VizWiz::LocateIt - Enabling Blind People to Locate Objects in Their Environment

Jeffrey P. Bigham[†], Chandrika Jayant[∮], Andrew Miller[γ], Brandyn White[‡], and Tom Yeh[‡]

[†]University of Rochester CS
Rochester, NY 14627 USA
`jbigham@cs.rochester.edu`

[‡]University of Maryland CS
College Park, MD 20742 USA
`bwhite, tomyeh@umiacs.umd.edu`

[γ]University of Central
Florida CS
Orlando, FL 32816 USA
`amiller@ucf.edu`

[∮]University of Washington CSE
Seattle, WA 98195 USA
`cjayant@cs.washington.edu`

Abstract

Blind people face a number of challenges when interacting with their environments because so much information is encoded visually. Text is pervasively used to label objects, colors carry special significance, and items can easily become lost in surroundings that cannot be quickly scanned. Many tools seek to help blind people solve these problems by enabling them to query for additional information, such as color or text shown on the object. In this paper we argue that many useful problems may be better solved by direclty modeling them as search problems, and present a solution called VizWiz::LocateIt *that directly supports this type of interaction. VizWiz::LocateIt enables blind people to take a picture and ask for assistance in finding a specific object. The request is first forwarded to remote workers who outline the object, enabling efficient and accurate automatic computer vision to guide users interactively from their existing cellphones. A two-stage algorithm is presented that uses this information to guide users to the appropriate object interactively from their phone.*

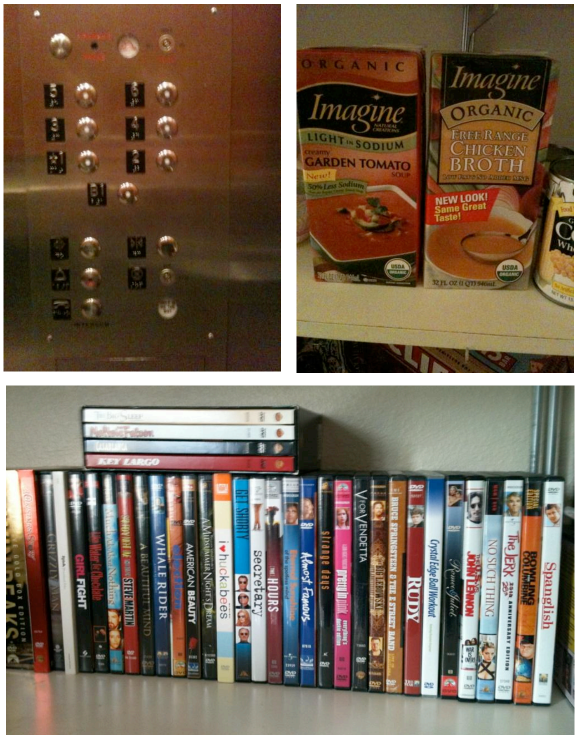


Figure 1. Many real world problems that blind people face can be framed as visual search problems. For instance, finding the "cancel call" button on an elevator otherwise labeled with Braille, finding the right soup from two tactually identical boxes, or finding a favorite DVD from among many on a shelf.

1. Introduction and Motivation

Blind people face challenges when interacting with their environments because so much information is encoded visually. For example, to find a specific object, a blind person may use various applications which attempt to translate encoded visual information such as text and color, which can provide verification but does not assist in finding a starting point for the search (Figure 1). Items can easily become lost in surroundings that cannot be quickly scanned.

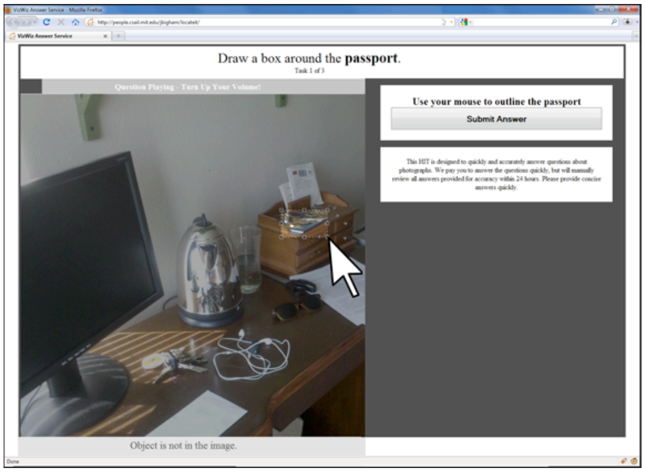
Existing technology reveals information that blind people may be missing, but many of the problems that blind people face may be better framed as search problems. The question "What does this can say?" is usually asked as part of the process of finding the desired can. A blind person may serially ask for the label from one can after another until they hear the answer they want (e.g., corn). Similarly, "What color is this button?" may be asked because technology does not yet directly support the underlying information need: "Which button do I press to record?" In this paper, we present *VizWiz::LocateIt*, a system that can help blind people find arbitrary items in their environments.

VizWiz::LocateIt has three logical components. The first is an iPhone application that lets blind users take a picture and ask for an item that they would like help finding. Both the picture and the audio from the requested item get sent to the second component: a remote server that puts the picture and question on a web page and recruits human workers to outline the object in the picture (Figure 2). The server component interfaces with an existing service called Mechanical Turk provided by Amazon Inc [1], and is specially designed to get answers back quickly (generally in less than a minute). The third component is an interface again built into the iPhone application that uses relatively simple computer vision running in realtime on the iPhone along with the description of the region selected by the remote worker to help interactively guide users to the desired object. VizWiz effectively outsources the as of yet unsolved parts of computer vision to remote workers and uses highly accurate recognition on the phone to automatically and interactively guide human users.

Although not all of the information needs of blind people can be framed as a search problem and solved by VizWiz, its unique architecture and clever reframing of problems means that many can. Because humans are identifying the objects with the photographs, users can ask questions that require both visual acumen and intelligence. For instance, in Figure 2, humans need to know that passports are generally little navy booklets and are likely to be lying on a desk. Once they have outlined this example of a passport, many techniques can be used to locate that example again in the same lighting and orientation.

1.1. Motivating Scenario

Consider the following motivating scenario. Julie, a blind woman, goes to her local grocery store to pick up, among other things, cereal for breakfast. From prior visits, she's learned where the cereal aisle is. Once in the aisle, she wants to find Frosted Flakes, her favorite cereal, but does not know exactly where it is located because its precise location often changes as new products are added. She remembers that cereal is on the left side and takes a picture on her cell phone. She says "Frosted Flakes?" to VizWiz::LocateIt and moments later, her phone vibrates to tell her that it is ready to guide her to the appropriate box. The application first guides her toward the direction of the correct cereal box using audible clicks that increase in frequency when the phone is pointed in the right direction. Once she gets close to the shelf, the phone switches to using a similar audible clicking feedback to tell her when her phone is over the right cereal box. Julie picks up the cereal, puts it into her cart, and continues shopping.



This HIT is designed to quickly and accurately answer questions about photographs. We pay you to answer the questions quickly, but will manually review all answers provided for accuracy within 24 hours. Please provide concise answers quickly.

Figure 2. The interface for remote workers to identify and outline objects. Because real people are tasked with finding the object, the object specification can rely on both their intelligence and contextual knowledge. In this example, the workers needs to know what a passport is and about what size it would be. The worker may also benefit from knowing that passports in the United States are a deep navy color.

1.2. Contributions

This paper offers the following contributions:

- We motivate and present VizWiz::LocateIt, an accessible mobile system that lets blind people take a picture and request an item that they would like to find, and interactively helps them find that item.
- We show how human-powered services, such as Mechanical Turk, can be used in concert with automatic computer vision to enable advanced computer vision in realtime.
- We present a two-stage algorithm that can run on the iPhone that uses both accelerometer and realtime computer vision to help users locate a desired object.

2. Related Work

Work related to VizWiz generally falls into the following two categories: (i) the mobile devices that blind people already carry, and (ii) work on leveraging humans remotely as part of computational processes. We will provide a brief survey of these areas to provide a landscape upon which cheap, mobile applications like VizWiz have the potential to be very useful.

2.1. Mobile Devices for Blind People

Most mainstream cellphones are not accessible to blind people. Smartphones often provide the best access through separate screen reading software like Mobile Speak Pocket (MSP) [13]. Though somewhat popular, the uptake of such software among blind users has been limited due to its high price (an additional $500 after the cost of the phone). Google's Android platform and the Apple iPhone 3GS now include free screen readers [20, 5]. The iPhone has proven particularly popular among blind users, which motivated us to concentrate on it for VizWiz. Apple's stringent controls on the applications available on its online store and tighter integration of its screen reader (VoiceOver) with the operating system has resulted in a large number of accessible applications. Touchscreen devices like the iPhone were once assumed to be inaccessible to blind users, but well-designed, multitouch interfaces leverage the spatial layout of the screen and can even be preferred by blind people [29].

Applications for general-purpose smartphones are beginning to replace special-purpose devices, but blind people still carry devices such as GPS-powered navigation aids, barcode readers, light detectors, color identifiers, and compasses [30]. Some accessible applications that use the camera on existing phones include currency-reading applications and color identifiers [33, 19]. Because VizWiz connects users to real people, it can potentially answer all of the questions answerable by many costly special-purpose applications and devices.

Talking OCR Devices: Of particular interest to blind people is the ability to read text, which pervasively labels objects and provides information. Both the kNFBReader [10] and the Intel Reader [7] are talking mobile OCR tools. Human computation could have an advantage over tools such as these because humans can still read more text written in more variations than can automatic approaches. When OCR works, however, it is faster and can be used to transcribe large text passages. Human workers are slower but this may be partially offset by their ability to take instructions that require intelligence. For example, an OCR program can read an entire menu, but cannot be asked, "What is the price of the cheapest salad?"

Other Automatic Computer Vision for Mobile Devices: Several research projects and products expose automatic computer vision on mobile devices. Photo-based Question Answering enables users to ask questions that reference an included photograph, and tackles the very difficult problems of automatic computer vision and question answering [41]. Google Goggles enables users to take a picture and returns related search results based on object recognition and OCR [6]. Although these projects have made compelling progress, the state-of-the-art in automatic approaches is still far from being able to answer arbitrary questions about photographs and video. Although not on mobile phones, GroZi is another project using vision; it is a grocery shopping assistant for the visually impaired which uses Wiimotes, gloves with LED lights, a laptop, a wireless camera, and spacialized audio and feedback cues [25].

Interfacing with Remote Services: Most mobile tools are implemented solely as local software, but more applications are starting to use remote resources. For instance, TextScout [17] provides an accessible OCR interface, and Talking Points delivers contextually-relevant navigation information in urban settings [26]. VizWiz also sends questions off for remote processing, and these services suggest that people are becoming familiar with outsourcing questions to remote services.

2.2. Human-Powered Services

VizWiz builds from prior work in using human computation to improve accessibility [22]. The ESP Game was originally motivated (in part) by the desire to provide descriptions of web images for blind people [40]. The Social Accessibility project connects blind web users who experience web accessibility problems to volunteers who can help resolve them, but 75% of requests remain unsolved after a day [39]. Solona started as a CAPTCHA solving service, and now lets registered blind people submit images for description [15]. According to its website, "Users normally receive a response within 30 minutes." VizWiz's nearly real-time approach could be applied to other problems in the accessibility space, including improving web accessibility. Locating objects, as in VizWiz::LocateIt, is one such application.

Prior work has explored how people ask and answer questions on their online social networks [35]. While answers were often observed to come back within a few minutes, response time varied quite a lot. The "Social Search Engine" Aardvark adds explicit support for asking questions to your social network, but advertises that answers come back “within a few minutes.” [37]

Mechanical Turk has made outsourcing small paid jobs practical [1]. Mechanical Turk has been used for a wide variety of tasks, including gathering data for user studies [31], labeling image data sets used in Computer Vision research [38], and determining political sentiments in blog snippets [28]. The Amazon Remembers feature of its iPhone application lets users take pictures of objects, and later emails similar products that Amazon sells [2]. It is widely suspected that Amazon outsources some of these questions to Mechanical Turk. The TurKit library enables programmers to easily employ multiple turk workers using common programming paradigms [32].

2.3. Connecting Remote Workers to Mobile Devices

Some human-powered services provide an expectation of latency. ChaCha and KGB employees answer questions asked via the phone or by text message in just a few minutes [4, 8]. Other common remote services include relay services for deaf and hard of hearing people (which requires trained employees) [36], and the retroactive nearly real-time audio captioning by dedicated workers in Scribe4Me [14]. A user study of Scribe4Me found that participants felt waiting the required 3-5 minutes was too long because it “leaves one as an observer rather than an active participant.”

Existing Use of Photos and Video for Assistance: Several of the blind consultants whom we interviewed mentioned using digital cameras and email to informally consult sighted friends or family in particularly frustrating or important situations (e.g., checking one’s appearance before a job interview). Back in 1992, remote reading services for the blind were proposed using low cost fax equipment and sighted remote readers. Compressed video technology allowed very low framerate, high-resolution video transmission over ordinary telephone lines [23]. oMoby is an iPhone application similar to Google Googles, but instead of an automated database lookup, human computation is used. The Soylent Grid CAPTCHA-based image labeling system requires remote human annotation for CAPTCHA images then included in a searchable database [24].

LookTel is a soon-to-be-released talking mobile application that can connect blind people to friends and family members via a live video feed [11]. Although future versions of VizWiz may similarly employ video, we chose to focus on photos for two reasons. First, mobile streaming is not possible in much of the world because of slow connections. Even in areas with 3G coverage, our experience has been that the resolution and reliability of existing video services like UStream [18] and knocking [9] is too low for many of the questions important to blind people. Second, using video removes the abstraction between user and provider that VizWiz currently provides. With photos, questions can be asked quickly, workers can be employed for short amounts of time, and multiple redundant answers can be returned.

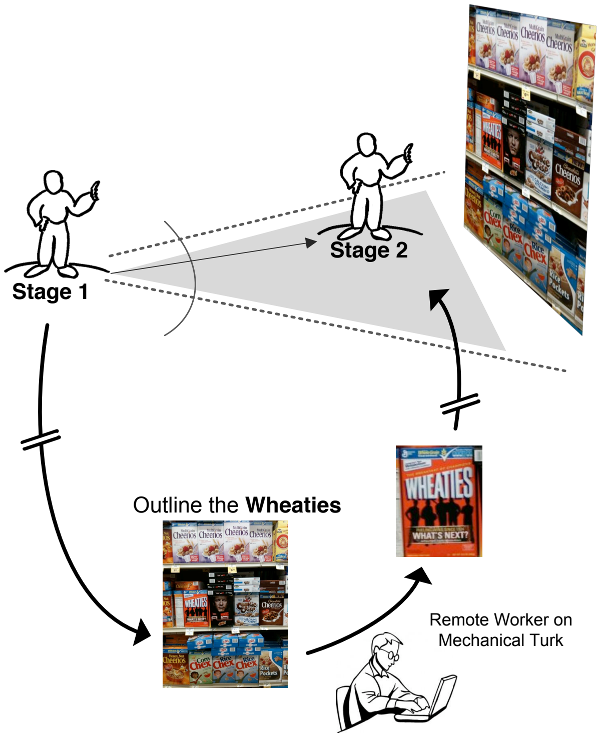


Figure 3. To use VizWiz::LocateIt users first take a picture of an area in which they believe the desired item is located, and this is sent to remote workers on Mechanical Turk who outline the item in the photograph. During this stage, VizWiz uses the accelerometer and compass to direct the user in the right direction. Once users are closer to the objects, VizWiz switches to using a color histogram approach to help users narrow in on a specific item.

3. VizWiz::LocateIt

Here we present our work on VizWiz::LocateIt, a prototype system that combines remote human work with automatic computer vision to help blind people locate arbitrary items in their environments (Figure 3). To support object localization we created the following two components: a web interface to let remote workers outline objects, and the VizWiz::LocateIt mobile interface consisting of the *Sensor* (zoom and filter) and *Sonification* modules.

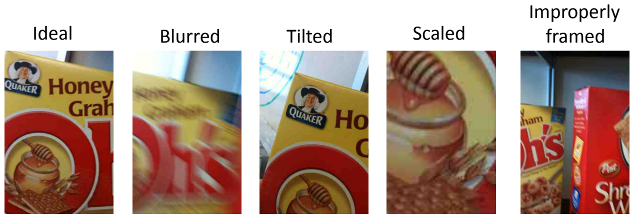


Figure 4. Frames captured by blind users during the Filter stage exemplifying computer vision challenges.

3.0.1 Sensor Module

In the zoom stage (stage 1), the Sensor module estimates how much the user needs to turn in the direction of the target object (left, right, up, or down). It first uses the object's image location (u, v) indicated by the remote worker to calculate the 3D position (x, y, z) of the object relative to the user's current position. The construction of such a mapping function typically requires knowledge of a set of camera parameters that are extrinsic (e.g., camera orientation) and intrinsic (e.g., focal length, lense distortion). We estimate intrinsic parameters by camera calibration once per device and compute extrinsic camera parameters directly from the device's built-in compass (heading angle) and accelerometer (gravity vector) once per camera movement. Note that extrinsic parameters change whenever the camera moves whereas the intrinsic parameters stay constant and only need to be computed once per device. Once the 3D position of the target object is known, we can also compute the 3D position (x', y', z') toward which the camera is currently pointing using a similar procedure. The angular cosine distance between the two resulting vectors indicates how much the user needs to turn. This difference is measured as an angular cosine distance, and is passed to the Sonification module to generate appropriate audio cues.

In the filter stage (stage 2), the Sensor module uses computer vision to determine how close the current camera view is to the object outlined by the remote worker. This task is non-trivial because input images are often blurred, tilted, varied in scale, and improperly framed due to blind users being unable to see the image they are capturing (Figure 4). We implemented two visual matching schemes based on invariant local speeded up robust features (SURF) and color histograms respectively. In the first scheme, a homography between each captured frame and the overview image is computed based on the correspondences of SURF features [21]. Based on the homography, the object's location (u, v) in the overview image is mapped to a location (u', v') in the current frame. The distance between (u', v') and the center of the current frame is computed. The smaller the distance, the more "centered" the target object is in the current frame.

We found that local features were quite susceptible to problems related to lighting and blur and so we also created a visual matching scheme based on color histograms that is more robust to these problems. A color histogram h of the object outlined by the remote helper is computed. Then we divide each input frame up into N blocks and compute a color histogram h_i for each block i, which improves robustness to improper framing. We then compare the computer histogram to the target color histogram h, and calculate a distance measure d_i using L1. The total distance D is the minimum distance of contiguous subsets of the N individual block differences. The smaller the D, the more "similar" the object in the current frame is to the target object. To provide users with a consistent sense of distance, the distance is normalized by the smallest D observed during the k most recent interactions with the system. The normalized distance is then passed to the Sonification module to generate audible feedback.

3.0.2 Sonification Module

The Sonification Module inputs computed distances from the *Sensor* component and generates audio feedback to inform the user how close she is to the goal. In the zoom stage, the goal is a particular direction to "zoom in" (i.e., walk closer to). In the filter stage, the goal is a particular object for which we have implemented three different sonification options. The first two are based on the pitch of a tone and frequency of clicking, respectively. The third scheme is a voice that announces a number between one and four, which maps to how close the user is to the goal.

3.1. User Study

We conducted a preliminary within-subjects lab-based user study of VizWiz::LocateIt in which we asked participants to find a desired cereal box using (i) LocateIt (color-histogram version) and (ii) a commercially-available barcode scanner with a talking interface (Figure 5(b)). We prepared three shelves each with five cereal boxes (Figure 5(a)). All cereal boxes were unique and unopened, and they reflected a diversity of sizes, colors, and weights. We recruited seven participants (two females, four totally blind, three low vision) aged 48 years on average (SD=8.7). Only one participant owned an iPhone, four others had experience with an iPhone, and five had previously taken photographs on inaccessible cell phone cameras.

Participants were trained using both methods (approximately 10 minutes). Participants then completed three timed trials using each method. For the LocateIt

(a) (b)

Figure 5. (a) Our mock grocery store shelf stocked with 15 different cereal boxes. (b) The ID Mate II talking barcode scanner from Envision America.

trials, the zoom and filter stages were timed separately. For the purposes of this study, researchers answered requests in order to concentrate on the user interaction with the application, although our experience has been that workers on Mechanical Turk can quickly answer questions requiring them to outline objects. For all six trials, participants started 10 feet in front of the shelves, and boxes were randomized after each trial.

3.2. Results

Participants used LocateIt and the barcode scanner very differently. LocateIt enabled users to zero in on the right part of the shelf quickly like a visual scan, whereas the barcode scanner required each box to be serially scanned. The time required for each tool was similar, although LocateIt produced many more errors. LocateIt took an average of 92.2 seconds (SD=37.7) whereas the barcode scanner took an average of 85.7 seconds (SD=55.0), although the researchers answered questions in approximately 10 seconds as compared to the almost 30 seconds that we would expect workers on Mechanical Turk to require. Participants found the correct box in all cases using the barcode scanner (since it clearly spoke the name of each box), but found the correct box using LocateIt on their first try in 12 of 21 cases and in their second try in 7 out of 21 cases.

Interestingly, the zoom stage of LocateIt correctly led users to the correct area of the wall in only 30.7 seconds on average (SD=15.9). We informally tried using the first stage of LocateIt to direct users to approximately the right part of the wall, and then had them switch to the barcode scanner for identification. This ended up being slower, primarily because of how cumbersome it was to switch between devices. In future work, we will explore how to better integrate both human-powered and automatic services together. None of the participants wanted to carry around a bulky barcode reader, or even a smaller portable one, because of their high prices and inconvenience. All participants said, however, that they would use an accessible barcode reader on their phone if one was available.

In summary, our first LocateIt prototype was comparable to barcode scanner in terms of task completion time but produced more errors. However, LocateIt's advantages include being useful for general visual search problems, not requiring objects to be tagged in advance, and potentially scaling much better. From the observations and results we draw three lessons related to cues and orientation that will inform this work as we go forward.

First, participants used many cues other than the audio information from LocateIt and the barcode scanner: shaking the boxes, having prior knowledge of box size, or using colors (low vision participants).

Second, all participants liked the clicks used in the zoom stage of our application. For the second stage, many alternatives were brought up, including vibration, pitch, more familiar sounds (e.g., chirps and cuckoos crosswalk signal sounds), verbal instructions, or a combination of output methods, many of which are used in other applications for blind people [27, 16, 12].

Finally, three participants had difficulty walking in a straight line from their beginning position to the shelf once a direction was indicated, desiring a more continuous clicking noise to keep them on track. Most of the participants had trouble keeping the phone perpendicular to the ground. In the up-close stage, all fully blind participants had trouble judging how far back from each cereal box they should hold the phone in order to frame the cereal boxes.

To design with human values in mind, we asked participants how comfortable they would feel using an application such as LocateIt in public. All participants said they would likely be comfortable with such an application if it worked in nearly real-time, but wondered about the reaction of bystanders. In practice, LocateIt feedback could be delivered by a headset, although vibrational output might be preferred as to not interfere with the user's primary sense.

4. Discussion

We have motivated modeling the problems that blind people face in their environments as visual search problems, and proposed a novel two-stage algorithm that uses on-board sensors and input from remote workers. This approach lets blind people start benefiting from the technology before automatic computer vision technology can achieve all of the necessary functionality and we can start building a corpus of types of tasks for blind people to help motivate future research.

VizWiz::LocateIt combines automatic computer vision with human-powered vision, offloading the vision not yet possible to do automatically to humans, while retaining the benefit of quick response times offered by automatic services. This allowed us to prototype an interaction that would not have been possible otherwise and easily begin a participatory design process to see if this type of interaction is desirable or even useful. This project highlights the potential of nearly real-time human computation to influence early designs.

5. Conclusion and Future Work

We have presented VizWiz::LocateIt, a mobile system that enables blind people to locate objects in their environment using a unique combination of remote human computation and local automatic computer vision. This project represents a novel change in how assistive technology works, directly inspired by how blind people overcome many accessibility shortcomings today – ask a sighted person. Our approach can make this easier while keeping the users in control.

As we move forward, we plan to directly engage with blind and low-vision people to help test our approach after another iteration of design informed by our current work. We will start with formal lab studies to guage its effectiveness and then release the application to many users to get results from the field. Engaging the user population in this way is of equal importance to the success of the project as getting the technology right. As just one example of why this is vital, users may need to collectively help one another imagine uses for VizWiz::LocateIt and how to restructure the problems they face as search problems.

In future work, we plan to study in more depth how to best enable blind people to take pictures appropriate to the questions they seek to ask- not just for object location, but for general questions about the environment as well. Taking good photographs is of general interest, as often blind people want to share photographs with friends and family, just like everyone else. Taking pictures, and in particular framing and focusing photographs can be difficult. This, however, has not stopped blind photographers from taking and sharing photographs [3]. We might be able to provide software supports to help them take pictures more easily. We also plan to explore the limits of what can be done to receive answers even more quickly from remote human computation. For example, active techniques might attract turkers to our Human Intelligence Tasks (HITs) before they are needed and keep them busy with other work until a realtime task comes in that could be inserted at the front of the queue.

VizWiz presents an excellent platform for new research. We believe that low-cost, readily available human computation can be applied to many problems face by disabled users on the go.

6. Acknowledgements

We would like to thank Jennifer Asuscion, Rob Miller, Greg Little, and Aubrey Tatarowicz.

References

[1] Amazon mechanical turk. http://www.mturk.com/.

[2] Amazon remembers. http://www.amazon.com/gp/. 2010.

[3] Blind with camera: Changing lives with photography, 2009. http://blindwithcamera.org/.

[4] Chacha, 2010. http://www.chacha.com/.

[5] Eyes-free shell for android, 2009. http://code.google.com/p/eyes-free/.

[6] Google goggles, 2010. http://www.google.com/mobile/goggles/.

[7] Intel reader. http://www.intel.com/healthcare/reader/. 2009.

[8] Kgb, 2010. http://www.kgb.com

[9] Knocking live video. ustream, 2010. http://knockinglive.com/.

[10] Knfb reader. knfb Reading Technology, Inc., 2008. http://www.knfbreader.com/.

[11] Looktel, 2010. http://www.looktel.com

[12] Miniguide us. http://www.gdp-research.com.au/minig_4.htm/.

[13] Mobile speak screen readers. Code Factory, 2008. http://www.codefactory.es/en/products.asp?id=16.

[14] Scribe4me: Evaluating a mobile sound transcription tool for the deaf. In *UbiComp 2006: Ubiquitous Computing*, 2006.

[15] Solona, 2010. http://www.solona.net/.

[16] Talking signs. http://www.talkingsigns.com/, 2008.

[17] Testscout- your mobile reader, 2010. http://www.textscout.eu/en/.

[18] UStream ustream, 2010. http://www.ustream.tv/.

[19] Voice for android, 2010. www.seeingwithsound.com/android.htm.

[20] Voiceover: Macintosh os x, 2007. http://www.apple.com/accessibility/voiceover/.

[21] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Proceedings of Computer Vision and Image Understanding (CVIU 2008)*, volume 110, pages 346–359, 2008.

[22] J. P. Bigham, C. Jayant, H. Ji, G. Little, A. Miller, R. C. Miller, A. Tatarowicz, B. White, S. White, and T. Yeh. Vizwiz: Nearly realtime answers to visual questions. In *Proceedings of the 7th International Cross-Disciplinary Conference on Web Accessibility (W4A 2010) - Submitted*. Available at http://www.cs.rochester.edu/u/jbigham/papers/challengevizwiz/vizwiz.pdf.

[23] J. Brabyn, W. Mendec, W. Gerrey. Remote Reading Systems For The Blind: A Potential Application Of Virtual Presence. In Engineering in Medicine and Biology Society, Vol.14, 1992.

[24] P. Faymonville, K. Wang, J. Miller, and S. Belongie. CAPTCHA-based Image Labeling on the Soylent Grid HCOMP '09: Proceedings of the ACM SIGKDD Workshop on Human Computation.

[25] G. Sze-en Foo. Summary 2009 Grocery Shopping for the Blind/Visually Impaired. Client: National Federation of the Blind

[26] S. Gifford, J. Knox, J. James, and A. Prakash. Introduction to the talking points project. In *Assets '06: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*, pages 271–272, New York, NY, USA, 2006. ACM.

[27] D. Hong, S. Kimmel, R. Boehling, N. Camoriano, W. Cardwell, G. Jannaman, A. Purcell, D. Ross, and E. Russel. Development of a semi-autonomous vehicle operable by the visually-impaired. *IEEE Int Conf Multisensor Fusion Integr Intell Syst IEEE Intl. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pages 539–544, 2008.

[28] Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: a study of annotation selection criteria. In *HLT '09: Proc. of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pages 27–35, Morristown, NJ, 2009.

[29] S. K. Kane, J. P. Bigham, and J. O. Wobbrock. Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques. In *Assets '08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, pages 73–80, New York, NY, USA, 2008. ACM.

[30] Shaun K. Kane, Chandrika Jayant, Jacob O. Wobbrock, and Richard E. Ladner. Freedom to roam: a study of mobile device adoption and accessibility for people with visual and motor disabilities. In *Assets '09: Proc. of the 11th Intl. ACM SIGACCESS Conf. on Computers and accessibility*, pages 115–122, New York, NY, 2009.

[31] Aniket Kittur, Ed H. Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI 2008)*, pages 453–456, 2008.

[32] Greg Little, Lydia Chilton, Robert C. Miller, and Max Goldman. Turkit: Tools for iterative tasks on mechanical turk. http://glittle.org/Papers/TurKit.pdf, 2009.

[33] X. Liu. A camera phone based currency reader for the visually impaired. In *Assets '08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, pages 305–306, New York, NY, USA, 2008. ACM.

[34] Tara Matthews, Janette Fong, F. W.-L. Ho-Ching, and Jennifer Mankoff. Evaluating visualizations of non-speech sounds for the deaf. *Behavior and Information Technology*, 25(4):333–351, 2006.

[35] Meredith Ringel Morris, Jaime Teevan, and Katrina Panovich. What do people ask their social networks, and why? a survey study of status message q&a behavior. In *Proc. of the conf. on Human Factors in computing systems (CHI 2010)*, To Appear.

[36] M. Power. Deaf people communicating via sms, tty, relay service, fax, and computers in australia. In *In Journal of Deaf Studies and Deaf Education*, volume 12, 2006.

[37] Hadi Bargi Rangin. Anatomy of a large-scale social search engine. In *Proc. of the World Wide Web Conf. (WWW 2010)*, To Appear. Raleigh, NC, 2010.

[38] A. Sorokin and David Forsyth. Utility data annotation with amazon mechanical turk. *CVPRW '08. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops.*, pages 1–8, June 2008.

[39] H. Takagi, S. Kawanaka, M. Kobayashi, T. Itoh, and C. Asakawa. Social accessibility: achieving accessibility through collaborative metadata authoring. In *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility (ASSETS 2008)*, pages 193–200, Halifax, Nova Scotia, Canada, 2008.

[40] L. von Ahn and L. Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI '04)*, April 2004.

[41] Tom Yeh, John J. Lee, and Trevor Darrell. Photo-based question answering. In *Proc. of the Intl. Conf. on Multimedia (MM '08)*, pages 389–398, Vancouver, BC, Canada, 2008.