Integrating Asynchronous Interaction into Real-time Collaboration for Crowdsorced Creation

Sang Won Lee

Computer Science & Engineering[1]
University of Michigan, Ann Arbor
snaglee@umich.edu

Abstract

My dissertation aims to enable dynamic crowds of web workers to collaborate to create complex artifacts; developing a software, sketching new ideas, playing music together, or writing a novel. To support this open-ended process, we introduce a real-time collaboration systems that are domain-specific. First we address the inherent challenges of crowdsourcing in supporting the model. We aim to develop a novel set of techniques to make the system easy to use, to support self-coordination, and to sustain real-time collaboration, inspired by the asynchronous and synchronous CSCW disciplines and remixing them for crowdsourcing.

Introduction

My dissertation seeks to understand how we can coordinate real-time collaboration, especially for complex tasks that involve an open-ended process. We present a set of cross-domain tools and techniques inspired by asynchronous collaboration in CSCW to coordinate and facilitate crowdsourcing for creative practices, ranging from real-time performing arts, through programming, writing, to designing interactive prototypes. The other goal is to involve non-expert users in the collaborative process coordinated by the system and guided by the other people.

While crowdsourcing provides a scalable solution to some problems that cannot be easily automated with machine intelligence, its applications are limited to the well-defined problems. However, many of the real-world problems are not well-defined, and in this case, the process needs to be flexible, agile and iterative. Collaboration is an essential component for such processes. Furthermore, in some cases, crowdsourcing is used to engage a more general audience for various purposes: education, public events, or participatory arts, which neither has an apparent problem nor a goal-oriented task to be solved/addressed.

Crowdsourcing poses a set of inherent challenges which makes it especially difficult to accomplish such open-ended tasks via live collaborative process. Building a real-time collaborative system in crowdsourcing will be different from following set of design implications found in synchronous groupware. We aim to address the problems, and we can generalize our understanding in facilitating real-time collaboration in the demanding situation. My research focuses on adapting and improving existing disciplines on asynchronous CSCW systems in response to the challenges unique to this setting. Findings and techniques that support creative collaboration at scale will benefit not only expand the domain of crowdsourcing to the broader applications and also can be applied to participatory practice in today's web.

Real-time Collaboration in Crowdsourcing

The most established approach to crowdsourcing involves decomposing a task into microtasks that non-expert crowd workers can solve independently, without having to collaborate. This method is effective in settings where there exists a definable problem-solving process. For open-ended tasks, microtasking fails as there is no well-defined steps of solving the problem. Instead, real-time collaboration—that goes beyond just the "collection of wisdom"—can enable the crowd to work on a bigger-sized task without well-structured workflows.

One potential way to support the collaborative model is to apply the traditional team-based project management in crowdsourcing by having expert crowd workers as in (Retelny et al. 2014). In this work, an end user can create a workflow from scratch (depending on the project) and change it on the fly. This method preserves the advantages of dynamic recruitment procesess while the ad-hoc team can quickly accomplish complex tasks such as designing an interactive system. However, the actual collaboration within an individual sub-task is hidden behind the workflow and left to the ad-hoc team.

The other approach is to design domain-specific tools in which multiple people can collaborate on an open problem in that domain (e.g. shared documents in writing). For example, Apparition exemplifies this method of real-time collaboration in crowdsourcing in the domain of user interface design (Lasecki et al. 2015). This work chooses to take this approach of presenting domain specific systems in various fields, finding cross-domain solutions to the problem.

Challenges of Real-time Collaboration in Crowdsourcing

There exist a set of challenges that make live collaboration in crowdsourcing difficult.

1. the variability in expertise of each worker
2. limited on-boarding and coordination time
3. the transient nature that workers come and go

First, as crowd workers may have a diverse background, it will be essential to design the system easy enough so that it is open to the larger pool of non-expert workers. Second, the anonymous nature of crowdsourcing makes it difficult for the crowd to coordinate their efforts unless the specific roles of individual workers are assigned in advance. Thus there emerge needs in the system for a crowd to self-coordinate the collaboration. Lastly, the transient nature of crowd worker as they are free to leave and join easily creates a critical problem for real-time collaboration. Crowd workers dislocated in time should be able to quickly catch up with ongoing live collaboration. These challenges are especially critical to the real-time collaboration model whereas the microtasking approach is tolerant to these.

Description of Proposed Research

To address these challenges above, we draw insights from CSCW in traditional setting the design implications found in CSCW from two different collaborative forms: asynchronous and synchronous, a classification method based on whether users of the groupware are working together at the same time (*synchronous*) or different times (*asynchronous*) (Rodden 1991). The descriptive framework of awareness and coordination of CSCW has been developed separately for synchronous groupware (Gutwin and Greenberg 2002) and asynchronous groupware (Tam and Greenberg 2006).

Engineering of existing findings in real-time groupware will fail for many reasons: difference in scale, temporary time-frame, and the lack of additional communication. The proposed research originates from the idea that the challenges of real-time crowd collaboration can be regarded as the problems in asynchronous collaboration found in CSCW. The techniques to support asynchronous collaboration and it needs to be adapted to address the challenges in the crowd setting.

Here, three research questions are proposed.

1. How can integrating asynchronous and synchronous collaboration keep real-time collaboration accessible to non-expert users?
2. How can asynchronous workflows can be parallelized and coordinated concurrently?
3. How can the transient nature of crowd workers be mediated with the adaptive use of asynchronous change awareness?

The hybrid usage of asynchronous design in real-time collaboration setting will create a novel techniques permitting creative practice to be solved in crowdsourcing and general social computing domain.

Planned Methodology and Preliminary Works

We examine our research questions in a variety of domains to understand how our techniques generalize. In particular, we explore four different domains: programming, writing, user interface design, and audience participation in music performance. The common tasks in these areas well reflect the needs for collaborative efforts. i) They are complex tasks with inherently interdependent sub-components, ii) the processes required are often not fixed, but open-ended, iii) rapid iteration is emphasized in practice, and iv) measuring a traditional productivity is insufficient.

Currently, we are in the process of testing our hypotheses of reinventing collaborative techniques inspired by asynchronous collaboration (e.g. record-and-replay) to address the challenges, which we think can leverage related approaches across domains. **[Live Writing]:** This project discusses the effects of the real-time playback of individual writing/programming sessions in order to assist context recovery for other collaborators. Having real-time access to past interactions and identifying summarized history based on the interaction log can compensate for the lack of communication and clear responsibility (Lee and Essl 2015). **[Demonstrate-and-Remix Interactive Behaviors]:** This work extends the existing crowd-powered UI sketching system (Lasecki et al. 2015). In this scenario, an end user describing the interactive behaviors of a UI and crowd workers recognising and creating the behaviors in the sketch. The additional functionality lets crowd workers "demonstrate" the required changes for a particular interactive behavior and "remix" their demonstrations. The refined interactive behaviors can be documented and triggered leveraging on human intelligence. A typical recording function for asynchronous collaboration is repurposed to transform the process of composing interactive behaviors in a sketch extremely easy and fast. **[Crowd in C[loud] - Audience Participation in Music Performance]:** In this interactive music performance, each audience member is given a microtask to compose a tune on their mobile phones. The aggregated result of each playing a short tune creates a texture of a musical instrument. The mobile musical application that audience uses is built as a shared environment, using the metaphor of online dating websites. A tune composed by a participant acts as a personal profile on an online dating website. Browsing tunes composed by others mimics the activity of scanning online profiles of others (de Carvalho Jr, Lee, and Essl 2016).

My dissertation research expands the understanding of CSCW disciplines developed in the temporal dimension to the domain of crowdsourcing systems for collaborative tasks and more general social computing.

Research Issues

My previous and preliminary work has been practice-based research where creative artifacts are the basis of the contribution, especially in the computer music domain. These lack a formal user study to rigorously validate the hypotheses that stem from our research questions. One of the biggest challenges that I face is that, due to the compound nature of domain-specific tasks that I chose, standard productiv-

ity measures may not be sufficient to understand the impact of our techniques. Other than exploring time efficiency, how can I measure complex qualities such as collaboration, creativity, exploration, expressivity, or engagement? How would I differentiate these varieties independent from the overall performance? What would be the baseline condition of solving these complex, open-ended and exploratory tasks?

Background and Related Works

Utilizing real time collaboration in crowdsourcing systems has been attempted in various domains. Legion enables crowd workers to control an existing user interface and tested different strategies to mediate various crowd responses (Lasecki et al. 2011). However, the nature of the given tasks was simple that a collaboration structure or coordination among workers was not required. Apparition is a direct precursor of this work where the crowd workers are asked to collaborate on a task of prototyping a user interface as the end user describes and draws its specifications (Lasecki et al. 2015).

Self-coordination in real-time collaboration is challenging and draws ideas from existing works of encouraging structured collaboration. Kim et al. finds that the self-assignment of roles helps participants complement each other's skills, leading to better collaboration (Kim, Cheng, and Bernstein 2014). André et al. demonstrate that sequential workflow becomes more effective than simultaneous work as the size of a collaborative group increases (André, Kraut, and Kittur 2014). Apparition avoids conflicts on a shared canvas through a write-lock mechanism and in-progress markers (Lasecki et al. 2015).

Programming is a challenging domain for crowdsourcing as it involves a complex task where decomposition is difficult due to the interdependency within code. There has been various approaches to solve the problem. Topcoder leverages a community of programmers using a competitive model where contributors participate in programming contests (Lakhani, Garvin, and Lonstein 2010). Latoza et al. suggest a systematic approach to decomposing a complex programming task into a set of microtasks that can be quickly solved by individual crowd-workers (LaToza et al. 2014). However none of the above support agile development.

There emerged a new artistic practice of online crowdsourced art using the Internet as a participatory platform to directly engage the public in the creation of visual, musical, literary, or dramatic artwork (Literat 2012). Musicians often involve the audience in a concert hall as part of the music making process with the aid of technologies, typcially mobile music applications (Lee and Freeman 2013). Ensemble explores crowdsourcing in the domain of creative writing, facilitating online-coauthoring the open-ended task via two different collaboration models (Kim, Cheng, and Bernstein 2014).

References

André, P.; Kraut, R. E.; and Kittur, A. 2014. Effects of simultaneous and sequential work structures on distributed collaborative interdependent tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 139–148. ACM.

de Carvalho Jr, A. D.; Lee, S. W.; and Essl, G. 2016. Understanding cloud service in the audience participation music performance of crowd in c[loud]. *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)* 1001:48109–2121.

Gutwin, C., and Greenberg, S. 2002. A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work (CSCW)* 11(3-4):411–446.

Kim, J.; Cheng, J.; and Bernstein, M. S. 2014. Ensemble: exploring complementary strengths of leaders and crowds in creative collaboration. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, 745–755. ACM.

Lakhani, K. R.; Garvin, D. A.; and Lonstein, E. 2010. Topcoder (a): Developing software through crowdsourcing.

Lasecki, W. S.; Murray, K. I.; White, S.; Miller, R. C.; and Bigham, J. P. 2011. Real-time crowd control of existing interfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 23–32. ACM.

Lasecki, W. S.; Kim, J.; Rafter, N.; Sen, O.; Bigham, J. P.; and Bernstein, M. S. 2015. Apparition: Crowdsourced user interfaces that come to life as you sketch them. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 1925–1934. ACM.

LaToza, T. D.; Towne, W. B.; Adriano, C. M.; and van der Hoek, A. 2014. Microtask programming: Building software with a crowd. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, 43–54. ACM.

Lee, S. W., and Essl, G. 2015. Live writing: Asynchronous playback of live coding and writing. In *Proceedings of International Conference on Live Coding*.

Lee, S. W., and Freeman, J. 2013. echobo: A mobile music instrument designed for audience to play. *Proceedings of the International Conference on New Interfaces for Musical Expression* 1001:48109–2121.

Literat, I. 2012. The work of art in the age of mediated participation: Crowdsourced art and collective creativity. *International Journal of Communication* 6:23.

Retelny, D.; Robaszkiewicz, S.; To, A.; Lasecki, W. S.; Patel, J.; Rahmati, N.; Doshi, T.; Valentine, M.; and Bernstein, M. S. 2014. Expert crowdsourcing with flash teams. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, 75–85. ACM.

Rodden, T. 1991. A survey of cscw systems. *interacting with Computers* 3(3):319–353.

Tam, J., and Greenberg, S. 2006. A framework for asynchronous change awareness in collaborative documents and workspaces. *International Journal of Human-Computer Studies* 64(7):583–598.

Supplemental Paragraph

Motivation: I have been actively doing computer music research with the intersection of HCI and software engineering. In the computer music community, typically researchers seek for new ways to express music. One theme that I consistently pursued for last 6 years was to scale the music making process and by default music performance is real-time performing art Naturally, crowdsourcing is an ideally challenging test-bed that I can confirm if the findings from the previous work generalize in other domains. To that end, I think it will be mutually beneficial for me to attend in the Doctoral Consortium for me to seek for advices from the established researchers in Human Computation for the community to bring diversify the research topic and blurs the definition of crowdsourcing.

The status of your dissertation proposal : Writing

Expected Defense Date : July, 2017