A Personal Videocasting System with Intelligent TV Browsing for a Practical Video Application Environment

Sang-Kyun Kim, Jinguk Jeong, Hyoung-Gook Kim, and Min Gyo Chung

In this paper, a video broadcasting system between a home-server-type device and a mobile device is proposed. The home-server-type device can automatically extract semantic information from video contents, such as news, a soccer match, and a baseball game. The indexing results are utilized to convert the original video contents to a digested or arranged format. From the mobile device, a user can make recording requests to the home-server-type devices and can then watch and navigate recorded video contents in a digested form. The novelty of this study is the actual implementation of the proposed system by combining the actual IT environment that is available with indexing algorithms. The implementation of the system is demonstrated along with experimental results of the automatic video indexing algorithms. The overall performance of the developed system is compared with existing state-of-the-art personal video recording products.

Keywords: Video indexing, video shot classification, audio shot classification, video navigation, video digest, video highlight.

Manuscript received Mar. 9, 2008; revised Oct. 18, 2008; accepted Dec. 10, 2008.

This work was supported by a special research grant from Seoul Women's University (2008), Rep. of Korea.

Sang-Kyun Kim (phone: +82 31 330 6443, email: goldmunt@gmail.com) is with the Computing Engineering Department, Myongji University, Gyeonggi-do, Rep. of Korea.

Jinguk Jeong (email: jinguk.jeong@samsung.com) is with Digital Media R&D Center, Samsung Electronics, Suwon, Rep. of Korea.

Hyoung-Gook Kim (email: hkim@kw.ac.kr) is with the Wireless Communications Engineering Department, Kwangwoon University, Seoul, Rep. of Korea.

Min Gyo Chung (email: mchung@swu.ac.kr) is with the Department of Computer Science, Seoul Women's University, Seoul, Rep. of Korea.

I. Introduction

The deluge of video contents from digital broadcasting and the emergence of an industry involving personal video recorders (PVRs), electronic program guides (EPGs), and large-size storage are changing the paradigm of how to watch TV. For example, Tivo and several Internet TV broadcasting services with a time-shift function are already changing TV viewing habits by allowing viewers to watch any content anytime. Viewers can record any broadcasting contents through an EPG and watch them in a digested or arranged format. Since many recorded contents take much time to watch after recording, there is a substantial demand from viewers to see digests or storyboards of recorded contents for quick browsing.

Furthermore, there is a strong industry trend (e.g., DLNA: Digital Living Network Alliance) toward a system involving a home server, a powerful PC, and a PVR, utilizing a fast networking capability that controls home appliances and automatically stores and indexes multimedia data. In addition, there are strong network infra-developments, such as ZigBee, ultra-wideband (UWB), WiFi, high-speed downlink packet access (HSDPA), and wireless broadband Internet (WiBro), which enable mobile devices to be easily connected to home devices through the Internet. Thus, more convenient personalized media services are expected to be provided to fulfill users' needs to enjoy contents anytime, anywhere, and in any amount. Developed functionalities should provide more active and convenient tools for mobile TV watchers than existing mobile TV services, like digital media broadcasting (DMB) service or LocationFree TV from Sony.

Video indexing is used to automatically extract and analyze video content information. Research on video indexing has a

long history. There have been studies on classifying video shot types, such as video scenes and event detection. Extraction of semantic information has shown promise for sports and news contents. For example, there have been studies on the detection of important events from sports programs [1], [2] and the generation of a navigation map of scene units by detecting article boundaries in news programs [3]-[6]. The main goal of these studies has been to minimize the gap between video semantic information sought by users and information from the low-level features extracted from videos.

In this paper, a video broadcasting system between a personal home-server-type device and a hand-held device is proposed. A home-server-type recorder or PVR can automatically extract some semantic information (video indexing) from video contents, such as news, a soccer match, and a baseball game. The indexing results are utilized to convert original video contents to a digested or arranged format. Though many video indexing algorithms have been proposed [1]-[11], the use of these algorithms is quite limited in the market as an actual consumer device. To implement functionalities for devices with low CPU power and memory capacity, such as PVRs, the indexing algorithms are optimized to have minimum complexity. Both visual and audio information are extracted to index videos efficiently. Since video indexing technologies are well saturated, it is worth focusing on the actual implementation in a practical consumer electronic environment rather than introducing a new and complex video indexing algorithm.

From a mobile device, a user can make recording requests to the home-server-type devices and can then watch and navigate recorded video contents in a digested form using the provided mobile graphic user interface.

In section II of this paper, simple and practical shot type detection algorithms are proposed to analyze soccer, baseball, and news video contents. Section III explains how the proposed personal videocasting system architecture is implemented. In section IV, experimental results are demonstrated to prove the reliability of the proposed algorithms. In addition, the overall performance is compared with existing state-of-the-art PVR products. Concluding remarks are given in section V.

II. Automatic Video Indexing Algorithms

Shot type analysis utilizes an intrinsic characteristic of a video shot segment, for example, an anchorperson shot from a news program, a penalty area shot in a soccer match, or a pitching shot in a baseball game. The shot type of an audio signal plays an important role in characterizing the semantic information of a video segment. Four types of audio shots, namely, silence, music, speech, and excitement, are presented. Finally, a method to detect important events using information from video/audio indexing is presented in this section.

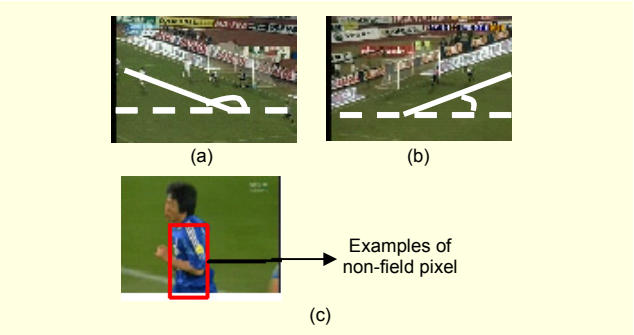

Fig. 1. (a), (b) Examples of penalty lines and (c) a key frame of a close-up shot.

1. Shot Type Analysis of Soccer Video

In a soccer match, most important events, such as a goal kick happen in the vicinity of penalty areas. Moreover, the camera usually takes close-up shots of a soccer player who has scored a goal along with the spectator areas right after the goal event. Therefore, the penalty area shot and close-up shot play an important role in extracting important events of a soccer match. Close-up shots include magnified shots of players or referees, and camera shots of non-field areas containing spectators or coaches.

Penalty areas can be detected by using penalty lines in a soccer field. Because the locations of cameras capturing a soccer match are usually fixed during the match, the penalty lines are skewed by a fixed angle as shown in Fig. 1. By measuring the angles empirically, the penalty lines are found to be skewed by 145 to 175 degrees when a goal post appears to the right in a scene, as shown in Fig. 1(a). They are skewed by 5 to 34 degrees when a goal post appears to the left, as shown in Fig. 1(b). Twenty soccer videos were observed to determine the penalty areas angles.

$$T = \frac{\sum_{i=0}^{N \times N} Y(i)}{N \times N} \times a, \qquad (1)$$

where a is a threshold constant.

Penalty area shots can be detected by using the angles of the penalty lines. The proposed algorithm first binarizes key frames and then calculates the angles of detected lines using a Hough transform. Because lines on a soccer field are usually bright white, a binarization is performed using brightness values (Y). A key frame is divided into $N \times N$ blocks, and then the brightness threshold (T) is calculated per block (1). The

binarization ends by comparing each pixel with these brightness thresholds. The threshold for the binarization is determined by pixels in a 32×32 local window which are 1.2 times higher in contrast than the average. The binarized frames are supplied to the Hough transform to calculate the angles of the lines. The Hough transform extracts the biggest accumulated (θ, ρ), which is obtained by substituting each input coordinate to a line equation, $x\cos\theta + y\sin\theta = \rho$. If the number of pixels satisfying the maximum (θ, ρ) is greater than a pre-determined threshold, and the detected line angle satisfies the angle conditions of a penalty line, then the shot is classified as a penalty area shot.

A close-up shot usually magnifies players, spectators, or coaches so that pixels from the close-up shot contain a relatively large amount of non-field colors as shown in Fig. 1(c). In addition, a player's upper body and spectators are often located in the lower portion of the image frame.

The proposed algorithm to detect a close-up shot is implemented as follows. First, the dominant colors of key frames are extracted to obtain a field color. The most prevalent color is considered the field color. It is then compared to a pre-trained field color model (HSV ranges such as H[0.18, 0.4], S[0.1, 1], V[0.2, 1]) [12]. If the difference between the color model and the most prevalent color is substantial, the shot is judged to be a close-up shot. Otherwise, the most prevalent color is used to extract the ratio of field in the spatial window. The spatial window slides through the bottom half of a frame. If at least one of the spatial windows contains non-field colors of more than a predefined threshold, the shot is judged to be a close-up shot. Shots other than the detected close-up shots are classified as long-view shots.

2. Shot Type Analysis in Baseball

A baseball game is composed of play segments and break segments. In baseball, a play segment starts with a pitching shot and ends with a close-up shot magnifying players or spectators. Therefore, detecting the pitching shots and close-up shots is a key factor in analyzing a baseball game video.

A pitching shot can be taken from behind the pitcher toward the catcher. However, it is difficult to detect pitching shots using a pre-defined model due to differences in baseball parks, field colors, and weather (illumination conditions). Here, a pitching shot model is adaptively generated for each game in order to overcome such problems. Figure 2 shows a flow diagram of a pitching shot detection algorithm. First, key frames of shots are clustered using an SOM or *K*-means algorithm. Features for clustering are extracted using an edge component histogram (ECH) [13]-[15], which can effectively describe the spatial edge distribution of an image frame. After a

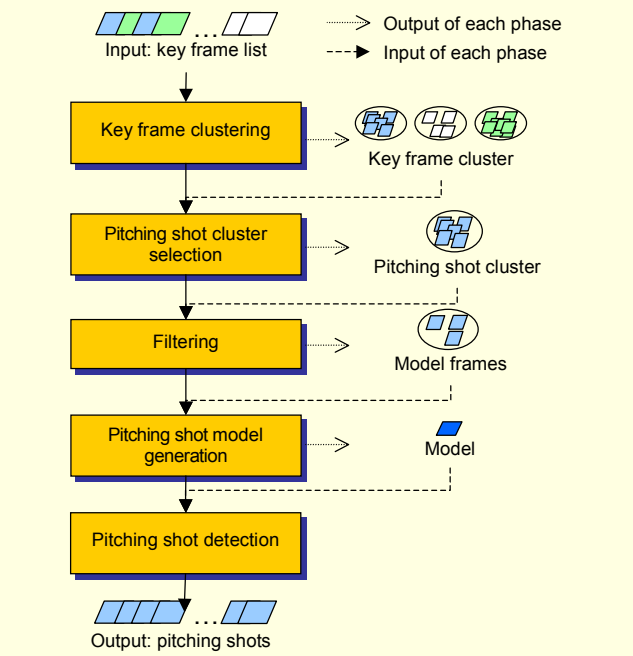

Fig. 2. Flow diagram of pitching shot detection algorithm.

clustering process, a cluster containing a pitching shot is selected using the characteristics of pitching shots that recur with a regular time interval and that are taken from a similar camera angle during the game period. The ECH and color structure descriptor (CSD) [13] are extracted to choose key frame sequences to satisfy the above conditions. In a filtering process, only key frames with similar features are selected. The pitching shot model is generated using the weighted average of the ECH and CSD from the key frames obtained during the filtering process. Finally, the generated pitching shot model is used to detect pitching shots among all of the key frames.

The close-up shot detection method is analogous to that of soccer match analysis which was described in the previous section. The only difference is the field color extraction method. In a baseball video, a field model should include a grass color model along with a ground color model. These models can be extracted from the detected pitching shots. To extract these colors, a key frame from a pitching shot is divided in half and then dominant colors are extracted from the lower half of the frame. The two most prevalent colors are selected and are used to generate a field color model for the corresponding baseball game. Finally, a spatial window slides though the key frames and detects close-up shots possessing a high ratio (more than 70% in a spatial window) of non-field colors. The high ratio was derived by analyzing more than 20 baseball videos.

3. News Anchorperson Shot Detection

Anchorperson shots are the focus of news video indexing.

An anchorperson shot is usually a starting point of a news story unit [6], [16], [17]. Faces of anchorpersons are key information for finding them in news videos. Face detection and feature extraction techniques are described in detail in [18]. After face feature extraction, face features are clustered by their similarities. Face feature clustering and face model generation are carried out as follows.

Step 1. Similarity Calculation. Calculate similarities on n face features composing $n \times n$ normalized correlation matrix:

$$PES(FD_i, FD_j) = \frac{FD_i \cdot FD_j}{\|FD_i\| \cdot \|FD_j\|}. \quad (2)$$

Step 2. Initial Cluster Creation. Among n nodes, if the similarity distance of any two nodes is more than a pre-defined threshold (0.85, which was determined by analyzing more than 20 news videos), two nodes are connected. A node can be included in several clusters.

Step 3. Face Cluster Merge. If a node is included in several clusters, the clusters are merged into one cluster.

Step 4. Removal of Single-Node Clusters. Clusters with a single node are removed from anchorperson face candidates.

Step 5. Selection of Anchorperson Model. The face feature that is close to the center point of each cluster is selected as a representative model of the cluster.

Each anchorperson cluster candidates can be confirmed by checking the regularity of anchorperson shot emergence. Since there are various kinds of anchorperson shots, the following more robust way to confirm the anchorperson shots is proposed.

Step 1. Sort anchorperson cluster candidates (M) by the number of nodes: $C_1 > C_2 > C_3 > \cdots > C_M$.

Step 2. Select the largest anchorperson cluster candidate: C_1.

Step 3. Calculate a standard deviation of C_1: $STDEV_1$.

Step 4. Anchor cluster = C_1.

Step 5. Select the next largest anchorperson cluster candidate: C_2.

Step 6. Calculate a standard deviation of C_1+C_2: $STDEV_2$.

Step 7. If $STDEV_2 < STDEV_1$, then anchor cluster = C_1+C_2; else, anchor cluster = C_1.

Step 8. Repeat steps 5 to 7 until no candidate cluster remains.

If an anchorperson shot cluster candidate is obtained from real anchorperson shots, the shots in it occur regularly throughout the news segment. Therefore, newly inserted shots reduce the standard deviation after they are merged with the current anchorperson cluster. Thus, the cluster would be confirmed as an anchorperson shot cluster. However, if a candidate cluster originates from other shots (such as report shots), these shots occur irregularly in the news segment and increase the standard deviation after they are merged with the current anchorperson cluster. Consequently, it is not considered an anchorperson cluster.

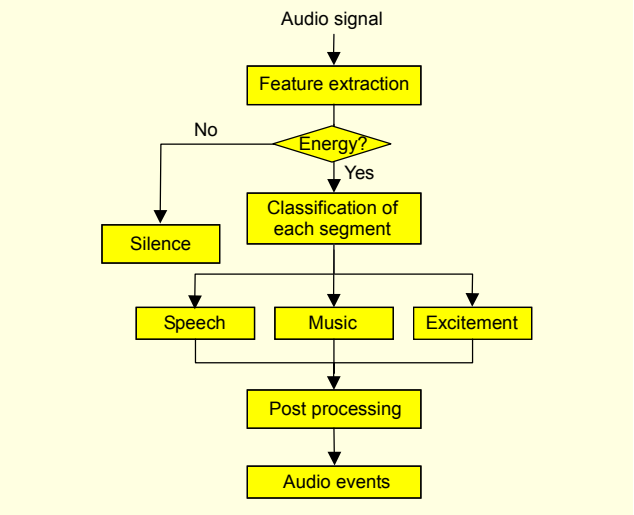

Fig. 3. Block diagram of audio events detection.

4. Audio Type Analysis

Audio analysis information plays a major role in video indexing due to its fast processing speed [8]-[10]. In this paper, audio signal segments are classified into the categories of music, speech, silence, and excitement that can be used to detect primary audio events in video indexing. Figure 3 explains the flow of audio segment classification.

A. Audio Features and Classifier

The input audio signal is divided into frames of 24 ms. Audio features are extracted from each frame. An audio segment is composed of 40 frames, totaling 0.96 s in length. The audio features from all the frames in a segment and their mean and variance are used as input to the audio segment classifier. For the audio classifier, a Gaussian mixture model (GMM) and a support vector machine (SVM) are used. The SVM classifier demonstrates better performance.

The audio features which are extracted from 40 frames in a segment and used as input to the classifier (together with their mean and variance) are the following [7], [11].

- Twenty mel-frequency cepstrum coefficients (MFCCs)
- Zero-crossing rate (ZCR)
- Short-time energy (STE)
- Spectral centroid (SC), which is the first-order moment of the spectrum at each frame
- Spectral rolloff (SR), defined as the frequency below which 85% power is concentrated
- Spectral flux (SF), which is the squared difference

between two adjacent frames of spectrum

- Mean of energies in five frequency sub-bands, which we define to be 0 Hz to 630 Hz, 630 Hz to 1,720 Hz, 1,720 Hz to 3,500 Hz, 3,500 Hz to 5,000 Hz, and 5,000 Hz to 8,000 Hz.

In this paper, SVM is used for audio classification. SVM was originally designed for binary classification [19]. Here, it is extended for K-class classification by constructing $K(K\text{-}1)/2$ "one-against-one" binary classifiers and combining them using the rule of majority voting. Each binary classifier is trained on data $\boldsymbol{w}_t$ (t=1~N, the number of training samples) from class i and class j by solving the following binary classification problem:

$$\min J(\boldsymbol{w}, e, b) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_t e_t \tag{3}$$

subject to

$$\begin{aligned} &\boldsymbol{w}^T\Phi(\boldsymbol{x}_t) + b \geq 1 - e_t, && \text{if } \boldsymbol{x}_t \in \{\text{class } i\},\\ &\boldsymbol{w}^T\Phi(\boldsymbol{x}_t) + b \leq -1 + e_t, && \text{if } \boldsymbol{x}_t \in \{\text{class } j\},\\ &e_t \geq 0. \end{aligned}$$

After training, the decision function is given by

$$\text{sign}(\boldsymbol{w}^T\Phi(\boldsymbol{x}) + b). \tag{4}$$

In combining the decisions of the $K(K\text{-}1)/2$ binary classifiers, the sample is assigned the class in which at least k classifiers agree on the identity, where

$$k = \begin{cases} \dfrac{K+1}{2}, & \text{if } K \text{ is odd},\\ \dfrac{K}{2}, & \text{if } K \text{ is even}. \end{cases} \tag{5}$$

If the two classes have identical votes, the one with the smaller index is selected.

5. Important Event Detection

The exciting moments in a soccer broadcast can be derived by combining close-up shots, penalty area shots, and excited speech segments. The exciting moments would contain goal scenes or nearly-missed-goal situations. Figure 4 shows typical video shot changes during exciting moments.

Figure 4 exemplifies the play scenes of a baseball game. Since a play scene starts with a pitching shot and ends with close-up shots, a pitching shot is detected using the pitching shot detection algorithm, field colors are extracted from the detected pitching shot, and finally, the close-up shots are detected using extracted field colors. Exciting moments in a baseball game usually contain the excited speech of announcers during events like a hit or a homerun. Therefore, a play scene including excited speech would be the exciting moment temporal segment.

Fig. 4. Video shot type change in exciting moments of soccer and a baseball play scene.

To calculate the weight for each event, the Bayes theory is used, and $P(I|E_i)$ is a probability that the i-th video/audio event (E_i) belongs to an important event (I). Based on the Bayes theory, $P(I|E_i)$ is proportional to $P(E_i|I)$. The weight (W_i) of the i-th video/audio event is calculated as

$$W_i = \frac{P(E_i \mid I)}{\sum_x P(E_x \mid I)}. \tag{6}$$

For instance, if a baseball indexing uses three features, such as the length of a play scene, audio events, and audio energy, the weight (W_i) of the i-th video/audio event is calculated as follows.

The importance level of the length of a play scene is

$$F(L) = \frac{\text{End}_i - \text{Start}_i}{\text{Max}_L}, \tag{7}$$

where Start_i and End_i are the start and end times of the i-th scene, and Max_L is the maximum length of the play scene in the entire game. The level of audio energy is

$$F(A) = \frac{A_e}{\text{Max}_A}, \tag{8}$$

where A_e is the average energy in the play scene, and Max_A is the maximum (average) audio energy of the play scene in the entire game. If there is an (audio) exciting moment in the play scene, $F(E)$ is 1.0; otherwise, it is 0.3. Finally, the weight is calculated as

$$\begin{aligned} W_i &= \frac{P(L \mid I)}{P}\cdot F(L) + \frac{P(A \mid I)}{P}\cdot F(A) + \frac{P(E \mid I)}{P}\cdot F(E), \\ P &= P(L \mid I) + P(A \mid I) + P(E \mid I). \end{aligned} \tag{9}$$

This procedure can be easily applied to soccer videos as well. The main features of a soccer match are audio exciting moments, penalty shots, close-up shots, and audio energy.

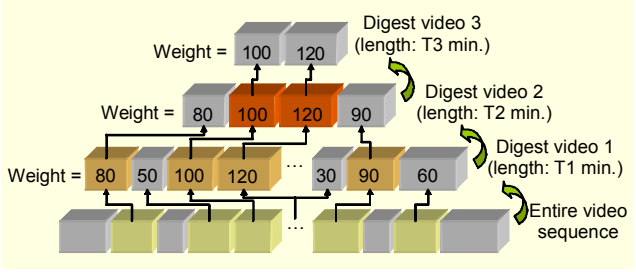

Fig. 5. Digest generation scheme based on weights.

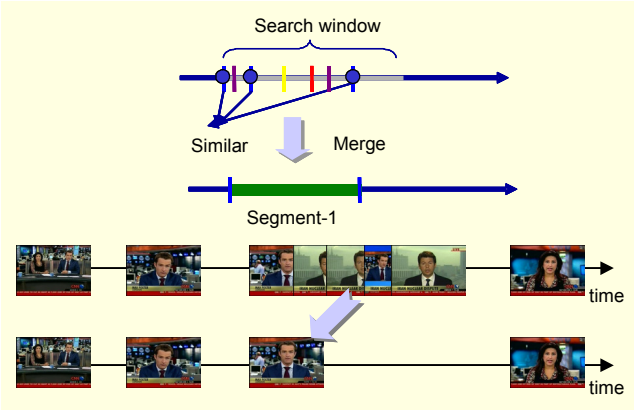

Fig. 6. News story segment generation.

When the digest length is changed, the scene is selected based on the weight as shown in Fig. 5.

To generate news scene segments, the positions of the anchorperson shots detected should be refined. In news, repeated shots of the identical anchorperson often occur within a short time period, as in interview shots between an anchorperson and reporters/interviewees. These shots should be adequately merged into one news article scene using the occurrence of the identical anchorperson in a predefined search window (for example, five shots long or a 30-second window). After this refinement process, the position of each anchorperson shot becomes the starting point of a news article scene (Fig. 6). An important news scene is composed using the detected anchorperson shots.

III. Personal Videocasting System Architecture

The proposed algorithms were implemented in an actual personal videocasting system. The main purpose of the system is to provide the functions of remote recording, video digesting and navigating, and real-time event content broadcasting between a home-server-type device and a mobile device.

For example, a mobile phone can be connected to home appliances, such as set-top boxes, PVRs, PCs with TV receivers, or home servers that remotely record programs broadcast over the Internet. While the selected home device records video contents, it can automatically and simultaneously generate a content digest or navigation information. A mobile device can stream a user's recorded contents with a digest and navigation information so that a user can watch only the part that he or she desires.

A feasible usage scenario is as follows. A user, Mr. Kim, is fanatical about soccer. He would like to see the final match of the national cup in the evening, but he is abroad due to a business trip. Because he cannot see the match in real-time, he connects to his PVR at home instead, and makes a recording reservation for the match using an EPG with his mobile phone. When the match begins, the PVR at home automatically starts recording and indexing the match. After indexing, the PVR at home converts the recording status of the corresponding match to the on-stage. Mr. Kim can check the recording status of the match using his mobile phone, and he can then watch a digest of the match or navigate through important moments of the match using a media player on his mobile phone. He can watch the match even though he is abroad.

There are three major advantages in this system. First, hand-held devices do not need high CPU power. Since a home media server is responsible for the expensive indexing task, a hand-held device with a reasonable display size (such as 320 × 240), a media decoder, and Internet connection capability would be enough to accomplish the scenario. Second, since a user can select only the part of the video that he or she wants to watch through the digest or the navigation function, the short battery life of a hand-held device would no longer be a limiting problem. Third, commercial PVRs strictly prohibit copying contents (such as HD video contents) saved in the storage since they can be reused commercially. Therefore, downloading (that is, copying) video digest clips from a PVR to a mobile device must be prohibited as well. However, the proposed scheme is free from such a problem, since recorded video contents and their digests are preserved in the home devices and are only streamed live to the hand-held devices.

To realize the system, currently available technologies were combined. First, it was assumed that a user can use a web browser of a notebook PC, a PDA, or a smart phone, which enables a WLAN connection. With these mobile devices, a user can connect to a PVR or a home server and browse a pre-designed web page as shown in Figs. 8(a) and (b). The web-based interfaces can provide a device-independent and flexible user interface. Database (DB) tables contain the EPG and digest/navigation information.

The system is composed of three parts as shown in Fig. 7. The first part is related to data sources, such as a tuner device and an external EPG server. The existing conventional system was employed for the first part.

The second part is the main processing part of home devices.

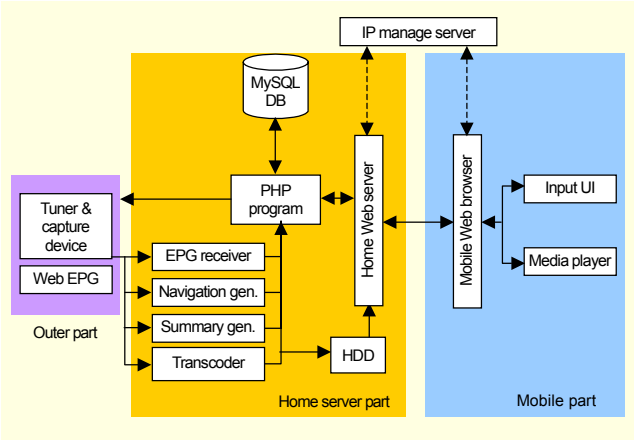

Fig. 7. Personal videocasting system diagram.

Fig. 8. (a) EPG; (b) play page user interface; and (c), (d), and (e) navigation page user interface.

For this part, a recording program and an EPG data receiver program were implemented. The EPG data receiver can obtain EPG data from the Web and store it in a DB using an implemented DB application programming interface (API) layer. The genres and titles of video contents are extracted from EPG data. The recording program is a main part of this system. It uses a direct-show filter technology to process, transcode, and store AV streams. The recording reservation included in the DB is regularly checked by this program. If a reservation to record a program matches with the current time, recording and the generation of a digest and navigation information begins. The proposed algorithms are only applied when the extract genre and title match with one of the three video subjects. Digested videos and important scene information for navigation are stored to provide streaming service. In our realization of the system, a transcoding filter was implemented, and a streaming service function from Microsoft was employed.

The last part of the system is the user interface displayed on the mobile device. The Web program provides three interface pages. The first is a program recording reservation page which contains an EPG list as shown in Fig. 8(a). This page can be sorted by times, genres, or reserved program names. The second is a play page which is used to check recording status and to select contents as shown in Fig. 8(b). It shows reserved program names and their recording status. A recorded program can be played by clicking its name while its digested version can be played by clicking the "digest" button beside its name. When a "navigation" button is clicked, a new page for navigation can be seen. The navigation page is genre dependent. Currently, there are two types of navigation pages shown in Figs. 8(c) to (e). For baseball video content, inning information is displayed as in Fig. 8(c) so that a user can directly select an inning of interest. In contrast, news and soccer use the same interface, displaying representative thumbnail images of each scene, each of which is the first frame of a detected important event. News can be navigated by individual articles, while soccer can be navigated by important match events.

IV. Experimental Results

To evaluate the shot type analysis, recall and precision were used. As in (10), N_c is the number of correct shots detected, N_m is the number of shots missing from a ground truth, and N_f is the number of false alarms detected:

$$\text{Recall} = \frac{N_c}{N_c + N_m}, \quad \text{Precision} = \frac{N_c}{N_c + N_f}. \tag{10}$$

The ground truth set for each video was generally made by one person and was double-checked by four others. To determine the pre-defined thresholds and color models used in the algorithms, at least 20 videos for sports and news videos were observed and examined for training. Those training videos were not included in the test videos.

For a shot type analysis test of a soccer broadcast, 10 soccer matches were tested, for a total duration of 1,178 minutes of 352×240 MPEG-1 and 720×480 MPEG-2 videos. The 10 matches included soccer matches of Korean, Spanish, German, English, and Italian leagues broadcast by various stations.

Table 1 shows the results of the shot type classification for tested soccer broadcasts. The proposed algorithm demonstrates robustness against various conditions, as shown in the table.

Table 1. Experimental results of shot type detection from soccer broadcasts.

Video name	Close-up shot		Penalty area shot	
	Recall	Precision	Recall	Precision
K League 1	0.99	0.94	0.97	0.91
Primera Liga 1	0.99	0.87	0.97	0.83
Euro 2004	0.99	0.97	0.99	0.83
K League 2	0.85	0.97	0.99	0.84
AFC Champ.	0.98	0.95	0.99	0.86
Primera Liga 2	0.98	0.96	0.94	0.83
Bundesliga 1	0.97	0.95	0.93	0.90
Seria A 1	0.99	0.87	0.99	0.93
Primi. League 1	0.99	0.96	1.00	0.85
Primera Liga 3	0.99	0.92	0.98	0.82
Total	0.98	0.94	0.98	0.86

Table 2. Shot type detection experimental results for baseball broadcasts.

Video name	Pitching shot		Close-up shot
	Recall	Precision	Accuracy
Korean Playoff	1.00	0.97	0.98
Korea League 1	0.99	0.98	0.98
Korea League 2	1.00	1.00	0.97
Korea League 3	0.98	0.98	0.97
Korea League 4	0.98	0.99	0.97
Korea League 5	0.99	1.00	1.00
Korean Series 1	0.98	1.00	0.98
Korean Series 2	0.99	1.00	0.98
Major League 1	0.98	0.99	0.95
Major League 2	0.93	1.00	0.93
Total	0.98	0.99	0.97

The precision rate for penalty shots was lower than the recall rate since there are many lines that have an angle similar to that of the penalty line. Also, the recall rate of K League 2 was relatively low because the players' uniform color was similar to the field color.

To evaluate the shot type analysis for baseball broadcasts, an accuracy measure was employed along with the recall and precision measurements. The accuracy can be obtained as shown in (11), where P_t is the total number of play scenes, and P_c is the number of correct shot type detections:

$$\text{Accuracy} = \frac{P_c}{P_t}. \tag{11}$$

Table 3. Experimental results of anchorperson shot detection.

Video name	No. of shots (A)	Detected shots (B)	Missed shots (C)	False shots (D)	Prec = B/(B+D)	Recall = B/A
KBS1	73	72	1	1	98.6	98.6
MBC	30	30	0	0	100	100
SBS	28	27	1	0	100	96.4
KBS2	73	71	2	1	98.6	97.3
YTN	24	24	0	0	100	100
BBC	34	33	1	1	97.1	97.1
CCTV4	61	56	5	1	98.2	91.8
CNN	32	30	2	9	76.9	93.8
NHK	45	34	11	9	79.1	75.6
Total	400	377	23	22	94.5	94.3

Ten baseball games were recorded to test the baseball shot type analysis. The recordings had a total duration of 1,183 minutes and included 352×240 MPEG-1 videos and 720×480 MPEG-2 videos. The 10 games were obtained from Korean and American major leagues and were produced by various broadcasting stations.

Again, as shown in Table 2, our proposed algorithms for detecting baseball shot types work superbly and robustly in spite of the varying conditions of stadiums, broadcasting stations, weather, and so on.

The performance of the anchorperson shot detection algorithm was measured using recall and precision. The news recordings were 352×240 MPEG-1 videos. Various news programs were obtained from the following broadcasting stations and countries:

- Korean news (KBS, MBC, SBS, YTN): 10 hours total of news videos
- BBC World Business News: 1 hour total of news videos (30 min/news)
- CCTV4 China News: 2 hours total of news videos
- CNN World News Asia: 1 hour total of news videos
- NHK News 10: 2 hours total of news videos

Table 3 shows the results for the anchorperson shot detection experiment. Precision results for CNN and NHK news are relatively low because of false alarms due to the repeated occurrence of non-anchorpersons and some complications regarding the news composition, such as a news story commented upon by two or more anchorpersons. The recall results of NHK are also relatively low because of the high miss rate due to face poses or small face size of anchorpersons.

The database for each audio class was collected by recording broadcasting programs, such as news, shows, and sports, along with MP3 music files for the performance test of audio

Table 4. Audio database for general audio classification test.

Type	Length (s)	Event types
Noise	3,696	Noise DB, background noise
Speech	2,729	Clean speech, low-level noisy speech
Music	22,736	Song, instrument sound
Excitement	1,758	Excited speech, applause

Table 5. Audio database for sports events detection test.

Genre	No. of games/matches	Events for ground truth
Baseball	26	Hit, home run, strike-out, nice field play, etc.
Soccer	17	Shooting, goal, etc.

Table 6. Audio classification confusion matrix by SVM classifier (unit: %).

	Noise	Speech	Music	Excitement
Noise	96.7	0.0	3.2	0.1
Speech	0.6	87.8	9.7	1.9
Music	0.9	0.9	97.3	0.9
Excitement	0.1	2.1	12.6	85.2

Table 7. Excitement segment detection results for sports.

Sports genre	Recall	Precision
Baseball	0.81	0.94
Soccer	0.70	0.63

classification as shown in Table 4.

A ground truth set of major event segments for baseball and soccer was composed to evaluate the performance of the excitement segment detection using only audio information. The event database for sports is shown in Table 5. Five videos were used for training and others were used for testing.

The audio type classification algorithm was applied to the database for noise, speech, music, and excitement classification every 0.96 seconds. The results are shown in Table 6.

The excitement segments were detected using the audio classifier, and the experimental results are shown in Table 7. The SVM was used for the classifier.

The proposed algorithm was applied to realize a practical video storyboard application interface between a PVR device (450 MHz CPU with 50 MB of main memory) and a mobile device (520 MHz Intel Bulverde CPU and 57.03 MB of main memory). To index a 1 hour video, it took 694 seconds for soccer, 537 seconds for baseball, and 456 seconds for news on average.

Table 8. Performance comparison results of a 2006 pro-baseball game (Yomiuri vs. Yokohama).

	Sony		Hitachi		SAIT	
	Recall	Precision	Recall	Precision	Recall	Precision
5 min	0.19 (4/21)	0.20 (53/259)	0.24 (5/21)	0.3 (90/300)	0.33 (7/21)	0.39 (120/305)
	4 min 19 s (actual digest length)					
15 min	0.62 (13/21)	0.21 (225/1076)	0.48 (10/21)	0.19 (173/900)	0.91 (19/21)	0.33 (304/911)
	17 min 56 s					
30 min	0.67 (14/21)	0.14 (233/1621)	0.57 (12/21)	0.11 (205/1800)	1.00 (21/21)	0.18 (326/1807)
	27 min 1 s					

Table 9. Performance comparison results of a 2006 World Cup semi-final soccer match (Germany vs. Italy).

	Sony		Hitachi		SAIT	
	Recall	Precision	Recall	Precision	Recall	Precision
5 min	0.15 (7/46)	0.36 (125/349)	0.20 (9/46)	0.47 (141/300)	0.30 (14/46)	0.67 (225/337)
	5 min 49 s (actual digest length)				5 min 37 s (actual digest length)	
15 min	0.24 (11/46)	0.24 (171/721)	0.39 (18/46)	0.28 (255/900)	0.61 (28/46)	0.46 (436/953)
	12 min 1 s (actual digest length)				10 min 53 s (actual digest length)	
30 min	0.35 (16/21)	0.19 (262/1390)	0.63 (29/46)	0.24 (437/1800)	0.85 (39/46)	0.31 (617/2007)
	23 min 10 s (actual digest length)				33 min 27 s (actual digest length)	

Viewers could easily select important event segments using a TV remote control or a mobile device keypad. Viewers could play a selected segment to watch in detail or stop as they want.

The performance of detecting important events of games or matches was compared with existing state-of-the-art PVR products. Since the implementation was merely a laboratory level system, direct comparison of complexity and timing between the proposed system and the state-of-the-art systems was not feasible. The algorithms were tested using one baseball game (a 2006 pro-baseball game, Yomiuri vs. Yokohama) and one soccer match (a 2006 World Cup semi-final, Germany vs. Italy). Recall and precision were measured as in (12), where N_d was the number of important events detected, N_e was the number of all important events in the game or match, T_e was the actual time length of a digest clip generated by the digest function, and T_d was the actual duration of the important events amongst the generated digest clip:

$$\text{Recall} = \frac{N_d}{N_e}, \quad \text{Precision} = \frac{T_d}{T_e}. \tag{12}$$

A digest was automatically produced in the target digest lengths of 5, 15, and 30 minutes, respectively. For the baseball game, 21 important events (such as hits and homeruns) were selected by a human observer, while 46 important events (such as goal scenes or nearly-missed-goal situations) were selected for the soccer match. Tables 8 and 9 demonstrate that the proposed algorithms outperform the state-of-the-art PVR digest generator.

V. Conclusion

In this paper, a personal videocasting system between a PVR home server and mobile devices has been demonstrated along with simple video indexing algorithms. The proposed system would allow users to enjoy TV programs in a more convenient way using their mobile devices without any indexing complexity or content rights management problems.

The algorithms can be further investigated to index other program genres. More general feature extraction development for indexing should be investigated to automatically index a general genre broadcasts. The personal videocasting system can be further improved by combining text descriptions of video programs to obtain more semantic information.

References

[1] D. Tjondronegoro and Y.P.P. Chen, "Integrating Highlights for More Complete Sports Video Summarization," *IEEE Trans. on Multimedia*, vol. 11, no. 4, 2004, pp. 22-37.

[2] D.A. Sadier and N.E. O'Connor, "Event Detection in Field Sports Video Using Audio-Visual Features and a Support Vector Machine," *IEEE Trans. on CSVT*, vol. 15, no. 10, 2005, pp. 1225-1233.

[3] J.W. Choi and D.S. Jeong, "Storyboard Construction Using Segmentation of MPEG Encoded News Video," *Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems*, vol. 2, 2000, pp. 758-761.

[4] M. Bertini, A. del Bimbo, and P. Pala, "Content Based Indexing and Retrieval of TV News," *Pattern Recognition Letters*, vol. 22, 2001, pp. 503-516.

[5] X. Gao, J. Li, and B. Yang, "A Graph-Theoretical Clustering Based Anchorperson Shot Detection for News Video Indexing," *ICCIMA*, 2003, p.108.

[6] S.K. Kim et al., "An Effective News Anchorperson Shot Detection Method Based on Adaptive Audio/Visual Model Generation," *Image and Video Retrieval, 4th International Conference Proceedings* (*Lecture Notes in Computer Science*), CIVR, vol. 3568, 2005, pp. 276-85.

[7] S. Srinivasan, D. Petkovic, and D. Ponceleon, "Towards Robust Features for Classifying Audio in the Cue Video System," *Proc. of ACM Multimedia*, 1999, pp. 393-400.

[8] R. Radhakrishan et al., "Generation of Sports Highlights Using a Combination of Supervised and Unsupervised Learning in Audio Domain," *International Conference on Pacific Rim Conference on Multimedia*, vol. 2, 2003, pp. 935-939.

[9] K. Wan and C. Xu, "Efficient Multimodal Features for Automatic Soccer Highlight Generation," *International Conference on ICPR*, vol. 4, 2004, pp. 973-976.

[10] M. Furini and V. Ghini, "An Audio-Video Summarization Scheme Based on Audio and Video Analysis," *International Conference on CCNC*, vol. 2, 2006, pp. 1209-1213.

[11] T. Zhang and C.C. Kuo, "Audio Content Analysis for Online Audiovisual Data Segmentation," *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 4, 2001, pp. 441-457

[12] M. Luo, Y.F. Ma, and H.J. Zhang, "Pyramid-Wise Structuring for Soccer Highlight Extraction," *PCM*, Dec. 2003, pp. 945-949.

[13] B.S. Manjunath et al., "Color and Texture Descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, June 2001, pp. 703 -715.

[14] C.S. Won, D.K. Park, and S.J. Park, "Efficient Use of MPEG-7 Edge Histogram Descriptor," *ETRI Journal*, vol. 24, no. 1, Feb. 2002, pp. 23-30.

[15] S.M. Kim, S.J. Park, and C.S. Won, "Image Retrieval via Query-by-Layout Using MPEG-7 Visual Descriptors," *ETRI Journal*, vol. 29, no. 2, Apr. 2007, pp. 246-248.

[16] J.G. Kim et al., "Multimodal Approach for Summarizing and Indexing News Video," *ETRI Journal*, vol. 24, no. 1, Feb. 2002, pp. 1-11.

[17] J.H. Lee et al., "Automatic Video Management System Using Face Recognition and MPEG-7 Visual Descriptors," *ETRI Journal*, vol. 27, no. 6, Dec. 2005, pp. 806-809.

[18] W.J. Hwang et al., "Multiple Face Model of Hybrid Fourier Feature for Large Face Image Set," *Proc. IEEE Computer Vision and Pattern Recognition* (CVPR), New York, USA, June 2006, pp. 1574-1582.

[19] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*, Cambridge University Press, 2000.

Sang-Kyun Kim received the BS, MS, and PhD degrees in computer science from the University of Iowa in 1991, 1994, and 1997, respectively. In 1997, he joined the Samsung Advanced Institute of Technology as a research staff member. He was a senior research staff member as well as a project leader of the Image and Video Content Search Team in the Computing Technology Lab until 2007. He is an assistant professor with the Department of Computer Engineering of Myongji University. His research interests include digital contents (i.e., image, video, and music) analysis and management, fast image search and indexing, MPEG-7, MPEG RoSE, and multi-modal analysis.

Jinguk Jeong is a researcher with Samsung Electronics. He received the BS, MS, and PhD degrees in computer science from Sogang University, Seoul, Korea, in 1998, 2000, and 2004, respectively. In 2004, he joined the Samsung Advanced Institute of Technology as a research staff member. He is a research staff member of the Digital Contents Management Team of the Core S/W Component Lab. His research interests include video/image analysis and search.

Hyoung-Gook Kim received the Dipl.-Ing. degree in electrical engineering and the Dr.-Ing. degree in computer science from the Technical University of Berlin, Berlin, Germany. From 1998 to 2002, he worked on mobile service robots at Daimler Benz, speech recognition at Siemens, and speech signal processing at Cortologic, Germany. From 2002 to 2005, he served as an adjunct professor of the Communication Systems Department of the Technical University of Berlin, where he was a project leader of the MPEG-7 Annotation of Video Sequences Project. From 2005 to 2007 he was a project leader responsible for the Audio Content Indexing and Retrieval Project of Samsung Advanced Institute of Technology, Korea. Since 2007, he has been an associate professor with the Wireless Communications Engineering Department of Kwangwoon University, Seoul, Korea. His research interests include audio signal processing, audiovisual content indexing and retrieval, speech enhancement, and speech recognition.

Min Gyo Chung received the BS degree in computer engineering from Seoul National University, Seoul, Korea, in 1985, the MS degree in computer science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1987, and the PhD degree in computer science from the University of Iowa in 1996. He worked with Korea Telecom until 2000 and Vivcom Inc. until 2002. He is now an assistant professor with Seoul Women's University. His current research interests include computer vision, pattern recognition, video/image compression, biometrics, and digital watermarking.