Trinetra: Assistive Technologies for Grocery Shopping for the Blind

Patrick E. Lanigan, Aaron M. Paulos, Andrew W. Williams, Dan Rossi and Priya Narasimhan
Carnegie Mellon University, Pittsburgh, PA 15213, USA.
Contact*: priya@cs.cmu.edu*

Abstract

Trinetra aims for cost-effective, assistive technologies to provide blind people with a greater degree of independence in their daily activities. The overall objective is to improve the quality of life for the blind by harnessing the collective capability of diverse networked embedded devices to support grocery shopping, transportation, etc. This paper describes our research and development of the Trinetra system, a barcode-based solution comprising COTS components, such as an Internet- and Bluetooth-enabled cell phone, text-to-speech software and a portable barcode reader. We describe our experiences with the first deployment of Trinetra at the Carnegie Mellon University's campus store, Entropy.

1. Introduction

According to the American Foundation for the Blind's statistics [1], every 7 minutes, someone in America will become blind or visually impaired; almost 8 million Americans have a visual impairment with difficulty in reading. Basic activities such as riding the bus or buying groceries are just some of the challenges that these blind people face every day.

We were fortunate to have access to a blind mentor and user, Dan Rossi (also a co-author on this paper), who worked with us actively from conception to deployment. He has been blind since the age of seven, is literate in Braille, and is also an expert with technology. He provided invaluable feedback, not just from the viewpoint of a blind user, but also in giving us insights into the quality-of-life and cost-of-living issues that blind people face. His personal observations and previous experiences with assistive technologies were some of the driving principles in the design of Trinetra:

(a) Independence in one's daily activities, without requiring/requesting assistance from a sighted person, is of the highest priority.
(b) Blind users should be intrinsically involved *a priori* in the design of assistive technologies, and not simply brought in after-the-fact for testing.
(c) As sighted people, we tend to be tempted to "insert" assistive enhancements into a blind person's cane – this might be unwise because additional weight or functionality to a cane might adversely influence its everyday usage.
(d) Products targeted specifically at blind people tend to be significantly more expensive than those for the sighted, and keeping costs down in assistive technologies is important to their adoption.

The initial scope of the Trinetra system was to assist the blind and visually impaired with the task of grocery shopping. The first deployment target of this system was the Carnegie Mellon University (CMU) campus convenience store, Entropy.

2. The Problem Statement

There are many kinds of food, such as fruits and vegetables, which can be identified by touch. A common problem for blind and visually impaired people is to differentiate between containers that feel the same but that have different contents. This becomes more serious if the products are hazardous, e.g., identifying a glue-stick vs. a stick of lip balm.

Consider the following question posed by a blind person: "Have you ever opened a can, hoping to add tomatoes to your spaghetti sauce, and then been faced with the dilemma of figuring out how to incorporate green beans into the menu?" [10] The solution of using a Braille writer and labeling tape to affix custom labels onto grocery items has its clear advantages – it allows a blind user to identify various items (beyond groceries, to include appliances) to allow for faster and more accurate identification, without requiring assistance from a sighted person at the time of identification.

Unfortunately, there are drawbacks. First, a sighted person is still required at the time of product-tagging in order to generate the correct Braille labels. Moreover, Braille literacy is still not widespread (only 8-10% of the blind population in the U.S.). Finally, this is not an economically feasible solution – the U. S. Council for Better Business Bureau observes that, "for most grocery stores, putting all price labels in Braille could not be done without significant expense." [3]

Most of the suggestions provided by the National Federation of the Blind for grocery shopping [9] invariably involve the assistance of a sighted guide, typically, a friend, neighbor or a store clerk. However, a blind shopper must wait for a store clerk to become free, and to assist him/her in then fetching the product. The blind person might not have the luxury to browse products unassisted. Also, this scenario inevitably hampers the independence of blind shoppers, requires them to plan their shopping trips around times that the store is likely to be more or less empty, and makes them more reluctant to purchase a large number of items for fear of overly bothering the store clerks.

2. 1. Target Deployment Environment

Entropy is a small convenience store that stocks different items (e.g., snacks, health products) that students and others on the CMU campus are likely to need often. The store has a small number of shelves, two main aisles, and is often manned by two cashiers behind the counter. We selected Entropy as our initial deployment target for Trinetra, not only because of its proximity and access to us, but because our blind user, Dan Rossi, tends to frequent the store to make various purchases on campus. We felt that targeting Entropy would serve his immediate needs, would allow us the flexibility to experiment with the various structural elements (aisle, rack, shelf) of the store, and give us a chance to understand the day-to-day operations of a relatively small store before we moved forward to address the larger-scale needs of supermarkets.

As with most stores, the employees at Entropy tend to re-stock the items and re-locate them based on demand, e.g., during winter, brands of flu medication might be more prominently displayed on entering the store. In our discussions with these employees, we came to realize that any solution that required more effort in the re-stocking process was not a feasible alternative. For instance, if we chose to tag an aisle or shelf containing cookies with an electronic label that encoded this information, then, each time those cookies were re-located within the store, we would require significant cooperation from the store employees to move the electronic labels with the cookies, and to re-tag the aisles and/or shelves that represent the new locations of those cookies. While most stores might be willing, in principle and in spirit, to cooperate with us in this manner, there is no guarantee that the re-labeling process will actually occur or will be done correctly.

2.2. Objectives and Design Considerations

The technical objectives of the Trinetra system, in the context of grocery shopping, are to allow for a cost-effective, independent shopping experience for the blind. The blind shopper should not need to ask for assistance from a store clerk, and should be able to locate and browse items in the store on his/her own.

The idea is to provide for enough levels of identification within the store to allow a blind shopper to find the right aisle and shelf containing the product of interest (e.g., soup) and then to discriminate between different products of the same type (e.g., between chicken soup vs. tomato soup) once the overall shelf location of the product of interest has been identified.

This use of the Trinetra technology need not be restricted to grocery shopping alone, but can also apply to the dilemmas encountered by blind people in daily tasks such as cooking. Without a way to discriminate between identical containers with different products, a blind person might end up opening a can of green beans rather than the can of tomato sauce that a recipe calls for, but would have to either live with the outcome or start all over again to seek the right can.

Leveraging existing infrastructure: One of our aims was to ensure that the resulting solution was cost-effective to the blind user and the store, and that it did not require maintenance on the store's part.

Individual products within grocery stores are already tagged with UPCs or barcodes by product manufacturers. While the use of radio-frequency identification (RFID) tags for merchandise identification has been touted for its advantages (such as the reprogrammability of RFID tags, their ability to hold more product information, no requirement of line-of-sight reading of the tags, accuracy of localization, etc.), they are not yet prevalent in grocery stores because of concerns such as privacy and cost [12].

Instead, from a deployment perspective, *we deliberately focused on practical, achievable, cost-effective solutions that could be deployed and used by blind shoppers today*, and not on futuristic RFID-based solutions that might be attractive but that would not

Figure 1: Hardware Components.

help blind shoppers with their immediate needs. Our barcode-based solution does not require any investment on the store's part. Our RFID solution, while feasible, would certainly require investment on the store's part.

While we discuss our solution for dealing with RFID-based aisle- and shelf-level identification of products, the first deployed prototype of the Trinetra system focuses on product-level identification, assuming that a blind person has been able to navigate to the right shelf and aisle. While this might still require a blind person to seek assistance to get to the right aisle and shelf in an unfamiliar store (which even sighted people often require assistance with), once the blind person has arrived at the shelf containing for instance, cans of soup, he/she can decide whether to purchase chicken or tomato soup. The blind person can also browse the various items located in the region to make spontaneous, unplanned purchases of, say, a new kind of soup that has become available on the market.

Leveraging COTS components: As another aspect of cost-effectiveness, we aimed to use commercial off-the-shelf (COTS) components that were not necessarily developed with blind and visually impaired people in mind. Products targeted specifically for the blind are more expensive than their counterparts for the sighted population because of the additional effort and cost that is required to adapt and test technologies specifically for the blind population. Moreover, economies of scale do not exist to keep manufacturing costs low [11]. Thus, *if Trinetra used as many COTS products as possible, our resulting solution was likely to be more cost-effective and widely adoptable* (and potentially even useful for sighted people).

To this end, we used a standard off-the-shelf barcode-scanning embedded device, a standard cell phone, and a standard Bluetooth headset. To integrate these devices effectively for the blind consumer, we exploited standard protocols, such as TCP, Bluetooth and HTTP. On the software side, we used standard operating systems, such as Symbian and Windows.

Another off-the-shelf software component was text-to-speech software that was used to convert text displayed on the cell phone into speech output. This software is available commercially for the Symbian operating system, and while the software has discounts for blind users, it is not marketed only to the blind. For instance, this can also be used by sighted people to listen to incoming email safely, while driving.

Usability: We wanted to ensure that the Trinetra user interface employed a format that is common or intuitive to the user, that the resulting technology was portable, and that it could be used without attracting undue attention to the user (e.g., avoiding noises or awkward, bulky gadgets). The use of text-to-speech software on the cell phone ensured the first of our goals – our blind user was already used to his cell phone providing him output in the form of speech; conveying the product information in the form of speech merely leveraged his phone's capabilities. Portability was possible through the use of the cell phone and a barcode-scanning pencil-like device that is a mere 42 grams in weight – both devices can easily fit into the user's pocket, and do not require any wires because Bluetooth was used for communication. By providing our user with a small Bluetooth head-set, we conveyed speech from the cell phone unobtrusively and directly into the user's ear.

3. Design and Architecture

3. 1. Hardware Components

Our system is completely built and designed using COTS hardware, including, Nokia's 6620 smart-phone, a Bluetooth wireless headset, Baracoda's IDBlue RFID-scanning pen, Baracoda's barcode-scanning BaracodaPencil and generic high-frequency RFID tags from Texas Instruments, most of which are shown in Figure 1. A desktop computer (not shown here) acts as a remote server for specific operations.

The Nokia 6620 runs the Symbian 7.0s operating system, works with Enhanced Data GSM Environment (EDGE) and General Packet and Radio System (GPRS) networks, contains 11MB of memory, and 32MB of persistent storage on a Secure Digital multimedia card. We initialized our Nokia 6620 using the T-Mobile service provider, and we currently use GPRS for data connectivity over the Internet. The Nokia 6620 supports both Java Mobile Information Device Profile (MIDP) development and native Symbian programming using the Series 60 2nd Edition, Feature Pack 1. The BaracodaPencil, a Bluetooth-enabled device, is capable of scanning and relaying UPC information to master devices. The BaracodaPencil comes in a small form-factor resembling a thick pencil as shown in Figure 1. A sister product from Baracoda, the IDBlue pen, is capable of scanning low-frequency RFID tags and marshalling data over Bluetooth in a similar manner.

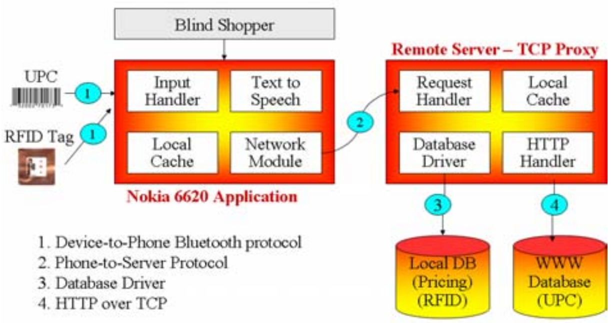

Figure 2: Software Components.

3.2. Software Components

Figure 2 shows the architecture diagram involving the integration of the various protocols, software components, and hardware components that the Trinetra system uses. The input to the Trinetra system on the Nokia 6620 cell phone can come from three sources: (i) scanned barcodes, (ii) scanned high-frequency RFID tags, or (iii) user keypad-events. The system's output to the user is in the form of synthesized speech that is generated by the off-the-shelf text-to-speech software, Nuance Communication's TALKS, which can be installed on the cell phone as a third-party product. TALKS converts the text displayed on a Symbian-powered cell phone into speech, using the underlying Eloquence speech engine. Also shown in Figure 2 is an online, publicly available UPC database, located at http://www.upcdatabase.com.

Every barcode is associated with a numerical sequence of 8-12 digits. If this numerical sequence is then looked up in the UPC database, a text string can be retrieved that describes the product, including brand name and type. If an HTTP query is sent to the UPC database with the numerical sequence as an input, the resulting output over HTTP will be the corresponding text string.

Scanned input (from a UPC or an RFID tag) is sent via Bluetooth to the Trinetra Symbian C++ application residing on the Nokia 6620. The application checks a most-recently-used (MRU) local, first-level, cache for a product match. In the case of a cache hit, a product description will be returned to the user. In the case of a cache miss, the Trinetra network module onboard the Nokia 6620 communicates over TCP to the remote concurrent desktop server that forms a part of the Trinetra system. The Trinetra remote server checks its much larger, user-centric MRU cache for a product entry. In the case of a cache hit at this second-level cache, the results are returned to the Trinetra Nokia 6620 application via TCP. The Trinetra code onboard the cell phone updates its local, first-level cache and returns the resulting value to the blind user through TALKS. Note that the first-level cache is persistent, i.e., its contents persist over phone restarts.

In the case of a cache miss at the remote, second-level cache, Trinetra will either (i) check a local store-level inventory management system for pricing or RFID information, or (ii) perform an HTTP request to the publicly available UPC product database. In the case of (i), the inventory management system or RFID access, connectivity is accomplished by a database driver provided by the local vendor. In the case of (ii), after UPC requests over HTTP, the resulting HTTP responses are parsed and stored locally in the remote, second-level cache. Likewise, in the case of an RFID lookup, the response is stored in the remote, second-level cache. In both cases, the response is relayed to the originator of the request, and is stored in the local, first-level cache onboard the Nokia 6620, and output to the user through TALKS.

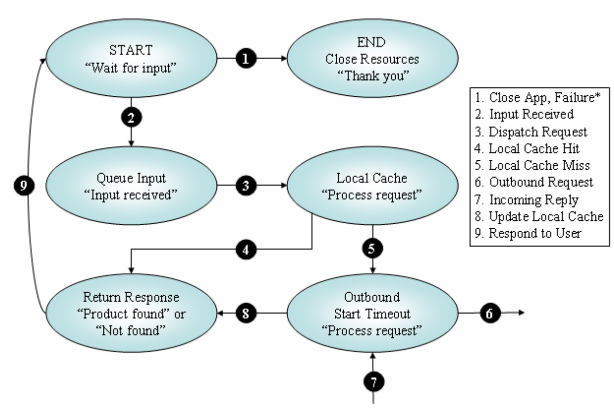

Figure 3: Trinetra Onboard the Cell Phone.

3.3. Trinetra's and the User's Perspectives

Figure 3 demonstrates the usage lifecycle from the

Trinetra cell phone application's perspective. Starting from the top left-hand corner, in the state labeled "START", the user launches the Trinetra application on the cell phone (as any other application on the cell phone would be normally launched, through a combination of keypad inputs), and receives a verbal notification indicating the system is ready for input. The user is allowed to shut down the Trinetra application at any point during the lifecycle. The termination of the application is indicated in the top right-hand corner, and is labeled as the "END" state. When the user closes the application, or in cases where a "trap"-able error occurs, leading to the same result, the application will release system resources and exit gracefully with a verbal notification of termination.

On receiving scanned input from either the BaracodaPencil or the IDBlue pen, the application will verbally announce that an input has been received. At this point, the local, first-level cache is examined, and in the case of a cache miss at this level, the user receives another notification that the request is under processing. We envision this notification can ultimately be some type of familiar, audible pattern that will let the user know to expect longer latencies in receiving a response because Trinetra is using the GPRS network to obtain the result. Finally, the user will receive an audible message describing the product name along with any brand/type description, or alternatively, a message indicating the product could not be located in the global UPC database. The user can then terminate the Trinetra application or continue scanning more items, as indicated by arrow #9 that transitions to the START state ("wait for input").

Figure 4 shows the typical interactions between the user and Trinetra. The vertical arrow represents the march of time. The text strings displayed in the ovals in the left-hand side of Figure 4 represent the audible messages that Trinetra returns to the user; the text strings displayed in the ovals to the right-hand side of Figure 4 represent the steps in the user's interaction.

4. Implementation Details

End-to-end, the Trinetra system can be considered to be a three-tier distributed system, as shown in Figure 2; the BaracodaPencil or IDBlue pen devices constitute the first tier, the Trinetra application onboard the cell phone represents the middle tier, and the Trinetra remote server represents the back-end, or third, tier.

The purely event-driven Symbian C++ Trinetra application in the middle tier reads asynchronous, unidirectional, serial input from a Bluetooth

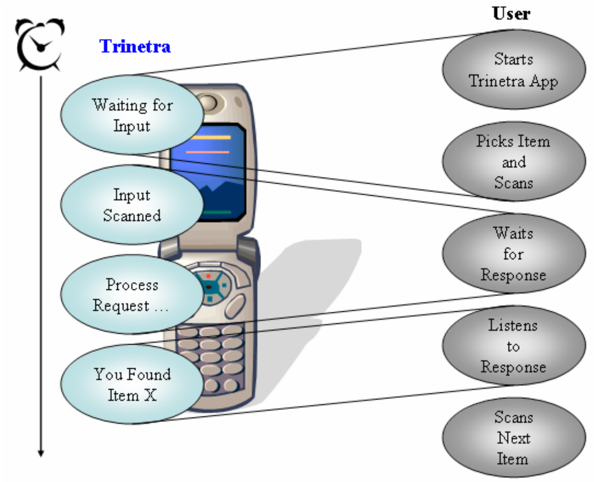

Figure 4: Mapping User Actions to Trinetra.

connection to the Baracoda scanning devices. In the event of a local, first-level cache miss, the Trinetra application protocol is used over TCP to access our third-tier, remote concurrent I/O-multiplexing TCP proxy server. The phone-to-server application protocol is synchronous (where each request blocks, awaiting a response, before the next request can be processed) and leverages the reliable delivery guarantees that TCP provides. Blocking until the completion of any requests to the back-end tier, the middle-tier Trinetra application is suspended until the remote server returns a positive acknowledgement or an "Item No Found" message. The remote server uses either a database driver, or an HTTP GET request, to query its back-end data source.

4.1. Device-to-Phone Interaction Protocol

The Nokia 6620 communicates with the two Baracoda scanning devices using a serial communication protocol over a Bluetooth wireless link. The Baracoda scanning devices are capable of operating in a number of different modes when connected to a host. For our implementation, the devices are set to an "asynchronous" operating mode. In this asynchronous mode, the user initiates the scanning process by pushing a button on the scanning device. The device will then send packets to its host device (in this case, the Nokia 6620 cell phone) on the successful scan of a barcode or an RFID tag. The host-side code running on the Nokia 6620 listens for incoming data, reconstructs the packet structure from the serial byte-stream, and parses the scanned data.

In the case of the BaracodaPencil, the scanned data is sent as a string of ASCII characters that correspond to the numerical UPC code. The IDBlue device sends an 8-byte binary stream that corresponds

Figure 5: Text output of barcode scanning.

to the unique tag ID embedded in ISO15693 RFID tags. This data is further processed by the host-side receiving code to convert it into a string of 16 ASCII characters that correspond to the hexadecimal encoding of the original binary stream.

4.2. Phone-to-Server Infrastructure Protocol

The Trinetra phone-to-server protocol is comprised of five operations supporting two request types, for UPCs and RFIDs. Currently, this protocol supports a variable payload size of up to 128 bytes. While larger payloads can certainly be sent over GPRS, most product labels should not require more than 128 bytes. UPC and RFID payloads are ASCII strings, which represent an adequate data-type for accessing information stored in a local inventory management system or in the global UPC database The payloads do not require endian conversions.

Upon accepting a new connection from a cell phone, the remote server launches a new thread to handle the invocations from that phone. This thread-per-connection concurrency model is adequate for our current deployment considering Entropy's maximum capacity of well under 50 customers within the store at any time. Both UPC- and RFID-request types are immediately processed by performing a lookup on the server's second-level cache. Successful lookups are returned using the respective UPC/RFID reply-type. Second-level cache misses require a subsequent access to the respective operation's back-end data store. A UPC-type request is marshaled over HTTP to the global UPC database in the form of a GET query. The query result is then parsed, stored in the server's second-level cache, and then returned to the phone. An RFID-type request, although not part of the deployment for Entropy, is handled locally using the store's inventory-management that is populated with descriptions of the RFID tags inside the store; in this case, access is achieved using a remote connection and the appropriate database driver. Unlike the global UPC database, there does not currently exist, to the best of our knowledge, a maintainer of a global or standardized RFID database for grocery items. Through the use of RFID scanning, we can incorporate the grocery's inventory-management pricing and provide local product lookup if they are available.

4.3. User Interface (UI)

In an effort to not overwhelm the blind shopper, the user interface (UI) mostly consists of short pop-up messages displaying the system state, along with two hotkey inputs. The main enabling component, the TALKS software, effectively enumerates the text contained in Symbian UI elements, such as `CAknGlobalNote`, `CAnkPopupList` and `CAknTitlePane`, and then reads aloud the interface options. A UI element in focus is read aloud by TALKS. This functionality allows blind users to navigate fairly complex UIs with minimal effort.

The Trinetra interface makes use of the two main hotkeys on the Nokia 6620 cell phone. The right hotkey is used for exiting the Trinetra application gracefully at any time. The left hotkey contains a short menu with options to connect/disconnect the two Baracoda Bluetooth input devices. After a connection is made, the application returns a "Connection Successful" message. The input is now driven from the connected devices. After scanning a UPC/RFID, the application notifies the user that input has been received. Finally, the application returns an "Item Not Found" message, or the results, as shown in Figure 5.

To keep the UI functions simple in our first implementation, we do not permit other applications to run concurrently with the Trinetra application onboard the cell phone. We expect to eliminate this simplifying assumption in future prototypes.

5. Evaluation

Figure 6: Trinetra in Use at Entropy.

Our testing has so far been limited to a single blind user, Dan Rossi. Figure 6 shows Dan Rossi using Trinetra at Entropy. In his left hand, he holds the BaracodaPencil, and in his right hand, he holds the Nokia 6620 cell phone. A Bluetooth head-set (in his right ear, not shown here) provides him with audible notifications from Trinetra. In this first test-run, he was able to distinguish between various packages of cookies to select the one to his liking. A short video demonstrating Trinetra at Entropy is available at http://www.ece.cmu.edu/~trinetra/publications.html.

One of the frequent questions that we have been posed is how a blind user locates the UPC or barcode on a grocery item, given that the BaracodaPencil requires physical contact with the barcode on the item to provide any scanned output. Our experience has led us to not underestimate a blind person's tactile senses and capabilities. Within minutes of handling a new grocery item, our user was able to locate the barcode through various length-wise and breadth-wise surface scans of the item; he subsequently recalled this information to locate the barcode on other packages of the same type, e.g., a different kind of cookie from the same vendor.

Usability: We provide below, a quote from Dan Rossi, printed here with his permission.

Yesterday afternoon I stood in Entropy and picked up a bag of cookies off the shelf and scanned it. I tested a number of products during meetings with the team and I could scan a product and have the phone tell me what the product was. This is amazing! You cannot imagine the real impact of this. When shopping with a store assistant, it is nearly impossible to browse products. I ask for what I need, and they take me to that product, and that's it. A device that can tell me what just about anything in the store actually is, is incredible.

You have no idea of how frustrating it is for me, as a blind person, to stand in my kitchen in front of the cupboard, two identical cans in my hand, knowing that one contains the black beans I need to finish a chili recipe, and the other contains green beans which do not have their place in a good chili. The bar code reader solves that problem simply and elegantly.

The fact that the system leverages so much existing technology and infrastructure puts it miles ahead of most devices for the blind. The use of an off-the-shelf cell phone, an off-the-shelf bar code reader, a publicly available UPC database, and a popular screen reader for the phone, means this product is not limited by having to defray large manufacturing costs over a small number of units. There is no need for building and populating a huge database. No need for designing and maintaining a large infrastructure for a limited use. All this has been done already.

The design of the system lends itself nicely to customizations or upgrades such as replacing the bar code reader with a more powerful version, a smaller version, or eventually switching to an RFID reader.

Is it perfect? Not yet. Due to limited finances and time, the team could not evaluate every type of bar code reader on the market. An omnidirectional scanner would make scanning faster and easier. However, the known omnidirectional scanners are larger and more expensive. That's not to say we know of every omnidirectional scanner out there. Would RFID tags work better? Possibly, but until RFID tags are as pervasive as bar codes, the point is moot. The Trinetra system is practical and useful. [Dec. 2005]

Latencies: A first-level cache hit retrieves the product description in under less than 1 sec, while a first-level cache miss (requiring Internet communication with the UPC database) takes a few more seconds to retrieve the product description, depending on the latency of contacting the global database. We are in the process of completing a more thorough experimental evaluation of Trinetra's latencies for product identification (under cache misses and hits) for a multi-day series of shopping trips undertaken by our blind user. We expect that the results will enable us to allocate cache capacity and structure cache-item replacement better. Another aspect will be to evaluate the RFID vs. UPC infrastructural mechanisms in terms of their respective latencies. We also intend to extend our evaluation to a wider range of blind users.

Cost: The cell phone costs approximately $200 and clearly has uses beyond grocery shopping. The TALKS software costs $200, but with the right service provider, it is available for free. The Baracoda Pencil costs $200 and the IDBlue pen costs $500. We are in the process of designing a custom, low-cost, barcode scanning device with omnidirectional capabilities.

Footprint: The additional footprint due to Trinetra onboard the cell phone is a mere 37kB, excluding the cache. To store each cache item onboard the cell phone might require up to 1kB per item, depending on the length of the text string describing the product.

6. Related Work

Autonomous indoor navigation for various contexts (e.g., object finding, robotic soccer) has been explored in the robotics community. We do not cover all of this work here, but focus on one example that targets grocery shopping needs. The RoboCart [7] from Utah State University provides for a robotic assistant to aid the blind in the task of grocery shopping. The principal focus of this technology is to assist the blind in navigating through a store and in carrying their purchased items. RoboCart relies on RFID tags that are embedded at various locations in the store, and is not intended for individual ownership, but rather, requires an investment on the part of the store. The cost for each unit is estimated to be between $3,500-4000.

Other systems have focused on providing enhanced navigation capabilities for the blind. Drishti [6], one such system from the University of Florida, uses GPS location information with building maps and relevant spatial information to provide directions to buildings and locations within a campus environment. Other technologies have aimed at something similar through the use of smarter canes. GuideCane [2] onsists of a special cane mounted on a two-wheeled axle unit with onboard sensors for obstacle avoidance. Guido [5] from Haptica Corporation is a robotic walking frame that also targets obstacle avoidance. HARINOBU-6 [8] from Yamanashi University is a smart wheel-chair equipped with sensors for similar purposes. To the best of our knowledge, none of these was designed specifically to target the needs of blind shoppers.

I.D. Mate [4] is a talking barcode scanner that sells at up to $1559 per unit. It is a portable, 2 lb. electronic device that scans barcodes and labels of various items at the grocery store. On board the device is a UPC database of almost 1 million items to enable the identification of scanned items. The omni-directional scanner eliminates the need to visually and precisely locate the barcode on any grocery item.

7. Conclusion

The Trinetra system aims to develop assistive technologies for the blind to provide a greater degree of independence in their daily activities. The first implementation was intended to be a proof-of-concept exercise in assisting blind shoppers at the CMU store, Entropy. We have evaluated the technology with the help of a blind user who was instrumental in developing the system.

As a part of future work, we would like to evaluate the effectiveness of RFID vs. UPC solutions. This would also involve developing techniques for labeling and navigating RFID-enriched store environments. In keeping with the original spirit of our work that aimed for adoptability, we intend to ensure that our end-result is portable and cost-effective. A second thrust will involve examining alternative technologies, such as optical character recognition (OCR), instead of barcode or RFID scanning. Several cell phones can be enhanced with higher-resolution macro-mode lenses and off-the-shelf OCR software.

Finally, we intend to address other targets for assistive technologies for the blind – one of our immediate targets is to address and explore transportation needs for the blind by targeting the CMU campus shuttle.

10. References

[1] American Foundation for the Blind, *Statistics and Sources for Professionals*, 2005.

[2] J. Borenstein and I. Ulrich, “The GuideCane - a computerized travel guide for the active guidance of blind pedestrians*,” IEEE International Conference on Robotics and Automation*, San Diego, CA, 1994.

[3] Council for Better Business Bureau, *Grocery Stores: Access Equals Opportunity*, June 2003.

[4] En-vision America, Inc., *I.D. Mate*.

[5] Haptica Corporation, *Guido*.

[6] A. Helal, S. E. Moore and B. Ramachandran, “Drishti: An Integrated Navigation System for the Visually Impaired and Disabled,” *International Symposium on Wearable Computers,* Zurich, Switzerland, October 2001, pp. 149-156.

[7] V. Kulyukin, C. Gharpure and J. Nicholson, “RoboCart: Toward Robot-Assisted Navigation of Grocery Stores by the Visually Impaired,” *IEEE/RSJ International Conference on Intelligent Robots and Systems,* Edmonton, CA.

[8] H. Mori and S. Kotani, “Robotic Travel Aid for the Blind: HARUNOBU-6,” *European Conference on Disability, Virtual Reality, and Assistive Technology,* Sovde, Sweden, 1998.

[9] National Federation of the Blind, *Shopping Ideas*.

[10] E. Ringlein, “101 Ways to Use Braille,” *The Braille Spectator*, National Federation for the Blind, Summer 1998.

[11] V. Smith, “The Blind Struggle as Gadgets Proliferate,” *Assistive Technology News,* vol. 107, May 2005.

[12] V. Stanford, “Pervasive Computing Goes the Last Hundred Feet with RFID Systems,” *IEEE Pervasive Computing*, April-June 2003, pp. 9-14.