Using Real-time Feedback to Improve Visual Question Answering

Yu Zhong
University of Rochester
252 Elmwood Ave.
Rochester, NY 14627 USA
zyu@cs.rochester.edu

Walter S. Lasecki
University of Rochester
252 Elmwood Ave.
Rochester, NY 14627 USA
wlasecki@cs.rochester.edu

Phyo Thiha
University of Rochester
252 Elmwood Ave.
Rochester, NY 14627 USA
pthiha@cs.rochester.edu

Jeffrey P. Bigham
University of Rochester
252 Elmwood Ave.
Rochester, NY 14627 USA
jbigham@cs.rochester.edu

Grant He
University of Rochester
252 Elmwood Ave.
Rochester, NY 14627 USA
ghe3@u.rochester.edu

CHI'12, May 5–10, 2012, Austin, Texas, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract
Technology holds great promise for improving the everyday lives of people with disabilities; however, automated systems are prone to errors and cannot handle many real-world tasks. VizWiz, a system for answering visual questions for blind users, has shown that crowdsourcing can be used for assistive technology in such domains. Our work extends the VizWiz model by enabling users to interact with the crowd via a real-time feedback loop. We introduce Legion:View, a system that enables such a real-time feedback loop for visual questions between users and crowd workers. Legion:View sends audio questions and streaming video to the crowd, and forwards feedback about the position and orientation of the camera and answers to questions back to users.

Keywords
Assistive Technology, Blind, Real-time Crowdsourcing, Human Computation

ACM Classification Keywords
K.4.2 [Social Issues]: Assistive technologies for persons with disabilities

General Terms
Human Factors, Design

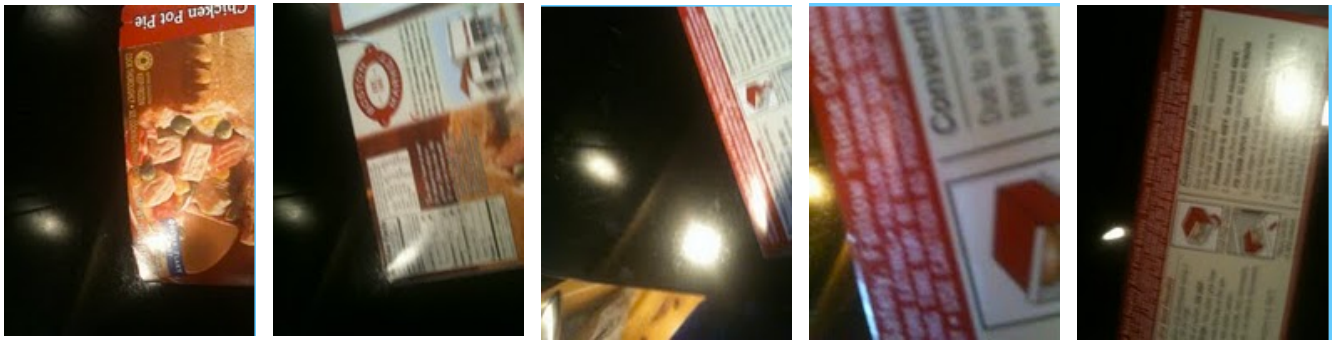

Figure 1. Sequence of pictures taken with VizWiz to answer the question "How do I cook this?" Each question submission was answered quickly, but it took the crowd workers about 10 minutes to help the user frame the right part of the box before answering the question.

Introduction

One role of assistive technology (AT) is to help people with disabilities interpret the world around them. This often requires solving complex Artificial Intelligence (AI) problems, such as computer vision, which machines cannot yet do reliably and accurately. One approach for achieving the promise of AT now is to have a dynamic group of workers available on-demand, which has proven to be a valuable tool for developing deployable AT systems. Using the crowd leverages human computation to handle problems that automated methods cannot yet reliably solve. Workers can be recruited at low cost from services such as Amazon's Mechanical Turk or CrowdFlower. This allows those developing AT to provide affordable systems without having to wait for AI research to reach a stage where it is able to solve all of the necessary problems.

VizWiz [2] is a mobile application that enables users to ask questions about their visual environment by first taking a picture, then recording an image and submitting it to the crowd, their social network, or an automated image recognition system. Answers from workers are then forwarded directly back to the user. Figure 1 shows that even though individual answers are obtained quickly, answering sequences of questions can be time consuming.

In this paper, we describe Legion:View, a system designed to provide real-time answers and feedback to audio questions about a user's visual environment. Unlike VizWiz, Legion:View forwards a video stream to crowd workers, allowing for real-time guidance in taking pictures and for follow-up audio questions to be submitted. This avoids the need to resubmit questions and re-recruit workers in the event of a problem with

the image or audio. It also means that subsequent questions are distributed to workers with knowledge of the domain and task.

We first describe the design of the system and its current implementation, followed by preliminary usability tests. We conclude with a discussion of our findings on how workers interact with Legion:View, and future work to extend the system in navigation tasks.

Background

Legion [3] was the first system to engage crowd workers in a closed-loop fashion, and perform continuous tasks in real-time. This is accomplished by creating a virtual network computing (or remote desktop) style interface with the crowd. Multiple workers are shown a video stream of the interface and all contribute input simultaneously. Legion then aggregates their input in real-time using an *input mediator* and forwards it back to the user's interface as a single control stream. Legion:View uses this model, but implements an input mediator that is customized to handle the two classes of worker input separately.

VizWiz [2] used the quikTurKit script for TurKit [4] to maintain a queue of workers who completed older tasks until a new one was submitted. This effectively sped up the response time by increasing the number of workers available. This method of engaging workers inspired the closed-loop approach used in Legion. Adrenaline [1] shows that these crowds can be recruited in less than two seconds and kept around until a task is ready.

EasySnap [6] is an application that helps blind users take photographs by first having them frame the image at close range, then move progressively farther away until they choose to take the picture. Machine vision is used to identify the primary object in the initial frame, then track it as the user zooms out, giving the user audible cues to follow in order to keep it centered in the frame. Legion:View uses a similar approach to cue the user, but does not rely on machine vision, which is prone to losing track of the object, and cannot tell if appropriate information is being captured in the frame.

Design

In order to improve on visual question answering for blind users, Legion:View adds the concept of a closed-loop interface between users and the crowd. Figure 2 shows the layout of Legion:View. Users capture streaming video and a series of questions via the mobile interface, and workers provide feedback, in the form of orientation corrections and answers, to users.

Blind users often have difficulty framing images. This is particularly true when no tactile clues regarding the orientation of an object are available. In Legion:View, continuously streaming video to the crowd and receiving feedback in real-time removes the need for repeatedly submitting the same question, a procedure that could take a significant amount of time using VizWiz (as can be seen in Figure 1).

Another issue is that we must promote convergence to a single action in order to use multiple workers at once, which helps ensure reliability and optimality in the response, and prevents malicious users from abusing this feature. We do this by providing a predefined set of camera adjustment controls for workers to suggest. As in VizWiz, answers are forwarded directly to the user, who is able to judge their validity by comparing them to subsequent responses.

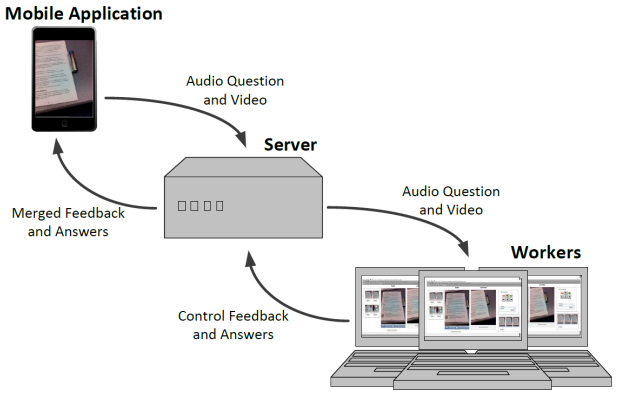

Figure 2. The mobile application forwards audio questions and video stream to the server, which then distributes it to the workers. Workers request adjustments to the camera angle as needed, which are then combined into a single feedback by the input mediator running on the server. When able, workers submit answers that are directly forwarded to the user.

Instead of allowing only a single audio question per submission, which reduces the benefits of our approach, we let users record an initial question, and then re-record a new one at any point. This allows users to modify a question or ask follow-up questions to the same crowd of workers who know the answers to previous questions.

System

Legion:View is composed of a mobile application, server, worker interface, and Legion input mediator.

Mobile Application

The mobile platform is an iOS application (Figure 3) that allows users to record audio and stream video, and uses VoiceOver to read screen elements and answers.

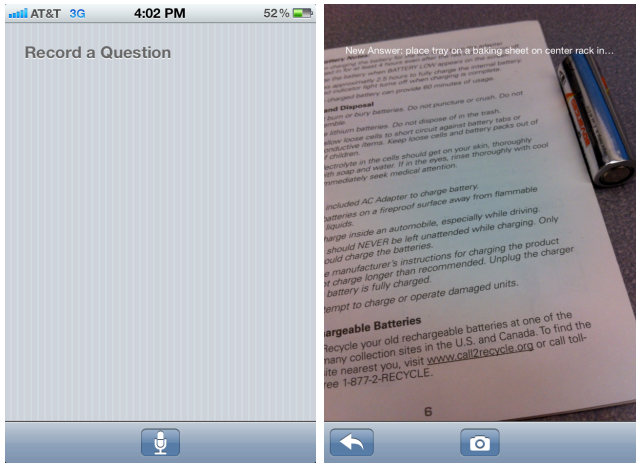

Figure 3. The mobile application interface. Users can record a verbal question (left) and begin streaming video (right). The application then forwards the video stream and question to the server. At any point, the user can pause the video stream, re-record a new question, or end the session. VoiceOver is used to read the incoming controls aloud to users.

Server

When the Legion:View mobile application is started, the server recruits a crowd of workers from Mechanical Turk. The server then forwards the audio question and the video stream from the user to recruited workers via the worker interface. Responses are then aggregated using the input mediator, and returned to the user.

Worker Interface

The crowd workers are given an interface that lets them view the video stream, listen to the most recent question, select from a set of camera adjustments, and submit answers to the user (Figure 4). Workers can

also capture screen shots from the video stream with a single key press, allowing them to review images later.

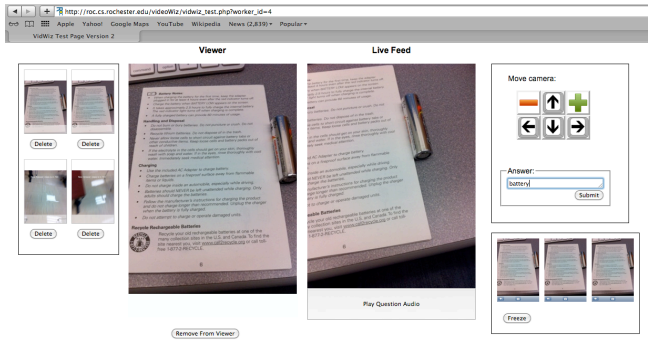

Figure 4. The worker interface. The streaming video with audio control is displayed in the center right. The worker can capture the three most recent frames (bottom right), save them as thumbnails (left side), and then view them in larger format in main view (center left). The worker can send direction and zoom information using the buttons on the top right, and answer queries using the answer box below.

Input Mediator

Legion uses input mediators to merge worker input into a single control stream in real-time. Legion:View implements an input mediator that merges the camera adjustment feedback as they arrive by taking a vote over short time intervals, or *epochs*, and forwarding the most agreed upon response to users. Answers to audio questions will not be aggregated at this stage because convergence to a single response is difficult to accomplish in real-time in such an open-ended domain.

The input mediator in Legion:View are different from those originally used in Legion because we do not want to view inactivity as being indicative of a lazy or missing worker, as Legion did. Instead, no input for a period of time may indicate that the worker agrees with the current camera angle or course of action.

Requiring the worker to explicitly submit feedback to maintain the current orientation would be cumbersome since it is unnatural, and distracting to the answering task. Instead, we assign a default null action to any worker not providing input during a given time window.

Experimental Setup

We conducted an informal evaluation of the Legion:View worker interface using five students as crowd workers. Workers were asked to help a user perform the three tasks described below.

In the first test, we explore questions regarding a single object. The user holds a canned good that could be one of three vegetables: green beans, corn, or peas. The user frames the can correctly, but begins with the can's label facing the wrong direction. The user then gets feedback from the crowd until the orientation is correct and the question can be answered.

In the second test, the crowd answers questions about a visual environment. The user attempts to find a sign regarding a relocated course in a large room containing obstacles. The user begins facing a blank wall, and asks the crowd if there is any indication as to why no one is in the room. The crowd can then guide the user to a sign located on the far wall, and read it to them.

The third test asks the user to find a specific TV dinner in a freezer and then get the cooking instructions for it. The user first asks a series of simple questions about

the type of the meal, then once they locate the correct one; ask what the cooking instructions for it are.

Preliminary Results

We found that with Legion:View, the user was able to accomplish each of the tasks in approximately 2-3 minutes, compared to over 7 minutes when using VizWiz. Workers suggested many modifications to the interface such as making it easier to capture screenshots and adding video playback. Workers also lacked information about the user's current status. We plan to address this by notifying workers when the user is recording a new question, or when the user received an acceptable answer.

Future Work

Legion:View effectively circumvents the need to resubmit the questions in different sessions, which is the main inefficiency encountered in VizWiz. However, it does not yet provide the same assistance available if a single person were co-located with the user, answering questions as they arose.

We plan to achieve this type of assistance by enabling the user to have a *conversation with the crowd*. This requires improving the feedback from the crowd to allow them to respond to user queries as if they were a single, reliable, worker. For example, a blind user could ask where a nearby Starbucks is, then using Legion:View, locate the store in the distance, and walk across a plaza to it without needing co-located assistance. To do this, the crowd must remember the goals of a task since users will most likely ask a series of follow-up questions, and provide consistent, reliable advice throughout. Enabling interactions with the crowd that can span longer periods of time, and encompass multiple goals furthers the idea of true collaboration with the crowd. This idea can be extended to crowd assistants that all users can benefit from.

References

[1] Bernstein, M., Brandt, J., Miller, R. and Karger, D. (2011). *Crowds in Two Seconds: Enabling Realtime Crowd-Powered Interfaces*. In Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2011). Santa Barbra, CA. p33-42.

[2] Bigham, J.P., Jayant, C., Ji, H., Little, G., Miller, A., Miller, R.C., Miller, R., Tatarowicz, A., White, B., White, S. and Yeh, T. (2010). *VizWiz: nearly real-time answers to visual questions*. In Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2010). New York, NY. p333-342.

[3] 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 ACM Symposium on User Interface Software and Technology (UIST 2011). Santa Barbra, CA. p23-32.

[4] Little, G., Chilton, L.B., Goldman, M. and Miller, R.C. (2010). *TurKit: Human Computation Algorithms on Mechanical Turk*. In Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2010). New York, NY. p333-342.

[5] White, S., Ji, H. and Bigham, J. (2010). *EasySnap: Real-time Audio Feedback for Blind Photography*. In the Adjunct Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2010). New York, NY. p409-410.