WebTracer: A new web usability evaluation environment using gazing point information

Noboru Nakamichi a,*, Makoto Sakai b, Kazuyuki Shima c, Jian Hu d, Ken-ichi Matsumoto a

a Software Engineering Laboratory, Nara Institute of Science and Technology, 8916-5 Takayama, Ikoma, Nara, 630-0101, Japan
b SRA Key Technology Laboratory Inc., Marusho Bldg. 5F, 3-12, Yotsuba, Shinjuku-ku, Tokyo, 160-0004, Japan
c Hiroshima City University, 3-4-1 Ozuka-Higashi, Asaminami-ku, Hiroshima, 731-3194, Japan
d Hitachi Ltd., Hitachi System Plaza Shinkawasaki, 890 Kashimada, Saiwai-ku, Kawasaki, Kanagawa, 212-8567, Japan

Received 18 March 2005; received in revised form 17 August 2005; accepted 26 January 2006
Available online 20 March 2006

Abstract

WebTracer is a new usability evaluation environment that supports recording, replaying, and analysis of a gazing point and operation while a user is browsing a website. WebTracer can record a user’s gazing point and operation compactly.

Results of an experimental evaluation showed that the size of the operation history recorded by WebTracer was from 1/10 to 1/20 of the size of data recorded by an MPEG-2 or MPEG-4 format. With its compact form, the results of usability testing with the gazing point can be efficiently shared. It is expected to easily share empirical data between researchers. Also, evaluators can easily send the testing results as a feedback to the developers over the Internet.

Moreover, the results show a possibility that gazing points related to usability. For example, when the menu of a Web page is divided into two panes, gazing points move more quickly. It seems that WebTracer efficiently improves usability evaluation, with gazing point data helping to identify problems on Web pages.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Web usability evaluation; Quantities; Gazing point

1. Introduction

We know experientially that the state of mind affects movement of the gazing point, the focus of the eyes. If a state of mind can be determined from gazing point information, we can use this correlation for usability evaluation. We can use gazing point information to find out about a user’s impression, or to directly evaluate effects in some cases [6,7].

Thus, there are various possibilities to use gaze-point information. But research using gaze-point information is seldom done in the field of Web usability evaluation. The usability of a Web site is so important that it can influence the amount of sales [9], because users are unwilling to read Web pages with low-usability, such as pages that are hard to operate or understand or pages reacting differently than expected. To create easy-to-use Web pages requires an evaluation of usability. Among various methods for usability evaluation, usability testing is widely used [4]. Specifically, the utterance analysis method, which has subjects speak about an impression during operation, has mainly been performed [13]. But subjects often do not speak well or fully. So an evaluator needs to ask questions which elicit comments from a subject in many cases [2,6,7,13]. An evaluator with special knowledge is required for such a conventional evaluation method.

In this paper, we examine “WebTracer”. WebTracer is an environment which supports analysis by quantitative data and various replay functions on the data. It can record a subject’s gaze point and operations used to browse a
website, and can replay them like video. The conventional recording environment of the gaze point has some problems: (1) the resolution is low; (2) the recorded amount of data is huge; (3) the data total for every Web page is not easy, and analysis is difficult. WebTracer solved these problems by including a delay capture system and browser ability. Also, there are few burdens to a subject since what they look at is automatically captured. The problem of the usability testing evaluation environment is eased since quantitative data are obtained.

Results of an experimental evaluation showed that the size of the operation history recorded by WebTracer was from 1/10 to 1/20 of the size of data recorded by an MPEG-2 or MPEG-4 format. Thus, with its compact form, the results of usability testing with the gazing point can be efficiently shared. It is expected that we can easily share empirical data between researchers. Also, evaluators can easily send the testing results as a feedback to the developers.

2. Related work

2.1. Web usability evaluation

Web application development is increasing quickly. The improvement in usability of Web application is becoming an important problem. However, usability evaluation of Web applications is not easy. Usability evaluation is categorized as either usability testing evaluated by operation of the Web application or usability inspection [4]. Usability inspection is a method where a subject does not operate the Web application. An evaluator points out problems on screens using a checklist, etc. Since usability inspection does not actually operate the application, it does not need a special environment. Moreover, it is easy to use to point out general usability problems. However, application-specific problems are more difficult to uncover. On the other hand, usability testing is a method of discovering a problem based on operation of the application by a subject. We have to prepare an execution environment, subjects, and tasks. Since a user actually uses the application, important problems are discovered. There is a thinking aloud protocol used as the method of general usability testing. The thinking aloud protocol allows an evaluator to analyze problems based on the user’s comments. To some extent, a user has to interrupt operations, when commenting. This approach has some problems. Evaluators are expected to have special knowledge. Subjects do not necessarily comment [2,13]. Also a method based on quantitative data is desired.

The Web usability evaluation based on quantitative data has been studied for many problems. Evaluation has been performed using specific operation information, such as statistical analysis of the layout structure in a Web page by the gazing point data on a screen, analysis of the operation events in the Web page by Java script, and analysis of the access log of a Web server.

Schroeder analyzed the relation between the layout structure and the number of times of reference in a page based on the gazing point on a screen [16]. Schroeder divided the Web page into an upside button bar, a upside search engine, left-hand side, a center, right-hand side, and the bottom. And he analyzed the part which a user observes many times. Gazing point data are used for analysis of the structure which can be classed in low resolution. But he was not analyzing the structure of a Web page in detail.

Paganelli analyzed the execution situation of a task from a user’s operation event recorded by Java script [11]. He is supporting the analysis based on quantitative data, such as page reference time and task execution time. He analyzed the usability of a Web page based on task execution time totaled for every Web page.

There are researches which evaluated the structure of a Web page using the access log of a Web server. The research using the evaluation environment called WET (Web Event-logging Tool) investigated the relation between a benchmark test and user satisfaction [12]. There is a limit of the event log recorded by a Web server. But it has been shown that it is effective in usability evaluation. It can be visualized as two-dimensional information by time series, the client host, etc. [3]. These are used in collecting and analyzing the data about Web page structure, and for evaluating the usability about Web structure.

These researches collect only a part of Web application operation information in a lot of user operation information, and are difficult to use in analyzing usability. To analyze usability from various angles and to develop better Web applications, various data must be unified and a recordable environment is required. Conventional environments have some problems. (1) The screen of high resolution is not recorded efficiently. (2) Web page information at the time of execution and User operation information cannot be unified, replayed and analyzed. (3) Evaluating the usability of Web pages based on quantitative data is difficult.

2.2. Movie recording mode

In the conventional collection analysis environment of an operation log, video is used for recording a computer screen and gazing points. There is a method of compounding the replay screen and the recorded operation log which does not use the picture at the time of execution. However, a replayed display may differ from the display of the timing which recorded the gazing point. Web application constitutes display data based on the window size and the data which were obtained from the network at the time of execution. For this reason, it is difficult to analyze an operation history without recording the picture at the time of execution. Then, after changing the computer screen into an NTSC system video signal, a gazing point picture is superimposed and it is recorded as a moving image in many cases. The CAESE (Computer Aided Empirical Software
Engineering) environment records gazing point data as a video picture [10]. And analysis of the operation log is supported by interlocking the recorded video picture with other operations data on a computer. Thus, in order to interlock the video picture of a gazing point and a screen with operations data, such as a keyboard and a mouse, a video picture must be able to be used on a computer. In statistical analysis of the software of character user interfaces like an editor and screen structure, practical analysis was possible by using the video picture which lowered and compressed resolution. However, when a computer uses the video picture of high resolution, the amount of record data is too large, and it is not so practical.

There are many systems for video compression. Generally video compression has structure as shown in Fig. 1 [15]. The Web page actually displayed is shown in Fig. 1(a). The image of the video compression at that time is shown in Fig. 1(b). The compressed video is constituted by the intra frame and the delta frame. The intra frame consists of only data in the frame which compressed a certain frame. A delta frame is constituted by the compressed data and the interpolated data between frames. Since the original picture is constituted, the last intra frame and subsequent delta frames are required. In edit work, random access earlier than real time is needed. It is required to reduce the influence of the data lack by the communication noise or recording-media error. For this reason, an intra frame with random access performance and error tolerance is required. For example, DV format used for recording of a home video consists of only intra frames in consideration of edit work. Moreover, MPEG-2 format used for broadcast and DVD is inserting per second one intra frame in order to lessen influence of errors [14]. MPEG-4 format used for communication in narrow band inserted one intra frame in 8 s, and has balanced the amount of data and error tolerance [8].

A Web application displays web pages one by one due to user operation. There is little change of the data currently displayed on the Web page. We think that the Web page is fit for compression between frames performed with a delta frame. The information displayed in the same Web page scrolls or a menu changes due to a mouse click or keyboard input. However, it is rare that many other changes are displayed per page. Flash and animation GIFs are examples of display changes within a Web page. In most Web pages where these are not used, the display changes only by mouse operation. If a Web page can be recorded on an intra frame and the change information in a Web page can be recorded as a delta frame, we think that efficient video compression is possible. However, when recording and analyzing an actual Web application screen, the conventional video compression technology has the following problems:

1. There is no environment which records a screen on high resolution. In the case of the video signal of an NTSC format, perpendicular oriented resolution of over 525 dots of screen ruling is not obtained.

2. The amount of data becomes huge. Web pages download the data displayed from a server through a network. From the display start to the completion of download at the time of a page change, many frames are required. In this period, since the change of the contents of a display is large, efficient data compression is difficult.

3. Since the operational information on a screen is analyzed earlier, an intra frame cannot be reduced. It is necessary to analyze in detail the area which the user is gazing at from the user operation compounded on the screen in the case of data analysis. In the conventional video compression format, to display the picture of a certain position on a screen for a short time, intra frames of a fixed ratio are required.

### 3. The delay capture system and WebTracer

In this section, we describe the delay capture system and WebTracer. The delay capture system is a new system for recording and replaying a high resolution screen in time.
series by fewer data using the features of a Web application display. WebTracer is an environment which supports recording, replay, and analysis using information integrated with the picture recorded by the delay capture system and other operation events.

3.1. The delay capture system

The delay capture system (see Fig. 2) captures the picture at the time of page changes. The picture is used as a screen of the Web page before changing. We use the picture captured just before changing to the following Web page because it has very little change. The download time of display data is a very big value as compared with the display speed of one frame of an animation (1/30 s). However, the time is not a problem for users due to the spread of broadband in recent years. It is not a big problem to compress the screen of each Web page to one picture except for changing information like a mouse or a keyboard in many cases.

It is impossible to detect certainly completion of display of a Web page. However, web pages display downloaded data one by one from a network. So, in the delay capture system, the screen is captured using the following procedure:

(1) Start the display of a Web page and record the time \( T \).
(2) Receive a user's operational event.
(3) When an event from which a screen display changes is received, capture the screen and save it as a picture from time \( T \) to the present.

Thus, it is not necessary to know completion of download by capturing at the moment a screen changes. Events from which a screen display changes are shown below:

- The mouse click for changing to the following Web page.
- The input of the line feed code or a tab code by the keyboard when determining the contents of a display.
- The mouse click and wheeling at the time of page scrolling.
- The mouse click at the time of a menu list display.

The captured picture differs from the picture under download of display data a little. However, the screen layout hardly changes and the interval of capture is short. Therefore, the captured picture can be used as a screen of the interval to capture. It is reproducible as an actual screen by laying a gaze point and mouse operation on top of the captured picture.

The delay capture system differs from the conventional method of using the intra frame and delta frame at regular time intervals. The system compounds and displays the screen of each page and other data. Therefore, it can record a high definition picture in a small amount of data. Construction of the environment unified with other data, such as an operational history, becomes easy.

3.2. WebTracer

WebTracer is an environment for recording and analyzing the user's operations in Web pages. It has the following three features:

(1) The screen of high resolution is recordable.
(2) Web-specific data such as change information on a Web page is recordable.
(3) Information required for analysis is unified and it can be replayed and analyzed.

WebTracer is a Windows application developed using Visual Basic. It is developed using the Web component for displaying the Web page attached to Internet Explorer, the A/D translation library made by Contec for inputting gazing point coordinates, the Handle component for
acquiring Windows events, and the JPEG save library for saving a picture in JPEG form (see Fig. 3).

WebTracer has the following three functions:

1. The record function of user operation.

   The screen data, gazing points, mouse operation, keyboard operation, and the display events of a Web page are recorded by the delay capture system during a Web page display. WebTracer captures a browsing screen using the delay capture system, and compresses and saves it using a JPEG library. Moreover, the picture file is recorded as data the first time it is used in time series. One line consists of a time stamp which consists of generating time, a sequence number, and a date, and a user operation data sequence. Each data is structured by the parenthesis and is recorded. JPEG file name, coordinates of the gazing point obtained from the A/D library, mouse events, keyboard events, the events of a Web component, and operations to WebTracer are recorded on the list in time series. WebTracer records the coordinates of gazing points every 0.1 seconds via eye tracking equipment. Fixations are eye movements which stabilize the retina over a stationary object of interest. Relevant statistical measures of fixations include their duration range of 150–600 ms, and the observation that 90% of viewing time is devoted to fixations [5]. Since the shortest time which detects fixation is 0.15 s, when analyzing a gaze position, it is enough in 0.1 s.

2. The replay function of user operation.

   The recorded data and the pictures of Web pages are integrated and displayed. As shown in Fig. 4(a), the user’s operations are replayed by displaying an icon at the coordinates of a mouse (arrow) and a gazing point (circle arrow) on the screen of a Web page. As shown in Fig. 4(b), gazing points can be shown as a circle on the picture of a Web page. Moreover, it is also possible to display the trails of a gazing point and a mouse as shown in Fig. 4(c). The mouse click and wheeling which were recorded display generation of an event by the icon on a screen. In a lower right replay control dialog box, replay control is possible. Also the events and keystroke data recorded at the time is displayed. By the replay control button, usual replay, high-speed replay, and reverse replay, high-speed reverse replay, stop, slow replay, and step replay is possible. Moreover, movement at arbitrary time is possible by using slide control. Thus, by various replay controllers, an evaluator can analyze the series of operations to observe in detail.
The analysis function of user operation.

The analysis function of user operation has a time-series-analysis function and a total data analysis function. A time-series-analysis function supports detailed analysis of user operation using a replay control dialog. Notes can be written in when a characteristic scene is discovered while replaying operation of a user using the various replay methods. And only the time at which notes are written in can be displayed on right order or the order of reverse. A total analysis function can indicate the total data of an operation history by list for every page. As shown in Fig. 5, the list of the reason for changes, the download time of data (a favorite menu, a link, key, etc.), browsing time, the distance of mouse movements, the number of scrolling, the number of clicks, the number of keystrokes, the distance of eye movements, and speed of eye movements (=the distance of eye movements/browsing time) can be displayed for every page. Moreover, these can be displayed also as graph. By using these, an evaluator can analyze how analysis of the features of each Web page and the Web pages were changed.

3.3. Comparison with the conventional research

Comparison between WebTracer and conventional research is shown in Table 1. Symbol × means unavailable analysis function and symbol ○ means available analysis function. WebTracer can be analyzed in detail using a time series analysis and total data analysis. It includes almost all functions except for change of the picture on a Web page. WET [12] and Hochheiser [3] are collecting page change histories. WebTracer can analyze a page change history similarly in time series. Hochheiser [3] can analyze the number of times of reference of the same page by analysis based on page structure. Paganelli [11] collects operation events and analyzes page reference time and browsing times. But, it has only a few kinds of events as compared with WebTracer. Gazing point information can be analyzed by Schroeder [16], MPEG-2 [14], and MPEG-4 [8]. Schroeder [16] totaled gazing point information like WebTracer, and uses it for analysis. Although he is not analyzing movement of a gazing point, the number of times of reference for every area is analyzed. MPEG-2 [14] and MPEG-4 [8] can analyze the operation to change of a screen and movement of a gazing point by recording the output of eye tracking equipment. As for these researches, the candidate for analysis differs from WebTracer.

However, WebTracer can record the gazing point information, page changes information, a user event, and a screen image currently studied, respectively. Moreover, it can integrate and analyze these information at the time of a user browsing. Moreover, it is also possible to support the analysis studied until now in the future. However, WebTracer makes the amount of record data small by not displaying the animation by screen change while loading.

![Fig. 5. Collected data analysis function.](image)
display data, Flash, and animated GIF. Therefore, WebTracer is not suited to analysis of the picture change on such Web pages. Because of the spread of broadband in recent years, the necessity of analyzing the screen display during loading is low. The structure on Web pages, such as which part on a Web page is observed, and analysis of changes between Web pages can be evaluated only by WebTracer. Thus, we think that WebTracer is effective in analysis of the whole Web page, and usability evaluation with the prototype from which a screen does not change.

The conventional analysis environment allows use of any browser when recording. However, WebTracer requires use of IE (Internet Explorer) component. In WebTracer, since there is no processor of an analog video signal, the picture of high resolution is recorded. Analysis by a time-series data list, and the random access and order replay by MPEG-2/4 form is possible for the conventional time series analysis. On the other hand, entry and various searches of notes are also possible for WebTracer. It is possible to analyze the feature of user operation in detail by these functions.

3.4. Evaluation experiment

In order to verify the effectiveness of WebTracer, we made the following evaluation experiment:

(1) Quality-of-image comparison with the picture of WebTracer, and the video output from a scan converter.

(2) Comparison of the amount of record data of WebTracer, and the MPEG-2 format and MPEG-4 format which are a general-purpose video compression system.

(3) Comparison of the random access performance of WebTracer and MPEG-2 format and MPEG-4 format.

(4) Study of the relation of a characteristic eye data and Web pages.

The outline of an evaluation experiment is shown below:

<table>
<thead>
<tr>
<th>Task</th>
<th>A subject looks for the home electric appliances which have a specific function by the Internet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task execution time</td>
<td>362 s</td>
</tr>
<tr>
<td>Screen size</td>
<td>1024 × 768 pixels</td>
</tr>
</tbody>
</table>

**The record method**

<table>
<thead>
<tr>
<th>WebTracer</th>
<th>Direct record</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPEB-2, MPEG-4</td>
<td>After recording on DV (digital video), it encodes after the experiment</td>
</tr>
</tbody>
</table>

| Record screen size | WebTracer 853 × 717 pixels (window size at the time of execution) |

The evaluation experiment performed WebTracer with the composition shown in Fig. 3. And the data of MPEG-2 format and MPEG-4 format was recorded on the display output through the scan converter. However, in order to optimize the compression parameter of MPEG-2 format and MPEG-4 format, while checking the output of a scan converter, it recorded in DV (Digital Video) format. Video data was compressed in MPEG-2 format and MPEG-4 format after the experiment. At that time, the compression parameter was set up so that noise might not be conspicuous and the amount of record data might become small.

When checking the video output under experiment visually, distinction of a small character was impossible. We think that it is because a video signal cannot obtain the vertical resolution of 525 pixels and over which is the number of scanning lines. There was some noise by JPEG compression on the replay screen of WebTracer. However, it was possible to have read all including a fine character. The result of an evaluation experiment of the amount of recorded data and random access is shown in Table 2. The amount of recorded data by WebTracer is the smallest compared with other format. It is about 1/20 of MPEG-2 format, about 1/3 of MPEG-4 format of 720 × 480 dots, about 1/3 of MPEG-4 format of 1024 × 768 dots. Moreover, the random access time of reproduction data is as quick as that of MPEG-2 format when the rate of data compression is low. When we consider that WebTracer has various functions, it is an environment which can support analysis work most effectively for an experiment.

In looking at time required for random access, MPEG-4 format with the setting ratio of a frame the lowest is the largest.

We consider time to decode arbitrary screens from an intra frame and a delta frame as mostly fixed. The result of random access is dependent on reproduction environment. Therefore, improvement in performance of hardware may improve this. However, the CPU load at the time of

<table>
<thead>
<tr>
<th>Screen resolution (pixels)</th>
<th>WebTracer 853 × 717</th>
<th>MPEG-2 720 × 480</th>
<th>MPEG-4 720 × 480</th>
<th>MPEG-4 1024 × 768</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record data size (MB)</td>
<td>14.0</td>
<td>272.0</td>
<td>48.5</td>
<td>140.1</td>
</tr>
<tr>
<td>Random access time (s)</td>
<td>1</td>
<td>1</td>
<td>1-2</td>
<td>5</td>
</tr>
</tbody>
</table>
WebTracer reproduction was almost the same as the MPEG-2 format in this experiment environment.

Moreover, movement and speed of a gazing point show the features of a Web page. Movement and speed of the gazing point in an experiment are shown in Fig. 6. The solid line shows the move distance of a gazing point, a dotted line shows the move speed of a gazing point, X-coordinate show a page number, and Y-coordinate show the relative value of each value in a figure. The 18th page with a quick speed of a gazing point had a menu in left-hand side and the center, and was a layout with an unclear selection item. The 2nd, the 9th, and the 10th page with a slow speed of a gazing point were the input screens of a search engine and error display screens with few display items. The 3rd, the 5th, and the 13th page with long movement of a gazing point were a menu page and a PDF page, and were a page with much amount of information. Thus, we think that the recorded quantitative data represent variable features of a Web page, and may be able to support usability evaluation.

3.5. Case study

We have conducted an experiment as a case study, to confirm the effectiveness of WebTracer in a Web usability evaluation. In the experiment, we asked the three subjects to find objective information within a company Website. Then, with the support of WebTracer, the subjects were interviewed and their comments were recorded.

Experimental setting
Display: 21 in.
(viewable screen size: H30 × W40 cm)
Resolution: 1024 × 768 pixels
Distance from subject’s face to display: approximately 50 cm
Device for measurement of sight line: NAC, EMR-NC
(view angle: 0.28, resolution on the screen: approximately 2.4 mm)
Recording and playing of sight-line data: WebTracer
(sampling rate: 10 times per second)
Subjects: 3 subjects who often use the WWW

First, we asked the three subjects to do tasks, and we recorded the operational data using the WebTracer. Tasks for the subjects included gathering the following five pieces of information from a company Website:

Task1: finding the way to a certain place in the company;
Task2: finding out the number of employees;
Task3: finding information about compensation benefits of the company;
Task4: finding specific news about the company;
Task5: finding a technical method of construction.

Second, we interviewed each subject about the ease of using an object Website after the end of a task. The interview is divided into four phases, and the outline of each phase is as follows:

Phase1: An interview based on statistics obtained from recorded data.
WebTracer shows the summarized browsing history. We interviewed each subject based on the summary and the eye movement graph as shown in Fig. 2. We had the subject point out the difficulty of use in focusing on a web page with both the large moving distance and moving speed of the subject’s gazing point.

Phase2: The interview based on the replay screen in which the eyemark was placed.
We had the subject point out the difficulty of use with the replay screen as shown in Fig. 3 regarding the web page from which the comment was obtained by phase 1.

Phase3: The interview based on the memory of the subject.
We conducted an interview based on the memory of the subject for whom the whole task was performed. Comments which were obtained and which overlapped by phase 2 are not included in the number of comments obtained by this phase.

Phase4: The interview based on the fast forward replay screen.
We had the subject point out the difficulty of use with the fast-forwarding replay screen. Comments which were obtained and which overlapped by phase 3 are not included in the number of comments obtained by this phase.
The average execution time per task in the experiment was 2 min and 48 s. Also, the average time of the interview was 19 min. Therefore, the total time spent for entire process was 21 min and 48 s.

The average number of comments given by the three subjects is summarized in Table 3. For execution time, we obtained 16 comments on average. The comments include usability problems, thinking during operation. We assume that the unit of a comment is every sentence uttered by a subject. The number of comments becomes material data of the usability problem for evaluator judges.

As shown in Table 3, about 95% of the entire comments are obtained from the graph and replay (Phases 1, 2 and 4) using the WebTracer. Moreover, about 74% of the entire comments are obtained from Phase 2 by replaying of WebTracer. We can see that the replay function of the user operation which added the eyemark in WebTracer is effective from the experiment results. Subjects can remember their operations, can point out the points, which is hard to use, and a evaluator can ask an interview more easily by replaying of WebTracer.

We believe that the WebTracer made the phases of this evaluation experiment possible. For example, we consider the case where “Thinking aloud protocol” carries out the interview phases. The “Thinking aloud protocol” can be applied to Phase 2 and Phase 4, if the evaluator records the subjects’ operations with a video camera. However, by replay of the video camera, we still cannot know a user’s gazing point. Therefore, we assume that the number of comments obtained from subjects decreases compared with the WebTracer. Phase 3 is also made possible by performing a questionnaire-based evaluation at the end of the task. However, “Thinking aloud protocol” cannot be applied to Phase 1, since no quantitative result is available. “Thinking aloud protocol” cannot record a detailed user’s operational data and show the summarized data. Consequently, 9.3 comments in Table 3 would not be available. As a result, the “Thinking aloud protocol” would miss 11.6% of the entire comments shown in Table 3.

4. Discussion

In this section, we consider the advantage of a delay capture system, and the possibility of Web usability evaluation using gazing point measurement.

First, we consider the advantage of a delay capture system from comparison with the conventional video compression system. MPEG-2 format and MPEG-4 format record an intra frame and a delta frame at regular time intervals. An intra frame is recorded in the JPEG format which is a compression system of a picture. The information compressed between frames is recorded as a delta frame. By the delay capture system, only when change is shown in a display is a screen recorded in JPEG format. Moreover, the event of a mouse and key operation, and the coordinates of the gazing point sampled at regular time intervals are recorded. A delta frame is binary data. The event information on WebTracer is text format to it. Therefore, it is not necessarily an efficient form. However, the conventional animation compression system must record data at regular time intervals. Except for the coordinates of a gazing point, only event-driven information are recorded, reducing the number of times to record to a necessary minimum in WebTracer. We think that the whole amount of data is decreased this way. The advantage of WebTracer can be maintained if a capture picture is saved in the format even if a more efficient animation compression system is developed.

Moreover, the picture recorded by WebTracer has high resolution in required size without using a specific video format. Therefore, the amount of data recorded can be made small and usability evaluation becomes easy using the picture of high resolution. When some features are looked at by user’s gaze point and operation, it is expected that many reasons are shown in a screen. Conventionally, an evaluator analyzes from a video picture with low resolution using the information shown on a screen. When it is difficult for an evaluator to analyze, an evaluator has to display the Web page again to check it. However, even if it is the same address, when the contents are displayed again, an evaluator cannot analyze it. Since WebTracer can save a screen by high resolution, an evaluator can check contents from the saved screen directly.

Also, when adding operation information and the analyzed comments of a user, you have to compound this with video data conventionally using an authoring tool. In WebTracer, recording and replay of various data, such as an event of a mouse or a keyboard and notes, also use the same text file. Therefore, extension of data is easy and various customizations are possible. For example, the screen is captured every time the data on a Web page downloads, and biological data such as the heart rate is recorded. Only the events of the mouse and keyboard, and screen data, are recorded in an environment without eye tracking equipment. As for WebTracer, the function has actually been extended in this way.
Next, we consider the possibility of Web usability evaluation using look measurement. Research of eye movement related to usability has been done in the field of psychology [1]. When man recognizes vision information, high-speed eye movements called saccade and fixation are performed repeatedly. During saccade, the input of information is very limited with little perception. If there is too much information which must be recognized, saccade will increase. And it becomes difficult to understand information. On a page with quick movement of a gazing point, there was much information as a result of an evaluation experiment. It is difficult to separate between saccade and other eye movements at the input interval of the gazing point of WebTracer. However, it is expected that the page with a quick move speed of a gazing point is a page with more saccade than fixation. We think that it is difficult for a subject to understand such a Web page.

The information used for evaluation this time is only the amount of eye movements in a page unit. From now on, analysis including other metrics will also be required. The metrics in a page unit cannot express change within a page. Therefore, precious information may be covered using other information. It is necessary to conduct an evaluation experiment and to analyze the metrics which shows change of the look within a page and a relation with other operation information. If the relation between eye information and a Web page becomes clear, we will think that it becomes possible to point out a usability problem using eye information.

Although the distance of the gazing point movement, the duration of stay of the gazing point, and the distance of the mouse movement are larger on average for the cases with low usability, the differences are not statistically significant. The reason why those values are larger for the cases with low usability is presumably because the subjects cannot make a quick decision what to do on those pages and those pages contain too much content. However, when a page contains too much content, the distance of the gazing point movement, the duration of the stay of the gazing point, and the distance of the mouse movement will be large without yielding an impression of low usability. This would be the reason why there was no significant difference for those values.

The speed of the gazing point for each Web page is correlated with the usability, but the speed of the mouse movement is not. When searching Web pages for some information, each subject keeps the gazing point moving. Some subjects put the mouse pointer to their gazing point, while other subjects put the mouse pointer outside of the Web page window. For this reason, no correlation was observed between the speed of mouse movement and usability.

WebTracer may be able to be used for analysis of the relation during pages and page changes. When evaluating a website, it is also important to consider the structure which the designer intended. It was conventionally evaluated by the log analysis of a Web server. If the information shown by the total analysis function of WebTracer is used, it may be able to support this more effectively.

5. Conclusion

In this paper, we proposed WebTracer: a record analysis environment of the Web page operation, for supporting usability evaluation. WebTracer uses a delay capture system which records the features of a Web display at the time of user operation. This allows the display of high resolution, reduction of the amount of recorded data, and the integrated display of a screen and operation data. Moreover, it is possible to collect all the Web browsing data even on pages that use a Web component. WebTracer can record the screen of high resolution in a smaller amount of data than the conventional video compression system as shown by an evaluation experiment. Moreover, we showed that the features of a Web page had appeared in the eye information.

A web application is the face of a company and sales can change a lot with better usability. Many reviews using usability guidelines are already carried out so that users may be satisfied. However, as compared with reviews, usability testing which can discover serious problems is not carried out. The conventional heuristics method requires subjects and much time. Moreover, usually there is no support environment based on quantitative data.

Recording and replay of the Web operations of a user, and the total of quantitative data per Web page, are possible for WebTracer. Web usability may be able to be efficiently evaluated by using the eye information per page. For example, it is possible to interview after recording data without interrupting operation of a user. Only the Web page which may contain a problem need be checked. Moreover, it may also become possible to evaluate usability without an interview based only on the Web operation data.

Acknowledgments

This work was supported by the Comprehensive Development of e-Society Foundation Software program of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) and Grant-in-Aid for Japan Society for the Promotion of Science (JSPS) Fellows (Research No. 16005035).

References