Self-Initiated Intermission: A Better Way of Handling Interrupting Events

Shreejana Prajapati, Koichi Yamada and Muneyuki Unehara

Graduate School of Engineering, Nagaoka University of Technology,
Nagaoka, Niigata, Japan
Email: s105087stn@nagaokaut.ac.jp, yamada@kjs.nagaokaut.ac.jp, unehara@kjs.nagaokaut.ac.jp

***Abstract*—The aspect of handling interruption plays a vital role in human computer interaction. Interruptions become disruptive when they occur at inappropriate times. This is the reason that researchers are paying more attention towards predicting interruptions and their effects over the past few decades. We came up with an idea of finding self-initiated intermission to realize fewer distraction in human-computer interaction. Self-initiated intermission is an initiation to report oneself as being available for interaction. It gives the privilege of choosing appropriate time to handle interruptions, so that it would not hamper the current active tasks. Along with self-initiated intermission, interruptibility levels at the time of application switching and at regular intervals are studied. Then, the interruptibility levels in these three conditions are compared to find the most appropriate moment for handling interrupting events. The study shows that self-initiated intermission is the best moment for interrupting user which is less annoying than at the time of application switching and at regular intervals.**

Keywords–self-initiated intermission; interruption; regular intervals; application switching.

I. Introduction

An Interruption, as defined by O'Conaill and Frohlich [1], is "a synchronous interaction which is not initiated by the recipient, is unscheduled, and results in discontinuing of recipient's current activity". It is a prominently occurring phenomenon in human computer interaction which drives users' attention away from their regular work. Whenever interruptions occur, they constitute disruptions, i.e., disrupt work flow or make compromises on users' productivity [1][2]. In general, there are two ways of giving information to people working on a computer, i.e., either through pull or push service.

Interruptions might occur as a result of positive or negative feelings [3] of task progress and goal attainment. In most of the cases, interruption forces users to switch their task. On the other hand, users spend very little continuous time on any single task [4][5] before they move to another task. Users keep on interrupting themselves in a certain time interval even if they do not get any external interruptions. A term self-interruption [3][4] has been introduced to discuss this internally motivated interruption. In-depth study of self-interruption [3] shows that self-interruption increases at the time of negative triggers such as frustration, exhaustion, and obstruction than at the time of positive triggers, whereas Jin and Dabbish [4] give the classification on self-interruptions not only as an internally motivated interruptions, but also as a factor of environmental causes.

Regardless of numerous research on interruptions, none of the existing literature has focused on finding the effect of interruption for a short break or intermission. Though they focused on understanding and managing interruptions using external devices, interruption at self-initiated intermission has not been given enough attention. The self-initiated intermission is different from self-interruption mentioned above. Our study seeks to provide interruption management through self-initiated intermission. In this intermission, users can start communicating with a computer without hampering their current work. It gives users the privilege of handling interrupting events at their preferred time. In addition to self-initiated intermission, users' interruptibility levels at application switching [6][7] and at regular intervals [6][8] are studied for comparison.

In human-human communication, people wait for an opportune moment to start communication, but in the case of human computer interaction, such as, in Instant Messenger (IM), communication takes place instantaneously without concerning about the preferable time. In order to manage interrupting events with minimal negative impact, we propose the self-initiated intermission to interrupt users. In this paper, users' interruptibility levels at three different conditions, viz. at the time of application switching, at regular intervals, and at self-initiated intermission are compared to find the best among them. A sequence of experiments are set up where users need to perform two predefined tasks. While users are performing the tasks, they get interrupting messages in the middle of their work, depending on the conditions of interruptions. From the users' response to interruptions, it is found that users' interruptibility level changes depending on the conditions of interruptions. The results of the experiments indicate that users' interruptibility level is higher at the time of self-initiated intermission than at the time of application switching and at the regular intervals.

The rest of the paper is organized as follows. Section II presents the discussion of related work. Section III describes the idea of self-initiated intermission in detail, and shows that self-initiated intermission is different from self-interruption. Section IV, then, proceeds with presentation of experimental methods, the task being assigned, and the procedure carried out for the experiments. Section V demonstrates the results of the experiments and the statistical test. Section VI describe the discussion based on experimental results. Finally, Section VII presents our conclusion and future work.

II. Related Work

Studies on interruptions were started by examining their effects in different aspects. One aspect is based on finding proper moments for the interruption. Czerwinski et al. [9] anticipates the moment of interruption at the beginning, at

 ISBN: 978-1-61208-356-8

the middle, and at the end of the task. Likewise, Miyata and Norman [10] predict that interruption after an important action, or between task execution and evaluation would be less harmful than interruption occurring at other times. But, Gillie and Broadbent [2] discovered that being able to detect one's position in the main task does not protect one from the disruptive effects of an interruption.

Some studies used machine learning techniques [11]–[13] to create predictive statistical models for interruption prediction. CRISP [14], an interruption management algorithm used machine learning algorithm to automatically model users' preference for interruption. It used a rule based algorithm to identify breakpoints, and k-nearest neighbor to find people who behave in a similar way in order to identify the situations that users might encounter while working. Based on the finding of these two algorithms, it decides appropriate time for interruption. Some focused on sensor based interruptibility prediction [8][15][16]. Both the sensor based and machine learning approaches were only concerned with the physically observable interruptible states and not with the intellectually interruptible conditions such as thinking.

Some past work focused on handling interruption that appears in terms of notification [17]–[21]. A haptic notification system [19] is designed for time management during oral presentation, which provides time alert to the presenter by generating different vibration clues. Muller et al. [20] focused on displaying information by employing an ambient timer using LED light along with vibration [21]. It gives gradually increasing cue through light, to inform the subjects that they are running out of the allocated time. Since it uses light as an alternative way for representing notification or pop-up messages, it becomes visible to the co-worker in group work. Sometimes it leads to privacy issues.

Another notification management approach is to schedule notifications at break points [17][18]. It reduces frustration and reaction time as compared to delivering them immediately. Relevance of notification content determines the granularity of breakpoint at which it should be delivered [18]. The core concept of scheduling notifications at breakpoints fitted well with how users prefer notifications to be managed [17]. Breakpoints along with the application specific knowledge are used in sensing appropriate time for notification [22]. It manipulates the running applications based on the granularities and available application specific knowledge. Then, it extracts the information, such as, expected breakpoints, forecasting the incoming breakpoints, and the target application that is most likely to get attention. There are three existing granularities of breakpoints in users' tasks, namely: coarse, medium, and fine [7][23]. And users' interruptibility level increases depending on the granularities of breakpoints [18][24]. However, the model with the breakpoints struggles to differentiate the granularities of breakpoints, and is only useful for detecting them without differentiating the granularities [18]. In spite of this problem, the breakpoints are considered to be more acceptable for delivering notification than during continuous work [6]. Tanaka and Fujita [25] propose a secretary agent to mediation interaction between users and others based on interruptibility estimation at breakpoints [6] along with the concept of avatar. The avatar continuously face towards user's computer screen and appeals for interaction by turning its face to user when breakpoint occurs, and time taken for interaction initiation is calculated [26].

III. Approach of Self-Initiated Intermission

In this section, an idea of self-initiated intermission is proposed to find an appropriate time for interruption. Self-initiated intermission is a way of managing interrupting events at self-initiated time. In this intermission, users deal with interrupting events by taking a short break from their ongoing work. Users are supposed to provide information on their leisure instead of managing the events for that time period. In other words, it is a conscious deviation from the current work. The focus here is to find a proper moment of delivering interrupting events that come in terms of the notification. Especially, we target our study with a long term goal of creating a system that handles notifications in IM and present them at the time of self-initiated intermission. Possible alternative ways of handling notifications or interruptions are: at the time of application switching, at regular intervals, at phase transition, e.g., from execution to evaluation phase and at the timing of the detail task coupling such as copy-paste.

Application switching is users' intentional switching of their working space [6], and in the case of continuous work, users keep interrupting themselves in a certain time interval because of intentionally motivated interruption. Therefore, an interruptibility comparison between the three ways of interruption management, namely: self-initiated intermission, application switching, and regular intervals is made. In our approach of finding the best moment of interruption, unlike detecting switching based on the operating history of the computer, and regardless of task being performed [6], users are provided with pre-defined task and are asked to take an initiative to report their intermission.

Intermission is the time when users' interruptibility becomes higher, which means interruptions are acceptable without any distraction at that time. A hypothesis is generated to study the conditions in which users' interruptibility becomes higher among the three ways of interruption management.

Hypothesis: *Users' interruptibility becomes higher at the time of self-initiated intermission than at the time of regular intervals and at application switching.*

IV. Experimental Method

Three experiments are designed to study users' availability for interruption. They are conducted to examine users' interruptibility at the time of 1) Application Switching (hereafter referred to as AS interruption), 2) Regular Intervals (RI interruption), and Self-Initiated Intermission (SI intermission). Users are interrupted at the time of every application switching, at every regular intervals, or on their intermission time in each experiment and are asked to report on their interruptibility levels.

A. Subjects

Twelve subjects participated in all three experiments. The subjects are familiar working with a computer and, are similar in working environment. Among them, three are females (25%) and nine are males. Their age ranges from 20-35 years. The

 ISBN: 978-1-61208-356-8

average age is 28.1 with a standard deviation of 3.3. All of the subjects are from technical field whose academic qualification ranges from undergraduate to post-graduate. One subject is from undergraduate program, 6 are from graduate program and 5 are from post-graduate program. Each subject needs to take all three experiments.

B. Task

The subjects are provided with two pre-defined tasks in each experiment. The first task is summarizing a document and the other task is solving a crossword puzzle. In document summarization, the subjects are provided with 5-6 pages of documents with approximately 2600 words. The documents is extracted from novels 'Luminance', 'One Night @ Call Center' and a book 'An Essay Concerning Human Understanding'. They are asked to summarize the given document within a page. Along with the summarization, they need to answer two questions related to the document. One of the question is simple while another is a tricky one for which the subjects need to go through the whole document. In the case of solving the crossword puzzle, some hints are given such as an initial letter, final letter and synonyms. They are asked to solve the crossword puzzle on their own. If they need help they can go through on-line dictionaries to search for words or information related to the given hints. In the experiments, document summarization force the subjects to concentrate on their task as they need to answer the questions and summarize the document. In the case of solving crossword puzzle, they need to think possible words in order to solve the puzzle, which can be thought of as a thinking task in the real environment.

Three sets of document summarization and crossword puzzle task are prepared with different content and assign one of them to the subjects randomly so that each subject would use a different set for the three experiments.

C. Procedure

Before starting, the subjects are briefed about the tasks they need to perform during the experiments. They are asked to perform the given tasks on their own pace. There is no time limitation for completing them because if they are given a time limit, they will race towards completing the tasks within the allocated time instead of concentrating on them.

The subjects' activities, both desktop and physical, are monitored constantly through a program and via video data throughout the experiment. They are informed about this prior to the experiment. The information monitored are key press events, mouse events, the current active application, and the switches between applications. When the subjects start the experiments, this program keeps running in the background and records the above mentioned information. The program generates interruption messages at the time of application switching, or at regular time intervals. The interruption messages ask the subjects to give their interruptibility levels. The interruptibility levels here indicate degrees of being available for interruption. In other words, this gives information about how much they are distracted when those messages appear. In our study, there are four levels of interruptibility, they are: Level 1 (Not at all), Level 2 (Interruptible), Level 3 (Un-interruptible), and Level 4 (Highly Un-interruptible).

During the experiments, in both AS interruption and RI interruption, if the subjects do not give their interruptibility level for thirty seconds then the interruption message disappears. The program considers the subjects' interruptibility level as "Highly Un-Interruptible". It assumes that they were busy with their work and could not provide their interruptibility level. If they "Cancel" or close the message instead of giving interruptibility level, it again considers the interruptibility level as "Highly Un-Interruptible". In the experiment for RI interruption, 12 minutes is considered as the time interval for the interruption, since subjects spend only about 12 minutes on a particular task before switching to the next task [4][5].

In the experiment for SI intermission, when the subjects start with an experiment, a notification icon will appear in the task bar. At the initial stage, the notification icon will appear in red color indicating "Un-interruptible" state, i.e., unavailable for interruption. It is made to represent "Un-interruptible" state at the beginning, so that, once the subjects turn into "interruptible state" they can change the notification icon into green by clicking it. It remains green for two minutes, and the program asks the subjects whether they are still in "Intermission". If the answer is yes, the icon remains green for another two minutes. Otherwise, the color goes back to red.

Similar to AS interruption and RI interruption, in SI intermission, if the subjects do not respond to the message for thirty seconds then it disappears. It considers the subjects' interruptibility level as "Un-Interruptible" state. In the case of SI intermission, the subjects' interruptibility levels are binary, that is, either in "Interruptible" or in "Un-interruptible" state.

Order of the three experiments as well as the tasks to be performed in the experiments are chosen randomly for every subject.

V. Results

The three sequential experiments are performed to study subjects' responses on interruptions in three different conditions. Based on the obtained data, subjects' interruptibility levels are analyzed in terms of being available and not being available for interruption.

A. User's Response on Interruptibility

The results of AS interruption, RI interruption, and SI intermission are shown in Tables I-II, III-IV, and V-VI, respectively. Tables I, III, and V show the minimum, maximum, and average time spent by the subjects for completing the assigned tasks in AS, RI, and SI interruptions/intermission along with the standard deviation. Tables II and IV show the minimum, maximum, and average frequencies of interruptions and interruptibility levels, which are caused by application switching and regular intervals, respectively. Table VI shows the minimum, maximum, and average frequencies of intermission in SI intermission. The interruptibility levels in SI intermission are binary as noted before.

The frequency of application switching is higher during crossword puzzle than during document summarization because of it's complexity. Most of the subjects find difficult to solve it on their own. So, in order to find the information

 ISBN: 978-1-61208-356-8

TABLE I: Time taken to complete the tasks in the experiment of AS interruption (in minutes)

Task	Min	Max	Average	Standard Deviation
Total Task	85	176	117.9	26.9
Document Summarization	32	94	61.3	16.9
Crossword Puzzle	22	118	56.4	26.1

TABLE II: Frequency of Interruption and Interruptibility Levels in the experiment of AS interruption

Frequency	Min	Max	Average	Standard Deviation
Highly Un-interruptible	21	252	97.0	67.5
Un-interruptible	0	38	11.6	13.3
Interruptible	0	8	3.9	2.8
Not At All	0	19	4.6	5.6
Interruption	32	283	117.1	77.8

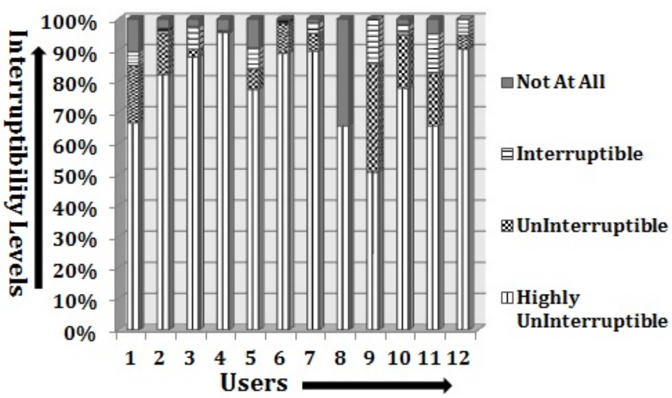

Figure 1. Subjects' responses on their Interruptibility levels in the experiment of AS interruption

TABLE III: Time taken to complete the tasks in the experiment of RI interruption (in minutes)

Task	Min	Max	Average	Standard Deviation
Total Task	52	144	97.9	29.7
Document Summarization	28	85	56.2	17.4
Crossword Puzzle	23	74	41.7	17.2

TABLE IV: Frequency Of interruption and interruptibility levels in the experiment of RI interruption

Frequency	Min	Max	Average	Standard Deviation
Highly Un-interruptible	2	10	6.5	3.0
Un-interruptible	0	4	1.8	1.1
Interruptible	0	6	2.0	2.2
Not At All	0	3	1.3	1.5
Interruption	4	12	8.4	2.8

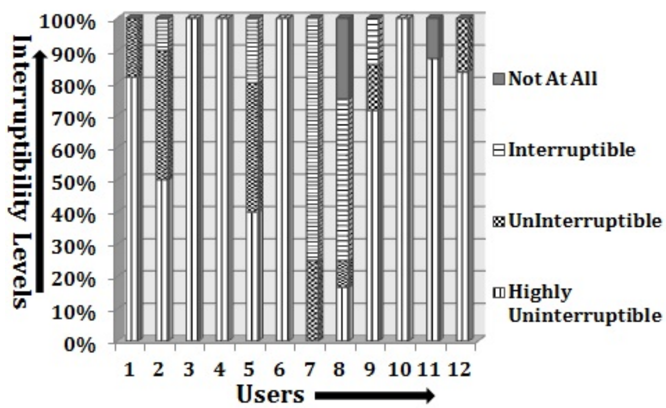

Figure 2. Subjects' responses on their Interruptibility levels in the experiment of RI interruption.

related to given hints, they search the Internet for which they need to switch the applications. In the case of document summarization, they switch the applications when they look at the original document to check for the keywords while writing the summary.

In percentage wise interruptibility level, as shown in Figure 1, the subjects considered themselves being in "Highly Un-interruptible" state for almost 80% of the time, when they get interruption messages during AS interruption. Around 5 to 15% of time they considered themselves to be in "Un-interruptible" state. They considered themselves either in "Interruptible" or "Not at all" state for remaining 5% of the time. This shows that there are very few chances for the subjects to be available for interaction at the time of application switching. Every application switching does not always indicate subjects are free for interaction or available for interruption.

In Figure 2, most of the subjects find themselves either in "Highly Un-interruptible" or "Un-interruptible" state whenever they get interruption messages. In this experiment, out of twelve subjects, four report themselves as "Highly Un-interruptible" all of the time throughout the experiment, six report either "Highly Un-interruptible" or "Un-interruptible" for almost 90% of the time. The rest of two report either "Interruptible" or "Not at all" for almost 80% of the time.

In the case of SI intermission, the time taken for completing the assigned task and the frequency of self-initiated intermission are shown in Tables V and VI, respectively. The subjects report on their intermission maximum 12 times and the minimum 4 times (Figure 3). In self-initiated intermission, whenever subjects take intermission, their interruptibility level is "Interruptible". In other words, reporting on intermission is nothing but reporting about their interruptible state.

Results of these three experiments demonstrate that interruptibility levels change based on the conditions of interruptions as shown by the graphical data in Figures 1, 2, and 3. Also, it is found that subjects take initiations to report on their intermission at least 4 times within the task completion period.

 ISBN: 978-1-61208-356-8

TABLE V: Time taken to complete the tasks in the experiment of SI Intermission (in minutes)

Task	Min	Max	Average	Standard Deviation
Total Task	74	174	115.7	26.9
Document Summarization	58	91	66.7	9.8
Crossword Puzzle	11	83	49.1	20.4

TABLE VI: Frequency Of intermission in the experiment of SI Intermission

Frequency	Min	Max	Average	Standard Deviation
Interruptible	4	12	7.3	2.5
Un-Interruptible	0	0	0.0	0.0
Intermission	4	12	7.3	2.5

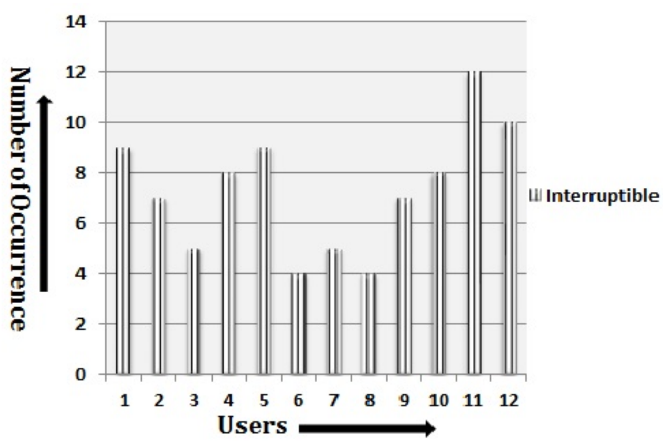

Figure 3. Subjects' responses on their Interruptibility levels in SI Intermission and also shows the frequency of intermission.

B. Evaluation based on Interruptible state

In the experiment of SI intermission, there are two levels of interruptibility, i.e., either "Interruptible" or "Un-interruptible". When the subjects report on their intermission, they are always in "Interruptible" state. On the other hand, when they are not in intermission they always seem to be in "Un-interruptible" state. Here in this section, the subjects' interruptibility level is analyzed statistically.

In order to compare the three ways of interruptions, viz. AS interruption, RI interruption, and SI intermission, repeated paired wise t-test is performed. The available four levels of interruptibility in the experiments of AS and RI interruption is transformed into binary to compare with the results of SI intermission experiment. The two interruptibility levels, level 4 (Highly Un-interruptible) and 3 (Un-interruptible) are transformed to Un-interruptible. The next two interruptibility levels, level 2 (Interruptible) and 1 (Not at all) are transformed into Interruptible. The value '0' is assigned for Interruptible and '1' for Un-interruptible, respectively. Then, the data of 12 subjects is mixed into a data set. Three data sets of the 12 subjects are obtained with 1405, 101, and 88 data, resulted from AS interruption, RI interruption, and SI intermission, respectively (shown in Table VII).

TABLE VII: Data sets from three different experiments

Experiments	Data	Average	Standard Deviation
AS Interruption	1405	0.93	0.26
RI Interruption	101	0.84	0.37
SI Intermission	88	0.00	0.00

The average values suggest that SI intermission is the most interruptible, whereas RI interruption is the 2nd most interruptible and AS is the 3rd. Since there is unequal variance, Welch's t-test is performed for every pair of the data sets. The results of one-sided t-tests also show that SI Intermission is more interruptible than AS and RI interruptions with significance level of 1%. The p-value is almost zero between the results of SI intermission and AS interruption, and 4.23×10^{-42} between the results of SI intermission and RI interruption. This supports the Hypothesis in Section III. As for the comparison between AS interruption and RI interruption, significant difference at 5% can be seen with p-value of the one-sided test being 0.0114, which concludes RI interruption is more interruptible than AS interruption.

VI. Discussion

In this study, the subjects' responses on their interruptibility levels are explored to find the best condition of interruption. From the experimental data and the statistical test, it is verified that SI intermission is the best moment for interruption management. In the interruptibility comparison, it is found that degree of availability for interruption is higher at AS interruption, at RI interruption, and at SI intermission in increasing order. This result contradicts with the suggestion from past study [6], which says application switching is a good approach of getting subjects' attention. In their study, regardless of the nature of the task they only considered computer operation records to check subjects' interruptibility during application switching. This might be one reason for getting application switching acceptable for the interruption. Whereas, in our case, the task being assigned require more concentration. They made an interruptibility comparison at the time of application switching and at the middle of continuous work. The interval of interruption at continuous work was very short (5 minutes), that might be another reason for getting application switching more acceptable for interruption than continuous work (regular intervals). In our study, the time of interruption for RI interruption is increased, which was 12 minutes. As studies [4][5] suggested that even if people do not get external interruptions, they spend only about 12 minutes of time before moving to another task because of the self-interruption. This was the reason to increase the time of interruption in RI to 12 minutes. Tanaka and Fujita [6] also generate interruptibility estimation rules based on the operating history of the computer. They made the estimation that physical activity such as keystroke and mouse operation reflect 'Interruptible state' and close coupling of tasks such as copy and paste reflect 'Un-interruptible state' but the accuracy was only around 50%.

 ISBN: 978-1-61208-356-8

There was a post questionnaire session to examine subjects' reactions towards the appearance of interruption messages at different conditions of interruptions. After the experiments, the subjects described that it was annoying to get messages at every application switching and at regular intervals. Similar to the fact mentioned in past studies [27]–[29], out of twelve subjects, ten reported interruption messages as annoying and distracting while two expressed reporting on their intermission was annoying. This demonstrates that when subjects are fully concentrated on their assigned task, they prefer pull service and when they are doing their work just for the sake of completing, they prefer push service, i.e., interruption.

Subjects found it more convenient when they could handle the incoming interruption messages at their preferable time. Among the four methods of coordinating interruptions [30]: Immediate, Mediated, Scheduled, and Negotiated, the approach of handling interrupting events at SI intermission supports the negotiated method of coordinating interruptions in human computer interaction.

VII. Conclusion and future work

In this paper, a new approach of self-initiated intermission was presented in order to handle the interruption related events or notification at the appropriate time without distracting users from their current work. Self-initiated intermission is a conscious diversion from current work. Unlike other interruptions, it leaves very less negative impact. Through the experimental data, it is proved that self-initiated intermission is the best way to handle interrupting events without creating annoyance and anxiety to users. This study is limited to the verification of our anticipation on how well self-initiated intermission works. We plan our future work on implementing self-initiated intermission for minimizing interruptions by presenting notification in an appropriate moment in an instant messaging system. We also plan to develop a system with self-initiated intermission to get users' attentions for interaction.

References

[1] B. O' Conaill and D. Frohlich, "Timespace in the Workplace: Dealing with Interruptions," in Proc. CHI 95 Conference Companion on Human Factors in Computing Systems, 1995, pp. 262–263, ISBN: 0-89791-755-3.

[2] T. Gillie and D. Broadbent, "What Makes Interruptions Disruptive? A Study of Length, Similarity, and Complexity," Psychological Research, 1989, pp. 243–250, ISSN: 0340-0727, 1430-2772.

[3] R. F. Adler and R. Benbunan-Fich, "Self-interruption in Discretionary Multitasking," Computer in Human Behavior, vol. 29, 2013, pp. 1441–1449.

[4] J. Jin and L. A. Dabbish, "Self-interruption on the Computer: A Typology of Discretionary Task Interleaving," in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2009, pp. 1799–1808, ISBN: 978-1-60558-246-7.

[5] V. M. Gonzlez and G. Mark, "Constant, Constant, Multitasking Craziness: Managing Multiple Working Spheres," in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2004, pp. 113–120, ISBN: 1-58113-702-8.

[6] T. Tanaka and K. Fujita, "Study of User Interruptibility Estimation Based on Focused Application Switching," in Proc. of the ACM conference on computer, 2011, pp. 721–724, ISBN: 978-1-4503-0556-3.

[7] P. D. Adamczyk and B. P. Bailey, "If Not Now When? The Effect of Interruptions of Different Moments Within Task Execution," in Proc. of the SIGCHI conference on Human factors in computing systems, 2004, pp. 271–278, ISBN: 1-58113-702-8.

[8] M. Nilsson, M. Drugge, U. Liljedahl, K. Synnes, and P. Parnes, "A Study on Users'Preference on Interruption When Using Wearable Computers and Head Mounted Displays," in In Communications IEEE PerCom'05, 2005, pp. 149–158, ISBN:0-7695-2299-8, doi:10.1109/PERCOM.2005.5.

[9] M. Czerwinski, E. Cutrell, and E. Horvitz, "Instant Messaging and Interruption:Effects of Relevance and Timing," in People and Computers XIV: Proceedings of HCI 2000, British Computer Society, 2, 2000, pp. 71–76.

[10] Y. Miyata and D. A. Norman, "Psychological Issues in Support of Multiple Activities," In: D. A. Norman and S. W. Draper (Eds.), User Centered Systems Design: New Perspectives on Human-Computer Interaction, Hillsdale: Lawrence Erlbaum Associatesn, 1986, pp. 265–284.

[11] J. Fogarty, S. E. Hudson, and J. Lai, "Examining the Robustness of Sensor-Based Statistical Models of Human Interruptibility," in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2004, pp. 207–214, ISBN:1-58113-702-8.

[12] S. E. Hudson, J. Fogarty, C. G. Atkeson, D. Avrahami, J. Forlizzi, S. Kiesler, J. C. Lee, and J. Yang, "Predicting Human Interruptibility with Sensors: A Wizard of Oz Feasibility Study," in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2003, pp. 257–264, ISBN:1-58113-630-7.

[13] E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and K. Rommelset, "The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users," in Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, 1998, pp. 256–265, ISBN:1-55860-555-X.

[14] T. Shrot, A. Rosenfeld, and J. Golbeck, "CRISP: An Interruption Management Algorithm Based on Collaborative Filtering," in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2014, pp. 3035–3044, ISBN: 978-1-4503-2473-1, doi:10.1145/2556288.2557109.

[15] P. Vorburger, A. Bernstein, and A. Zurfluh, "Interruptability Prediction Using Motion Detection," 2011, international Workshop on Managing Context Information in Mobile and Pervasive Environments MCMP-05, doi:10.1.1.87.380.

[16] M. Haller, C. Richter, P. Brandl, S. Gross, G. Schossleitner, A. Schrempf, H. Nii, M. Sugimoto, and M. Inami, "Finding the Right Way for Interrupting People Improving their Sitting Posture," in Human-Computer Interaction-INTERACT, 2011, pp. 1–17.

[17] S. Iqbql and B. P. Bailey, "Oasis: A Framework for Linking Notification Delivery to the Perceptual Structure of Goal Directed Tasks," ACM Transactions on Computer-Human Interaction, vol. 17, 2010, doi:10.1145/1879831.1879833.

[18] S. Iqbal and B. P. Bailey, "Effects of intelligent notification management on users and their tasks," in Proceedings of the ACM Conference on Human Factors in Computing Systems, 2008, pp. 93–102, ISBN: 978-1-60558-011-1.

[19] D. Tam, K. MacLean, J. McGrenere, and K. Kuchenbecker, "The Design and Field Observation of a Haptic Notification System for Timing Awareness During Oral Presentation," in Proc. CHI'13 ACM, 2013, pp. 1689–1698, ISBN: 978-1-4503-1899-0.

[20] H. Muller, M. Kazakova, A. Pielot, W. Heuten, and S. Boll, "Ambient Timer Unobtrusively Reminding Users of Upcoming Tasks with Ambient Light," in Pro. of Human Computer Interaction- INTERACT, 2013, pp. 2111–228.

[21] H. Muller, M. Pielot, and R. Oliveira, "Towards Ambient Notification," 2013, in Pheripheral Interaction: Embedding HCI into Everyday Life-Workshop at INTERACT.

[22] T. Okoshi, H. Tokuda, and J. Nakazawa, "Application as a sensor (AaaS) Approach for User Attention Sensing," in Proc. of The 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, 2013, URL: http://www.ubicomp.org/ubicomp2013/dc/okoshi-jr-ds-crc.pdf.

[23] S. Iqbql and B. P. Bailey, "Understanding and Developing Models for Detecting and Differentiating Breakpoints During Interactive Tasks," in Proc. CHI 2007, 2007, pp. 697–706, ISBN:978-1-59593-593-9.

[24] S. T. Iqbal and B. P. Bailey, "Leveraging Characteristics of Task Structure to Predict the Cost of Interruption," in Proceedings of the

SIGCHI Conference on Human Factors in Computing Systems, 2006, pp. 741–750, ISBN:1-59593-372-7.

[25] T. Tanaka and K. Fujita, “Interaction Mediate Agent Based on User Interruptibility Estimation,” Human Interface and the Management of Information. Interacting with Information, vol. 6771, 2011, pp. 152–160.

[26] K. Fujita and T. Tanaka, “Secretary Agent for Mediating Interaction Initiation,” in Proc. of Human Agent Interaction, 2013, URL: http://hai-conference.net/ihai2013/proceedings/pdf/II-2-p5.pdf.

[27] D. C. McFarlane and K. A. Latorella, “The Scope and Importance of Human Interruption in Human Computer Interaction Design,” Human Computer Interaction, vol. 17, 2002, pp. 1–61.

[28] D. C. McFarlane, “Interruption of People in Human-Computer Interaction: A General Unifying Definition of Human Interruption and Taxonomy,” 1997, naval Research Laboratory (NRL/FR/5510979870),Washington,DC.

[29] G. Mark, V. M. Gonzalez, and J. Harris, “No Task Left Behind, Examining the Nature of Fragmented Work,” in Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 2005, pp. 321–330, ISBN:1-58113-998-5.

[30] D. C. McFarlane, “Comparison of Four Primary Methods for Coordinating the Interruption of People in Human-computer Interaction,” Human Computer Interaction, vol. 17, 2002, pp. 63–139.