2016 Joint 8th International Conference on Soft Computing and Intelligent Systems and 2016 17th International Symposium on Advanced Intelligent Systems

Comparative Experiments on Models for Automatic Baseball Video Tagging

Komei Arasawa
Web Intelligence Time-Space (WITS) Laboratory
Graduate School of Engineering
Muroran Institute of Technology
27–1 Mizumoto-cho, Muroran, Hokkaido 050–8585, Japan
Email: 16043005@mmm.muroran-it.ac.jp

Shun Hattori
Web Intelligence Time-Space (WITS) Laboratory
College of Information and Systems
Muroran Institute of Technology
27–1 Mizumoto-cho, Muroran, Hokkaido 050–8585, Japan
Email: hattori@csse.muroran-it.ac.jp

***Abstract*—To enable us to select the only scenes that we want to watch in a baseball video and personalize its highlights sub-video, we require an Automatic Baseball Video Tagging system that divides a baseball video into multiple sub-videos per at-bat scene automatically and also appends tag information relevant to at-bat scenes. The previous paper proposed several Tagging algorithms using ball-by-ball textual report and voice recognition. To improve our system, this paper introduces a more refined model for baseball games, and performs comparative experiments on models with regard to their recall, precision, and F-measure.**

I. Introduction

Highlights videos are frequently used in sports news programs. Because most of those highlights videos are produced by the side of sports news programs, the same sport game has different highlights videos depending on sports news programs. Because highlight scenes are high spots in a sport video, one aim of producing a highlights sub-video of the sport video is to enable those who could not satisfyingly watch the sport video to enjoy it in a short time. However, scenes which we want to watch in a sport video are dependent on us, and a highlights sub-video of the sport video cannot have all of our wanted scenes. Therefore, highlights videos based on estimating the general needs of many viewers and produced by the side of sports news programs cannot meet their each individual needs completely.

The above-mentioned problem of a highlights video produced by a sports news program could be solved by enabling viewers to produce a highlights video by themselves. One method of enabling a viewer to produce a highlights video by her/himself is "s/he records a sports relay program previously and then edits its video by collecting the only scenes which s/he wants to watch". However, in general, such work requires a great deal of time because the viewer has to watch the whole video while editing the scenes which s/he wants to watch and fast-forwarding through the other scenes which s/he does not want to watch.

Let us imagine that a sport video has been already divided into multiple chapters per scene. Chaptering is the function to enable a viewer to easily move to the point of a sport video that s/he wants to watch by dividing the sport video into multiple sub-videos per scene and appending a caption to each scene of the sport video. Because the sport video has been already divided into chapters per scene with their caption, the viewer does not have to watch the whole sport video and all s/he needs to do is to collect the only chapters that s/he wants to watch by using their captions as a reference. That is to say, the viewer does not need to take a deal of time.

Providing the sport video that is previously divided into chapters per scene would enable viewers to watch their wanted scenes easily. We focus on the "Tagging" [1] that divides a video into sub-videos per scene with not only their caption but also their Tag information, which is their detailed information showing what event happened in each scene, and we are developing an automatic tagging system of baseball videos using ball-by-ball textual report and voice recognition [2]. Because the previous work cannot achieve the enough high tagging accuracy owing to insufficiently modeling for baseball games, this paper proposes a more refined model for baseball games, and performs evaluation experiments by comparing with the previous models for baseball games.

II. Tagging algorithm

A. An overview of the proposed system

To tag every scene of a baseball video, a tagging system requires two kinds of clues, "what events happened in the baseball video?" and "when did the events happen?". This research aims at an Automatic Baseball Video Tagging system that divides a baseball video into multiple sub-videos per at-bat scene automatically and also appends tag information relevant to at-bat scenes. In this paper, the former of these clues is called as **event**, and the latter is called as **event time**.

The at-bat events of a baseball game are extracted from its ball-by-ball textual report on the Web. To divide a baseball video into multiple sub-videos per at-bat scene, the system requires events E_i $(i = 1, 2, ..., N)$ of a baseball game per at-bat scene. This process is shown as Step 1 of Fig. 1. If a viewer wants to watch all scenes of a player A in the baseball video, the viewer collects the scenes whose tag information

978-1-5090-2678-4/16 $31.00 © 2016 IEEE
DOI 10.1109/SCIS&ISIS.2016.70

contains "player A". In this case, the system appends tag information relevant to an at-bat scene, which indicates how the player A participates in the scene, even if the scene is not an at-bat scene of the player A and the player A participates little in the scene. Therefore, as tag information for an event E_i, the system requires not only the batter-name of the event, the situation when the batter stepped to the plate, and the result of the event, but also the names of the other players who participated if only a little in the at-bat scene, and how they participated. The process of event extraction acquires these events every time a new batter steps to the plate, that is to say per at-bat scene, after the end of a baseball game which is a tagging target of the system. An instance of an acquired event E_i is "the batter B of the event E_i stepped to the plate in his third at-bat with two outs and the runner C on the first base, and grounded out to third. In addition, the pitcher D was pitching, the runner C stole the second base, and the third baseman E put the batter B out". Here, the system recognizes multiple scenes which the same player participated in as a batter to be different events.

Secondly, the system appends the event time, which consists of the event's start time and end time, to these events. Here, as the first precess, it appends the event's start time T_i $(i = 1, 2, ..., N)$ to each E_i of all acquired events. This process is shown as Step 2 of Fig. 1. As the second process, by adopting the event's start time T_{i+1} of the next event E_{i+1} as the event's end time of the event E_i, the event times of the events of a baseball video have been computed. When computing the event's start time of an event E_i, this paper focuses on a ball-by-ball voice showing that the batter of the event E_i is stepping to the plate. An instance of a ball-by-ball voice showing that the batter of an event is stepping to the plate includes "the batter is (the player) A". To enable the tagging per at-bat scene of a baseball video that contains its tag information, this paper employs AmiVoice as a voice recognition software when the system recognizes a ball-by-ball voice, and defines a ball-by-ball point and an at-bat ball-by-ball point as the following, which are used to append the event time to each event of a baseball game.

- Ball-by-ball point : It is the instant when a player name appeared in the results of the ball-by-ball voice recognition. And this paper defines $P(i, 1), P(i, 2), ..., P(i, j)$ as ball-by-ball points of the batter name of an event E_i in order of their appearing. Here, the ball-by-ball points $P(i, j)$ of a batter contain a point where a sports commentator called his name in a scene which is not his at-bat scene as well as his at-bat ball-by-ball points.
- At-bat ball-by-ball point : It is the ball-by-ball point that is contained in a ball-by-ball voice showing that the batter of an event is stepping to the plate. Here, an at-bat scene does not always have only one at-bat ball-by-ball point.

B. The addition of event time

1) The calculation of event's start time: Firstly, the event's start time T_1 of the first event E_1 is exceptionally calculated. Because this research supposes that the system will be loaded on a television recorder, the system calculates the start time of a baseball game in its recorded video and adopts its start time as the event's start time T_1 of the first event E_1, by recognizing the time when the record of its relay program started and extracting its start time from its ball-by-ball textual report on the Web.

Secondly, the calculation method of event's start time after the event E_2 is explained. If an event E_i $(i = 2, 3, ..., N)$ has multiple ball-by-ball points $P(i, j)$, the system has to search them for the at-bat ball-by-ball point showing that the batter of the event E_i is stepping to the plate. First, the system calculates the event's estimated start time $\hat{T}_i$ $(i = 1, 2, ..., N)$ when the event E_i happened in the baseball video. Second, the system adopts the guessed at-bat ball-by-ball point as the event's start time T_i, by searching for the ball-by-ball point(s) of the event E_i in the near-field region of the event's estimated start time $\hat{T}_i$ where is from $\hat{T}_i + \Delta t_1$ to $\hat{T}_i + \Delta t_2$ based on the parameters Δt_1 (min) and Δt_2 (min), and guessing that the firstly-appearing ball-by-ball point $P(i, j)$ is the at-bat ball-by-ball point of the event E_i. Here, if there is no ball-by-ball point in the near-field region of the event's estimated start time $\hat{T}_i$, the system adopts the $\hat{T}_i$ as the event's start time T_i of the event E_i. The calculation method of the event's start time T_i of the event E_i is shown in Fig. 2, which regards the mark "◯" as a ball-by-ball point of the batter of the event E_i and the mark "△" as a ball-by-ball point of the other players who participated in the baseball game.

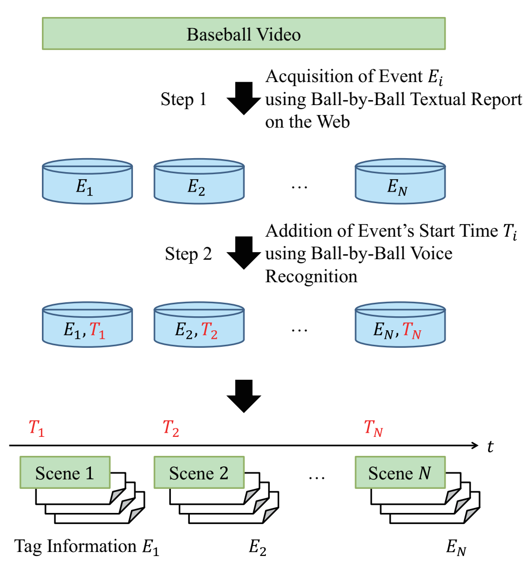

Figure 1. An overview of the proposed system.

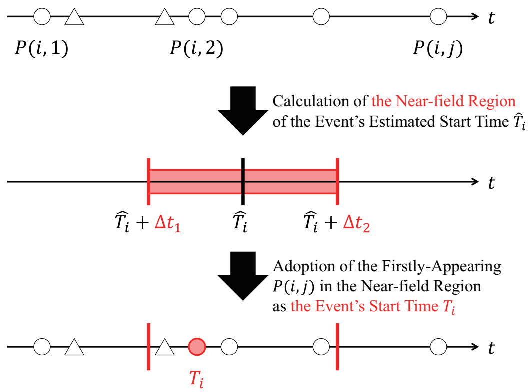

Figure 2. Calculation method of an event's start time.

2) The calculation of event's end time: As the event's end time T'_i $(i = 1, 2, ..., N-1)$ of an event E_i, the system adopts the event's start time T_{i+1} of the next event E_{i+1}. Here, there might be cases that $T_i \geq T_{i+1}$ because the system calculates the event's start time T_i of each event E_i independently without considering the context of each other events. In this case, as the event's end time T'_i of the event E_i, the system adopts the numerical value that is calculated by adding A, $B \times \beta_i$, $B' \times \beta_i$, $B_w \times \mathrm{W_l}(E_i) \times \mathrm{W_r}(E_i)$ based on each estimation method, (i), (ii), (iii), and (iv) to the event's start time T_i of the event E_i. In addition, because the last event E_N does not have the next event E_{N+1} and also the next event's start time T_{N+1}, as the event's end time T'_N of the last event E_N, the system adopts the numerical value that is calculated by adding A, $B \times \beta_N$, $B' \times \beta_N$, $B_w \times \mathrm{W_l}(E_N) \times \mathrm{W_r}(E_N)$, like the above-mentioned case.

III. Models for Baseball Games

A. The Previous Models

The event's estimated start time $\hat{T}_i$ of an event E_i is computed by considering two factors, "what scenes happened before the event E_i?" and "how long are the scenes?". These two factors determine a model for baseball games. The previous paper [2] proposed three kinds of models for baseball games to calculate the event's estimated start time $\hat{T}_i$ of an event E_i.

(i) The estimation based on mean time per unit event

First, this estimation method calculates the mean time A per unit event by the following formula (1) with the game time T that is extracted from its ball-by-ball textual report on the Web.

$$A = \frac{\text{(The game time of a baseball game)}}{\text{(The number of all events)}} = \frac{T}{N} \tag{1}$$

Second, this estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i by the following formula (2) with the mean time A per unit event.

$$\hat{T}_i = \hat{T}_{i-1} + A = T_1 + A \times (i-1) \tag{2}$$

(ii) The estimation based on mean time per unit pitch

First, based on the number of pitches β_i $(i = 1, 2, ..., N)$ that were thrown in an event E_i of a baseball game, which is extracted from its ball-by-ball textual report on the Web, this estimation method sums up these numbers of pitches to compute the number of all pitches of the whole baseball game.

And this estimation method calculates the mean time B per unit pitch by the following formula (3) with the number of all pitches of the whole baseball game.

$$B = \frac{\text{(The game time of a baseball game)}}{\text{(The number of all pitches)}} = \frac{T}{\sum_{i=1}^{N} \beta_i} \tag{3}$$

Second, this estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i whose number of pitches is β_i by the following formula (4) with the mean time B per unit pitch.

$$\hat{T}_i = \hat{T}_{i-1} + B \times \beta_{i-1} = T_1 + B \times \sum_{k=1}^{i-1} \beta_k \tag{4}$$

(iii) The estimation by applying the consideration of the changes of batting and fielding sides to the estimation method (ii)

This estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i based on the same calculation approach as the estimation method (ii). However, only if an event E_i is the preceded event by a change of batting and fielding sides, this estimation method adds the uniform necessary time of a change of batting and fielding sides as the parameter Δt_s (sec) to the event's estimated start time $\hat{T}_i$. Here, whether or not an event E_i of the baseball video is the preceded event by a change of batting and fielding sides is also extracted from its ball-by-ball textual report on the Web, and it is shown as the following function (5).

$$\mathrm{cs}(E_i) = \begin{cases} 1 & (E_i \text{ is preceded by a change of sides}) \\ 0 & (\text{otherwise}) \end{cases} \tag{5}$$

First, this estimation method calculates the mean time B' per unit pitch to which is applied the consideration of a change of batting and fielding sides by the following formula (6) based on the formulas (3) and (5).

$$B' = \frac{T - \Delta t_s \times \sum_{i=1}^{N} \mathrm{cs}(E_i)}{\sum_{i=1}^{N} \beta_i} \tag{6}$$

Second, this estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i by the following formula (7) with the amended mean time B' per unit pitch.

$$\hat{T}_i = \hat{T}_{i-1} + B' \times \beta_{i-1} + \Delta t_s \times \mathrm{cs}(E_i) \tag{7}$$

B. A Refined Model

To refine the previous models for baseball games, this paper proposes the following calculation method of the event's estimated start time by applying the consideration of the necessary time of a scene of changing a pitcher and the weighting of pitching-time per unit pitch. This estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i based on the same calculation approach as the estimation method (iii). However, only if an event E_i is the preceding event after changing a pitcher, this estimation method adds the uniform necessary time of changing a pitcher as the parameter Δt_p (sec) to the event's estimated start time $\hat{T}_i$.

And if an event happened in the late innings, and/or if there was a runner on a base when the batter of an event stepped to the plate, this estimation method adds the weight (w_l and/or w_r respectively) to pitching-time per unit pitch. Here, this paper defines that the late innings are after the seventh inning. And this estimation method is called the estimation method (iv), and is computed by the following process.

First, whether or not an event E_i is right after changing a pitcher is extracted from its ball-by-ball textual report on the Web, and it is shown as the following function (8).

$$\mathrm{cp}(E_i) = \begin{cases} 1 & (E_i \text{ follows after changing a pitcher}) \\ 0 & (\text{otherwise}) \end{cases} \tag{8}$$

Second, the inning in which each event of a baseball game happened is extracted from its ball-by-ball textual report on the Web, and the weight based on the parameter w_l (≥ 1) is added to the pitching-time per unit pitch by the following function (9) that shows whether or not the inning of an event is after the seventh inning.

$$\mathrm{W_l}(E_i) = \begin{cases} w_l & (E_i \text{ is after the seventh inning}) \\ 1 & (\text{otherwise}) \end{cases} \tag{9}$$

And the existence of runner(s) when the batter of an event of a baseball game stepped to the plate is extracted from its ball-by-ball textual report on the Web, and the weight based on the parameter w_r (≥ 1) is added to the pitching-time per unit pitch by the following function (10).

$$\mathrm{W_r}(E_i) = \begin{cases} w_r & (\text{there is a runner}) \\ 1 & (\text{otherwise}) \end{cases} \tag{10}$$

This estimation method calculates the weighted mean time B_w per unit pitch by the following formula (11) with the formulas (5), (8), (9), and (10).

$$B_w = \frac{T - \Delta t_s \times \sum_{i=1}^{N} \mathrm{cs}(E_i) - \Delta t_p \times \sum_{i=1}^{N} \mathrm{cp}(E_i)}{\sum_{i=1}^{N} \beta_i \times \mathrm{W_l}(E_i) \times \mathrm{W_r}(E_i)} \tag{11}$$

Finally, this estimation method calculates the event's estimated start time $\hat{T}_i$ of an event E_i by the following formula (12) with the weighted mean time B_w.

$$\begin{aligned} \hat{T}_i = \hat{T}_{i-1} + \Delta t_s \times \mathrm{cs}(E_i) + \Delta t_p \times \mathrm{cp}(E_i) \\ + B_w \times \mathrm{W_l}(E_{i-1}) \times \mathrm{W_r}(E_{i-1}) \times \beta_{i-1} \end{aligned} \tag{12}$$

IV. Comparative Experiment

This paper has proposed the tagging algorithms for a baseball video to acquire its events per at-bat scene and append their event time to them. This chapter evaluates only the event's start time and end time which are appended to each event. This evaluation defines the actual start time of an event as the instant when the batter of the event stepped to the plate and his name and season record were displayed on screen. And this evaluation defines the actual end time of an event as the actual start time of its next event. However, only if an event is the preceded event by a change of batting and fielding sides, this evaluation defines the actual end time of the event as the instant when the score sheet of the baseball game was displayed on screen while the change was going on. Based on these definitions of actual start time and end time, this paper compares the tagging algorithms with regard to their recall, precision, F-measure, and square error between the actual start time of an event and its computed start time. Here, this paper employs F-measure as the tagging accuracy to evaluate the whole system. The defined parameters, Δt_1, Δt_2, Δt_s, Δt_p, w_l, and w_r, are in the following range.

- $-15 \leq \Delta t_1 \leq 15$ (increments of 1 min)
- $\Delta t_1 \leq \Delta t_2 \leq 15$ (increments of 1 min)
- $0 \leq \Delta t_s \leq 300$ (increments of 30 sec)
- $0 \leq \Delta t_p \leq 300$ (increments of 30 sec)
- $1.0 \leq w_l \leq 1.5$ (increments of 0.1)
- $1.0 \leq w_r \leq 1.5$ (increments of 0.1)

First, the tagging system computes the event's estimated start time $\hat{T}_i$ of an event E_i. Second, it searches for the ball-by-ball points in the near-field region of $\hat{T}_i$ with the parameters Δt_1 and Δt_2, and adopts the firstly-appearing ball-by-ball point $P(i, j)$ as the event's start time T_i of the event E_i. In some case, there is some possibility that the adopted ball-by-ball point $P(i, j)$ is a point where a sports commentator called the batter name of the event E_i in a scene which is not his at-bat scene, because it is no more than a supposition that this adopted ball-by-ball point $P(i, j)$ is the at-bat ball-by-ball point showing that the batter of the event E_i was stepping to the plate. The near-field region of $\hat{T}_i$ has to be set smaller, because setting the search range oversize raises the possibility that the system fails to detect a ball-by-ball point where a sports commentator called in a scene which is not the at-bat scene of the batter of the event E_i rather than his at-bat ball-by-ball point to be searched for. In order to do that, the system needs to compute the event's estimated start time of each event accurately. Accurately computing the event's estimated start time of each event enables not only to reduce the possibility of incorrect detection of its at-bat ball-by-ball point but also to append more accurate event's start time even if there is no ball-by-ball point in the near-field region and the system exceptionally adopts the event's estimated start time $\hat{T}_i$ as the event's start time T_i of the event E_i. First, this chapter inspects the accuracy of event's estimated start times to be computed by the respective estimation methods, and performs an evaluation of the utility of refining a model for baseball games.

The recorded videos of relay programs of five baseball games are employed for the comparative experiments. Here, four recorded videos among these are the relay programs which fully broadcast the whole of a baseball game, and the remainder (Data 1) is the relay program which partway broadcast only the seven innings of a baseball game. And these videos contain television commercials. Table I shows the accuracy of the tagging that adopts the event's estimated start time of each event of a baseball game which is computed by the respective estimation methods as its event's start time, that is to say based on only the estimation of the event's start time without ball-by-ball voice recognition. Because the tagging accuracy rises along with the change from the estimation method (i) to the estimation method (iv), we can confirm that the proposed calculation of the event's estimated start time of each event is more accurate than the previous calculations. However, from the square error between the actual start time of an event and its computed start time, it is only striking that the square error of the tagging based on the estimation method (i) is larger than the square errors of the tagging based on the other estimation methods, and the square error does not change significantly along with the change from the estimation method (ii) to the estimation method (iv). From this, it can be also viewed that the proposed model for baseball games has not yet been refined enough, and we confirm that there is some possibility of refining it more.

Second, this chapter evaluates how the computing accuracy of the event's estimated start time of each event affects the whole tagging system by refining the modeling for baseball games, based on the evaluation of computing accuracy of the event's estimated start time of each event, that is to say the square error between the actual start time of an event and its event's estimated start time. Table II to VI show the tagging accuracy for the five baseball games depending on the respective estimation methods. And Table VII shows the mean of tagging accuracies for the five baseball games depending on the respective estimation methods. Table II to VI reveal that the tagging accuracy tends to rise along with the change from the estimation method (i) to the estimation method (iv), that is to say the refinement of the modeling for baseball games. In addition, it is also shown that refining a model for baseball games has some utility, because the square error becomes smaller along with the change from the estimation method (i) to the estimation method (iv). However, examining the mean of tagging accuracies for the five baseball games makes a difference of opinion. Table VII shows that even though the mean tagging accuracy seems to rise along with the change from the estimation method (i) to the estimation method (iv), the impact on the tagging accuracy by refining a model for baseball games seems to be smaller than our expectation because the improvement rate of the mean tagging accuracy along with the change from the estimation method (i) to the estimation method (iv) is low and the square error between the actual start time of an event and its computed start time is getting bigger along with the change from the estimation method (i) to the estimation method (iv).

This disappointing impact would be affected by the value-setting of each parameter in the refined model. The individual tagging accuracies for baseball games have become lower when the system sets the value that maximizes the mean tagging accuracy to each parameter, compared with when the system optimizes the parameters to each baseball game. The gap between the tagging accuracies based on the estimation method (iv) when the parameters are optimized to only Data 2 and when they are optimized to all data is the largest and 0.192. The remainder of this chapter discusses each parameter in detail to solve this issue.

Firstly, we discuss the parameters Δt_1 and Δt_2, which determine the near-field region of the event's estimated start time. Fig. 3 shows the F-measure for each baseball game (Data 1$\sim$5), and Fig. 4 shows the mean F-measure and the mean Square error depending on Δt_1. Here, the other parameters fix on the values that maximize the individual F-measures and the mean F-measure respectively, and Fig. 3 considers the relationship $\Delta t_1 \leq \Delta t_2$. The parameter Δt_1 has a great effect on the tagging accuracy because each curve of F-measure has a peak. And it has been able to be optimized satisfactorily because there is not a large gap between the values of Δt_1 that maximize the individual F-measures and the mean F-measure. Fig. 5 shows the F-measure for each baseball game, and Fig. 6 shows the mean F-measure and the mean Square error depending on the parameter Δt_2. Except for Data 4, each curve of the F-measure grows along with Δt_2 and converges almost to the greatest tagging accuracy. Therefore, the system needs to set a rather large value to the parameter Δt_2.

Secondly, we discuss the parameters Δt_s, Δt_p, w_l, and w_r, which involve the refined model for baseball games. Fig. 7 shows the F-measure for each baseball game, and Fig. 8 shows the mean F-measure and the mean Square error depending on the parameter Δt_s. And also Fig. 9 shows the F-measure of each baseball game, and Fig. 10 shows the mean F-measure and the mean Square error depending on the parameter w_r. The mean F-measures in Fig. 8 and 10 do not vary greatly depending on Δt_s and w_r respectively.

Table I
TAGGING ACCURACY BASED ON EVENT'S ESTIMATED START TIME.

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	–	–	–	–	–	–	0.162	0.131	0.145	306.043
(ii)	–	–	–	–	–	–	0.311	0.253	0.279	218.216
(iii)	–	–	90	–	–	–	**0.343**	**0.279**	**0.308**	**214.696**
(iv)	–	–	90	0	1.0	1.0	**0.343**	**0.279**	**0.308**	**214.696**

Table II
TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS. (DATA 1)

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-8	3	–	–	–	–	0.680	0.442	0.536	147.840
(ii)	-3	6	–	–	–	–	0.735	0.554	0.632	101.313
(iii)	-3	6	0	–	–	–	0.735	0.554	0.632	101.313
(iv)	-1	4	90	90	1.0	1.5	**0.774**	**0.614**	**0.685**	**51.246**

Table III
TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS. (DATA 2)

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-9	4	–	–	–	–	0.646	0.343	0.448	226.795
(ii)	-8	6	–	–	–	–	0.589	0.311	0.407	250.039
(iii)	-8	5	270	–	–	–	0.587	0.351	0.439	237.768
(iv)	-3	6	150	270	1.0	1.4	**0.712**	**0.501**	**0.588**	**129.531**

Table IV
TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS. (DATA 3)

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-6	12	–	–	–	–	0.656	0.363	0.467	273.179
(ii)	0	6	–	–	–	–	0.691	0.538	0.605	94.281
(iii)	0	6	0	–	–	–	0.691	0.538	0.605	94.281
(iv)	0	8	60	30	1.0	1.0	**0.742**	**0.567**	**0.643**	**93.313**

Table V
TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS. (DATA 4)

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-11	4	–	–	–	–	0.470	0.230	0.308	380.260
(ii)	-6	-1	–	–	–	–	0.510	0.331	0.401	204.679
(iii)	-7	-2	180	–	–	–	0.579	0.363	0.446	164.528
(iv)	0	1	30	150	1.0	1.1	**0.703**	**0.578**	**0.635**	**79.201**

Table VI
TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS. (DATA 5)

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-6	7	–	–	–	–	0.477	0.281	0.354	277.716
(ii)	3	5	–	–	–	–	0.489	0.337	0.399	225.825
(iii)	3	5	30	–	–	–	0.504	0.347	0.411	234.103
(iv)	-3	8	30	30	1.0	1.5	**0.564**	**0.347**	**0.430**	**201.514**

Table VII
THE MEAN TAGGING ACCURACY DEPENDING ON RESPECTIVE ESTIMATION METHODS.

Estimation method	Δt_1	Δt_2	Δt_s	Δt_p	w_l	w_r	Recall	Precision	F-measure	Square error
(i)	-9	6	–	–	–	–	0.559	0.304	0.394	289.940
(ii)	-3	6	–	–	–	–	0.502	0.363	0.421	**198.485**
(iii)	-4	7	60	–	–	–	0.524	0.360	0.427	199.403
(iv)	-3	10	90	30	1.1	1.2	**0.588**	**0.373**	**0.457**	202.628

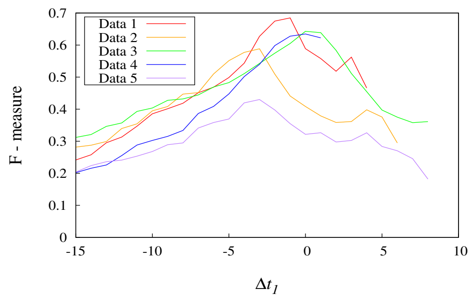

Figure 3. F-measure for each data based on Δt_1 [min].

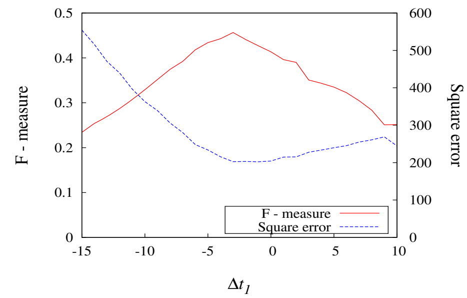

Figure 4. Mean F-measure and Square error based on Δt_1 [min].

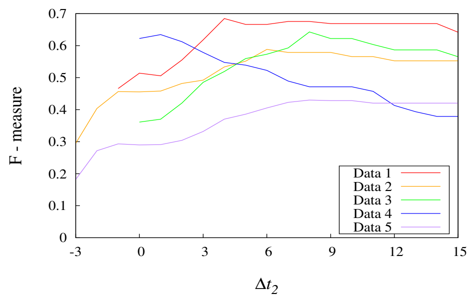

Figure 5. F-measure for each data based on Δt_2 [min].

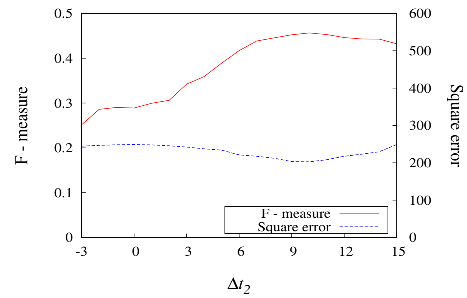

Figure 6. Mean F-measure and Square error based on Δt_2 [min].

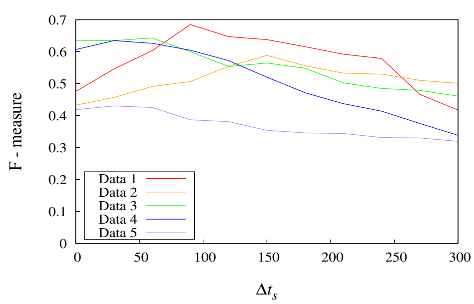

Figure 7. F-measure for each data based on Δt_s [sec].

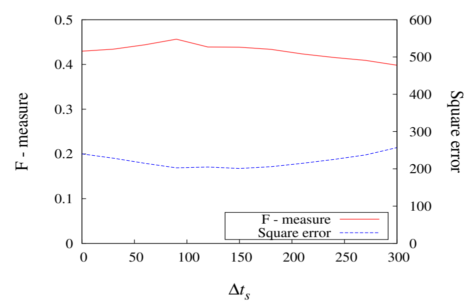

Figure 8. Mean F-measure and Square error based on Δt_s [sec].

However, Δt_s and w_r are effective parameters to improve the tagging accuracy compared with the previous models, whose $\Delta t_s = 0$ and $w_r = 1.0$, because the individual F-measures in Fig. 7 and 9 vary depending on Δt_s and w_r respectively. Therefor, our future system needs to enable the value-setting of these parameters Δt_s and w_r depending on each baseball game because the impact only by using the value that maximizes the mean tagging accuracy as each parameter is low unfortunately.

Fig. 11 shows the F-measure for each baseball game, and Fig. 12 shows the mean F-measure and the mean Square error depending on the parameter Δt_p. The parameter Δt_p has a similar tendency to Δt_s and w_r, but the mean F-measure varies much more greatly depending on Δt_p.

Fig. 11 reveals that the optimal parameter Δt_p that maximizes the individual F-measure for each baseball game has a variety of values, and each curve of F-measure has a sharper peak.

Therefor, our future system also needs to enable the value-setting of this parameter Δt_p depending on each baseball game because the impact only by setting the value that maximizes the mean tagging accuracy is not high unfortunately.

Fig. 13 shows the F-measure for each baseball game, and Fig. 14 shows the mean F-measure and the mean Square error depending on the parameter w_l. The parameter w_l has a tendency that the greater the parameter w_l is, the lower the individual F-measures are. Therefore, the system needs to set a rather small value to the parameter w_l.

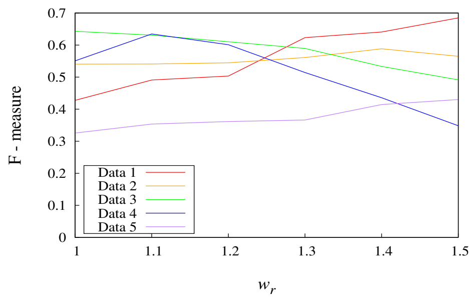

Figure 9. F-measure for each data based on w_r.

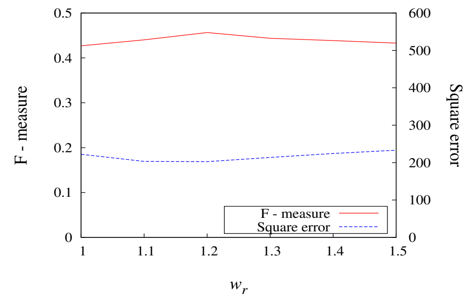

Figure 10. Mean F-measure and Square error based on w_r.

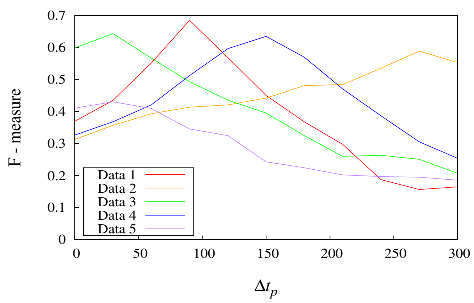

Figure 11. F-measure for each data based on Δt_p [sec].

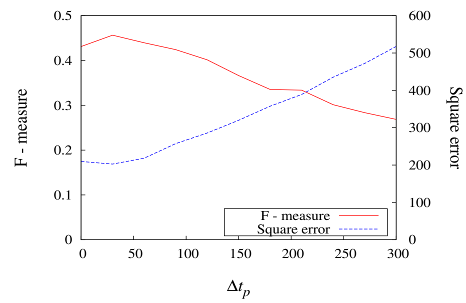

Figure 12. Mean F-measure and Square error based on Δt_p [sec].

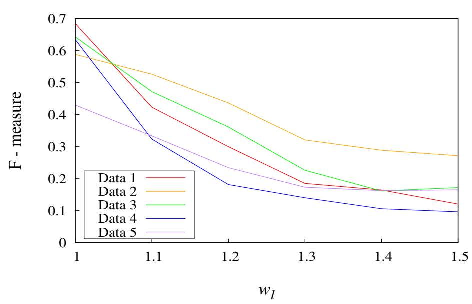

Figure 13. F-measure for each data based on w_l.

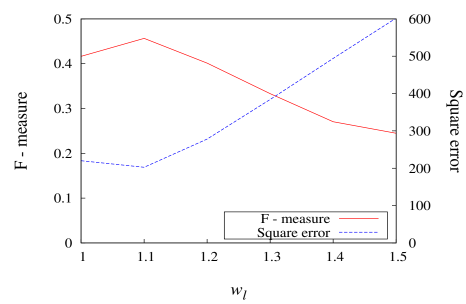

Figure 14. Mean F-measure and Square error based on w_l.

V. Conclusion

To develop an Automatic Baseball Video Tagging system, this paper has proposed the tagging algorithm that is introduced a refined model for baseball games into, which adopts the weight to the pitching-time per unit pitch based on the situation of an event and the necessary time of changing a pitcher, and has confirmed the tagging accuracy is improved by introducing the refined model and some useful knowledge to optimize the parameters that determine the near-field region of the event's estimated start time of an event when searching for ball-by-ball points could be acquired.

We plan to perform experiments by using many baseball games to inspect the optimization for each parameter, and examine more sophisticated models for baseball games to compute the event's estimated start time more accurately. And we aim to improve the tagging accuracy by inventing a novel tagging algorithm that consists of multiple techniques, for instance, not only ball-by-ball voice recognition but also tagging techniques based on the analysis of moving images and the tagging with the peculiarity of ball-by-ball textual reports on the Web.

References

[1] Chih-Yi Chiu, Po-Chih Lin, Sheng-Yang Li, Tsung-Han Tsai, and Yu-Lung Tsai, "Tagging Webcast Text in Baseball Videos by Video Segmentation and Text Alignment," IEEE Transactions on Circuits and Systems for Video Technology, Vol.22, Issue 7, pp.999–1013 (2012).

[2] Komei Arasawa and Shun Hattori, "Automatic Baseball Video Tagging Using Ball-by-Ball Textual Report and Voice Recognition," IEICE Technical Report, Vol.115, No.405, pp.1–6 (2016).