IEICE TRANS. INF. & SYST., VOL.E92–D, NO.6 JUNE 2009

PAPER

User and Device Adaptation in Summarizing Sports Videos

Naoko NITTA[†a)] ***and*** **Noboru BABAGUCHI**[†]**,** ***Members***

SUMMARY Video summarization is defined as creating a video summary which includes only important scenes in the original video streams. In order to realize automatic video summarization, the significance of each scene needs to be determined. When targeted especially on broadcast sports videos, a *play scene*, which corresponds to a play, can be considered as a scene unit. The significance of every play scene can generally be determined based on the importance of the play in the game. Furthermore, the following two issues should be considered: 1) what is important depends on each user's preferences, and 2) the summaries should be tailored for media devices that each user has. Considering the above issues, this paper proposes a unified framework for user and device adaptation in summarizing broadcast sports videos. The proposed framework summarizes sports videos by selecting play scenes based on not only the importance of each play itself but also the users' preferences by using the *metadata*, which describes the semantic content of videos with keywords, and *user profiles*, which describe users' preference degrees for the keywords. The selected scenes are then presented in a proper way using various types of media such as video, image, or text according to *device profiles* which describe the device type. We experimentally verified the effectiveness of user adaptation by examining how the generated summaries are changed by different preference degrees and by comparing our results with/without using user profiles. The validity of device adaptation is also evaluated by conducting questionnaires using PCs and mobile phones as the media devices.

key words: *user adaptation, device adaptation, video summarization, sports videos, metadata*

1. Introduction

As more and more video content becomes available, the importance of techniques to quickly search and browse only specific scenes has been widely recognized. As a solution to this problem, video summarization has attracted much attention. Video summarization is defined as creating a video summary which includes only important scenes selected from original video streams. Note that a video summary can be divided into two types: *dynamic* and *static* video summaries [1]. A dynamic video summary is a short video clip created by temporally arranging the video segments corresponding to only important scenes, while a static video summary is an organized list of scenes, each of which is represented by a static media such as image and text.

For both types of summary, the appropriate selection of scenes plays the main role in maximizing the semantic content and perceptual quality of a video summary. Sports videos have been very popular as a target content domain for automatic video summarization, since they tend to have much redundancy which can be significantly reduced by extracting only highlight scenes. Therefore, when targeted on sports videos, a video summary can be generated by selecting only highlight scenes based on the significance of each play scene.

However, there remain the following two problems [2], [3]. One is that the content of video summaries should be individual for every user since what is important depends on his/her preferences. For example, one who favors the Giants, a Japanese professional baseball team, wants to watch more of their play scenes than play scenes of their opponent. Thus, not only plays themselves but also players and teams should be taken into account to determine the significance of scenes based on each user's preferences. The other problem is that the same content should be presented for the same user whatever media devices he/she has. Since different devices such as PCs, PDAs, and mobile phones, have different constraints in the network traffic speed, display capability, CPU speed, etc., adapting how to present the generated video summaries such as changing the video frame rate and the resolution is of great importance while maintaining the users' levels of understanding of the content. As discussed above, user and device adaptation of video summaries is crucial.

This paper proposes a unified framework for user and device adaptation in summarizing sports videos. The proposed framework firstly selects important scenes by considering not only the importance of plays in the game but also the users' preferences as user adaptation. The user's preferences are described in a *user profile* as a set of pairs of a keyword and the user's degree of preference toward the concept represented by the keyword. As the semantic content of each scene is described in the *metadata* with keywords, the significance of each scene can be determined by using the preference degrees for its descriptive keywords. Then, the scenes selected according to their significance should be displayed on a media device in an appropriate way. Since the dynamic video summary presents video scenes sequentially, the resolution and the frame rate of the video are generally adapted. On the other hand, since the static video summary presents multiple scenes simultaneously, each scene should be presented using the suitable types of static media in the right position at the right time to effectively use the limited display space. Therefore, as device adaptation, this paper proposes to adapt how to present both types of summaries by changing the media types to present a scene and when

Manuscript received October 20, 2008.
Manuscript revised February 5, 2009.

[†]The authors are with the Graduate School of Engineering, Osaka University, Suita-shi, 565–0871 Japan.
a) E-mail: naoko@comm.eng.osaka-u.ac.jp

DOI: 10.1587/transinf.E92.D.1280

Copyright © 2009 The Institute of Electronics, Information and Communication Engineers

and where to present it according to the device information. The device information such as the device type and display screen size is obtained from a *device profile*. The effectiveness of user and device adaptation is verified through objective and subjective evaluations of the experimental results with baseball videos.

2. Related Work

This section discusses related work for video summarization for sports videos. Many existing methods have tried to extract highlight scenes for sports video summarization. For example, Ekin et al. [12] extracted score scenes by detecting goal-posts for soccer videos. Tjondronegoro et al. [13] used audio features such as spectral energy, loudness, and pitch for detecting the whistles and the crowd's or commentator's excitement, as well as visual features such as edges and shapes for detecting text displays of players' names and updated scores. Xiong et al. [14] proposed to use visual objects such as pitchers and catchers in addition to audio objects such as spectators' cheer for baseball highlight extraction. These methods use low- or mid-level features to analyze the semantic content of videos. However, it is hard to obtain the detailed semantic content such as more varied types of event and players, which should especially be necessary to consider users' preferences, from such low or mid-level features.

In order to realize user-adaptive video summarization, the external information is often used to consider the high-level semantic content. For example, in [6], a video is represented with event-based features which are extracted from manually created metadata. After a user selects his/her personal event highlights in a training set of soccer videos, a binary classifier is learned as a user profile to determine the highlights from a new game according to the user's preferences. Masumitsu et al. [7] proposed a framework for constructing personalized video summaries using the content profile which reflects the representative preferences of users with the same interest and the user profile, both of which describe the preference degrees for keywords and are created from manually prepared metadata of each scene. While these methods can generate summaries only with user profiles, Babaguchi et al. [5] consider the importance of plays themselves e.g., score plays are more important than non-score plays, in addition to user profiles. This technique can extract highlight scenes by not using the user profile and generate summaries suitable for general users. Then, the content of the generalized summaries can be changed so that they include more scenes preferred by the user by giving the higher importance to preferred scenes according to the user profile. However, in order to exclude specific scenes the user would rather not watch from the generalized summaries, the preference degrees need to be set for many keywords. For example, excluding the scenes of a specific player requires setting high preference degrees for every other player. In addition, these methods only generate dynamic video summaries. Static video summaries that allow nonlinear browsing of the video content are also necessary for users to understand the content more deeply.

For device adaptation, Tseng et al. [8], [9] proposed to utilize profiles not only about users' preferences, but also about device, network, delivery, and other environments to adapt the video for specific devices. As Chang et al. also pointed out in [10], the video properties such as the resolution and frame rate are generally changed to adapt dynamic video summaries for different media devices. As one of few techniques proposed for device adaptation of static video summaries, Ferman et al. [11] proposed a framework to generate a static video summary using the MPEG-7 color descriptors and the MPEG-7 user preferences. The scenes, each of which is represented by a key image frame, are presented in a hierarchical fashion and the number of simultaneously presented scenes can be decreased according to variations in network access and device properties. However, displaying several images simultaneouly can still require a large amount of data transmission and a large display space.

Based on the discussion above, the main challenges of the research can be stated as follows:

- User and device adaptation should be realized for both dynamic and static video summaries in a unified way.
- The content of the user-adapted video summaries should be changed more easily by considering not only what the user likes but also what the user dislikes.
- The data volume for presented scenes should be adaptively suppressed for efficient data transmission and display space usage, while maintaining the users' levels of understanding of the video content.
- The generated summaries should be evaluated to demonstrate the effectiveness of user and device adaptation.

Given these challenges, our main contributions are:

- We propose a unified framework implemented in a client-server architecture where the server uses user and device profiles to generate both dynamic and static video summaries of a suitable content for the client user and present them in a suitable way for the client device.
- For user adaptation, we prepare user profiles which include a set of keywords and the preference degrees that can represent positive preferences, no preference, and negative preferences toward different concepts represented by the keywords. Considering these three types of preference enables us to efficiently adapt summaries to each user simply by setting preference degrees for a few keywords.
- We prepare device profiles which describe the device information such as the device type and screen size for device adaptation. Specifically, we propose to change how to present both types of video summary by changing the media types, e.g. video, image, text, etc., to present a scene and their temporal and spatial display

positions according to the device information, so that the varied-sized display spaces of different devices can be effectively utilized.
- We verify the effectiveness of user adaptation by examining how the generated summaries are changed by different preference degrees and by comparing our results with/without using user profiles. We also subjectively evaluate the validity of device adaptation by conducting questionnaires to users using PCs and mobile phones as the media devices.

3. Framework Overview

This section firstly explains the basic ideas of our work. Firstly, the importance of each scene to a user should depend on the content of the scene and the user's preferences, for example, who is in the scene and who the user likes/dislikes. For user adaptation, we use user profiles composed of a set of pairs of a keyword k and the user's preference degree v_k for the keyword such as $< Tigers, -0.2 >$ and $< Matsui, 0.6 >$. Assuming the semantic content of videos is also described with keywords as metadata, setting higher positive/negative preference degrees for specific keywords makes the scenes described with the keywords to be more/less likely to be selected.

Secondly, the summaries should be displayed in such a way that the display space of each type of device can be effectively utilized. For device adaptation, in addition to changing the video properties such as the resolution and the frame rate, we change the media types such as video, image, and text to display scenes and their temporal and spatial display positions. Specifically, the larger the screen is, the more information can be displayed. For example, both images and texts can be displayed on a device with a large screen, while images should be eliminated for a device with a smaller screen to save the display space. Our framework uses device profiles composed of a set of pairs of the device's property name a and its value u_a, such as $< DeviceType, PC >$ and $< ScreenSize, 17inch >$. Considering the same type of device has a display of similar size, we refer the property name *DeviceType* to determine the media types and the temporal and spatial positions to display scenes.

Based on these ideas, our framework is designed as a client/server architecture as shown in Fig. 1. Since the user profile should be shared among all the devices used by a user, it is stored in the user profile server. On the other hand, since the device profile should be unique for each device, it is stored in each device. A user can view a video summary in our framework as follows:

(1) Once a client user requests for a video summary, the device profile and the user ID are transmitted to the application server from the client device.
(2) The application server accesses the user profile and metadata server to obtain the user profile of the obtained user ID and the metadata, respectively.
(3) In the application server, the significance of each scene is determined by referring to the metadata and the user profile.
(4) The application server accesses the media server and obtains the necessary video scenes according to their significance to generate a video summary.
(5) According to the device type described in the device profile, suitable media types and the temporal and spatial positions for displaying the selected scenes are determined.

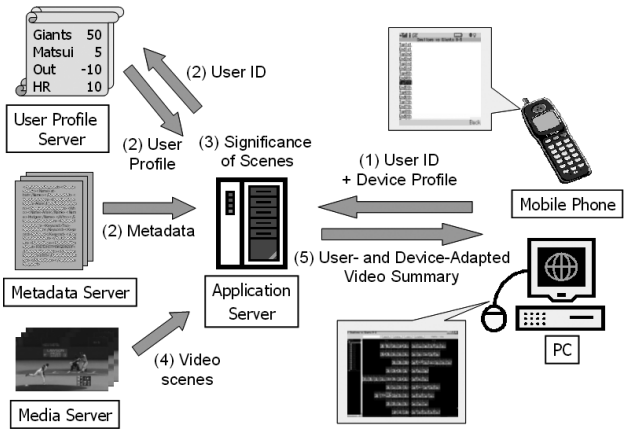
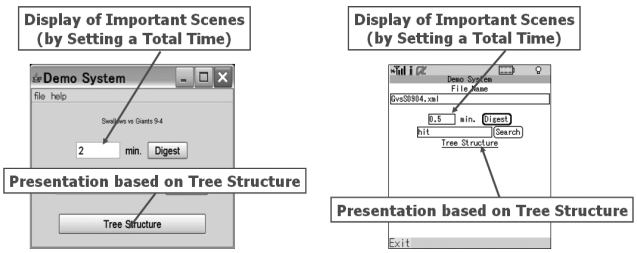


Fig. 1 Framework.



(a) PC (b) mobile phone

Fig. 2 Interface.

The generated summaries are viewed via an interface as shown in Fig. 2. Figures 2 (a) and 2 (b) show the example for a PC and a mobile phone, respectively. Note that, though being out of scope of this paper, the access to user profiles in the user profile server needs to be properly controlled to secure privacy.

The details of the metadata used in our framework and the functions provided via the interface are described below.

3.1 Metadata for Sports Videos

Metadata is the data to describe various characteristics of the data including the semantic information, and MPEG-7 has been standardized to describe the metadata for videos [15], [16]. In this paper, we assume the metadata, which is described with MPEG-7, is given to videos beforehand.

Now, we describe the relations between the structure of the metadata and the structure of a sports game. Sports games generally have tree structures according to their gen-

res, and a sports video can be structured based on the structure of the corresponding sports game. For example, Fig. 3 shows the game tree of a baseball video. A baseball game is composed of several innings, an inning is composed of several at-bats, an at-bat is composed of several plays, and a play is composed of several shots, each of which corresponds to a video segment filmed by one camera without interruption in a baseball video. Note that a play corresponds to a pitcher throw for a baseball game.

These tree structures are described in the metadata for sports videos using AudioVisualSegment tags. AudioVisualSegment tags denote each node such as game, inning, at-bat, or play in the tree structure. Additionally, for each play scene which corresponds to a play, five items of information, 1) the unit type, 2) the classification, 3) the players, 4) the events, and 5) the media time are described as shown in Fig. 4.

3.2 Function Descriptions

The following functions are provided for any type of media device via an interface as shown in Fig. 2.

[Display of Important Scenes:] Play scenes are selected based on their significance, and then only the play scenes with high significance degrees are displayed as either a dynamic or a static video summary. For the dynamic summary, its total length can be flexibly changed according to the time specified by a user. For the static summary, the user can specify the number of play scenes to be displayed.

[Presentation based on Tree Structure:] As another form of static video summaries, a list of play scenes is presented with certain types of static media according to the tree structure of the game. A specific scene can be viewed by hierarchically tracing the game tree.

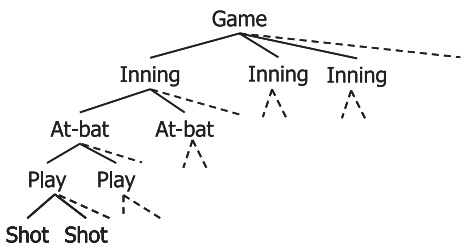


Fig. 3 Game tree of a baseball video.

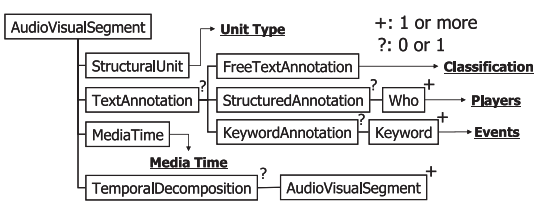


Fig. 4 Composition of the metadata.

4. User and Device Adaptation in Video Summarization

Figure 5 shows how user and device adaptation is realized in video summarization. For creating a dynamic or static video summary, highlight scenes, each of which corresponds to a play scene, should be selected. Therefore, a video is firstly divided into play scenes and the significance of each play scene is determined by considering the semantic content of the scene, which can be obtained from its metadata.

The significance of each play scene can generally be determined by the importance of the play in the game; however, the content of video summaries should be individual for every user since what is important also depends on his/her preferences. Therefore, users' preferences are also considered in determining the significance of scenes as user adaptation.

After selecting only the play scenes with high significance as important scenes, how to present both types of summary is determined according to the device type as device adaptation.

In the following, we firstly describe how to rank each play scene based simply on the importance of the play in the game and how to select important scenes. Then, user adaptation and device adaptation techniques are introduced.

4.1 Ranking Play Scenes

The significance of each play scene is firstly calculated based on three components: the play ranks, the play occurrence time, and the number of replays [2], [4].

1) Play Ranks
In this paper, we assume that a game is played between two teams, team A and team B, and that the team's goal is to get more scores than its opponent. Under this assumption, there are three states of the game situation: 'two-team tie,' 'team A's lead,' and 'team B's lead.' If a play can change the current state into a different state, we call it a State Change Play (SCP). It is evident that

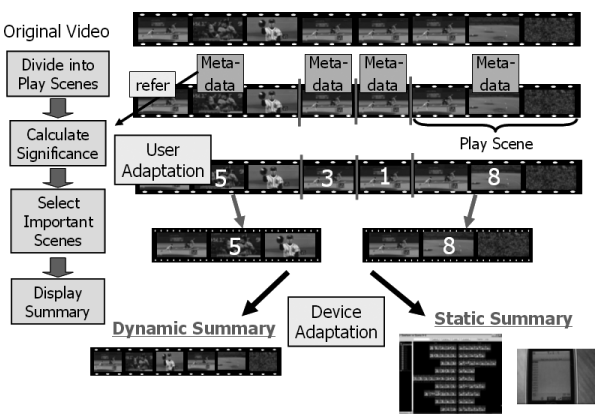


Fig. 5 User and device adaptation in video summarization.

SCPs are more significant than other plays.
The ranks of various plays are defined as follows:

Rank 1: SCPs.
Rank 2: score plays except SCPs.
Rank 3: plays closely related to score plays.
Rank 4: plays with score chance.
Rank 5: plays including a big play or the last play.
Rank 6: all other plays that are not in Rank 1-5.

Now, $s_r(p_i)$ $(0 \leq s_r \leq 1)$, the rank-based significance degree of a play scene p_i, is defined as

$$s_r(p_i) = 1 - \alpha \cdot \frac{r_i - 1}{5}, \tag{1}$$

where r_i denotes the play rank of the ith play scene p_i and α $(0 \leq \alpha \leq 1)$ is the coefficient to consider how much the difference in the play ranks affects the significance of play scenes.

2) Play Occurrence Time
The score play scenes which are close to the end of the game largely affect the game's outcome, especially when the two teams tie or have slight score difference. Thus, such play scenes are usually more attractive to users and of great significance. We define $s_t(p_i)$ $(0 \leq s_t \leq 1)$, the occurrence-time-based significance degree of a play scene p_i, as

$$s_t(p_i) = 1 - \beta \cdot \frac{N - i}{N - 1}, \tag{2}$$

where N is the number of all play scenes and β $(0 \leq \beta \leq 1)$ is the coefficient to consider how much the occurrence time affects the significance of play scenes.

3) Number of Replays
An important play scene has many replays and more important play scenes tend to have more replays than others. Thus, a play scene which has many replays is important. We define $s_n(p_i)$ $(0 \leq s_n \leq 1)$, the number-of-replays-based significance degree of a play scene p_i, as

$$s_n(p_i) = 1 - \gamma \cdot \frac{n_{\max} - n_i}{n_{\max}}, \tag{3}$$

where n_i denotes the number of replays of the play scene p_i, $n_{\max}$ is the maximum number of n_i, and γ $(0 \leq \gamma \leq 1)$ is the coefficient to consider how much the number of replays affects the significance of play scenes.

Then, $s(p_i)$, the significance degree of the play scene p_i, is given by

$$s(p_i) = s_r(p_i) \cdot s_t(p_i) \cdot s_n(p_i). \tag{4}$$

Changing the parameters of α, β, and γ enables us to control the composition of a video summary. For example, larger α can emphasize the significance of the play ranks. The other parameters behave in a similar manner.

4.2 Important Scene Selection

When the time length of a dynamic video summary L is given to the system with a function $\varphi\big(l(p_i)\big)$ $\big(0 < \varphi\big(l(p_i)\big) \leq l(p_i)\big)$ which changes the video length of a play scene p_i, the problem of selecting only important scenes can be formulated as follows:

select subset $P' = \{p_i \mid i = 1, 2, \ldots, m\}$ $(1 \leq m \leq N)$
from play scene set $P = \{p_1, p_2, \ldots, p_N\}$,
subject to $\displaystyle\sum_{p_i \in P'} s(p_i) \longrightarrow \max$

$$\sum_{p_i \in P'} \varphi\big(l(p_i)\big) \leq L,$$

where N denotes the total number of play scenes, $s(p_i)$ denotes the significance of a play scene p_i, and $l(p_i)$ denotes the video length of a play scene p_i. Thus, we can define this problem as the combinational optimization problem with constrained conditions.

As an approximation solution, we select play scenes in decreasing order of $s(p_i)$. $\varphi\big(l(p_i)\big)$ is determined as follows in order to put bounds to the video length of a play scene.

$$\varphi\big(l(p_i)\big) = \min\,[\, l(p_i),\ l_{th} + \delta \cdot L'\,], \tag{5}$$

where l_{th} denotes the threshold of the minimum time required for users to grasp the content of a play scene, L' denotes the current remaining time after subtracting the total video length of the selected play scenes from L, and δ is the coefficient to consider how much L' affects the video length of play scenes. Figure 6 summarizes the algorithm of the play scene selection.

After selecting play scenes, we rearrange the selected play scenes in the original temporal order, and then generate a dynamic video summary by concatenating the corresponding video segments.

For a static video summary, play scenes are selected with setting L to the number of scenes to be displayed and $\varphi\big(l(p_i)\big) = 1$, and the list of the selected play scenes is displayed with certain types of static media.

```
Input   L;   (the time specified by the user)
        N;   (the number of all play scenes)
        s(p_1), s(p_2), ..., s(p_N);   (the significance of p_i)
        l(p_1), l(p_2), ..., l(p_N);   (the time length of p_i)
Output  LIST;
(1)     LIST ← empty;   (initialization)
(2)     L' ← L;   (initialization)
(3)     sort out p_1, p_2, ..., p_N
        in the order of s(p_1) ≥ s(p_2) ≥ ... ≥ s(p_N)
(4)     for i = 1, 2, ..., N do
(5)         l(p_i) ← min [ l(p_i), l_th + δ · L' ];
(6)         if (l(p_i) ≤ L') then
(7)             PUT p_i into LIST;
(8)             L' ← L' − l(p_i);
(9)         end;
(10)    end;
```

Fig. 6 Algorithm for play scene selection.

4.3 User Adaptation

Generally, each play scene is ranked based on its significance to the game as described in Sect. 4.1. However, the significance of each play scene can be different depending on what each user likes/dislikes. Therefore, we also consider users' preferences in ranking play scenes as user adaptation.

We utilize a user profile which describes the user's preferences or interests with keywords and the user's preference degrees. Its items are as follows: a) team, players, or events, b) user's preference degree for each team, player, or event. In what follows, these are described as $(k,\ v_k)$. $s_p(p_i)$, the user's-preference-based significance degree of a play scene p_i, is calculated as

$$s_p(p_i) = \prod_{k \in F} \theta^{v_k}, \tag{6}$$

where F denotes the keyword set included in the user profile, v_k $(-1 \le v_k \le 1)$ denotes the user's preference degree of the keyword k, and θ $(\theta \ge 1)$ is the coefficient to consider how much users' preferences affect the significance of play scenes. $s_p(p_i)$ is determined such that

- Users' preferences can be ignored by setting $\theta = 1$.
- Regardless of the value of θ, no preference can be represented by setting $v_k = 0$.
- Stronger positive preferences can be represented by v_k closer to 1, while stronger negative preferences can be represented by v_k closer to -1.

As a consequence, $s_u(p_i)$, the user-adapted significance degree of a play scene p_i, is now given by

$$s_u(p_i) = s(p_i) \cdot s_p(p_i), \tag{7}$$

where $s(p_i)$ denotes the significance of the play scene considering only the importance of the play as in Eq. (1). Based on $s_u(p_i)$, the important play scenes are selected as described in Sect. 4.2 to generate a user-adapted video summary.

4.4 Device Adaptation

Each type of media device has different presentation capacity such as the display screen size and the network traffic speed. Therefore, we consider what kinds of elements need to be displayed on each device to present the generated summaries without changing the content.

Figure 7 shows the elements to be displayed for each type of summary. $[elements(if\ d = D_T)]$ represents the optional elements displayed when the device type d is D_T. $< elements,\ positions >$ represents the elements and where to display each of them. For example, given the device type d, a dynamic summary is a sequence of scenes, where the scene j is represented with the corresponding video segment and its text annotation described in the metadata, and they are displayed between $s_{d,j}$ and $e_{d,j}$ seconds after the summary started playing, where $s_{d,j} < e_{d,j}$. A list of static

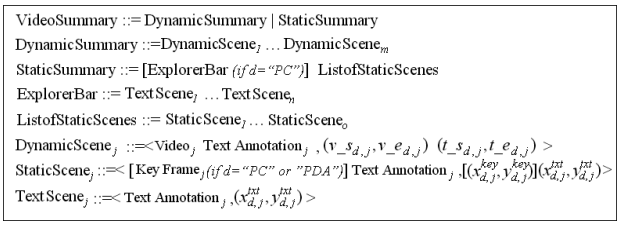


Fig. 7 Displayed elements.

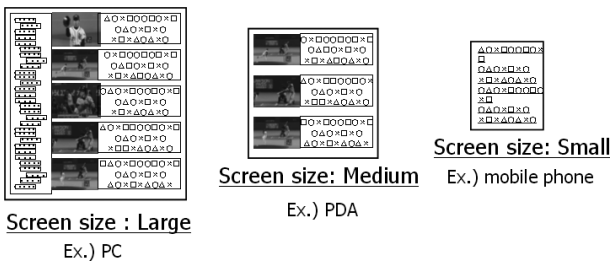


Fig. 8 Device adaptation.

scenes is a sequence of scenes, where the scene j is represented with an optional key image frame and its text annotation, and they are displayed at the coordinates $(x_{d,j}, y_{d,j})$ in the display screen.

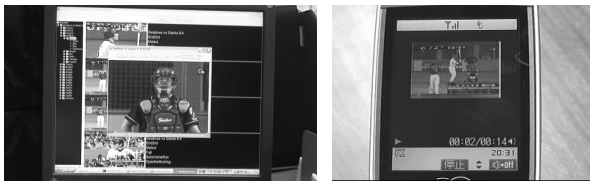
The key idea of device adaptation is that the optional elements are displayed only on the media device with a larger screen. Restricting the device types to PCs, PDAs, and mobile phones, we display the explorer bar only on PCs and the key image frames on PCs and PDAs. Figure 8 illustrates how a static video summary is presented on each device.

We use a device profile describing the device properties such as the device type, the screen size, and the network traffic speed in the form of $< a, u_a >$, where a represents the property name and u_a represents its value. The examples of the property name are *DeviceType*, *ScreenSize*, and *TrafficSpeed*, and the examples of their property values are *PC*, 17 *inch*, and 100 *Mbps*. After d is set to the value of *DeviceType* described in the device profile, the necessary elements are displayed at their temporal and spatial positions on each type of device as specified in Fig. 7.

5. Implementation and Demonstration

We have implemented our proposed framework. Figure 9 and Table 1 show the two types of client device, a PC and a mobile phone, used for demonstration and their specification, respectively.

Figure 10 shows how a tree structure-based presentation is actually displayed as a static video summary on these devices. When a tree structure-based presentation of a game is requested, all innings are displayed as shown in Figs. 10 (a) and 10 (b) on the PC and mobile phone respectively. On both devices, each row displays an inning scene using different types of media. Additionally, all at-bat scenes in each inning are also displayed sequentially in each row on the PC. That is, not only using different types



(a) PC (b) mobile phone

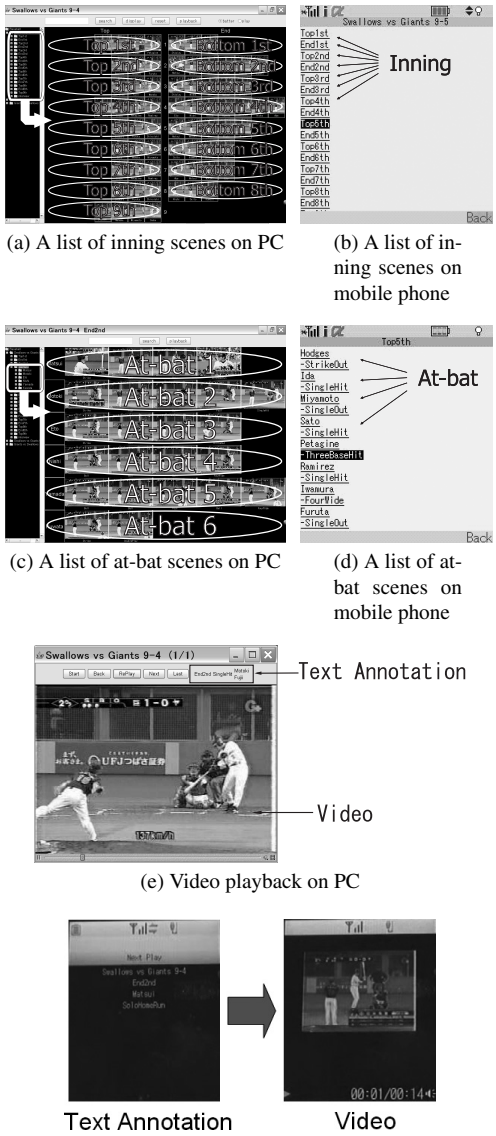
Fig. 9 Devices used for demonstration.

Table 1 Specification of devices.

mobile phone	model	Docomo FOMA N901iS
	display size	2.5 inch (240 × 345)
	video format	mpeg4
	video max. resolution	176 × 144
	video max. frame rate	15 fps
	max. traffic speed	384 kbps
	development kit	iappli Development Kit for DoJa-4.0(FOMA)
PC	model	Dell Precision 360
	display size	17 inch (1280 × 1024)
	video format	mpeg1
	max. traffic speed	100 Mbps
	development kit	Java 2 SDK, SE v1.4.2

of media to represent a scene, the content of a $StaticScene_j$ is different for each device: an inning for the mobile phone and an at-bat for the PC. The key image frame displayed on the PC is the first frame of the corresponding video segment. When an inning is selected in the explorer bar or in the list of static scenes, all at-bats in the selected inning are displayed as shown in Figs. 10 (c) and 10 (d), where each row displays an at-bat scene. Similarly, the content of a $StaticScene_j$ is an at-bat for the mobile phone, while it is a play for the PC. When displaying only important scenes selected in Sect. 4.2, the content of a $StaticScene_j$ corresponds to each play scene, and they are displayed in the same way as Fig. 10 (a) on the PC, according to the temporal order of the play in each inning, while they are displayed sequentially from top to bottom on the mobile phone.

Selecting a $StaticScene_j$ corresponding to a play scene activates the playback of the corresponding video segment. On the PC, a player window appears overlapping the static video summary as shown in Fig. 9 (a). The text annotation is simultaneously displayed while the corresponding video segment is played by setting $v\_s_{PC,j} = t\_s_{PC,j}$ and $v\_e_{PC,j} = t\_e_{PC,j}$. Once the video segment stops playing, the player window remains displayed for possible replay. On the mobile phone, the screen display switches from the static video summary to the video playback. The text annotation is displayed before the video segment is played to retain the readability of the text by setting $t\_e_{MobilePhone,j} = v\_s_{MobilePhone,j}$. Once the video stops playing, the screen display switches back to the static video summary. When playing a dynamic video summary, a $DynamicScene_j$ corresponds to a play and all play scenes are sequentially displayed in the same way. Figures 10 (e) and 10 (f) show how they are displayed. The videos are presented with the resolution and the frame rate suitable for each device: 320 × 240 and 30 fps for the PC and 176 × 144 and 15 fps for the mobile phone.



(a) A list of inning scenes on PC (b) A list of inning scenes on mobile phone

(c) A list of at-bat scenes on PC (d) A list of at-bat scenes on mobile phone

(e) Video playback on PC

(f) Video playback on mobile phone

Fig. 10 Demonstration.

6. Experiments

We prepared baseball videos as an example of sports videos. The parameters were experimentally determined as $\alpha = 0.8$, $\beta = 0.1$, $\gamma = 0.3$, $l_{th} = 14$, and $\delta = 0.02$, so that each play scene fully represents its content and more play scenes would be included both in our summaries, which are generated without user adaptation, and man-made summaries, which are broadcasted as highlights by the same TV stations as the original videos.

We firstly evaluated the effect of user adaptation. Changing the values of θ and v_k, we examined how the ranks of the play scenes corresponding to keywords in the user profile changed. The results are shown in Fig. 11. There are five play scenes which include the keyword, *Nioka*, in the

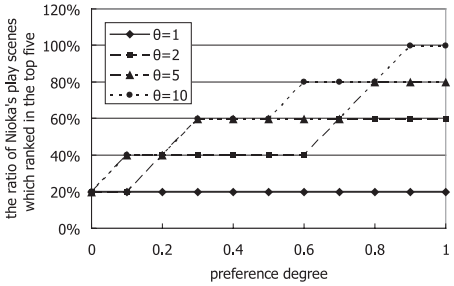


Fig. 11 The ratio of Nioka's play scenes which ranked in the top five when the user's profile includes the keyword *Nioka*.

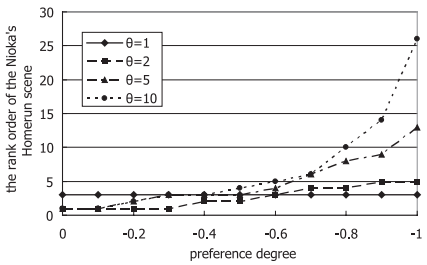


Fig. 12 Change of the rank order of the Nioka's Homerun scene.

whole video. In Fig. 11, the horizontal axis shows the user's preference degree v_{nioka}, and the vertical axis shows the ratio of Nioka's play scenes which ranked in the top five when the user's profile includes $< Nioka, v_{nioka} >$. $\theta = 1$ denotes no user adaptation. Since the user adaptation did not affect the summaries so much when $\theta = 2$, $\theta = 5$ or $\theta = 10$ is more suitable.

In addition, we verified the case where the users' preference degrees are negative. We used the profile which included $< Nioka, 0.5 >$ and $< Homerun, v_{homerun} >$, with changing $v_{homerun}$ from 0 to -1 by 0.1. Figure 12 shows the change of ranks of the Nioka's Homerun scene using this profile. The horizontal axis shows the user's preference degree $v_{homerun}$, and the vertical axis shows the rank of the Nioka's Homerun scene. At first, the Nioka's Homerun scene ranked first regardless of the value of θ because the profile included $< Nioka, 0.5 >$. The rank of the Nioka's Homerun scene went down as the user's preference degree $v_{homerun}$ became smaller, and the larger θ was, the faster the rank went down. From these experimental results, we can conclude that it is desirable to set θ to 5 or 10 for effective user adaptation, while v_k can be freely set from -1 to 1 by users.

Finally, the summaries generated for a game between Swallows and Giants considering the user's preferences with $< Swallows, 0.5 >$ and $< Giants, 0.5 >$ with setting $\theta = 5$ and the time length of a dynamic video summary to 110 seconds are compared with the summary generated without user adaptation in Fig. 13. Surrounded by a thick line are play scenes of Giants. The shaded play scenes were added in the summary as a result of user adaptation. The results have confirmed that more scenes of the favorite team were included in the video summaries by using user profiles.

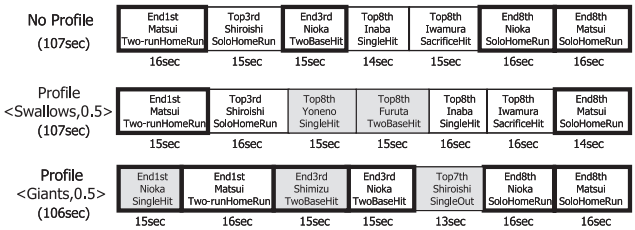


Fig. 13 Examples of generated summaries for a baseball video with/without user profiles.

Table 2 Questionnaire results (PC).

	1	2	3	4	5	average evaluation
operability	0	2	4	6	3	3.7
presentation based on tree structure	2	1	1	9	2	3.5
display of important scenes	0	1	2	3	9	4.3

Table 3 Questionnaire results (mobile phone).

	1	2	3	4	5	average evaluation
operability	0	1	4	8	2	3.7
presentation based on tree structure	0	3	3	5	4	3.7
display of important scenes	0	1	0	10	4	4.1

Next, in order to evaluate the validity of device adaptation, we gave 15 users the following questionnaires about the demonstration using the PC and the mobile phone.
Q.1: Is the operability of the interface good?
Q.2: Is each function convenient?
Q.3: Is there any advantage or disadvantage for each device?

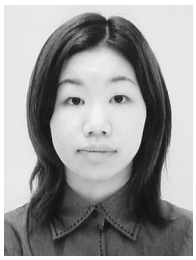
The results for Q.1 and Q.2 are shown in Tables 2 and 3. The users responded on a scale of 1-5 with 1 being very bad and 5 being very good. There was no big difference in the evaluation results for the PC and the mobile phone. The convenience of each function was not affected by the difference of the presentation style. According to the responses to Q.3, we confirmed that the user was able to effectively access the scenes they wanted and each device had no specific advantage or disadvantage especially for viewing the dynamic video summary. However, the following issues were pointed out for viewing the static video summary: 1) presenting the static video summary on the PC takes longer as the number of scenes increases due to the large data volume of images, 2) the key image frame which better expresses the content of a play scene should be selected to improve the understandability of the static video summary for the PC, and 3) it is easier to understand the tree structure of the game on the PC since the explore bar is presented and the scenes are laid out more properly to represent the tree structure on the PC, while the mobile phone merely presents the scenes sequentially from the top, as shown in Figs. 10 (a) and 10 (b).

7. Conclusion

In this paper, we proposed a unified framework for user and device adaptation in summarizing sports videos. Our method realized user adaptation by considering users' preferences in determining the significance of play scenes based on user profiles and the metadata. After selecting important scenes based on their significance, device adaptation was also realized by changing the media types and the temporal and spatial positions for displaying the selected scenes. Analyzing the content of the generated summaries has confirmed that user adaptation can be successfully achieved simply by setting positive/negative preference degrees for a few keywords in user profiles. Questionnaires to users also verified that there was no big difference in users' satisfaction levels with the understandability of video summaries which are displayed differently on a PC and a mobile phone. As future work, we need to investigate how to acquire users' preferences and need to evaluate our framework with other types of media device.

References

[1] A. Hanjalic, Content-based analysis of digital video, Kluwer Academic Publishers, 2004.
[2] N. Nitta, Y. Takahashi, and N. Babaguchi, "Automatic personalized video abstraction for sports videos using metadata," Multimedia Tools and Applications, vol.41, no.1, pp.1–25, 2009.
[3] Y. Takahashi, N. Nitta, and N. Babaguchi, "User and device adaptation for sports video content," Proc. IEEE ICME 2007, pp.1051–1054, July 2007.
[4] Y. Takahashi, N. Nitta, and N. Babaguchi, "Video summarization for large sports video archives," Proc. IEEE ICME 2005, July 2005.
[5] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, "Personalized abstraction of broadcasted American football video by highlight selection," IEEE Trans. Multimed., vol.6, no.4, pp.575–586, Aug. 2004.
[6] A. Jaimes, T. Echigo, M. Teraguchi, and F. Satoh, "Learning personalized video highlights from detailed MPEG-7 metadata," Proc. IEEE ICIP 2002, I, pp.133–136, Sept. 2002.
[7] K. Masumitsu and T. Echigo, "Meta-data framework for constructing individualized video digest," Proc. IEEE ICIP 2001, vol.3, pp.390–393, Oct. 2001.
[8] B.L. Tseng, C.-Y. Lin, and J.R. Smith, "Using MPEG-7 and MPEG-21 for personalizing video," IEEE Multimedia, vol.11, no.1, pp.42–52, Jan.-March 2004.
[9] B.L. Tseng, C.-Y. Lin, and J.R. Smith, "Video personalization and summarization system for usage environment," J. Vis. Commun. Image Represent., vol.15, pp.370–392, May 2004.
[10] S.-F. Chang and A. Vetro, "Video adaptation: Concepts, technologies, and open issues," Proc. IEEE, vol.93, no.1, pp.148–158, Jan. 2005.
[11] A.M. Ferman and A.M. Tekalp, "Two-stage hierarchical video summary extraction to match low-level user browsing preferences," IEEE Trans. Multimed., vol.5, no.2, pp.244–256, June 2003.
[12] A. Ekin, A.M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," IEEE Trans. Image Process., vol.12, no.7, pp.796–807, July 2003.
[13] D. Tjondronegoro, Y.-P.P. Chen, and B. Pham, "Integrating highlights for more complete sports video summarization," IEEE Multimedia, vol.11, no.4, pp.22–37, Oct. 2004.
[14] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T.S. Huang, "Highlights extraction from sports video based on an audio-visual marker detection framework," Proc. IEEE ICME 2005, July 2005.
[15] http://www.chiariglione.org/mpeg/
[16] J.M. Martinez, "Overview of the MPEG-7 standard (version 6.0)," ISO/IEC JTC1/SC29/WG11 N4509, Dec. 2001.



Naoko Nitta received the B.E., M.E. and Ph.D. degrees in engineering science from Osaka University, in 1998, 2000 and 2003, respectively. She is currently a Lecturer in Graduate School of Engineering, Osaka University. From 2002 to 2004, she was a research fellow of the Japan Society for the Promotion of Science. From 2003 to 2004, she was a Visiting Scholar at Columbia University, New York. She received Best Paper Award of 2006 Pacific-Rim Conference on Multimedia (PCM2006). Her research interests are in the areas of video content analysis and image/audio processing. She is a member of the IEEE and the ITE.

Noboru Babaguchi received the B.E., M.E. and Ph.D. degrees in communication engineering from Osaka University, in 1979, 1981 and 1984, respectively. He is currently a Professor in Graduate School of Engineering, Osaka University. From 1996 to 1997, he was a Visiting Scholar at the University of California, San Diego. His research interests include image analysis, multimedia computing and intelligent systems, currently content based video indexing and summarization. He has published over 100 journal and conference papers and several textbooks. He received Best Paper Award of 2006 Pacific-Rim Conference on Multimedia (PCM2006). He is a member of the IEEE, the ACM, the IPSJ, the ITE and the JSAI.