremely different since the highlights in basketball videos vary from each other.

TABLE IV: COMPRESSION RATES AND RECALL OF VIDEO SUMMARIZATION

Date	Game	Half	Compression Rates	Recall
2/17/2012	Kings vs. Pistons	first	13.881	0.767
		second	6.768	0.874
3/2/2012	Kings vs. Lakers	first	6.613	0.852
		second	5.449	0.883
2/6/2012	Suns vs. Hawks	first	10.065	0.837
		second	13.838	0.716

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a video summarization system for providing users with more rapid and convenient access to the interesting or important part of basketball videos. First, we detect special effects and extract inside replay highlights. Then, we detect landscape shots capturing the full dynamic basketball game in playing. Finally, the replay highlights and landscape shot highlights are integrated together to complete the video summarization. From the experimental results, we find that the accuracy on the special effect detection, landscape shot extraction, and landscape shot highlight detection is very high. Thus, the final video summarization has high recall values on highlight extraction.

So far, we only implement the video summarization method on basketball videos. In the future, we hope to explore the possibility of generalizing the approaches used in the video summarization system to sports or even generic videos.

REFERENCES

[1] J. Wang, E. Chng, and C. Xu, "Soccer replay detection using scene transition structure analysis," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2005, pp. 433-436.

[2] T. O. Cunha, F. G. H. de Souza, A. de A. Araújo, and G. L. Pappa, "Rushes video summarization based on spatio-temporal features," in *Proc. the 27th Annual ACM Symposium on Applied Computing*, 2012, pp. 45-50.

[3] T. Wang, Y. Gao, J. Li, P. P. Wang, X. Tong, W. Hu, Y. Zhang, and J. Li, "THU-ICRC at rush summarization of TRECVID 2007," in *Proc. the International Workshop on TRECVID Video Summarization*, 2007, pp. 79-83.

[4] M. Detyniecki and C. Marsala, "Video rushes summarization by adaptive acceleration and stacking of shots," in *Proc. the International Workshop on TRECVID Video Summarization*, 2007, pp. 65-69.

[5] D. D. Le and S. Satoh, "National Institute of Informatics, Japan at TRECVID 2007: BBC rushes summarization," in *Proc. the International Workshop on TRECVID Video Summarization*, 2007, pp. 70-73.

[6] H. Pan, B. Li, and M. I. Sezan, "Automatic detection of replay segments in broadcast sports programs by detection of logos in scene transitions," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2002, pp. 3385-3388.

[7] X. Tong, H. Lu, Q. Liu, and H. Ji, "Replay detection in broadcasting sports video," in *Proc. the 3rd International Conference on Image and Graphics*, 2004, pp. 337-340.

[8] M. Luo, Y. F. Ma, and H. J. Zhang, "Pyramidwise structuring for soccer highlight extraction," *in Proc. the 4th Pacific Rim Conference on Multimedia*, 2003, pp. 945-949.

[9] Z. Zhao, S. Jiang, Q. Huang, and Q. Ye, "Highlight summarization in soccer video based on goalmouth detection," in *Proc. the Asia-Pacific Workshop on Visual Information Processing*, 2006.

[10] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognition*, vol. 30, pp. 643-658, 1997.

[11] M. Mills, J. Cohen, and Y. Y. Wong, "A magnifier tool for video data," in *Proc. the ACM SIGCHI Conference on Human Factors in Computing Systems*, 1992, pp. 93-98.

[12] Y. Taniguchi, A. Akutsu, Y. Tonomura, and H. Hamada, "An intuitive and efficient access interface to real-time incoming video based on automatic indexing," in *Proc. the 3rd ACM International Conference on Multimedia*, 1995, pp. 25-33.

[13] A. Girgensohn and J. Boreczky, "Time-constrained keyframe selection technique," *Multimedia Tools and Applications*, vol. 11, pp. 347-358, 2000.

[14] S. Uchihashi, J. Foote, A. Girgensohn, and J. Boreczky, "Video manga: generating semantically meaningful video summaries," in *Proc. the 7th ACM International Conference on Multimedia*, 1999, pp. 383-392.

[15] ISO/IEC 15938-1, *Information Technology - Multimedia Content Description Interface-Part 1: Systems*, 2002.

[16] ISO/IEC 15938-2, *Information Technology - Multimedia Content Description Interface-Part 2: Description Definition Language*, 2002.

[17] ISO/IEC 15938-3, *Information Technology - Multimedia Content Description Interface-Part 3: Visual*, 2002.

[18] ISO/IEC 15938-4, *Information Technology - Multimedia Content Description Interface-Part 4: Audio*, 2002.

[19] A. Buturovic, "MPEG7 color structure descriptor," Technical Report, VizIR Project, pp. 1-2, 2005.

[20] D. Tjondronegoro, Y. P. P. Chen, and B. Pham, "Integrating highlights for more complete sports video summarization," *IEEE Multimedia*, vol. 11, pp. 22-37, 2004.

[21] J. H. Lee, G. G. Lee, and W. Y. Kim, "Automatic video summarizing tool using MPEG-7 descriptors for personal video recorder," *IEEE Transactions on Consumer Electronics*, vol. 49, pp. 724-749, 2003.

[22] J. Rong, W. Jin, and L. Wu, "Key frame extraction using inter-shot information," in *Proc. the 4th IEEE International Conference on Multimedia and Expo*, 2004, pp. 571-574.

[23] X. Zhu, A. K. Elmagarmid, X. Xue, L. Wu, and A. C. Catlin, "InsightVideo: toward hierarchical video content organization for efficient browsing, summarization and retrieval," *IEEE Transactions on Multimedia*, vol. 7, pp. 648-666, 2005.

[24] L. Y. Duan, M. Xu, Q. Tian, C. S. Xu, and J. S. Jin, "A unified framework for semantic shot classification in sports video," *IEEE Transactions on Multimedia*, vol. 7, pp. 1066-1083, 2005.

[25] S. F. Chang, T. Sikora, and A. Puri, "Overview of the MPEG-7 standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 688-695, 2001.

[26] S. Jeannin and A. Divakaran, "MPEG-7 visual motion descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 720-724, 2001.

Yin-Fu Huang received the B.S. degree in computer science from National Chiao-Tung University in 1979, and the M.S. and Ph.D. degrees in computer science from National Tsing-Hua University in 1984 and 1988, respectively. He is currently a professor in the Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology. Between July 1988 and July 1992, he was with Chung Shan Institute of Science and Technology as an assistant researcher. His research interests include database systems, multimedia systems, data mining, mobile computing, and bioinformatics.

Wei-Chung Chen received his B.S. degree in computer sciences from Shih Chien University and M.S. degree in computer sciences from National Yunlin University of Science and Technology in 2010 and 2012, respectively. He is currently serving in Chung Shan Institute of Science and Technology. His major areas of interests are multimedia systems and data mining.