2018
THE JOURNAL ON TECHNOLOGY AND PERSONS WITH DISABILITIES

Panoramik: Finding and Locating Objects via Panoramic Camera Techniques

Carl Colena, Nayancy Kashyap, Diana Yau, Zhigang Zhu and Celina Cavalluzzi

Department of Computer Science, The City College of New York

Goodwill Industries of Greater New York and Northern New Jersey

carl.colena@gmail.com, nayancyk@gmail.com, dyau95@gmail.com,

zhu@cs.ccny.cuny.edu, ccavalluzzi@goodwillny.org

Abstract

One of the areas that poses a challenge for individuals with significantly low sight and blindness is the dynamic nature of their work surroundings. Visually impaired individuals tend to prefer a very well organized and static environment because it allows them to know exactly where everything is at any given time. However, in a dynamic work environment, there are fellow co-workers, colleagues, and customers that may move or misplace key items and objects that visually impaired individuals may need in order to carry out their job. Because of these issues, we propose to build an affordable camera-based system to find and track key objects for visually impaired individuals. Our solution is to use a mobile panoramic imaging system that joins 360-degree surveillance with user-defined panoramic photos to create a self-aware environment. By tracking objects and item placement, the system will be able to assist visually impaired individuals locate objects using audio cues. For the software component, a phone application is developed that is Android friendly for users. The application consists of a classifier algorithm to locate objects and save the memory of these objects. The hardware component utilizes a 360 omnidirectional camera (or equivalent) and a smartphone.

Keywords

Blind and visually impaired; Object detection; Object localization; Panoramic vision

Introduction

There are 285 million visually impaired people in the world, and of them 39 million are considered totally blind; 37.7% of visually impaired people are actively employed in the workforce (WHO Facts, 2016), which means that there is a considerable barrier for visually impaired individuals to achieve and keep employment positions. One of the most prevalent challenges for working individuals with significantly low vision and blindness is the dynamic nature of their work surroundings (Beck, 2010). Visually impaired individuals tend to prefer a very well organized and static environment because it allows them to know exactly where everything is at any given time. This enables them to perform tasks quickly. However, in such dynamic work environments, there are fellow co-workers, colleagues, and/or customers that may move or misplace key items and objects that visually impaired individuals may need in order to carry out their job. When an object is misplaced, it makes their job harder to fulfill, reduces their productivity, increases their stress, and can make it much more difficult for visually impaired individuals to retain their jobs. Because of these issues, we propose to build an affordable and effective camera-based system to find and track key objects for visually impaired individuals. The system can be used in both a collaborative workplace, where a visually impaired user doesn't have the full control of the environment, or a private environment, where the user may be alone.

Problem Statements

There are not many applications in the market today targeted for the visually impaired or blind that perform spatial object detection and tracking. There are smartphone apps like iDentifi and TapTapSee that focus on specific image or scene detection from a still image from the smartphone (Coldewey, 2016; Taptapseeapp, 2016). These do not have any type of object

tracking ability, nor do they have any sort of spatial component to the detections. One approach that comes close is the O'Map approach, where an Object Map is generated from a sweeping panoramic image taken by a smart phone to generate spatial object detection regions (Alam, at al, 2015). While this presented a strong and viable solution to spatial object detection, it lacks the object-tracking dimension of the problem, and is therefore limited to single-point-in-time detection and location for objects. A handful of challenges come up with creating such a system. The first challenge is to make sure the system has a 360-degree view range of the user's environment. Generally mobile applications have a limited view range - often less than 180 degrees. It is also necessary that users are able to focus on a specified area and not just a general view provided by the system. This is necessary because a lot of times objects get misplaced behind a range of view so without a user customized range, it will be difficult to locate objects in areas not covered by the 360 degree system.

Another challenge is to add use verbal cues in the application so users don't have to rely on vision to interact with the system. To top it all, our system had to be an affordable solution so users from all backgrounds could have access and benefit from our project.

In any application or software system, processing speed plays an important role. Users will not use something that takes a long time to deliver an end result so we had to create a system that was as fast as possible without sacrificing object detection accuracy.

Rationale

Affordability is a crucial component of this project as 90% of visually impaired people live in low-income settings (WHO Facts, 2016), which makes expensive solutions impractical, such as 3D scanning and RF (radio frequency) tagging. By creating an inexpensive system that finds missing objects, we will be able to increase the number of people who can afford and use

the application.

Our solution, *Panoramik,* is to use a mobile panoramic imaging system that joins 360-degree surveillance with user-defined panoramic photos to create a self-aware environment, such as a room. Panoramik is created with hopes of making the visually impaired more independent and less stressed. To make sure visually impaired users do not have to rely on others to use our system, we have to build a simple, easy to use solution. By tracking objects and item placement, the system will be able to assist visually impaired individuals locate objects using audio cues. For the software component, a phone application is developed that is Android friendly for users. The application consists of a classifier algorithm to locate objects and save the memory of these objects. The hardware component utilizes a 360 omnidirectional camera (or equivalent, as of November 2017, a Ricoh Theta S 360 Camera we used cost about $200) and a smartphone (that users typically already own).

When a visually impaired person loses an object, they have to rely on others to find the misplaced object. This makes them dependent on another person. This system will help the visually impaired people become more independent. The application will also improve productivity of people. Searching for a missing object, especially for a visually impaired person takes time. By using the application, we can reduce the search time and increase the productivity. The system improves the organization ability of the visually impaired as well. When they know where each item is placed, they can track the items and reorganize them based on their preference. Finding an object increases stress in some people. In some cases if the missing item is valuable, it can increase the anxiety as well. The application will help reduce high stress and anxiety situation by helping the visually impaired find their misplaced object efficiently.

Discussion

Design

A high level overview of the client-server system is shown in Figure 1. The smartphone on the left represents the user-facing client. The 360 Camera client (Ricoh Theta S 360 Camera in Figure 1) is shown on the right. This client requires no interaction by users, as it will periodically take omnidirectional photos of the given indoor area at fixed intervals and send them to the Compute System. The 360 Camera client can either be activated manually by a supervisor or managing individual, or set automatically to activate once a day. After it is activated, the client is left to run continuously for the rest of the day. Having the 360 Camera Client start operating automatically allows for easy set up by the end-user (someone who is blind or has a visual impairment), which would give the user greater independence and autonomy. Since it is a 360-degree camera, there is less of a concern about aiming the camera correctly. The 360 camera client runs in the background, identifying objects, as well as monitoring the movements of these objects.

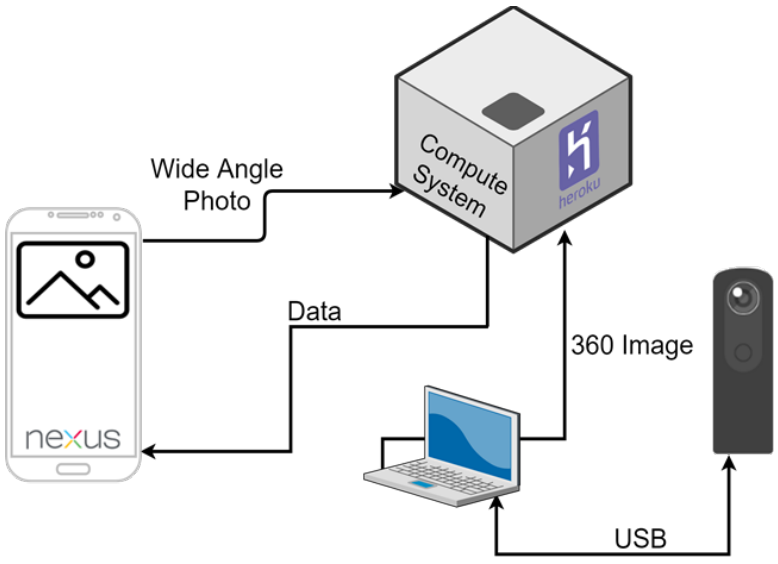

Fig. 1. A high-level overview of the Panoramik system.

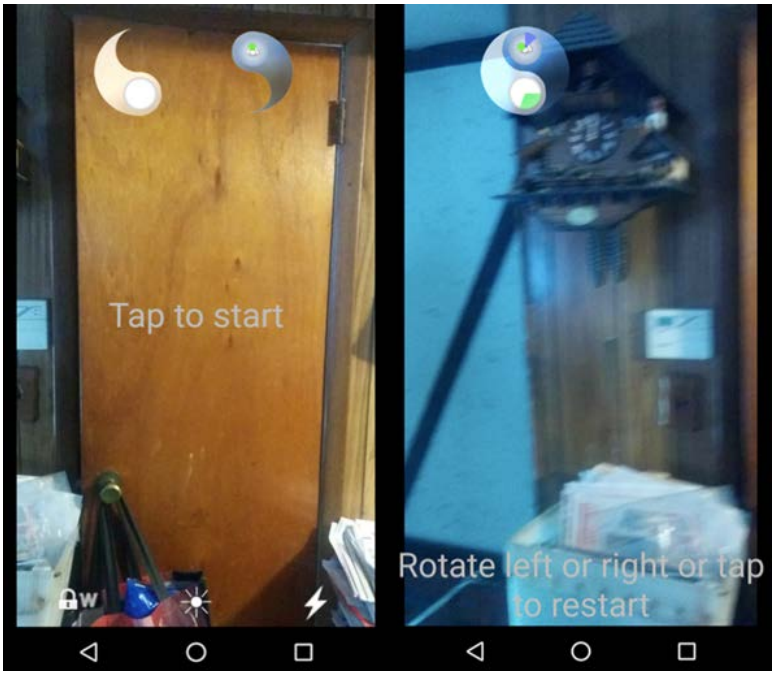

Fig. 2. Turning interface of taking panorama.

User Interaction and Interface

A standard case procedure of using the system is as follows: (1) User activates and launches the application on their phone. (2) Application greets user with voice-directed cues and menu options. (3) The user will speak the name of the item they are looking for to the phone. (4) The client will poll the compute system to see if the item had already been detected by the 360 Camera subsystem. (5a) When an item is found, the application will direct the user to the location of the item via voice cues. (5b) If item is not found, the user is presented with a wide-angle photo (Panoramic photo) interface (Figure 2), where the user is directed via voice cues to capture a wide-angle photo. (6) Upon completion of the wide-angle photo capture, the image is uploaded to the Compute System, where it is processed by the Object Detection subsystem. (7) Once a match is found, the server returns the location of the object to the user's smartphone client, which will navigate the user to the object.

Fig. 3. Capturing object placement and differences.

Motion Detection Algorithm

The motion detection algorithm takes 360 photos from the 360 Camera client input stream. The first image that this motion detection algorithm receives on startup becomes its *baseline* image. This means that every subsequent image will be compared against the baseline image. For particular environments, a reset threshold can be set and applied which will trigger a baseline image reset if a particular motion threshold is exceeded between the baseline image and the current image frame. Typically, the default reset for the baseline is done at the beginning of the work day and/or at the end of the work day, before the employees have arrived or after the employees have left. In Figure 3, we show an example of where objects have moved from their initial positions being detected by the motion detection algorithm. For each object that has moved we draw a bounding box around it using thresholding to get a binary black and white image. We then capture a cropped image of the object that had moved. These sub images are

then sent to the Object Detection Subsystem to be detected and given a tag name, which will be used to identify it when a user comes, asking to find an object.

Object Detection Algorithm

The object detection algorithm is the core component of how objects get identified in our system. This algorithm utilizes four commercial APIs for object recognition, identification, and classification. For this prototype, Cloudsight (Taptapseeapp, 2016) is primarily used because their API returns a single human-readable plaintext description of the object you give it. The primary disadvantage of Cloudsight is the speed taken to process and identify each object. A key issue is that APIs create bottlenecks due to usage restrictions, speed of processing, and pricing.

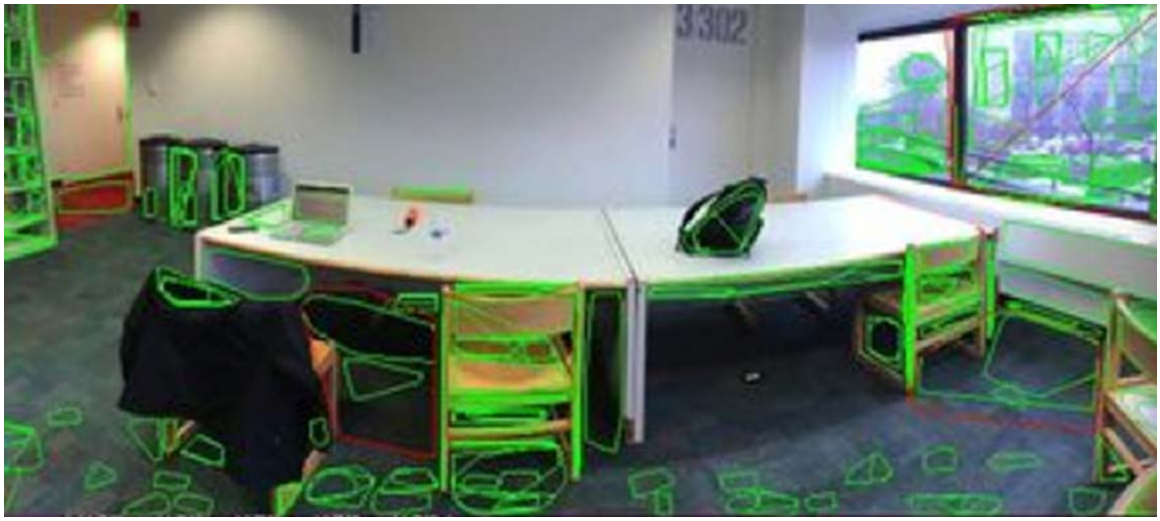

Fig. 4. Preliminary testing on MSER algorithm to fine-tune input parameters.

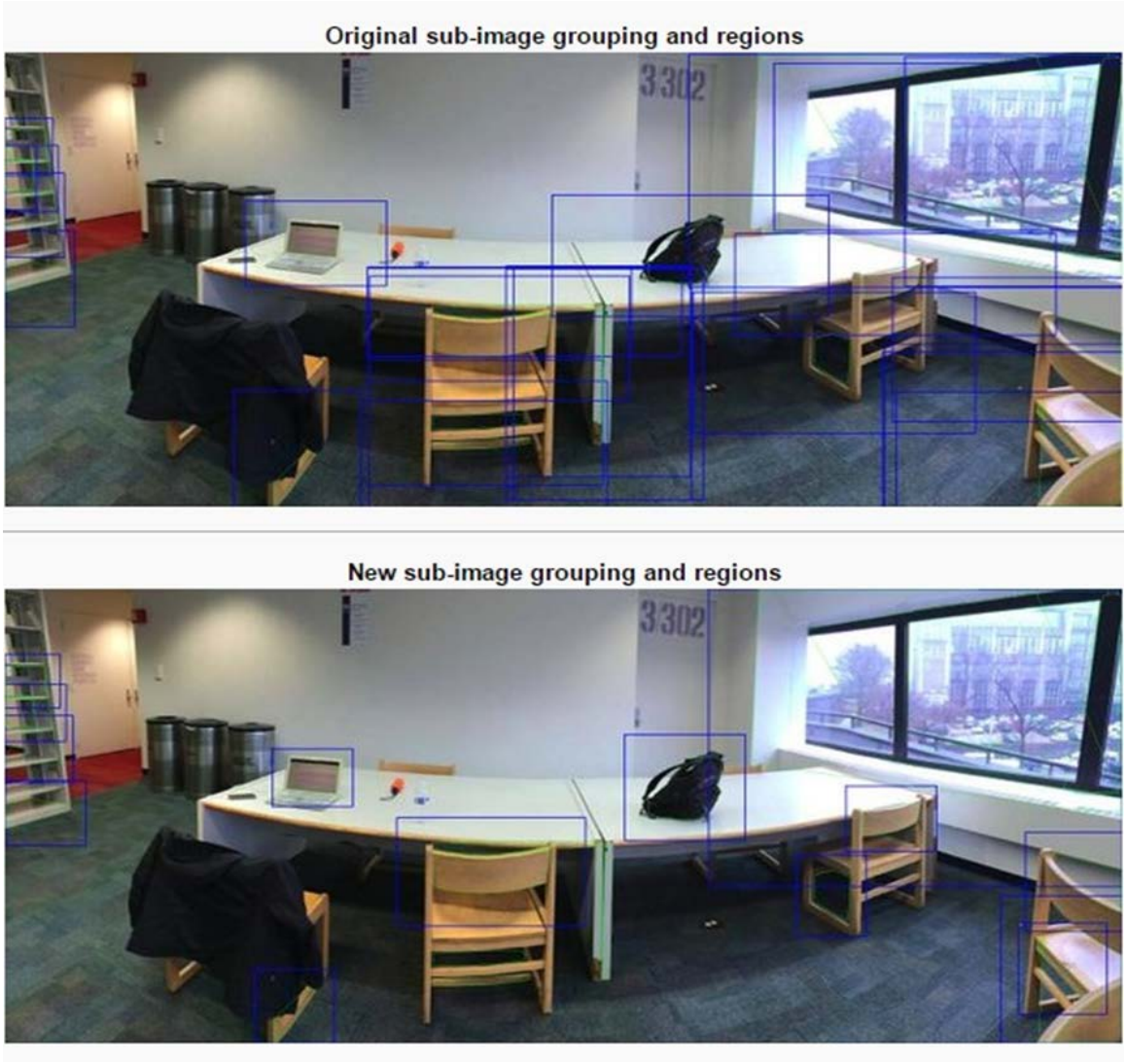

Fig. 5. Clustering effects, before and after.

In the final prototype, the image recognition will be done on-server using a templating system, which will be faster and more accurate to detect objects of interest. The algorithm gets its input from both the smartphone client and 360-camera client. For panorama images, they are pre-processed through a feature detector called MSER (Maximally Stable Extremal Regions) (Forssén & Lowe, 2007), which gives us regional data in the panorama image (Figure 4). From there, we take the region data, and begin preliminary filtering. We filter out regions which are too large (larger than 50% of the original image size) and too narrow (ratio of length to width or

vice versa less than 1:20). These regions are considered bad because the likelihood and fitness of these regions for detecting regions is very poor. After this point, we further filter our region data set through clustering, combining regions that share characteristics (e.g. overlapping regions of similar size and shape) so that we reduce the rate of redundant identifications. The clustering saves processing time and processing resources (Figure 5). At this point, regions are converted into sub-images, and join the 360-camera client input stream. These sub-images are then sent out one by one to the image recognition API's for recognition.

Implementation and Evaluation

A large bulk of time was spent on optimizing the MSER algorithm input parameters, as well as to resolve overlapping regions via clustering. Adjusting and fine-tuning the algorithm to run fast and flexibly was important for the object detection algorithm. It was also important to have it scalable, so that multiple users can interact with the system independently of each other.

We also worked on optimizing and combining multiple API's for object recognition. This included the 4 main APIs we looked into, which were Google Cloud Platform, IBM Bluemix, Microsoft Azure, and Cloudsight. An example of the results gathered from each of the APIs can be seen in Figure 6. Microsoft, Google, and IBM gave more generalized information such as tagging (and their respective confidence levels) and categories. We also began utilizing a motion detection algorithm for the 360 camera. OpenCV was used to cross-compare differences between two image frames. By this point our product was near finalization, and testing was needed to ensure that the system worked as intended.

We had the opportunity to present our work at the CREATE Symposium during the end of April at the Legislative Office Building in Albany, New York to many curious onsite onlookers. Many viewers were interested with how the omnidirectional 360-degree camera

worked together with the smartphone panoramic app.

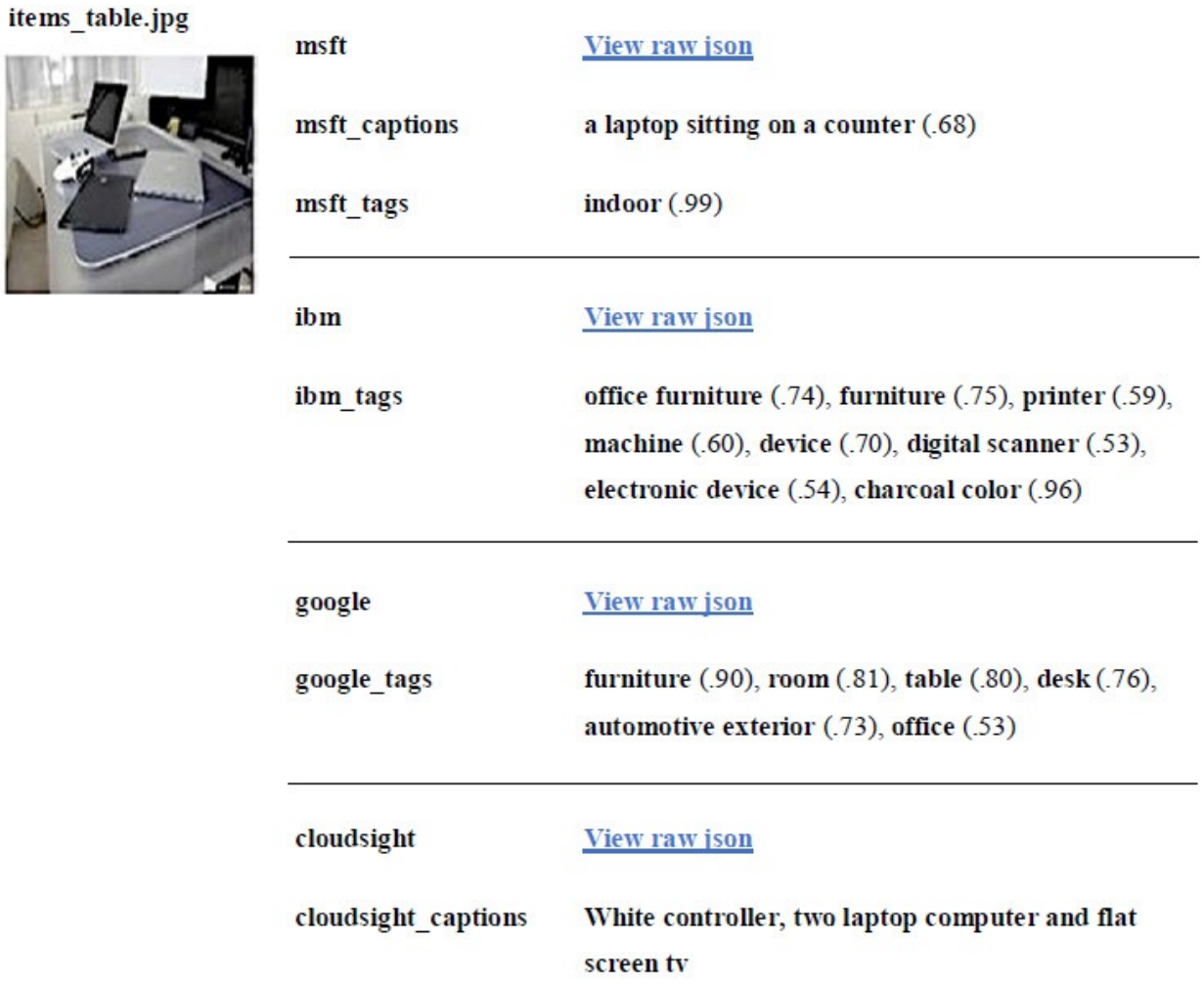

Fig. 6. Results from the 4 APIs for a sub-image.

We were also given the opportunity to visit Goodwill's Visually Impaired Program in Harlem to talk with visually impaired individuals about our project. Some suggestions included taking a panorama covering the floor angle because many of their misplaced items fell onto the floor and they cannot see what is on the floor. We generally received positive feedback as well as remarks about when the application would be on the market, the cost of using, and if we were developing an iOS application as well.

Conclusion

The camera-based system will be easy to use and affordable. Being both iOS and Android-compatible will further increase the target audience for the application. By using the panoramic feature in phones and the 360-degree field-of-view camera, we will be able to target and locate objects. This system will greatly benefit many visually impaired people by increasing productivity and independency. It will not only increase the efficiency in locating misplaced objects but also give visually impaired individuals the ability to become more comfortable in dynamic environments. By building this system, we plan on reducing every day struggles of the visually impaired and ease their mental and physical life.

The current implementation has focused on computer vision processing and combining results from multiple object recognition cloud services. We have also talked with end users about using such a system. In the future, we aim to focus on user requirements gathering with blind users, and on the evaluation of such a system with users who are blind. In doing these, we will be able to answer some questions with regard to the proposed system design, such as: How would users train the system to know which objects they care about in their environment? Can users give names to the things in their environment, e.g. "Bob's Laptop" ? What should be the appropriate voice cues, i.e. how do these guide someone to an object?

With the current modular framework in place, it is relatively trivial to replace the source of object recognition from using external cloud-based services to a localized template approach similar to that of Alam et al. 2015. Localized templating will address one of the key feedback requests we had gotten from visually impaired individuals at Goodwill as well as visiting visually impaired individuals from the Student Assistive Tech Exposition event at CCNY.

Acknowledgements

This work was supported by funding from NYSID CREATE. The project has also been supported by NSF GARDE Program for Course Development on Assistive Technology to Aid Visually Impaired People (Award #1160046), VentureWell (formerly NCIIA) Course and Program Grant on Human and Machine Intelligence - Perception, Computation and Action (#10087-12). We would like to thank Mr. Guillermo Cuadros at Goodwill's Visually Impaired Program in Harlem and the visually impaired individuals there for their feedback on users' needs and experience.

Works Cited

Alam, S., Anam, A. I. & Yeasin, M.(2015). O'Map: An Assistive Solution for Identifying and Localizing Objects in a Semi-Structured Environment. Journal on Technology and Persons with Disabilities , pp 204-231.

Beck, K.(2010). Challenges That Blind People Face. [Online] http://www.livestrong.com/article/241936-challenges-that-blind-people-face/ (accessed December 4, 2016).

Coldewey, D. (2016). Student's iDentifi app puts object recognition in the hands of the visually impaired. TechCrunch, [Online]. Available: https://techcrunch.com/2016/11/17/students-identifi-app-puts-object-recognition-in-the-h ands-of-the-visually-impaired/. [Accessed: 17- Dec- 2016].

Forssén, P.E. & Lowe, D.G. (2007). Shape descriptors for maximally stable extremal regions. IEEE 11th International Conference on Computer Vision.

Taptapseeapp (2016). TapTapSee - Blind and Visually Impaired Assistive Technology - powered by CloudSight.ai image recognition API. [Online]. Available: http://taptapseeapp.com/. [Accessed: 17- Dec- 2016].

WHO Facts (2016). International Agency for the Prevention of Blindness. IAPB, n.d. Web. 07 December.