Gaze cursor during distant collaborative programming: a preliminary analysis

Roman Bednarik [1,2] and Andrey Shipilov [1]
[1] School of Computing, University of Eastern Finland, Finland [2] School of Information Sciences, University of Pittsburgh, USA
roman.bednarik@uef.fi, ashipilo@cs.joensuu.fi

Abstract. The geographical distribution of working force creates a challenge for effective team collaboration. When face to face contact is missing, the quality of communication decreases. We study the ways of improving synchronous collaboration over distance by displaying the gaze locations of the collaborators in a shared workspace. In an experiment, we evaluated the effects of seeing the gaze location of an expert programmer when describing two algorithms to a group of novice programmers. Here we focus on an analysis of the global fixational measures. Our preliminary results show that seeing the gaze while comprehending had no effects on the outcome of collaboration. However, seeing the gaze cursor of the expert resulted in different behaviors than when it was not available.

Introduction

Eye-contact is a natural experience in face-to-face communication (Argyle and Cook, 1976). Indicated by the direction of the eyes, gaze and overt visual attention play a significant role in understanding others and their actions, in face-to-face interaction, communication and collaboration. Visual attention is indeed an important skill in many everyday tasks and joint attention is in fact critical in many social interaction processes, such as language learning, parent-child shared reading, collaborative problem solving, visual search, and referential grounding (Liu et al., 2011).

In collaboration, the communication channels are inherently multimodal, intricately intertwined and need to be carefully orchestrated (Flor, 2006). The range of the channels that are employed in collaboration depends on the context of the task and availability for communication of interactive processes; for example, synchronous face-to-face tasks completion benefits from a full range of modalities, when distant collaboration without shared visual space restricts the effectivity (Fussell et al., 2000).

When some of the natural ways to communicate are not available, the quality of collaboration is endangered. Such situation frequently occurs in geographically distributed collaborative environments where information about partners' gaze, as a proxy to their overt visual attention, is missing. Since the number of geographically distributed tasks and teams is growing rapidly, it is important to evaluate the effects of the lack of eye-contact and gaze direction information on collaboration and to investigate the ways of improving the collaboration when eye-contact and the direction of one's attention are not available.

In this work we focus on the situation where others direction of eye-gaze is not available for realtime collaboration over shared desktops. These typically include collaborative document editing, visual design, and other problem-solving activities. One of such many domains is also software development that nowadays happens in multinational distributed development environments (Herbsleb and Mockus, 2003).

Programmers need to communicate their design decisions, debugging strategies, requirement analysis, user interface design and other tasks that require coordination of shared information. Collaborative programming paradigms such as pair-programming, require participants to communicate about their problem-solving activities. Gaze and mutual gaze play an important role in these activities , and the lack of mutual gaze can eventually materialize as major problems in communication. We begin to uncover the effects of mutual gaze on collaboration by examining the effects of gaze display on the visual attention patterns.

Gaze in collaboration, programming, and collaborative programming

The spatial proximity or a lack of it has direct implications on many aspects of collaboration process and its outcomes, such as it results in coordination delay (Cummings et al., 2009) and lower teamwork quality (Hoegl and Proserpio, 2004). In order to have efficient collaboration, collaborators need to share common ground (Clark and Brennan, 1991). Not only the content and knowledge have to be coordinated and grounded, but also the process and the dynamics of it have to be grounded. All these activities require accurate and timely coordination of the resources for effective communication between the peers (Clark and Brennan, 1991).

In face-to-face collaboration, joint attention is used to help grounding. To successfully and effectively communicate, and to establish and maintain the common ground, collaborators need to be able to signal and infer what their partners are attending to (Kraut et al., 2003; Horvitz et al., 2003). Eye-gaze is important in coordinating turn-taking, information flow (Jokinen et al., 2009), and conversational feedback (Goodwin, 2000).

In distant collaboration, such as video conferencing, the lack of gaze and consequent difficulties to the grounding processes have been shown to have negative implications on the quality of collaboration (Chen, 2002; Grayson and Monk, 2003).

The increasing number of global software projects pose new challenges for communication and collaboration, affecting, for example, performance and development speed (Herbsleb and Mockus, 2003) and conflict (Hinds and Bailey, 2003). In this paper we aim to evaluate the role of real time gaze display in expert-novice distance collaborative learning situations. In these situations, establishing and maintaining the common ground is required for successful communication (Clark and Brennan, 1991; Schober, 1993). In particular, we focus on collaborative processes occurring during software development, where communication and collaboration have always been far from easy. We evaluate the effects of seeing other collaborator's gaze location on the visual attention patterns compared to the conventional situation of seeing only a mouse pointer.

Gaze in distant programming lecture: an initial experiment

The situation we investigate here is typical for collaborative episodes of sharing knowledge about a piece of code among a geographically distributed teams. Similarly, the distant programming lecture, as we call this episode, is characteristic to teaching programming, code co-comprehension, debugging, and pair-programming. A (more experienced) senior programmer explains a part of a code to the colleague, for example, when a new code maintainer begins working in a project. We simulate this situation and extend it so that the newly explained algorithm is applied immediately by the novice on a new set of input data.

In the present study we allow gaze display for only one direction, from the expert programmer to the novice.

Participants

Participation was voluntary and participants were recruited from computer science graduate and post-graduate students. There were two types of participant in this study, an expert programmer who assumed the role of the lecturer and a group of programmers that were assigned the role of students in this study. We recruited altogether thirteen participants, but due to technical problems and low-quality eye-tracking data two participants in the novice group could no proceed to complete the whole study.

The expert programmer had several years of programming experience with strong background in computer science, and was involved in the design of the experiment.

The group of student participants was composed of six females and four males. Average age was 26.1 years (SD = 4.2). Their self-rated efficacy and experience with programming was composed of three indicators. First, they rated how often they program, on a scale from 0 (never do programming) to 9 (every day programming). Second, the participants Java experience was rated on the same scale. Third, the familiarity with algorithms was collected.

Most of the participants had some previous exposure to programming (mean 5.8), Java experience was lower (mean 4.1), and algorithm familiarity was self-rated as 4.5.

Design and Procedure

The design of the study was a within-subject, with gaze-display (two levels, on and off) as an independent variable, and a set of dependent variables. We collected eye-tracking data for the novice participants, their performance on the subsequent test, and their responses on a user-feedback questionnaire. The order of conditions was balanced across participants.

The whole session has been divided into the learning passage, when the expert programmer explained the algorithm, and into the applied stage, when the learner applied the algorithm on a new data set.

Apparatus and Environment

In order to understand the second-to-second interaction between the peers we need to be able to analyze the direction of their attention concurrently. In the absence of generally available frameworks allowing joint and multiple gaze display and analysis of the data, we developed an open-source technology that makes the multiple gaze transfer available. The Multigaze framework (Bednarik et al., 2011) is an extension of the VNC protocol, allowing unlimited number of clients with or without eye-tracking device to connect to a shared environment. The data from all participants are aligned at the server, including mouse-clicks, keyboard input, and gaze-protocol. Clients can select whose gaze-mark they want to see on their own display overlaid on the shared desktop.

The Multigaze system was configured to display the gaze points as rounded, semi-transparent dots, with a diameter of about 0.5 degree of visual angle (30 pixels). In case of multiparty collaboration, the gaze-mark can be accompanied with a name of the emitting user.

Two Tobii eye-trackers (X120 and T120) were used to track the attention of the participants. The software provided by the manufacturers was set to record the gaze data, however, we also stored the aligned datasets onto a separate server computer that was running the shared environment. The size of the screens was 17 inches with resolution of 1280 x 1024 pixels.

Materials

Two target algorithms written in Java were presented to the novice participants, a breath-first and a depth-first implementations of searching for shortest path in a tree structure. Figure 1 presents the screenshot of the shared environment when presenting the algorithm. The depth-first search program made use of recursion, and the breath-first search was written in a sequential procedural style.

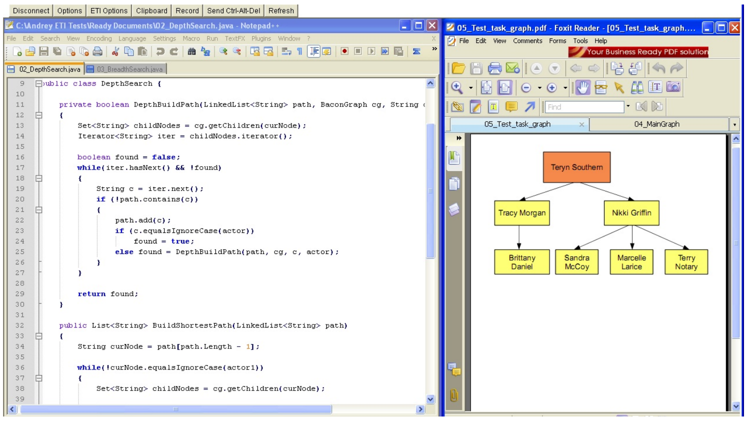

Figure 1. A screenshot of the shared environment with the depth-first search algorithm displayed on left, and test dataset displayed on the right..

A script containing the descriptions of the two programs was designed before the study. Each time the script was carefully followed by the expert when explaining the algorithms, and to keep the between-participant variations at minimum. Consequently, the duration of the learning passage was same for all sessions in the experiment. However, some variations in the proceedings of the experiment have occurred due to the eventual questions from the novice participants, such as whether the shared display can be scrolled, or whether it is ok to make notes. We allowed the questioning to keep the ecological validity high.

Analysis

In this paper we present an AOI type of analysis. The screen of the shared desktop was divided into two primary areas: the code and the data set window. The eye-tracking data were collected with respect to these areas. The data reported here concern the learning stage of the collaboration. The applied stage did not yield sufficient volumes of eye-tracking data, as the participants primarily focused their visual attention on the form to be filled.

Top 5% of the longest fixations were removed from the dataset. This filtration removed the artefacts of possible noise in the data, where multiple fixations would be merged into one extremely long fixation.

Results

Performance

Quantitatively, all participants were able to explain the algorithms and apply it to a new dataset. We have not observed any differences related to the intervention of this study.

Overall distribution of attention

Table I shows the distribution of eye-tracking measures across the two main areas of the display.

Table I. Overview of the eye-tracking dataset. Statistics are computed on the whole dataset across all participants.

Measure	Gaze		No Gaze	
	Code	Data	Code	Data
Total # of fixations	2005	1341	2137	1411
Total # of fixations %	59.9	40.1	60.2	39.8
Mean fix. duration (ms)	545.3	649.4	514.8	426.7
Mean fix. duration (SD)	497.2	621.0	465.9	333.1
Total fixation time (s)	1093	871	1100	602
Total fixation time %	55.7	44.3	64.6	35.4

Discussion

Effective communication is crucial in collaborative programming. Software development is an activity known for requiring large proportion of time spent on communication Herbsleb and Mockus (2003) and collaboration Herbsleb et al. (1995). It is thus important to support the collaborative activities and find ways to improve the processes when natural communication modes are inhibited. We investigate the role of gaze in the distant synchronous software development processes.

Conventional video conferencing has often been criticised for lack of eye-contact (Chen, 2002; Grayson and Monk, 2003). Gaze cursor seems to one potential candidate how to overcome the problem, and in this ongoing work we aim to evaluate its efficacy. Our preliminary results indicate that seeing other's gaze during distance teaching scenario has effects on visual attention of the novice. Perhaps due to ceiling effect, the analysis of quantitative performance results has not shown any differences contingent with the gaze cursor visibility. We are currently evaluating whether other, more qualitative, aspects to solutions could have been contingent on the presence or lack of gaze cursor. We however also believe that a more demanding task, for example a task with more frequent ambiguities, would allow observing the benefits of remote gaze tracking.

From the gaze-replays we observed that the novice participants were following the gaze cursor of the expert which resulted in more similarities in their behavior (Bednarik et al., 2011). However, the distribution of the fixations was similar across the two conditions.

On the other hand, the total fixation time and mean fixation durations show interesting differences. Having a gaze cursor influenced participants to look longer to the data window. In addition, the mean fixation duration was longer when gaze cursor was visible. Finally, while in the no-gaze condition the mean fixation duration was longer on code, opposite was true in the condition with the gaze-cursor visible.

Our future research will focus on the more detailed analyses of the gaze behavior. We predict that the effectiveness of gaze-cursor varies during different stages of communication during programming tasks. We thus want further deconstruct the datasets with respect to various subtasks. In addition, we plan to employ other tasks and vary their complexity.

References

Argyle, M. and M. Cook (1976): *Gaze and mutual gaze*. Cambridge University Press, Cambridge.

Bednarik, R., A. Shipilov, and S. Pietinen (2011): 'Bidirectional Gaze in Remote Computer Mediated Collaboration: Setup and Initial Results from Pair-Programming'. In: *In Procs. of CSCW'11*. New York, NY, USA, ACM.

Chen, M. (2002): 'Leveraging the asymmetric sensitivity of eye contact for videoconference'. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*. New York, NY, USA, pp. 49–56, ACM.

Clark, H. H. and S. A. Brennan (1991): 'Grounding in communication'. In: L. B. Resnick, J. M. Levine, and S. D. Teasley (eds.): *Perspectives on socially shared cognition*.

Cummings, J. N., J. A. Espinosa, and C. K. Pickering (2009): 'Crossing Spatial and Temporal Boundaries in Globally Distributed Projects: A Relational Model of Coordination Delay'. *Info. Sys. Research*, vol. 20, pp. 420–439.

Flor, N. V. (2006): 'Globally distributed software development and pair programming'. *Commun. ACM*, vol. 49, pp. 57–58.

Fussell, S. R., R. E. Kraut, and J. Siegel (2000): 'Coordination of communication: effects of shared visual context on collaborative work'. In: *Procs. of CSCW '00*. New York, NY, USA, pp. 21–30, ACM.

Goodwin, C. (2000): 'Action and embodiment within situated human interaction'. *Journal of Pragmatics*, vol. 32, no. 10, pp. 1489 – 1522.

Grayson, D. M. and A. F. Monk (2003): 'Are you looking at me? Eye contact and desktop video conferencing'. *ACM Trans. Comput.-Hum. Interact.*, vol. 10, pp. 221–243.

Herbsleb, J. D., H. Klein, G. M. Olson, H. Brunner, J. S. Olson, and J. Harding (1995): 'Object-oriented analysis and design in software project teams'. *Hum.-Comput. Interact.*, vol. 10, pp. 249–292.

Herbsleb, J. D. and A. Mockus (2003): 'An Empirical Study of Speed and Communication in Globally Distributed Software Development'. *IEEE Trans. Softw. Eng.*, vol. 29, pp. 481–494.

Hinds, P. J. and D. E. Bailey (2003): 'Out of Sight, Out of Sync: Understanding Conflict in Distributed Teams'. *Organization Science*, vol. 14, pp. 615–632.

Hoegl, M. and L. Proserpio (2004): 'Team member proximity and teamwork in innovative projects'. *Research Policy*, vol. 33, no. 8, pp. 1153 – 1165.

Horvitz, E., C. Kadie, T. Paek, and D. Hovel (2003): 'Models of attention in computing and communication: from principles to applications'. *Commun. ACM*, vol. 46, pp. 52–59.

Jokinen, K., M. Nishida, and S. Yamamoto (2009): 'Eye-gaze experiments for conversation monitoring'. In: *Procs. of IUCS '09*. New York, NY, USA, pp. 303–308, ACM.

Kraut, R. E., S. R. Fussell, and J. Siegel (2003): 'Visual information as a conversational resource in collaborative physical tasks'. *Hum.-Comput. Interact.*, vol. 18, pp. 13–49.

Liu, C., D. L. Kay, and J. Y. Chai (2011): 'Awareness of Partner's Eye Gaze in Situated Referential Grounding: An Empirical Study'. In: *In Proceedings of the 2nd Workshop on Eye Gaze in Intelligent Human Machine Interaction in Conjunction with IUI 2011, the International Conference on Intelligent User Interfaces*. pp. 33–38.

Schober, M. F. (1993): 'Spatial perspective-taking in conversation'. *Cognition*, vol. 47, no. 1, pp. 1 – 24.