UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

Transit Information Access for Persons with Visual or Cognitive Impairments

Permalink

<https://escholarship.org/uc/item/15m8h5t0>

Authors

Manduchi, Roberto
Flores, German
Cizdziel, Benjamin
et al.

Publication Date

2014-07-01

Peer reviewed

Transit Information Access for Persons with Visual or Cognitive Impairments

G. Flores, B. Cizdziel, R. Manduchi,
K. Obraczka, J. Do, T. Esser and S. Kurniawan

Department of Computer Engineering
University of California, Santa Cruz

Abstract. We are developing a location-based information delivery system to facilitate efficient and safe use of public transportation by people who have visual or cognitive impairments. This system comprises Wi-Fi beacons (access points) that are placed at bus stations and inside bus vehicles. Users of this system receive information on their cell phone, without the need for GPS or for Internet connectivity. The system allows one to receive information about an upcoming bus at a bus stop and to select a specific bus line. Once the desired bus arrives, the system automatically connects to the access point on the bus vehicle and remains connected while the user is riding the bus. The user can specify a desired bus stop, and the system informs the user (by a speech message) with enough advance notice when the bus is approaching the stop. A prototype system has been developed and tested inside the University of California Santa Cruz campus (UCSC), and we are now planning for user studies with blind participants.

1 Introduction

Public transportation is key to independence for those who, for various reasons, cannot drive. At the same time, independent use of public transportation can be challenging for large portions of these individuals. For example, individuals with cognitive disabilities may have problems organizing and executing independent trips [5, 9]. Individual with visual impairments are also at a disadvantage when taking public transit [6, 1, 3]. A blind person cannot access printed information at a bus stop; cannot read the number of a bus arriving at a bus station; and, once on a bus, may miss the desired stop if the bus driver does not call all stops, or if the ADA-mandated audible announcement cannot be heard, for example due to loud background noise. These problems were highlighted by a survey conducted jointly by the LightHouse for the Blind and the City of San Francisco in 2007 with more than 50 blind passengers [6].

We are developing a novel approach for conveying real-time, customizable, multi-modal travel-related information to any passenger, directly on his or her own cell phone. Unlike previous research addressing a similar problem [2, 9, 1], our system does not require access to the Internet, and thus does not demand

subscription to a data plan. Information is pushed to one’s cell phone from Wi-Fi beacons that are placed in the public transit vehicles or at bus stops. In addition, users of this system do not rely on GPS data from their cell phone (as, for example, in [8]). Note that GPS data can be inaccurate or unavailable in some situations (e.g. urban canyons), and continuous GPS usage quickly drains a cell phone’s battery.

In our proposed system, Wi-Fi beacons are placed at bus stops as well as within bus vehicles (see Fig. 1(a)). Upon arriving at a bus station, users of this technology receive an acoustic signal from their cell phone, indicating that a connection with the local beacon was established. At that point, the user can interrogate the system to obtain information about that bus stop and about incoming bus vehicles. As soon as the desired bus arrives, a new connection is established with the in-bus beacon and maintained while the passenger is riding the bus, and more information, this time related to the specific bus ride, is made available. For example, the user is informed well in advance when the bus is approaching the desired bus stop, in time to get ready to exit the bus. Previous work [7] used Bluetooth beacons placed at bus stops to alert the user about the arrival of a desired bus. Use of Bluetooth beacons was also considered in [4] to provide information to a blind pedestrian about the status of a traffic light.

This contribution reports on current accomplishments of this project for what concerns the technical development of the system. More specifically, we discuss the implementation of the Wi-Fi beacons, the development of the client software and of the user interface, and early experiments assessing the ability of the mobile system to connect to the beacons. User studies with blind individuals are planned for the near future.

Fig. 1. (a) Conceptual system representation. Wi-Fi beacons are placed at a bus stop and inside a bus, providing different types of information to a user carrying a client software in a mobile device. (b) The location of the bus stops considered in our experiments in the UCSC campus.

2 System Description

The system consists of two main units: a client and server. The server is the access point application that communicates with the client to transmit relevant bus information. The server was designed to transmit specific information when a client is within its transmission range and the user has requested information. Users of this system interact with a client application, written in Java and implemented in an Android mobile device (a Nexus 7 tablet). Both client and server have been designed and implemented to be modular, responsive, and intuitive, allowing the user to receive relevant information when desired in a convenient modality.

2.1 Server

In our current system, Wi-Fi beacons are implemented in TP-LINK routers re-programmed with OpenWrt, a Linux-based operating system that provides a writable root file system with package management and other configurable scripts and tools. Routers are configured as Access Points (APs), and a global static primary IP address is hardcoded into each of the APs.

For our current prototype, two types of APs were reprogrammed and reconfigured: a bus stop AP and an in-bus AP. These APs look and work exactly in the same way, except for the type of information sent to the client. Bus stop APs, which are placed at bus stops, send bus routes numbers and other information such as the address or the name of the bus stop. In-bus APs, which are placed inside a bus, send the bus identifier and information about the bus stops encountered in the route. The information sent by the APs helps the client recognize the type of AP that is within range and perform adequate actions such as prompting the user to select a bus or alerting the user that their selected bus is within range and about to arrive. Note that an in-bus AP traveling on a bus may come within range of other bus stop APs (located on one or both sides of the street), or other moving in-bus APs (see Fig. 2(a)). The client must be able to differentiate between these situations and perform appropriate actions. The information exchanged must be short and compact to allow for fast lossless data transmission even when several users may be using the client application at a bus stop or inside a bus at the same time.

2.2 Client

The client is an Android application written in Java that incorporates the following hardware and software technologies: Wi-Fi, touch and gesture detection, text-to-speech (TTS) engine, socket and message communication protocols, database management system (DBMS), and Android services. Upon arrival at a bus stop, the user and the system must perform several actions that include: Start the Android application; Scan and detect nearby access points; Provide guidance to the user in order to select a bus stop and a bus to board; Provide information

Fig. 2. (a) Possible mobile and static access point configurations. The black circles, B, C, and D, are access points that have been placed at a bus stop. The red and orange double circles, A and H, are access points that have been placed in a bus. Moving from left to right, a bus access point can be by itself, come near two bus stop access points, come near another bus access point, or arrive at a location where only one bus stop access point is present. (b) Sequence diagram of events to scan, connect, and send data between the client and the server.

about the bus routes and arrival time; Detect all of the users touch screen inputs: gestures or single and double taps; Query the local database to retrieve bus arrival information; Provide the user with auditory feedback; Listen for in-bus access points; Connect, disconnect and switch between bus stop and in-bus access points and vice versa; Stay connected to an access point; Listen for other bus stop or in-bus access points. Some of these actions must occur asynchronously and without interrupting other actions that are concurrently working or about to start. In order to fulfill this requirement, the client unit was subdivided into five main subcomponents: user activity, Wi-Fi manager service, schedule and instructions service, gesture recognition engine, and text to speech engine. The Wi-Fi manager service is the most complex component since it has to manage several threads (actions) and must be running at all times in order to listen, connect, and disconnect from access points that are within range.

These components must follow an order of execution determined by the user activity. For example, connection and initial communication with an access point must occur before transmission of data. Fig. 2(b) shows an example of a sequence of phases leading to a connection. In this scenario, it took exactly 1 second to find an access point, 1.5 seconds to fully connect to the access point, about 1/10 of a second to transmit handshake data, and 0.5 seconds to scan for selected access points. In general, the scan, connect, and data transfer phases take shorter times than in this example, depending on multiple factors including the distance to the access point and the presence of nearby obstacles.

3 User Interface

The client interface is designed to be effective at communicating proper instructions to blind users, guiding users to their desired task, and providing intuitive usage modalities. It uses multi-touch interaction techniques, text-to-speech, and tactile feedback. The user inputs data through single and double screen taps; simple instructions and information are facilitated via speech; and verbal Yes or

No words or non-speech sounds are used to provide feedback as the user single or double taps the touch screen.

Interaction with the system occurs in two main situations: when the user arrives at a bus stop and wants to be informed once a desired bus has arrived; and when the user is in the bus and wants to be informed about when to exit the bus. A typical information exchange in the first case would proceed as follows. Upon arrival at a bus stop, the client automatically detects nearby APs and iterates through each one of them, prompting the user to select one if multiple bus stop AP have been found. Once selected, the AP at the bus stop transmits relevant bus information such as the AP location and bus routes. The system then prompts the user to select one of the bus lines that he or she wishes to board. Once the bus line has been selected, the system listens and waits until the selected bus comes within range, at which point the system disconnects from the bus stop AP, connects to the bus AP, and alerts the user that the bus is arriving. In the second case, when the user is in the bus, the system is already connected to the bus AP. The system provides periodic updates such as arrival time, next stop name, and confirmation of arrival. In addition, the system allows the user to inquiry about the current route.

The set of instructions and confirmations that are used to guide the user during interaction with the system have been implemented in a hash map structure to allow for fast and easy retrieval and expansion. The client application sequentially iterates through these sets or dictionaries as a state machine, moving from a state to the next, and speaking the correct instruction, question, or confirmation given the current state of the system. For example, when the user opens the application, the system greets the user by speaking “*Welcome*” and then it asks the user “*Do you wish to connect to network X?*” Depending on the user response, the system provides a proper confirmation such as “*Bus N has been selected*” or “*The arrival time is ...*”. At any given state of the system, a phrase or word is grabbed from the dictionary, parsed to fill in any unknown information such as the bus number or the network name, and then sent to the TTS engine, which speaks it.

4 Experiments

To test the system and its various components under different conditions, we conducted a total of 41 trials comprising the following scenarios: (1) Walking from and to an access point in open and busy areas; (2) Remaining in the bus for at least 20 minutes; (3) Multiple situations at a bus stop.

Scenario 1. An AP was placed in open and in busy areas, with the user carrying the client system walking away and towards the access point. Open areas are areas with no buildings or large objects present (e.g., an open field). Busy areas are characterized by buildings and large objects surrounding the access point (e.g. a street surrounded by houses and trees). In both cases, we investigated any Wi-Fi connectivity issues that may occur due to the obstruction or disruption

of the radio signal, and obtained estimate ranges for access point discovery and connectivity.

Scenario 2. We placed the AP inside a bus, with the user remaining in the same bus while connected with the AP for extended periods of time. These tests were designed to ensure that the application remained connected to the access point continuously in realistic situations.

Scenario 3. We tested the system in four different situations with a potential user arriving at a bus stop (shown in Fig. 1(b)), and with a bus subsequently arriving at the same stop. More precisely, we tested (a) connection to a bus stop AP upon user arrival to the bus stop, (b) connection to a in-bus AP when no bus stop AP was present, (c) switch from the bus stop AP to a in-bus AP upon bus arrival, and (d) connection to a in-bus AP as the user entered the bus, then remaining connected to the in-bus AP for the duration of the trip. In all of the situations mentioned above, the system was tested for discovery and connection time, connection switching performance, and transmission range were measured. A single router, shown in Fig. 3(a), was placed at the bus stop, while another router was placed in a shuttle bus that came to the bus stop every 20 minutes. (The UCSC Dept. of Parking Transportation and the UCSC Police were notified in advance of the experiments.) The “open area” tests were performed at the UCSC West Field (a large open field without trees), while the tests in the “busy area” were performed at the Science Hill bus stop (Fig. 1(b)). Transmission range measurements from and to the access point were recorded for both environments.

The application was installed in a Nexus 7 tablet and a record of the trials was kept. A trial was declared successful if the client was able to connect to an access point within a reasonable amount of time and if the client and server were able to communicate with each other without a single data packet loss. In general, results show that there were no discovery, connection time and transmission range issues for 5 out of the 6 different conditions. Only in one condition (connection switch when the bus approaches the bus stop) the system was not able to communicate properly with the in-bus access point for the initial trials; proper revision of the software led to successful connection trials in subsequent trials.

An example of connection sequence with timing information is reported in Fig. 2(b). Typical transmission ranges are shown in Fig. 3(b). It was observed that the *effective range* (i.e. the range in which the client is able to connect to the server, send information, and remain connected) was of approximately 55 yards (165 feet). The *actual range* (i.e., the range in which the client is able to detect the server but not to connect or transmit information) was of approximately 70 yards (211 ft).

5 Conclusions

We have described the design and implementation of a system that will provide personalized, just-in-time public transit information to a person who, due to visual impairment or cognitive disability, may have difficulty using a bus. We are now planning for experiments with visually impaired participants who will test

Fig. 3. (a) Access point prototype. The system consists of a pre-configured router and a 12V battery. This system was placed in a bus stop and inside a bus. (b) Transmission range of a pre-configured access point. The effective range is the range in which the client is able to connect to the server, send information and remain connected; whereas the actual range is the range in which the client is able to detect the server but neither is able to connect nor transmit information. Any access point located at a distance greater than 211 ft is considered out of range.

the system. Measurements will be taken concerning the quality of user interface on the tablet, the ability to easily navigate the menus and input data, and the ability to seamlessly switch connection to the in-bus AP once the bus arrives. Based on the results with these tests, we will adjust the system parameters and fine tune the user interface, in preparation for more thorough tests involving multiple campus shuttles and bus stops at UCSC.

Acknowledgments

The work described in this paper was supported by a contract from the National Academy of Science, issued under the Department of Transportation Award No. DC-26-7321. Dr. J. Coughlan from SKERI, Dr. A. Karshmer from USF, and Mr. P. Cantisani, provided invaluable comments to an original draft of the project report.

References

1. Azenkot, S., and Fortuna, E. (2010). Improving public transit usability for blind and deaf-blind people by connecting a Braille display to a smartphone. Proc. ASSETS 2010, Orlando, FL.
2. Babeau, S., Labrador, J., Winters, P.L., Perez, H., and Georggi, N.L. (2008). The Travel Assistant Device: Utilizing GPS-enabled mobile phones to aid transit rides with special needs. Proc. 15th World Congress on Intelligent Transportation Systems, NY.

3. Banatre, M., Couderc, P., Pauty, J., and Becus, M. (2004). Ubibus: Ubiquitous computing to help blind people in public transport. Proc. MobileHCI 2004.
4. Bohonos, S., Lee, A., Malik, A., Thai, C., and Manduchi, R. (2007). Universal real-time navigational assistance (URNA): An urban Bluetooth beacon for the blind. Proc. ACM SIGMOBILE International Workshop on systems and networking support for healthcare and assisted living environments.
5. Cain, A. (2004). Design elements of effective transit information materials. FDOT Final Report, November 2004.
6. Lanzerotti, R. (2007). Blind and low vision priority project. Report from the Light-House for the Blind and San Francisco Mayors Office on Disability.
7. Lim, J. T. F., Leong, G. H., and Kiong, T. K. (2008). Accessible Bus System: A Bluetooth application. In *Assistive Technology for Visually Impaired and Blind People* (pp. 363-384). Springer London.
8. Silva, J., Silva, C., Marcelino, L., Ferreira, R., and Pereira, A. (2011). Assistive mobile software for public transportation. In Proc. UBICOMM 2011.
9. Winters, P.L., Barbeau, S., and Georggi, N.L. (2010). Travel Assistance Device (TAD) to help transit riders. Final report for Transit IDEA Project 52.