Trinetra: Assistive Technologies for the Blind

Patrick E. Lanigan, Aaron M. Paulos
Andrew W. Williams, Priya Narasimhan

May 1, 2006
CMU-CyLab-06-006

CyLab
Carnegie Mellon University
Pittsburgh, PA 15213

Trinetra: Assistive Technologies for the Blind

Patrick E. Lanigan Aaron M. Paulos Andrew W. Williams Priya Narasimhan
ECE Department, Carnegie Mellon University, Pittsburgh, PA, USA

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

1. Introduction

The Trinetra project was conceived in late 2004 at Carnegie Mellon University, as a part of an broad research effort to develop assistive technologies for blind and visually impaired people. The overall objective of Trinetra is to enhance the quality of life for the blind by providing them with greater independence in their day-to-day activities, such as grocery shopping and transportation, while keeping the resulting solution usable and cost-effective. In this technical report, we focus on the objectives and the architecture of the Trinetra system, along with our experiences in its development and subsequent deployment.

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 or are unable to read letters in regular print, even while wearing ordinary glasses. Basic activities such as riding the bus or buying a can of soup at the grocery store are just some of the challenges that these blind people face every day. Furthermore, as the population ages, the number of older individuals who are blind or have low vision will likely increase – approximately 1.5 million people within this group are considered to be legally blind. Generally, a significant number of individuals who are legally blind find large print or audio texts helpful, while only 8-10% use Braille as a reading medium. On a global scale, there are 37 million blind people; every second, a person becomes blind, every minute, a child becomes blind.

We were fortunate to have access to a blind mentor and user, Dan Rossi, who worked with us actively from the conception of the project to its deployment. Dan Rossi has been blind since the age of seven, is literate in Braille, and is also expert with technology. He provided invaluable feedback, not just from the viewpoint of a blind user of our system, but also in terms of 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 is the topmost priority for blind people – to be able to go about their daily activities, without requiring/requesting assistance from a sighted person would be invaluable.

(b) Blind users should be intrinsically involved *a priori* in the design of assistive technologies, and not simply brought in to test the resulting technologies well after the fact.
(c) As sighted people, we tend to be tempted to “insert” assistive enhancements into a blind person’s shoes or cane – in the end, this might encumber, more than, assist the end-user because adding more weight or functionality to a blind person’s cane might influence its torque and usage adversely.
(d) Products targeted specifically at blind people tend to be more expensive than those for the sighted – this meant that we needed to be sensitive to cost in developing our solution. Thus, we needed to consider commercial off-the-shelf technologies that were usable by both blind and sighted consumers. Thus, one way of ensuring cost-effectiveness was to leverage products and technologies that are mass-produced for the general market and not for the blind alone.

The scope of the initial Trinetra prototype, which is described here, 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 campus convenience store, Entropy. The remainder of this document focuses on the application of Trinetra in the context of grocery shopping.

2. The Problem Statement

There are many kinds of food that can be identified by touch, such as fruits and vegetables. But grocery items like cans of soup, cereal boxes, canned vegetables, etc. may be harder to identify. A common problem [9] for blind and visually impaired people is to differentiate between containers that feel the same but that have different contents. This is a much more serious problem if the products are hazardous, e.g., discriminating between a glue-stick and 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?” [16] The author of this article proposes the use of Braille as a way to identify various grocery items in order to work around this problem, using the combination of a Braille writer and labeling tape to affix custom labels onto purchased grocery items to mitigate the uncertainty in cooking and product usage. This solution has its clear advantages – it allows a blind user to create custom tags for various common items (beyond groceries, to include appliances and other household items) in order to allow for faster and more accurate identification at a later time, without requiring assistance from a sighted person at the time of identification.

Unfortunately, there are two drawbacks: first, the assistance of a sighted person might still be required at the time of product-tagging in order to generate the correct Braille labels, and secondly, Braille literacy is still not widespread (only 8-10%, as described above) in the blind community. Also, it is not economically feasible to apply Braille tags on every stocked product in a grocery store – 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.” [5]

The Council for Better Business Bureau goes on to offer the following alternative for grocery shopping for the blind: “A store employee could offer to assist customers who are blind or who have limited vision by describing the items and reading prices and labels unless an undue burden would result. When merchandise and price information are available on a display board at a deli, bakery, or other service counter, high-contrast signs or large print handouts are helpful for people with limited vision.” However, asking for a store employee’s assistance certainly robs a blind person of his/her independence; similarly, posting large-print or high-contrast signs is not really useful for completely blind individuals. Most of the practical suggestions provided by the National Federation of the Blind for grocery shopping [15] invariably involve the assistance of a sighted guide, typically, a friend, neighbor or a grocery-store clerk. Note that typical blind-navigation aids, such as a cane or a guide dog, are of little use in a grocery-store environment, because the stores often change the locations of stocked items, and because these aids do not really help to distinguish between various grocery items.

Figure 1 shows such a typical interaction scenario (which we also refer to as the *Before-Trinetra* scenario) where a blind person wanting to buy a jar of mustard must wait for a store clerk to have a free moment, and to assist him/her in then fetching the product. The blind person might not really have the luxury to browse products unassisted. Furthermore, 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.

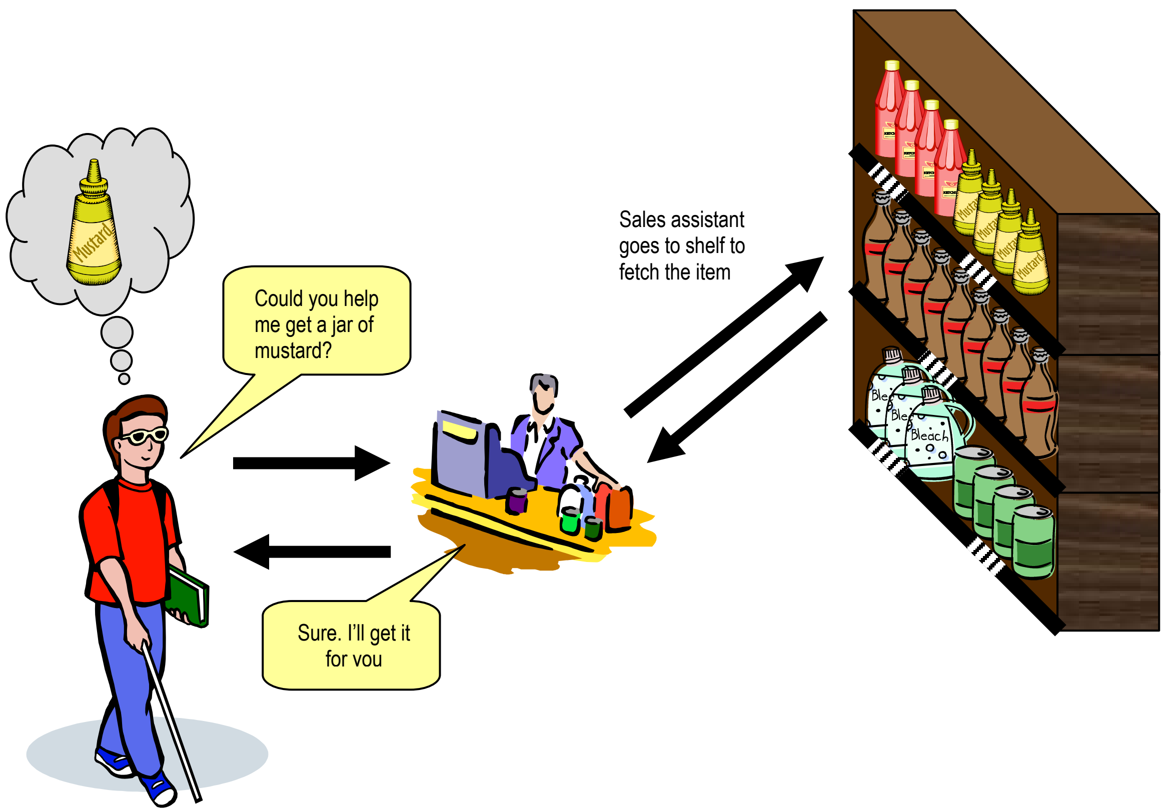

Figure 1: Before Trinetra

2.1. Target Deployment Environment

Our initial target environment for the deployment of the Trinetra system was the Carnegie Mellon University (CMU) campus store, called Entropy. It is important to describe how Entropy functions in order to understand some parts of our design.

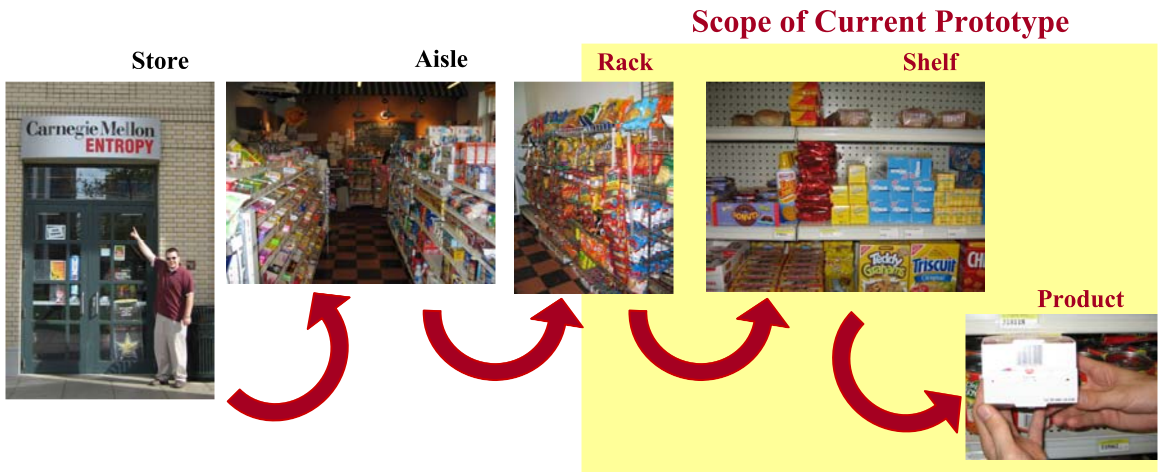

Figure 2: Various levels of grocery navigation at Entropy.

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 (one of which is shown in Figure 2), 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 *After-Trinetra* scenario shown in Figure 3 demonstrates the overall goals of this project. The blind shopper should not need to ask for assistance from a store clerk, and should be able to locate and browse products within the store on his/her own.

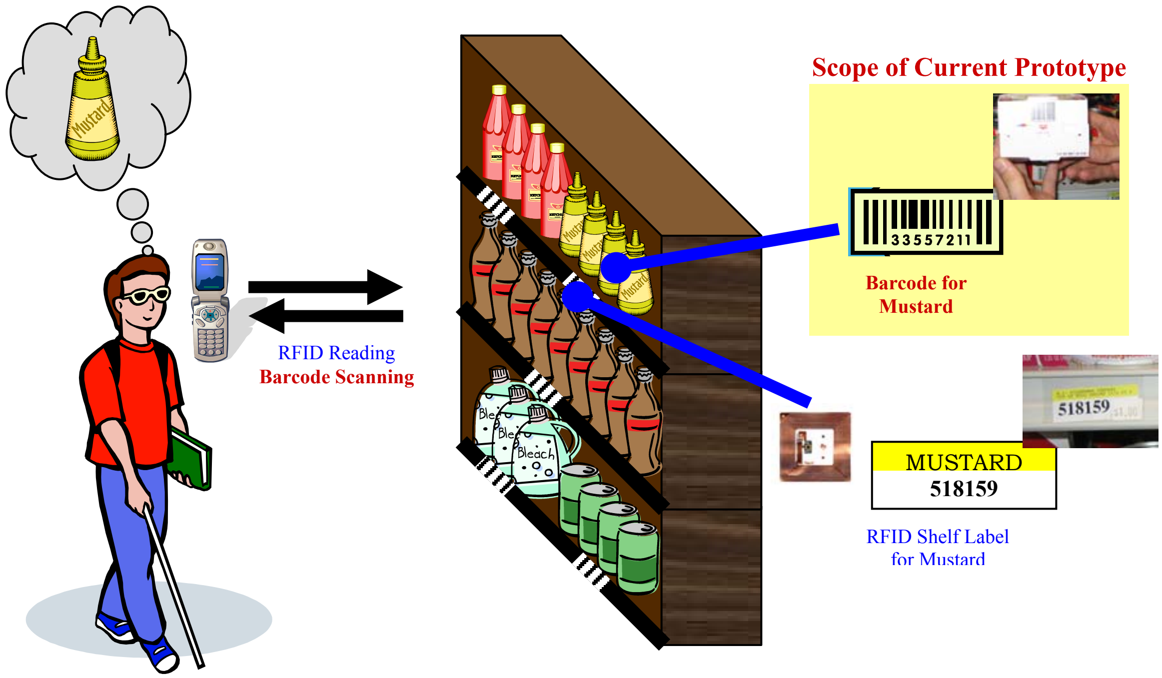

Figure 3: After Trinetra.

The idea is 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. To this end, we need aisle-level and shelf-level electronic tagging of products within the store, and once the shelf has been located, we can use the ubiquitous Universal Product Code (UPC), or barcode, already on the product to discriminate between various types of the same kind of product.

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 both the blind user and the grocery store, and that it was not onerous to the store in terms of requiring maintenance. To this end, we elected to exploit existing infrastructural elements that were commonly available in grocery stores and decided not to add additional infrastructural elements within grocery stores. Individual products within grocery stores are already tagged with UPCs or barcodes by the product manufacturers. While the use of radio-frequency identification (RFID) tags for merchandise identification has long been discussed and touted for the obvious 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 many concerns, including privacy and cost (up to $15 per tag, depending on capabilities) [18].

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 help blind shoppers with their immediate needs. Thus, our initial Trinetra prototype involves experimentation with barcodes alone. The barcode-based Trinetra solution does not require any investment on the part of the grocery store because of the ubiquitous nature of barcodes and the public, online availability of a barcode database that can map a barcode into a human-interpretable text string. Our next steps with deploying Trinetra will be to leverage both barcodes and RFID tags (as shown in the After-Trinetra scenario) so that our solution will be able to handle RFID capabilities when and if they ultimately emerge into the retail space.

While we discuss Trinetra's solutions and strategies for dealing with RFID-based aisle- and shelf-level identification of products in the following sections, the first deployed prototype of the Trinetra system is limited to 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 keeping our solution cost-effective, we aimed to use commercial off-the-shelf (COTS) components that were not necessarily developed with blind and visually impaired people in mind. We have come to realize that products targeted specifically for the blind are more expensive than their counterparts for the sighted population – this is because these products require additional effort and cost to be adapted for the blind, are manufactured in smaller quantities (because of the market size), and require more testing [17]. For instance, a blood-glucose monitor for the blind is often 10 times larger and 10 times more expensive than a standard one for sighted people. Thus, *as long as Trinetra used as many commodity COTS products as possible, our resulting solution was likely to be more cost-effective and widely adoptable* (and potentially even useful for sighted people).

This meant that we used a standard off-the-shelf barcode-scanning embedded device, a standard cell phone (that was recommended to us by our blind user, Dan Rossi, who uses this specific phone model every day), 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. None of these products or protocols was developed specifically with blind people in mind. Another off-the-shelf software component that was text-to-speech software that we required to convert any 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 and visually impaired users, the software is not specifically marketed only to the blind. For instance, this software can also be used by sighted drivers to listen to incoming email safely, without taking their eyes off the road.

Usability: Another technical objective for Trinetra was to make it usable to the blind user. This has many facets to it – ensuring that the Trinetra user interface employs a format that is common or intuitive to the user, ensuring that the solution was portable, and providing a way of using the system without attracting undue attention to the user (e.g., avoiding audible signals 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 with output in the form of speech; conveying the product information in the form of speech was merely leveraging his phone's existing capabilities. The portability of our solution was ensured through the use of the cell phone along with 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 to be carried around because Bluetooth can be used to integrate the two devices together wirelessly. The third goal was accomplished by providing our user with a small Bluetooth head-set (a popular lightweight device that fits snugly into one ear) that conveys speech information from the cell phone directly into the user's ear, without attracting any attention from bystanders.

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, generic low-frequency RFID tags from Texas Instruments, and a desktop computer acting as a remote server for specific operations. All of these hardware components are shown in Figure 4.

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 service provider T-Mobile, 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 4. 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.

3.2. Software Architecture

Figure 5 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 low-frequency RFID tags, or (iii) user keypad-events. The system provides output in the form of synthesized speech that is generated by the off-the-shelf text-to-speech software, TALKS, which can be installed on the cell phone as a third-party product. TALKS, a product of Nuance Communications, allows us to convert the text displayed on a Symbian-powered cell phone into speech, using the underlying Eloquence speech engine.

Also shown in Figure 5 is an online, publicly available UPC database, available at http://www.upcdatabase.com. Every barcode is associated with a numerical sequence of 8-12 digits that is often printed at the base of the barcode (in cases where this sequence is not printed, it can be scanned from the barcode representation). 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. Furthermore, 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.

Figure 4: Trinetra's Hardware Components.

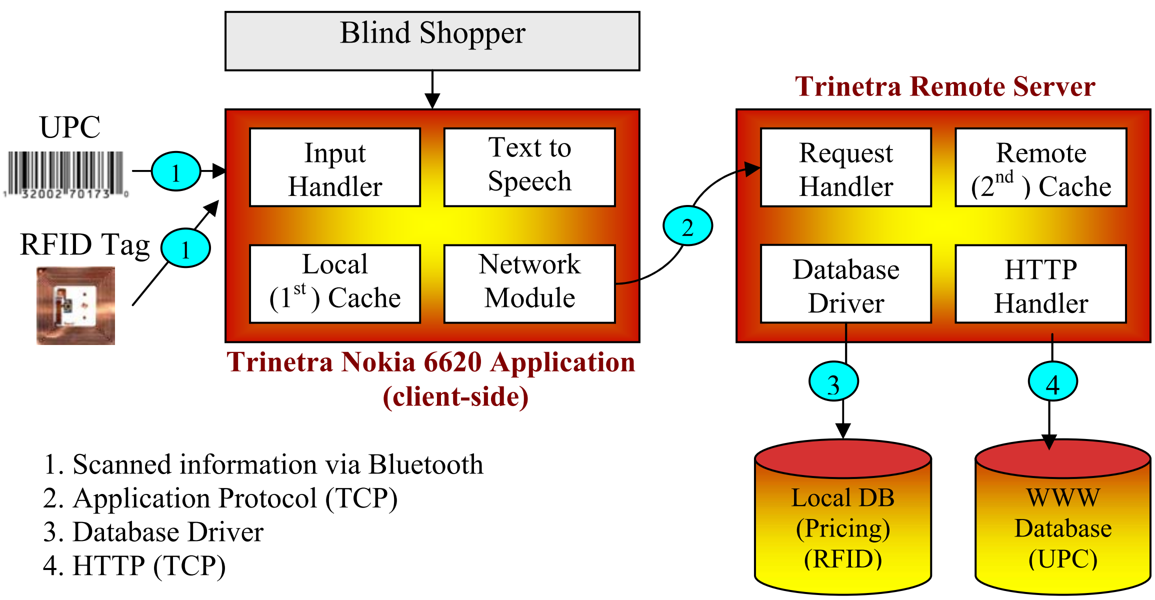

Figure 5: Trinetra's Software Components.

As shown in Figure 5, scanned input (from either 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 persistent, 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 end-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 we have developed and deployed as part of the Trinetra system. The Trinetra remote server checks its much larger, intelligent, user-centric MRU cache for a product entry. In the case of a cache hit at this second-level cache, the results will be returned to the Trinetra Nokia 6620 application via TCP. The Trinetra client-side code onboard the cell phone will update its local, first-level cache and subsequently return the resulting value to the blind user through TALKS.

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.

3.3. Trinetra's and the User's Perspectives

Figure 6 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 an exception occurs, leading to the same result, the application will release system resources and exit gracefully with a verbal notification indicating 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, similar to the common processing-status bar or hour-glass pattern used to indicate ongoing processing in most applications for sighted people. The intent of this notification is to let the user know to expect longer latencies in receiving a response because Trinetra is using the GPRS network to obtain the result. On Trinetra receiving the results via GPRS, the user will receive a final 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").

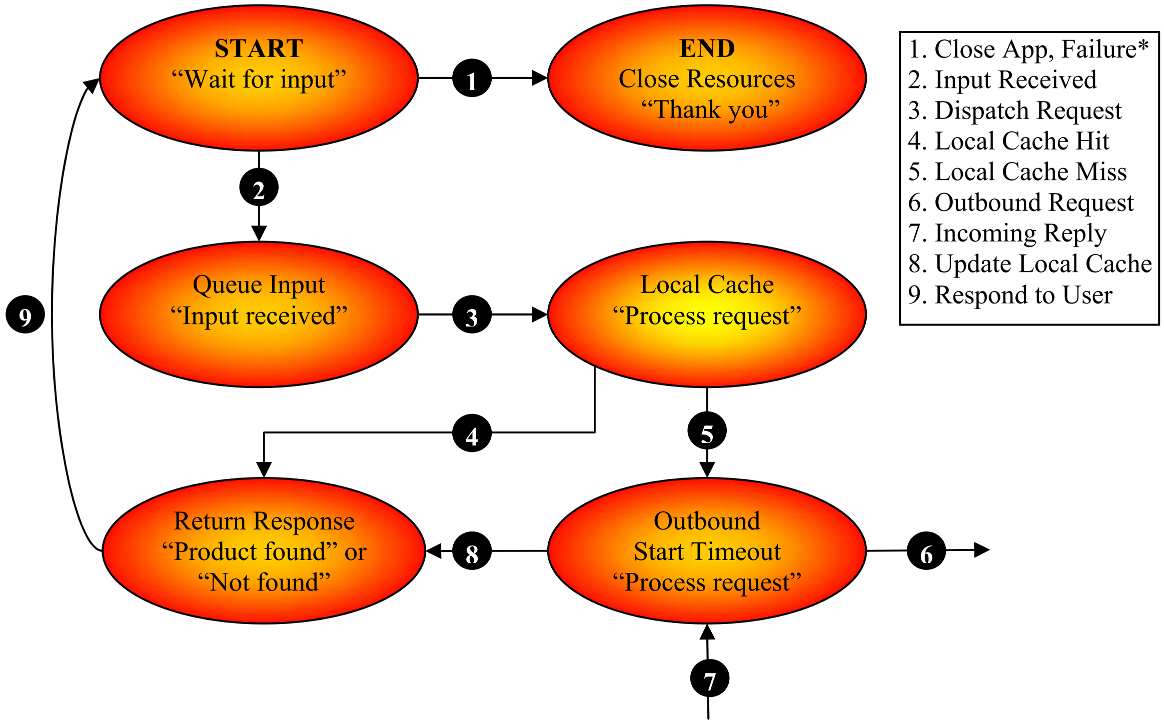

Figure 6: Trinetra Application Onboard the Cell Phone.

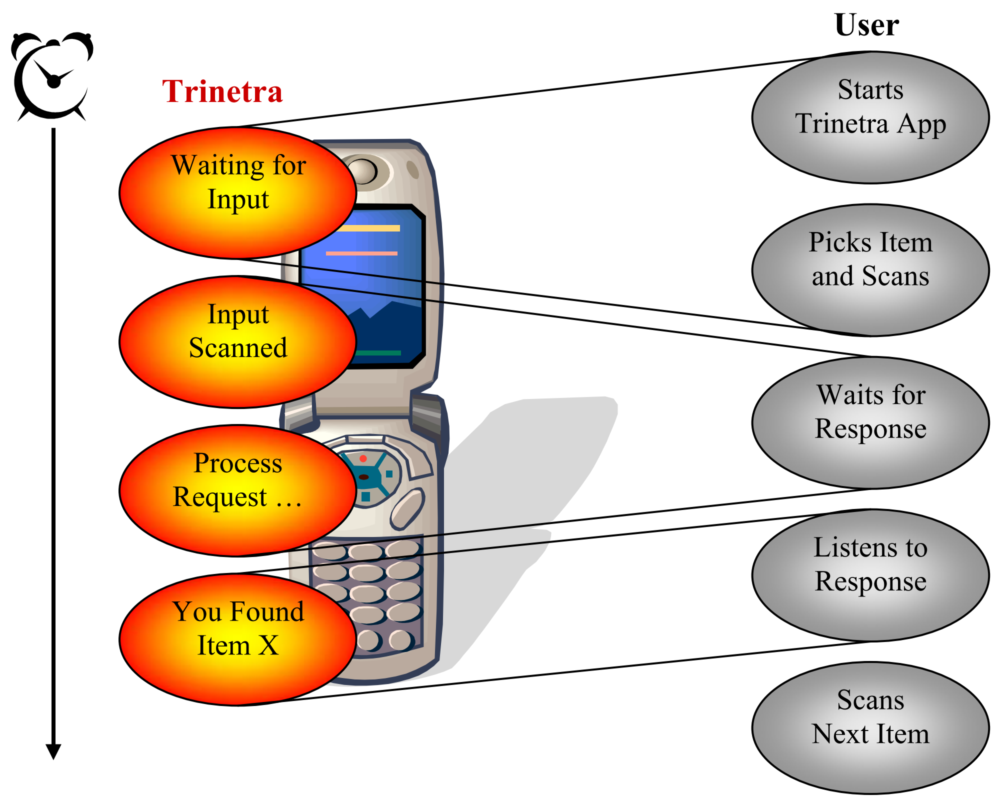

Figure 7: Mapping User Interactions to Trinetra Actions.

Figure 7 maps the user's perspectives to the Trinetra application's lifecycle, thereby detailing the typical interactions between the user and Trinetra. The vertical arrow in Figure 7 represents the march of time. The text strings displayed in the ovals to the left-hand side of the figure represent the audible messages that Trinetra returns to the user; the text strings displayed in the ovals to the right-hands side of the figure represent the steps in the user's interaction. Working from top downwards, the user launches the Trinetra application, listens for an input message ("Waiting for Input"), and then scans an item, receiving a notification when this is accomplished ("Input Scanned"). In the second phase, the user waits for a response from Trinetra while listening to messages indicating a request is undergoing processing ("Process Request …"). In the third phase, the user receives either an audible response, either a positive ("You Found Item X") or a negative result ("Item X Not Found", which is not shown in the figure) to the request. At this point, the user can proceed to scan the next item or shut down the Trinetra application.

4. Implementation Details

End-to-end, the Trinetra system implementation can be considered to be a three-tier distributed system, as shown in Figure 5; 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 connection to the BaracodaPencil. When using the IDBlue RFID reader, Trinetra employs a deferred-synchronous protocol (where the Trinetra expects a response or acknowledgement from the device, but does not block waiting for it) to query for pending data from the first-tier input device. 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 containing descriptive RFID or UPC information.

4.1. Trinetra 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 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. Trinetra Phone-to-Server Infrastructure Protocol

The Trinetra protocol from the phone to the server is comprised of five operations supporting two request types. The code snippet below lists the operation types supporting UPC and RFID inputs.

```
typedef unsigned char OpType;
const unsigned char OT_RequestRFID = 0x00;
const unsigned char OT_ReplyRFID   = 0x01;
const unsigned char OT_RequestUPC  = 0x02;
const unsigned char OT_ReplyUPC    = 0x03;
const unsigned char OT_Nack        = 0x04;
```

Currently, the application 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 in our current prototype.

Upon accepting a new connection from a cell phone, the remote server launches a new thread to handle the lifecycle of the invocations from that phone. This thread-per-connection concurrency model is adequate for our current Trinetra deployment needs 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 persistent cache. Successful lookups are returned using the respective reply operation-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, would be handled locally using the store's inventory-management that is populated with descriptions of the RFID tags embedded within the store; in this case, access would be 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. Our intent with supporting the use of RFID scanning within a store environment would be to integrate local inventory-management pricing and provide local product lookup, in the event that RFIDs become prevalent in grocery-store settings of the future. We acknowledge that our protocol does not support unique store-identification.

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 of our user interface is the TALKS software. TALKS effectively enumerates the text contained in Symbian user-interface elements, such as `CAknGlobalNote`, `CAnkPopupList` and `CAknTitlePane`, and then reads aloud the interface options. During user navigation, when a UI element comes into focus, it 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 closing the Trinetra application. The left hotkey contains a short menu with options to connect and 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. The application can be gracefully exited at any time during the application lifecycle by using the right hotkey. After scanning a UPC barcode or an RFID tag, the application will notify the user that input has been received. Finally, the application will return an “Item Not Found” message, or the positive results of the query. Figure 8 shows a return value from a successful lookup of the UPC on a bag of cookies.

We also note that, in our effort 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. Thus, while the Trinetra application is running, the user cannot receive other phone messages. We expect to eliminate this restriction in the future.

Figure 8: Trinetra's Results on Scanning an Item.

5. End-User Experiences

We acknowledge that we are yet to perform any systematic or comprehensive end-user testing of Trinetra. Our testing has so far been limited to a single blind user, Dan Rossi, who also guided us with valuable feedback and insight throughout the development of the Trinetra technology.

The picture to the left here shows Dan Rossi using the Trinetra technology at the CMU campus store, Entropy. In his left hand, he holds the BaracodaPencil, and in his right hand, he holds the Nokia 6620 cell phone that he already owns and uses. A Bluetooth head-set (in his right ear, not shown in the picture) provides with him audible notifications from the Trinetra system. 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, and including this test-run 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 before it can provide any scanned output. From our experience with our blind user, we have come to understand that we should never underestimate a blind person's tactile senses and capabilities. Within minutes of being given a new grocery item, Dan Rossi was able to locate the barcode through various length-wise and breadth-wise surface scans of the item; once he had identified the location of the barcode, he used this information to locate the barcode on other packages of the same type, e.g., a different kind of cookie from the same vendor.

We provide below, a quote from Dan Rossi (December 2005), 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 team designed, developed, and tested a concept that not only works, but is completely useful. This is not a sensor net vest of pager motors and scanning hat that is a neat idea, but not practical in the least. The Trinetra system is practical and useful and I commend the team on their work.

Having witnessed the value of our work to one blind user, we intend to carry out more comprehensive testing with other blind users, and to look for more systematic ways of evaluating Trinetra empirically.

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 [12] 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 has been estimated to be between $3,500-4000.

Other systems have focused on provided enhanced navigation capabilities for the blind. Drishti [11], 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 [3] onsists of a special cane mounted on a two-wheeled axle unit with onboard sensors for obstacle avoidance. Guido [10] from Haptica Corporation is a robotic walking frame that also targets obstacle avoidance. HARINOBU-6 [14] rom 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.

We cover below some of the commercial products that address the needs of blind users in grocery-shopping or other purchasing contexts.

I.D. Mate (shown here to the left) [6] is a talking barcode scanner that sells at up to $1559 per unit. It is a portable, 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. Of particular advantage is the omni-directional scanner which eliminates the need to visually and precisely locate the barcode on any grocery item. I.D. Mate also allows voice messages to be recorded and associated with any scanned item, in case the user wishes to recall important information. Another advantage is that custom barcodes can be generated and affixed to items without barcodes on them. The entire unit measures 11” x 4” x 4” and weighs less than 2 lbs.

There exist more products in the market-place to address the safety concerns of blind and visually impaired people with respect to identifying their prescription drugs [4]. More effort has been expended on this area than on grocery shopping because of the safety factor – prescription-related accidents have been estimated to cause about 125,000 deaths a year. All of the devices discussed below require the cooperation and involvement of sighted individuals in the product/prescription-identification process. We note that the prescription-related systems described below are not really appropriate for grocery-shopping scenarios.

ScripTalk (shown here to the left) [7] is a talking medication-label system that exploits radio-frequency identification (RFID) technology to tag and later identify prescription drugs for visually impaired individuals. When a ScripTalk user submits a prescription to a pharmacy, the pharmacy uses special ScripTalk software and a custom printer to generate and affix an RFID label containing information such as patient name, drug name, dosage, general instructions, warnings, prescription number, and doctor's name and telephone number. The ScripTalk user retrieves the information off the affixed label using a portable, electronic reader unit with a voice synthesizer to read the label details aloud. The reader sells at $325 per unit and the labels cost $1.75 each. There is also a significant investment on the part of the cooperating pharmacy in the ScripTalk software, dedicated RFID-label printer and the special RFID (rather than plain-paper) labels – the pharmacy might need to invest $4000-5000 for the system and training. Not every pharmacy might be willing to make this investment, even if the user purchases a ScripTalk reader.

The Talking RX system (shown here to the left) [19] contains a 60-second digital recording device that fits on the bottom of a standard prescription bottle. The pharmacist records the prescription-related information into the device using a recessed record button. The user can then listen to the recording later at home by pressing another button on the device. Talking RX can be reused and re-recorded over, and costs $24.95 per unit. Each prescription bottle requires its own unit, thereby requiring the user to buy as many units as the products that he/she wishes to identify in this manner. The use of Talking RX also requires the cooperation of the pharmacist; however, no dedicated equipment is required on the pharmacy’s part.

The MedivoxRX system (shown here to the left) [13] is similar to both the ScripTalk and the TalkingRX products. The prescription bottles are custom units, with built-in recording devices. The bottles cost $25 for a set of three, and are disposable, with a lifetime of 6 months. As with the other solutions, this requires cooperation from the pharmacist, but no investment of equipment on the pharmacy's part.

The Aloud Digital Audio Labeling System (shown here to the left) [2] consists of an audio-recording device that allows the pharmacist to generate an audio label for a prescription bottle. The user employs the Aloud player unit to listen to the audio labels by placing the tagged prescription bottle onto the player unit. As with the ScripTalk unit, this requires custom devices at the pharmacy's end, and requires the cooperation of the pharmacist in order to ensure effective usage. The user's Aloud player costs about $80, and the audio labels costs $5-10 each. Each prescription bottle requires its own audio label.

7. Looking Forward

We acknowledge the many limitations of our initial prototype. This was intended to be a proof-of-concept exercise in assisting blind shoppers. As we progress with the technology, we intend to incorporate many new features. For one, we would like to embed RFID tags into the Entropy environment (something we consciously avoided in this first prototype for cost and maintenance reasons) and to evaluate the effectiveness of RFID vs. UPC solutions. This would also involve developing techniques for labeling and navigating 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 future 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 magnifying lenses and with off-the-shelf OCR software. However, the costs and the effectiveness of the resulting solution remain to be evaluated, from the perspective of a blind user. A third, and significant, thrust will involve end-user testing with a larger population in order to understand the usability of our solution better. 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 through targeting the CMU campus shuttle.

8. Acknowledgements

We gratefully acknowledge the contributions of Dan Rossi, without whom this project would not have had the focused objectives that it did. Because of his involvement, we targeted the real needs of blind people, and we kept in mind the cost-effectiveness, portability and other issues that he raised. We also acknowledge partial funding support from the Pennsylvania Cyber Security Commercialization Initiative (PaCSCI).

References

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

[2] ASKO Corporation, http://www.askocorp.com/

[3] 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.

[4] E. Bryant, "Talking Medicine Identifiers," *Voice of the Diabetic,* vol. 13, no. 3, July 2003.

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

[6] En-vision America, Inc., *I.D. Mate*, http://www.envisionamerica.com/idmate.htm

[7] En-vision America, Inc., *ScripTalk*, http://www.envisionamerica.com/scriptalk.htm

[8] J. Gill, *Access-Ability: Making Technology More Usable by People with Disabilities*, 2004, http://www.tiresias.org/guidelines/access-ability/

[9] J. Gill, "The Forgotten Customers: One in Ten have Difficulty with Packaging," *Food, Cosmetics and Drug Packaging,* August 1995, http://www.tiresias.org/reports/packag.htm

[10] Haptica Corporation, *Guido,* http://www.haptica.com/walker.html

[11] 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.

[12] 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.

[13] MedivoxRX Technologies, http://www.rxtalks.com/

[14] 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.

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

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

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

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

[19] Talking Rx, Inc., http://www.talkingrx.com/