Exploring the Use of Visual Annotations in a Remote Assistance Platform

Mark Rice
Institute for Infocomm Research
A*STAR, Singapore, 138632
mdrice@i2r.a-star.edu.sg

Shue Ching Chia
Institute for Infocomm Research
A*STAR, Singapore, 138632
scchia@i2r.a-star.edu.sg

Hong Huei Tay
Institute for Infocomm Research
A*STAR, Singapore, 138632
hhtay@i2r.a-star.edu.sg

Marcus Wan
Institute for Infocomm Research
A*STAR, Singapore, 138632
marcus-wan@i2r.a-star.edu.sg

Liyuan Li
Institute for Infocomm Research
A*STAR, Singapore, 138632
lyli@i2r.a-star.edu.sg

Jamie Ng
Institute for Infocomm Research
A*STAR, Singapore, 138632
jamie@i2r.a-star.edu.sg

Joo Hwee Lim
Institute for Infocomm Research
A*STAR, Singapore, 138632
joohwee@i2r.a-star.edu.sg

Abstract
In this paper, we report on the evaluation of a remote assistance platform (RAP) that is designed to enable an expert to remotely assist a field operator. A user study with 16 participants was conducted to evaluate its usability with two assembly tasks that varied in their complexity. As part of the assessment, we compared the interaction behavior of our platform with a commercial instant messaging application, which lacked the ability to augment or view video imagery. The results identified differences in the completion times between the two conditions, as we examined the use of visual augmentation, including recommendations to improve the platform.

Author Keywords
Remote assistance; visual annotation; task assembly; instant messaging.

ACM Classification Keywords
H.5 Information Interfaces and Presentation (e.g., HCI): H.5.1 Multimedia Information Systems.

Introduction
Remote assistance can offer support in collaboration by enabling both experienced and low-skilled workers the

means to perform tasks through the virtual support of an expert. This is particularly useful in countries like Singapore, where in recent years there has been a revised focus in improving the productivity of service and manufacturing sectors. However, despite the potential for remote assistance tools to help support the restructuring of labor markets, Kim et al. argue that "*…most* [video conferencing] *systems have poor support for spatial cues, poor awareness of the remote user's context, and hard to share gesture communication cues"* [4, p. 83].

Within HCI, a number of studies have investigated the use of remote assistance via video streaming. These have varied from laser-guided projection [8], to eliciting user comparisons between operating camera and projector systems [6], hand and cursor control [7], and the use of tele-operated robotics [3].

In terms of annotating video, Fussell et al. [2] compared a video drawing tool, which was found to be faster at completing tasks than only viewing video. Through a user evaluation, graphical annotations were grouped into five categories, including pointing to objects, and indicating their angle of insertion, or orientation.

More recently, Palmer et al. [8] developed a remote guidance system where graphical annotations place on a live video feed could direct laser projected information. Designed for tele-health, the informal reporting of a user study indicated that the system performed sufficiently.

Researchers such as Kim et al. [5] have also conducted user research to understand the effectiveness of video auto-freezing to create annotations for remote assistance, in addition to comparing drawings on live video and snapshot images [4]. In the latter work, the authors found that drawing onto live video restricted viewing movement, while annotating onto snapshots caused some visual disorientation when returning to a live view [4]. Alternatively, Domova et al. [1] identified that taking video snapshots was found to be effective in a poor network environment, but gave few examples of how visual annotations were used for remote assistance tasks.

Subsequently, in extending these works, there still remain open questions over the use of visual annotations for remote guidance. Namely, their context of use, and how the types of visual gestures relate to the tasks performed. As such, the motivation for this work was to look at the use of visual annotations within the ongoing development of a remote assistance platform.

Similar to prior work [1, 4], the design of our platform currently works on the approach of taking snapshots from a live video, from which annotations can be included. The contribution of this paper is to provide a descriptive summary of the use of annotations for two assembly tasks, and to begin to understand how interactive behavior differs in their absence.

The technology

The remote assistance platform (what we describe as **RAP** in this paper) comprises of three main components: 1) a tablet application used by the field operator; 2) a web application used by the domain expert, and 3) a backend server that handles the communication exchange between the two parties.

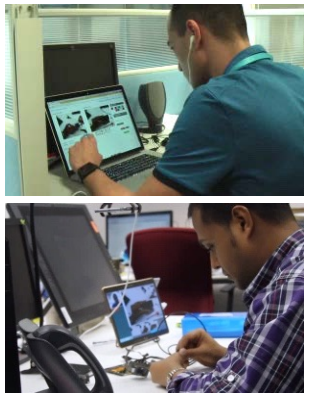

Figure 1: The RAP setup. *Top*, expert/instructor, *bottom*, remote field operator/assembler.

For the hardware setup, the field operator uses a Samsung smartphone and tablet, both running Android OS. The smartphone is placed in an overhanging position using a camera stand, where its back projected camera is able to capture a live video stream of the current workspace. Simultaneously, this view is displayed in the tablet to preview what the expert will see.

In real-time, the video stream is sent to a web application on a PC controlled by the domain expert. Audio communication is fully integrated. From the assembly video the expert is able to edit a video frame and send snapshots back to the field operator. Annotated features provide basic, yet a flexible range of functions to help the domain expert illustrate instructions. These include: 1) free hand drawings, 2) text/symbols, and 3) region selection using the RGB colors on a video image (Figure 2).

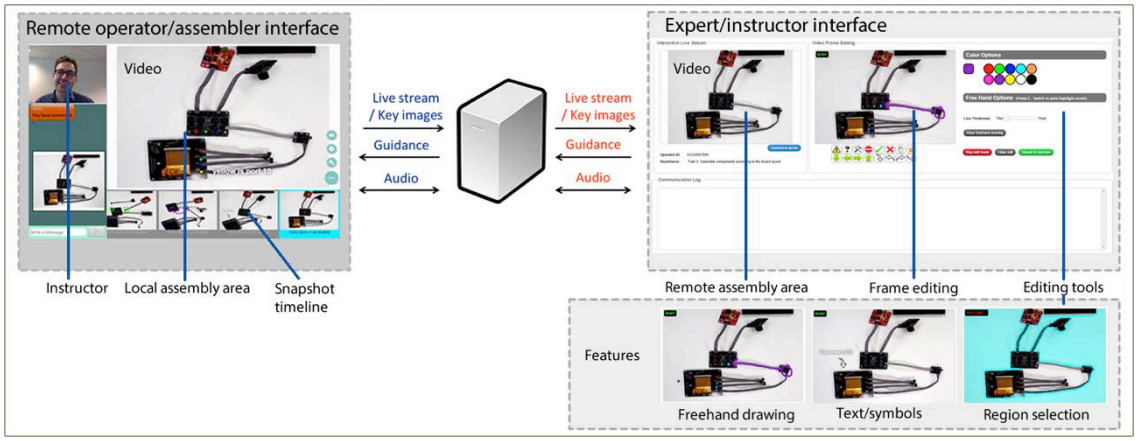

Figure 2: Illustration of the Remote Assistance Platform (RAP).

The field operator then views new snapshots on the tablet display. This interface retains a scrollable timeline of the received snapshots from which they can refer back to at any time. A video feed is also provided of the instructor.

A live video stream is achieved using WebRTC technology [9]. A signaling server is implemented using Java on a Spring framework. This server is used to connect the domain expert and the field operator through the use of a unique ID. A chat server is implemented using NodeJS, which separately handles the transmission of annotated images.

User study

A small-scale study was conducted to explore the usability of our prototype system, specifically the usefulness of the visual annotations. To do this we compared two conditions. In the first condition, tasks were completed on the RAP platform that included video streaming and image annotations. In the second condition we purposely removed these features by using a commercial instant messaging application, and constrained the interaction to voice and taking photographs of the assembly tasks.

Our experience has found that instant messaging applications are used to convey instructions in industries such as maintenance, where images are accompanied by text instructions to describe a task, or a scene. Subsequently, by using a subset of the instant messenger features, we wanted to begin to understand how task performance might vary between the use of video and visual annotations compared to those without.

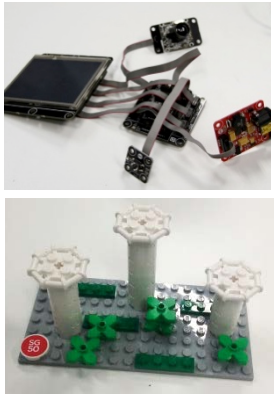

Figure 3: The tasks. *Top*, camera assembly; *bottom*, Lego assembly.

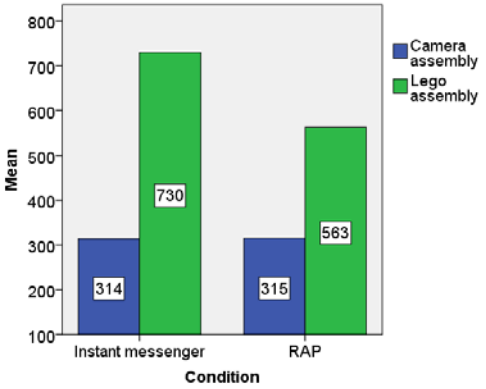

Figure 4: Average completion time per task.

In total, we recruited 16 participants (11 males; 5 females, with an age range of between 20 to 42 years) from the research institute. For the RAP, the setup was the same as the hardware previously described (see Figure 1). For the instant messaging condition, a PC version of the software was displayed on a laptop for the expert interface, while a mobile version on a smartphone was used for the field operator. For each pair, the testing was completed in separate rooms, and each session was recorded onto video.

Tasks

Using a between-subjects design, participants were asked to complete two tasks. In the first task, they were asked to assemble the electrical components of a camera (provided in 5 separate parts). In the second task, participants had to assemble an outdoor scene consisting of 28 Lego pieces (Figure 3). Each task took between 5 to 15 minutes to complete, as instructions were provided in the form of A4 handouts that visually illustrated the placement of assembly objects.

Procedure

Participants were randomly assigned to one of the two conditions, including for each pair their role as being either an *instructor* or *assembler*. As the names suggest, the instructor's role was to provide the instructions in the task, while the assembler to construct them. As both participants were novice users, these terms were felt to be a better description then calling them an expert and field operator.

On giving consent, the instructor was given 30 minutes to familiarize themselves with the interface and tasks, while the assembler with a simpler interface, a shorter 10 minutes to understand the remote setup. At no time could the instructor show the assembler the paper instructions they had received. They then completed the two designated tasks, followed by a 10-15 minute paired interview to review their experiences using the technology.

Given the small sample size, we focus on reporting descriptive statistics, in addition to a general summary of the interactive behavior identified from the video recordings, and annotations logged in the RAP system.

Preliminary findings

In comparing the RAP with the instant messenger, the average completion times were almost identical for the camera assembly (**RAP**, *M* = 315 secs; **instant messenger**, *M* = 314 secs). In contrast, a greater difference in performance was identified in the Lego task, as the RAP was noticeably quicker than the instant messenger (**RAP**, *M* = 563 secs; **instant messenger**, *M* = 730 secs) (Figure 4).

To account for these differences, using the instant messenger, the camera parts were fairly easy to distinguish, with labelled ports to check the numbers. In contrast, the uniformed shapes of the Lego pieces required more verbal directives, as pairs either adopted the use of grid coordinates, or physically counted the number of connecting circles on the board. Approaches that became more difficult to describe the more Lego pieces were positioned: "*One circle in from the top, and then two circles away from the green thing, which is behind the first tower you built. It should be one circle away from the edge*".

The interviews also suggested that one of the biggest constraints of not having a live video feed was the need

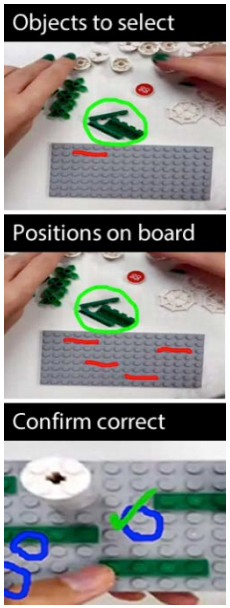

Figure 5: Examples of the different types of annotations used in the assemblies.

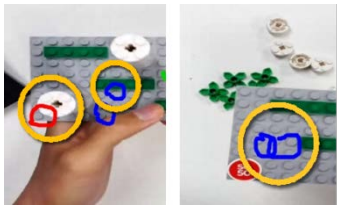

Figure 6: Examples of the annotation misalignment.

to mentally rotate objects when describing their position: *"The actual object, the capturing angle may be different, so this one I also need to do a conversion".* As such, without a visual reference point, it was common for the assembler to intermittently take photographs of the build to ask for confirmation on a task (averaging at 6 images for the Lego, compared to 2 images for the camera assembly). One instructor also indicated changing the use of terminology in the tasks to support their partner: "*Half way through I got confused. I was trying to direct him, two down and a few across, and I realised my terminology might be different. So I adapted to him using rows and columns*".

In comparison, feedback from the RAP was positively received, particularly in terms of knowing when to intervene in the tasks: *"It is very useful that you can just support the person with the task, and if you can see they are doing well you don't need to correct anything".* As such, with a video feed and the use of annotations, we noted that responses tended to be less instructive in the positioning of objects: *"Connect four of them, and put this on the place I marked on the image".* Corrections were also easier to implement (*"Shift it one piece to the right"*), as instructors gave more confirmation of correct actions.

Similar to variations in the number of photographs taken by the assembler in the instant messenger condition, on average there were a much higher number of annotated snapshots sent by the instructor in the Lego (16) compared to the camera assembly (2).

Annotation usage in the RAP

Based on the system log, 73 video snapshots with embedded annotations were made by the four RAP pairs. Within these images, we observed three types of annotations used by the instructor: 1) annotations to make visible an object to be selected (25%), 2) annotations to indicate an object's position (73%), and 3) annotations to confirm a correct assembly (2%) (see Figure 5).

In terms of the types of annotations used, freehand drawings far outweighed the use of symbols (appearing in 100% of the snapshots compared to 6%), suggesting that the types of symbols used may not have been context specific enough for the task. Moreover, while liked, at times the drawings lacked precision in their detail, which may partially reflect the size of the video and the use of a touchpad for input control. For example, we identified five instances where the instructor partially overlaid an annotation over a previous image, incorrectly giving the impression that one object was to be positioned on top of another (see Figure 6).

Annotated objects were also depicted differently in the tasks. In the camera task, a lose circle commonly denoted an area of interest, while for the positioning of the Lego pieces, they were either drawn as their actual size on the board, or central assembly position (see Figure 7). Given the nature of the Lego task, this could cause some confusion between perceived and actual representation: *"Sometimes there be like four things circled* [on the board], *but that thing* [object] *only has one* [pin]".

As such, it was recommended that a zoom feature be implemented in the instructor interface to improve the accuracy of positioning annotations, in addition to the precision of a drawing tool (e.g. a line ruler) for making

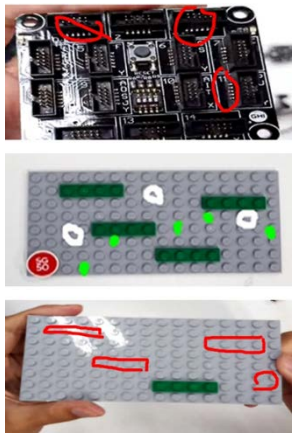

Figure 7: Annotation styles. General indication of proximity for the camera build (*top*), compared to the careful positioning of annotations to indicate the central placement point (*middle*), or complete object placement (*bottom*) for the Lego task.

symmetrical shapes. Design suggestions also included the use of grid co-ordinates to be displayed over the video to help direct the position of the objects.

When considering the layering of annotations over time, for the Lego task they appeared to work best when applied to a recent snapshot of a build. For example, without the visual reference to recently assembled objects, individuals suggested that additional effort was needed to trace the positioning of new annotations. Furthermore, suggestions by the instructors included an easier means to reuse previous annotations in the task, including those embedded beneath other drawings. For the image timeline, the assemblers also felt that annotated snapshots should be numbered in order to better draw reference too.

CONCLUSION

Our findings indicate greater value in the use of annotations where objects were identical in appearance, and lacked easily defined reference points. This suggests a need to better understand how visual annotations correspond to the type, or complexity of procedural tasks, including how interfaces should be designed to adapt to different gestural styles. Interestingly, in this study we see differences in the use of annotations not only between tasks, but also among individuals.

Recommendations from the study include improving the accuracy of positioning annotations in close proximity to one another, understanding how to reuse aspects of annotations when embedded in a sequence, and the added value of different forms of annotations, such as symbols over freehand drawings. Our future work recognizes the need to undertake a more controlled study to compare other interface features, including the use of a movable camera for a wider field of view. Fairer comparisons to understanding RAP's performance may also be better determined by weighing against commercial messaging applications that use live video.

References

1. Veronika Domova, Elina Vartiainen, and Marcus Englund. 2014. Designing a remote video collaboration system for industrial settings. In *Proc. ITS '14*, 229-238.
2. Susan R. Fussell, Leslie D. Setlock, Jie Yang, Jiazhi Ou, Elizabeth Mauer, and Adam D.I. Kramer. 2004. Gestures over video streams to support remote collaboration on physical tasks. *Human-Computer Interaction 19*, Sept 2004, 273-309.
3. Pavel Gurevich, Joel Lanir, Benjamin Cohen, and Ran Stone. 2012. TeleAdvisor: A versatile augmented reality tool for remote assistance. In *Proc. CHI '12*, 619-622.
4. Seungwon Kim, Gun Lee, Nobuchika Sakata, and Mark Billinghurst. 2014. Improving co-presence with augmented visual communication cues for sharing experience through video conference. In *Proc. ISMAR '14*, 83-92.
5. Seungwon Kim, Gun A. Lee, Sangtae Ha, Nobuchika Sakata, and Mark Billinghurst. 2015. Automatically freezing live video for annotation during remote collaboration. In *Proc. CHI EA '15*, 1669-1674.
6. Joel Lanir, Ran Stone, Benjamin Cohen, and Pavel Gurevich. 2013. Ownership and control of point of view in remote assistance. In *Proc. CHI '13*, 2243-2252.
7. Jane Li, Anja Wessels, Leila Alem, and Cara Stitzlein. 2007. Exploring interface with representation of gesture for remote collaboration. In *Proc. OzCHI '07*, 179-182.
8. Doug Palmer, Matt Adcock, Jocelyn Smith, Matthew Hutchins, Chris Gunn, Duncan Stevenson, and Ken Taylor. 2007. Annotating with light for remote guidance. In *Proc. OzCHI '07*, 103-110.
9. WebRTC. 2015. https://webrtc.org/native-code/