RSViewer: An Efficient Video Viewer for Racquet Sports Focusing on Rally Scenes

Shunya Kawamura[1], Tsukasa Fukusato[1], Tatsunori Hirai[1] and Shigeo Morishima[2]
[1]*Waseda University, Tokyo, Japan*
[2]*Waseda Research Institute for Science and Engineering / JST CREST, Tokyo, Japan*

Keywords: Video Summarization, Rally Shot Detection, Shot Evaluation, Fast-Forwarding, User Interface.

Abstract: This paper presents RSViewer, a video browsing system specialized for racquet sports, which reflects users' interests. Methods to support users in browsing racquet sports matches by summarizing video composed of important rally shots have been discussed in a previous study. However, the method is not practical enough because the auditory events should be manually annotated in advance to detect such scenes. Therefore, we propose an automatic rally shot detection based on shot clustering method using white line detection. Our system calculates the importance of rally shots based on audio features. As the result, the summarized video can facilitate users find and review the information they need. The result of experiments shows that our method is effective in an aspect of efficient video browsing experience. Furthermore, we propose a high-speed playback method customized to racquet sports video and realize more efficient video browsing experience.

1 INTRODUCTION

Many people enjoy watching sports video on TV or through the internet in their spare time. The development of broadcasting system made it possible for people to watch video whenever they want. However, watching a racquet sport video (RSV) is time-consuming due to the following reason: many games are held as a tournament and they are often long matches (e.g., an average tennis match is over two hour). As the result, viewers who do not have enough time cannot watch several matches. Moreover, when they watch one match, frequent and unimportant scenes to understand the whole game, such as change court and break time to play, make them weary. Accordingly, a system that enables users to efficiently watch RSV is necessary.

In this paper, we propose a RSV browsing system employing unsupervised rally shot detection based on visual features and rally-rank evaluation based on subjective evaluation. A system overview of our important rally detection is shown in Figure 1. This process is composed of the following two parts: (Step 1) rally shot detection is performed to generate the summary composed of only rally scenes. Since rally scenes can be detected by finding scenes which contain white lines, our system detects clusters composed of rally shots in an unsupervised manner. (Step 2)

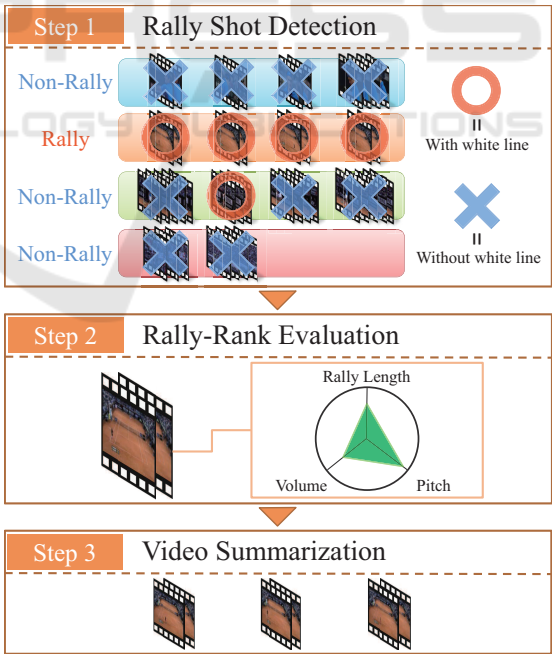


Figure 1: System overview of our important rally detection. All of the process is automatic without any pre-information while state-of-the-art method needs manually labelled audio information.

The importance of each detected rally shot is evaluated on the basis of audio- and content-based fea-

Kawamura, S., Fukusato, T., Hirai, T. and Morishima, S.
RSViewer: An Efficient Video Viewer for Racquet Sports Focusing on Rally Scenes.
DOI: 10.5220/0005670802470254
In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 2: IVAPP*, pages 249-256
ISBN: 978-989-758-175-5

tures. Given the results of rally-rank evaluation, users can watch the resulting short video summary that is composed of only important rally shots. Additionally, we present an efficient viewing system "RSViewer," a double meaning of "RSV viewer" and "Rally Scene Viewer." With its functions of important rally scene playback and its fast-forwarding, users can gain their satisfying experience to watch RSV focusing on rally scenes.

2 RELATED WORK

There are two approaches to efficiently watch a video: scene-based summarization and fast-forward viewing. Scene-based summarization aims to generate the highlight video called "video summary" and is applied to various sports video (Liu et al., 2009), (Tjondronegoro et al., 2004), (Zhao et al., 2012). Liu et al. proposed a method for rally shot detection based on unsupervised shot clustering and supervised audio classification using Support Vector Machines (Liu et al., 2009). This method can detect rally shots with high accuracy. However, video editors must manually annotate labels of audio information for the first 30 minutes to create the summary. Tjondronegoro et al. proposed a highlight scene detection for various sports video based on a cheer, whistle and text information (Tjondronegoro et al., 2004). Zhao et al. extracted the replay scene by searching logos that located before and after replays (Zhao et al., 2012). Highlights and replays can attract viewers and are very important scene because they are selected by skilled editors. Such scenes enable viewers to attentively watch a specific motion, while it is difficult to understand the tide of the game by the lack of information such as scores. Hence, highlights and replays are inadequate for the video summarization with the understanding of the game.

Fast-forward viewing approach aims to let viewers watch all of the video in a short time without removing any scene (Cheng et al., 2009), (Kurihara, 2012). Cheng et al. presented a system to watch a video with positively or negatively accelerating the playback speed (Cheng et al., 2009). Users can watch their interested scenes on low speed playback and skip their insensitive ones on high speed playback. However, this system has some limitation; for example, if users do not understand the scene structure of the video, to control the playback speed is difficult since they cannot predict when their interested scenes start. Kurihara proposed two-level fast-forwarding system for movies based on subtitles (Kurihara, 2012). However, this method is not suited to RSV because usually there is no subtitle in RSV and fewer speech than in a movie content.

Table 1: Mainly representative scenes on broadcast RSV.

Period	Scenes
broadcast start $\sim$ before game start	commentator's talk, player introduction, practice, fan
game start $\sim$ game set	rally, change court, fan, replay, player's zoom, court maintenance
after game set $\sim$ broadcast end	commentator's talk, fan, interview, ceremony

As the related work of racquet sports recognition, methods to detect events, such as services and net play, have been proposed (Chang et al., 2012), (Chen and Zhang, 2006), (Huang et al., 2012). While event detection is helpful for evaluating rally importance, it is inadequate for RSV summarization because considering lots of events makes the summarizing process much complicated. To detect events in RSV, rally scene detection is essential and has been discussed in previous studies (Kijak et al., 2003), (Liu et al., 2009), (Zhong and Chang, 2001). However, these methods require models which are adapted for an input video. Our approach overcomes such problem and enables automatic detection and summarization of rally scenes only by video input.

Compared with the prior studies, our main contributions are following two points: i) our method is fully automatic while preserving the quality of the video summary and ii) our system is the first work to present an user interface specialized for RSV including summarization and fast-forwarding functions.

3 RSV'S STRUCTURE

We will describe the structure of RSV treated in this study. On our observation, the period of broadcasting RSV is divided into three parts: "broadcast start $\sim$ before game start," "game start $\sim$ game set," "after game set $\sim$ broadcast end." Table 1 shows the scenes mainly included in the three periods. In table 1, the scenes that allow viewers to understand the racquet sports match are "rally scene" and "replay scene." In rally scenes, a scoreboard is always displayed and all of players' ball hits are included, whereas replay scenes have only a few ones without any scoreboard. Therefore, we assume that the most important scene for understanding the match is a rally scene, and aim to generate the video summary composed of only important rally scenes.

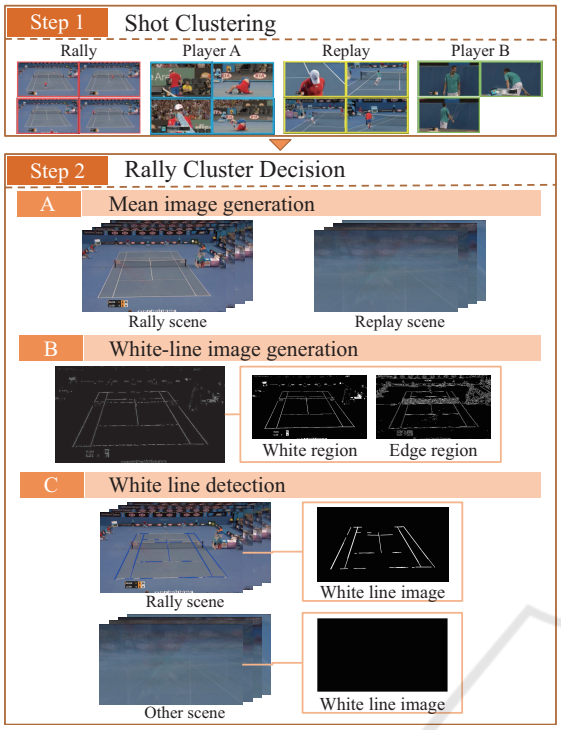


Figure 2: Overview of our rally shot detection. This method is composed of two parts: shot clustering and rally cluster decision. After the shot clustering, rally scenes are detected on the basis of white line detection for mean images.

4 RALLY SHOT DETECTION

We assume that the rally scene is most important to understand racquet sports matches. First of all, we automatically detect rally shots for summarizing various RSV. Figure 2 shows the overview of our rally shot detection.

4.1 Method

4.1.1 Shot Clustering

Outline. First, after the segmentation of the input video into shots based on Lian's technique (Lian et al., 2010), we perform Liu's unsupervised shot clustering (Figure 2, Step 1) (Liu et al., 2009) . This method enables to automatically decide the number of clusters by the merging stop criterion and is suitable for generating the video summary without any pre-information. The main idea is the repetition of the following three steps: (i) calculate the distance (scene similarity) of every two shots (or clusters) on the basis of HSV histogram. (ii) merge the most similar two shots into one cluster. (iii) when the criterion, discussed in detail later, reaches the minimum, the merging process is stopped and the clusters composed of similar shots are generated (otherwise, repeat these process from (i)).

Process. Firstly, key frames k_l^i ($l = \{1, L\}$) are extracted from the i-th shot at equal intervals (in this paper, $L = 5$). From the key frames, HSV color histograms are computed to evaluate the shot similarity, quantized into 256 color bins (H*S*V = 16*4*4). Using these features, the distance (scene similarity) of every two shots (or clusters) is calculated as follows.

$$SD(s_i, s_j) = \frac{1}{2}\left(M + \hat{M}\right) \tag{1}$$

$$M = \min\left\{d\left(k_{l_1}^i, k_{l_2}^j\right)\right\} \tag{2}$$

$$d(k_{l_1}^i, k_{l_2}^j) = \frac{1}{256}\Sigma_{b=1}^{256}|H_{l_1}^i(b) - H_{l_2}^j(b)| \tag{3}$$

where s_i, s_j are respectively i-th shot and j-th one, M and $\hat{M}$ are respectively the minimum and second minimum values, $d(k_{l_1}^i, k_{l_2}^j)$ is the HSV histogram distance between key frames $k_{l_1}^i$ and $k_{l_2}^i$, H_l^i is a color histogram value in k_l^i, and b is a number in the 256 color bins. The resulting closest pair of shots (or scenes) is merged into one new scene cluster n. Then, to update the key frames' histograms of the cluster n, Equation (4) is applied.

$$H_l^n = \frac{N_i * H_l^i + N_j * H_l^j}{N_i + N_j} \tag{4}$$

where N_i and N_j are the number of frames in i-th and j-th shot, respectively. By the repetition of the above process, we perform the shot clustering. Finally, the number of repetition times are decided by calculating the minimum of Equation (5).

$$E_n = J_n + k_n \tag{5}$$

$$\begin{aligned} J_n &= \frac{\Sigma_{c=0}^{K_n} J_w^c}{J_{inter}} \\ &= \frac{\Sigma_{c=0}^{K_n}\Sigma_{i=0}^{N_c}\|H_{L/2}^i - H_{mean}^c\|}{\Sigma_{i=0}^{N}\|H_{L/2}^{i_{init}} - H_{mean}\|} \end{aligned} \tag{6}$$

$$k_n = \frac{K_n}{N} \tag{7}$$

where J_{inter} is the total inter-cluster scatter before clustering, J_w^c is the intra-cluster scatter of scene cluster c, N and N_c are the total shot numbers before clustering and of scene cluster c, respectively. $\|\bullet\|$ is the Euclidean distance, $H_{L/2}^{i_{init}}$ and $H_{L/2}^i$ are the i-th shot before clustering and in scene cluster c, respectively.

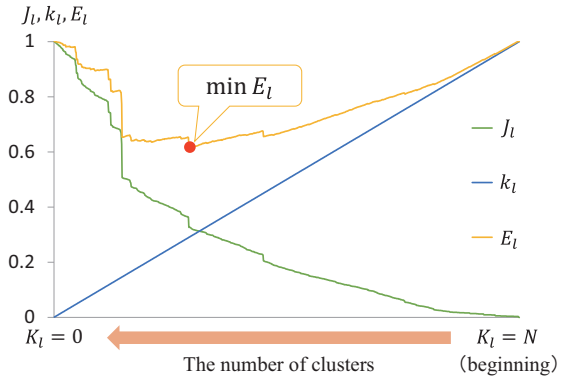


Figure 3: Instances of the transition of evaluation functions (E_l, J_l and k_l) for the number of clusters. When E_l is a minimum value, the clustering process is stopped and the number of clusters are decided.

H_{mean} and H^c_{mean} are the mean feature values of the shots before clustering and in scene cluster c, respectively. k_n is the ratio of the scene number to the total number of shots before clustering. Figure 3 shows the transition of evaluation function (E_l, J_l and k_l) for the total number of clusters. Since the total number of clusters get fewer, J_l value increases and k_n decreases on the clustering process. With this tradeoff between J_l value and k_n, we empirically try to find a stop point of the clustering. When E_l reaches a minimum value, the clustering process is stopped and the number of clusters are decided. The resulting clusters are composed of shots which have similar color. In RSV, we can generate scene clusters such as rallies, replays and players' zoom, as shown in Step 1 of Figure 2.

4.1.2 Rally Cluster Decision

We describe a method to detect rally clusters without supervised learning. First, clusters composed of only a few shots are considered as non-rally ones because the rate of rally shots in RSV is more than 10%. To determine rally clusters from the remaining clusters, we focus on court lines which always exist in rally scenes. In rally scenes, court lines are continually projected on the same position since camera position and direction are generally fixed. In other scenes (non-rally scenes), court lines move on the screen since camera position and direction often change. Therefore, by generating a mean image from each shot, we can detect rally scenes because white lines remain on the images only from rally shots and are blurred from the others (Figure 2, Step 2.C). White lines on the mean image are detected, and if the rate of the number of shots with white lines to that of shots in the cluster is high (over 60% in this paper), all shots in the cluster are decided as rally ones.

4.2 Experiment and Result

We tested the performance of rally shot detection. First, we detected rally scenes included in approximately two-hour tennis video (clay court) with our unsupervised method and Liu's supervised method (Liu et al., 2009). In this experiments, we used criteria called "Precision" and "Recall" rate. They are defined as

$$Precision = \frac{C}{D} \quad (8)$$

$$Recall = \frac{C}{T} \quad (9)$$

where T is the actual number of rally shots, D is the detected number of them, and C is the correctly detected number of them. Table 2 shows the comparison of the detection accuracy. Our method and Liu's one can detect the rally scene with same accuracy (precision and recall rate) because they detect the same clusters as rally ones. Since we are able to automatically detect rally shots using an unsupervised method, our method is superior to Liu's method. Furthermore, Table 3 shows the performance for six RSV. While the precision rate of table tennis video are lower because lots of court-view shots such as practice in pre-match and court-maintenance scenes are included, almost all of the recall rate is high. Therefore, this detection accuracy is sufficient for watching most of all rallies when the video summary is generated.

Table 2: Comparison of the accuracy of rally shot detection (Liu's method and our system). The tested video includes 192 rallies in 962 shots.

Method	Recall	Precision
Liu et al., 2009 (supervised)	0.984	0.900
Our method (**unsupervised**)	**0.984**	**0.900**

5 RALLY-RANK EVALUATION

5.1 Method

We quantitatively calculate each rally's importance using the following three features: volume V, pitch P and rally length L. We evaluate the cheer volume, which becomes large value in case of the exciting rally or the important point. Pitch means the voice pitch, which gets higher when spectators are excited. This value greatly varies whether a player scores or not. In a tennis video, for example, P value is very small when a player services a first fault. Moreover, we consider rally shot length L, assuming that longer

Table 3: Accuracy of our rally shot detection method for various RSV.

Input video	Tournament	Number of shots		Recall	Precision
		Rally	Total		
Tennis [clay]	Rome Semi-Final 2012	192	962	0.984	0.900
Tennis [hard]	Indian Wells Semi-Final 2012	157	744	1.000	0.981
Badminton 1	Mumbai Masters 2013	94	408	1.000	0.949
Badminton 2	BWF Championships Final 2013	109	789	0.972	0.972
Table tennis 1	World Table Tennis Championships 2013	70	290	1.000	0.642
Table tennis 2	London Olympics Semi-Final 2012	88	629	1.000	0.688
Total	–	710	3822	**0.992**	**0.864**

rallies are more interesting. Given the volume V_r, pitch P_r, and rally length L_r, each rally rank I_r for the r-th rally is calculated as

$$I_r = \alpha V_r + \beta P_r + \gamma L_r + \delta \quad (10)$$

where pitch and volume are respectively calculated by mean values (during 5 seconds) of the spectrum centroids (i.e., pitch) and the maximum cheer volume between the second half of a rally shot to 5 seconds after its end. The weights (α, β, γ and δ) are determined by multiple regression analysis of the subjective experiment values. The subjective experimentation was performed for each type of racquet sports and almost all of the participants had no experience with racquet sports. More detail about the experiment will be described in the following Section 5.2. The video summary in a certain time is interactively generated by adjusting a threshold for the evaluated rally importance.

5.2 Experiment and Result

5.2.1 Experiment

We performed the subjective investigation to calculate the weights in Equation (10). In this experiment, seven people (including two people with experience in racquet sports) in their twenties participated. They watched 120 rallies and subjectively answered each rally's excitement as integer values between 1 and 5 (higher scores mean more interest). Rallies that participants watched were randomly chosen from two badminton, two table tennis and two tennis matches (20 rallies per video) without considering the progress of matches. The weights of Equation (10) were calculated from mean values of the subjective evaluation. As the result, the evaluation values among people with non-experience and with experience in racquet sports are similar. Hence, by using Equation (10), we can generate the video summary regardless of experience in racquet sports.

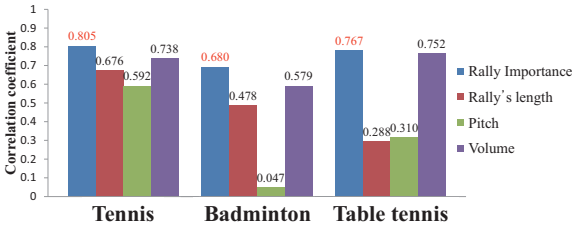


Figure 4: Correlation coefficients between mean subjective value and evaluated values (importances of rally and each feature's value. Correlation coefficients of rally importances which are calculated by our rally-rank evaluation function are highest in all kind of racquet sports).

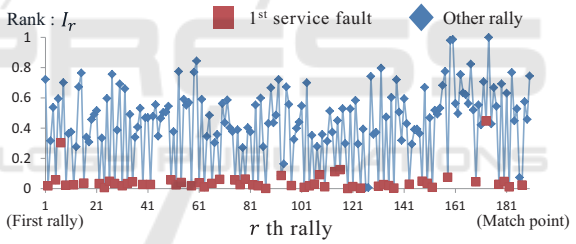
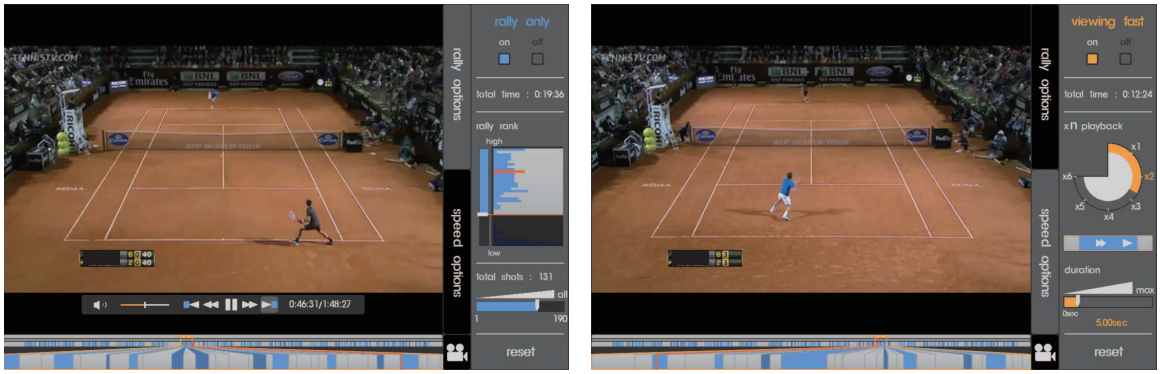


Figure 5: Example of rally importance for our evaluation function. A red point means a first service fault, and blue one is other rally. There are two red points that are relatively high because of spectators' cheers by their misunderstanding the faults as service ace.

5.2.2 Result

To verify that our rally-rank evaluation can reflect viewers' interest, we consider experimental results. We calculated correlation coefficients between mean subjective value and evaluated values, i.e. importance of rally and each feature's value (Figure 4). The result shows that correlation coefficients between subjective values and rally importance by our rally-rank evaluation function are highest in all kind of racquet sports. Therefore, the function evaluates rallies' importance on the basis of viewers' interest. Additionally, the subjective evaluation values among people with non-experience and with experience in racquet sports are similar. Hence, the resulting video summary is re-



(a) (b)

Figure 6: User Interface. The lower slider bar on the interface shows rally shots as blue bars. Right area is settings; (a) rally options for rally importance and (b) speed options for fast-forwarding.

gardless of experience in racquet sports if our rally-rank evaluation function can reflect viewers' interest.

An example of the ranking result for an approximately two-hour tennis video is shown in Figure 5. This result shows that first service faults are evaluated in lower importance (red points: 56 shots, in Figure 5). Therefore, our evaluation function is effective to exclude unimportant rallies. Additionally, for higher important rallies for Equation (10), essential rallies to understand the match result, i.e., game points and match points, are also evaluated higher. In brief, both essential rallies and exciting rallies are evaluated as important ones by considering the excitement in our evaluation function. However, this function tends to calculate short and important scenes (e.g., service aces) as low values. To correctly evaluate the importance of a specific scene, we will improve our evaluation function, for example, by considering some image features such as player's movement.

6 SYSTEM IMPLEMENTATION

This section presents a system that allows users to efficiently watch RSV. This system has useful functions based on rally-rank evaluation and fast-forward such as "SmartPlayer" and "CinemaGazer" (Cheng et al., 2009), (Kurihara, 2012).

6.1 Interface's Functions

The user interface for RSV is shown in Figure 6. This interface is more useful than major video players such as *QuickTime* and *Windows Media Player* in the following four points: i) users can easily seek rally scenes by using rally skip buttons. ii) the slider bar on this interface visualizes all rally shots in the RSV (below on the interface, blue bars: rally shots). This visualization is helpful for users who have watched RSV to seek a specific game and set in a short time. For example, if they want to watch a game point, they should select the rally before a long non-rally period. iii) by adjusting a threshold for the rally importance, users can watch the video in time they want. iv) fast-forwarding function enables users to watch the video in shorter time. More detail is described in the following.

6.1.1 Fast-forwarding

On our observation of some RSV, we assume that there are the following three factors to understand rallies: *ball trajectory*, *player's movement*, and *point*. *Ball trajectory* means the viewer's understanding of ball speed and trajectory in a rally. Watching ball trajectory can assist viewers to understand rally points. *Player's movement* means the viewer's understanding of player's running after and hitting a ball in a rally. This factor is a criteria that viewers assess player's superior or inferiority in a rally and a match. *Point* means the viewer's understanding of which a player scores in a rally. This is an important factor that allows viewers to understand the result of the match, especially the winner. Thus, by investigating the relationship between each factor and playback speed, we can reveal the limit speed of playback with retaining the understanding level of the match. The relationship between the understanding of rallies and the playback speed is obtained by subjective evaluation such as Kurihara's experiment (Kurihara, 2012). All of the participants answered the understanding of the three factors as integer values between 1 to 5 (higher scores mean their well-understanding) when they watched each 6 rallies randomly chosen from

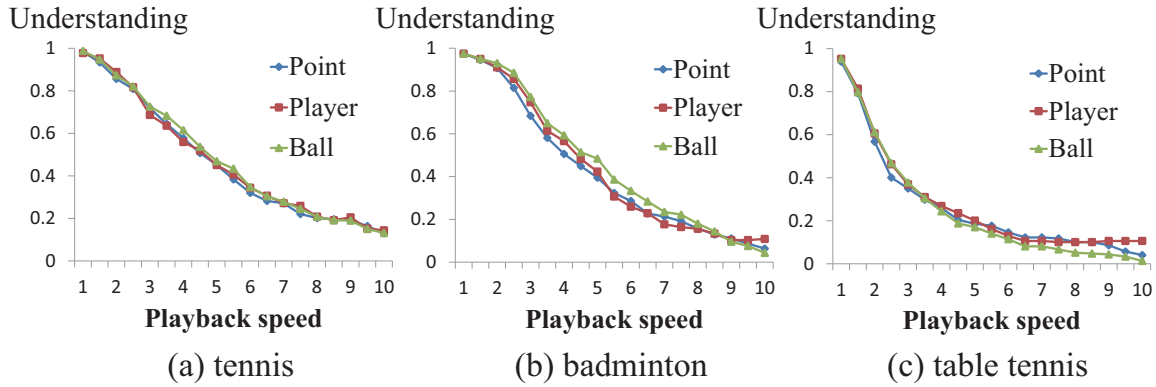


Figure 7: Relationship between the playback speed and the understanding of ball trajectory, point, and player's movement: (a) tennis, (b) badminton, (c) table tennis. Faster playback speed equally decreases the all factors' understanding regardless of the kind of racquet sports.

one badminton, one table tennis and one tennis video. Each rally scene was played from $\times$ 10.0 to $\times$ 1.0 in decrements of $\times$ 0.5. Figure 7 indicates the relationship between the playback speed and the understanding level (average) answered by all participants. This result shows that faster playback speed equally decreases the understanding of all factors regardless of racquet sports. Therefore, we implement the three-level playback speed according to the understanding level (100%, 80% and 50%). In addition, with the understanding of the last few hits in a rally, this system plays at normal-speed for the last several period. This system is based on the following two tendencies in RSV: replay scenes contain only a few hits in a rally because of their importance, and we can understand rallies even if the sound is inaudible during high-speed playback.

6.2 User Experiment

Our interface has been used to watch the RSV summary, and was also evaluated by users who provided individual feedback and comments. In this experiment, participants watched some matches of badminton, table tennis and tennis after they received the explanation of our system. Participants performed the following three tasks; to watch i) only rally scenes, ii) high-ranked ones by adjusting a threshold to rally rank, and iii) ones with fast-forwarding function. First, we asked participants to watch only rally scenes in a video. Some participants said "*when I watched RSV, I often became boring because of the long non-rally period. However, this function is excellent because I can enjoy only rally scenes.*" In other opinion, *while it is possible for some viewers, who want to watch rally scenes focusing on the detail of player's movement, to need some replay scenes, the summary composed of only rally scenes is enough for others who want to watch only good rallies.*

In the video summary, which is generated by their operation for our summarization system using rally-rank evaluation, our system can help users to generate the video summary by interactively adjusting a threshold to rally rank. Especially, in tennis video, users can watch the match without discomfort when first service faults and lets are removed in the summary video. In future, we plan to add editing functions for detected rally scenes. In our fast-forwarding system, some users stated that *this function was effective to emphasize the moment that a player scores.* We assumed that this function was for users who wanted to watch the match in a short time since they sensed the timing that a rally ended by shifting from fast-forward to normal playback. In a negative opinion, *few people are uncomfortable when the playback speed changes*. In future, we need to consider how to change the playback speed.

7 CONCLUSION AND FUTURE WORK

We proposed "RSViewer," a rally scene viewing system for racquet sports. We presented rally shot detection that enabled to automatically detect rallies with high accuracy without any annotation. However, it is difficult to apply our method to other kind of sports video since our system employed rally scenes' feature that the camera position and direction is generally fixed. We constructed an interface to efficiently watch a RSV based on rally-rank evaluation and fast-forwarding. This evaluation metric made it possible to exclude unimportant rallies from a video summary. On the other hand, the scenes such as service aces,

that often attract viewer's interest, are evaluated as less important scene since the length is short. To overcome such a problem, we will consider the players' movement in rally and replay scene for generating more informative video summary. Our user interface enabled users to watch the video summary according to their preference. Furthermore, by using our fast-forwarding function, users can watch the video summary in much shorter time without losing the content understanding. In future, we will investigate the perception-based features considering the difference between people with experience and non-experience in sports, thereby aiming to implement the system that allows users to watch the video efficiently and sufficiently. Moreover, we are planning to take users' experience of the sports into account. In this way, individuality of perceiving the content of a video might be considered. We will continue to implement our system to realize a better user experience of watching RSV.

ACKNOWLEDGEMENTS

This work was supported by OngaCREST, CREST, JST.

REFERENCES

Chang, C., Fang, M., Kuo, C., and Yang, N. (2012). Event detection for broadcast tennis videos based on trajectory analysis. In *Proc. of 2nd International Conference on Communications and Networks (CECNet)*, pages 1800–1803.

Chen, W. and Zhang, Y. (2006). Tracking ball and players with applications to highlight ranking of broadcasting table tennis video. In *Proc. of IEEE International Conference on Computational Engineering in Systems Applications*, volume 2, pages 1896–1903.

Cheng, K., Luo, S., Chen, B., and Chu, H. (2009). Smartplayer: user-centric video fast-forwarding. In *Proc. of SIGCHI Conference on Human Factors in Computing Systems*, pages 789–798.

Huang, Q., Cox, S., Zhou, X., and Xie, L. (2012). Detection of ball hits in a tennis game using audio and visual information. In *Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–10.

Kijak, E., Oisel, L., and Gros, P. (2003). Temporal structure analysis of broadcast tennis video using hidden markov models. In *Proc. of SPIE Storage and Retrieval for Media Databases*, volume 5021, pages 289–299.

Kurihara, K. (2012). Cinemagazer: a system for watching videos at very high speed. In *Proc. of AVI'12*, pages 108–115.

Lian, S., Dong, Y., and Wang, H. (2010). Efficient temporal segmentation for sports programs with special cases. In *Proc. of the Advances in Multimedia Information Processing – PCM 2010 PT I*, volume 6297, pages 381–391.

Liu, C., Huang, Q., Jiang, S., Xing, L., Ye, Q., and Gao, W. (2009). A framework for flexible summarization of racquet sports video using multiple modalities. *Computer Vision and Image Understanding*, 113(3):415–424.

Tjondronegoro, D., Chen, Y., and B, P. (2004). Integrating highlights for more complete sports video summarization. *IEEE Trans on. Multimedia*, 11(4):22–27.

Zhao, F., Dong, Y., Wei, Z., and Wang, H. (2012). Matching logos for slow motion replay detection in broadcast sports video. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1409–1412.

Zhong, D. and Chang, S. (2001). Structure analysis of sports video using domain models. In *Proc. of IEEE International Conference on Multimedia and Expo*, pages 713–716.