*Location-aware information delivery with *comMotion*

Natalia Marmasse, Chris Schmandt

MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139, USA
{nmarmas, geek}@media.mit.edu

comMotion is a location-aware computing environment which links personal information to locations in its user's life; for example, *comMotion* reminds one of her shopping list when she nears a grocery store. Using satellite-based GPS position sensing, *comMotion* gradually learns about the locations in its user's daily life based on travel patterns. The full set of *comMotion* functionality, including map display, requires a graphical user interface. However, because it is intended primarily for mobile use, including driving, the core set of reminder creation and retrieval can be managed completely by speech.

1 Introduction

It is evidence of our hectic and mobile lives that many of us carry portable computers and wireless communication devices everywhere we go. Yet, these devices provide few services which are responsive to location, and location matters; I want my list of books recommended by friends when I have time to browse and am at the bookstore. *comMotion* seeks to address this problem, so that we are reminded about the important meeting on the way to work and told that we need to buy milk as we are about to drive by the grocery store on the trip home, thus providing just in time information delivery. The user interface is critical for systems which are meant to be always on and available; *comMotion* presents both graphical and speech interfaces to its core set of functions.

comMotion knows its latitude and longitude from the satellite-based Global Positioning System (GPS). But coordinates must be translated into positions that are relevant to the user, and these obviously vary greatly from person to person. Users neither know nor care about such coordinates; rather they identify "home", "work", "school", "post office", etc. Although a map could be used to specify such points, why should users spend valuable time filling out detailed property lists for a system which has yet to prove its value to them? Instead, *comMotion* learns salient locations by

* This paper appears in the HUC 2000 Proceedings, pp.157-171, © Springer-Verlag

observing its user's travel over time, and periodically inviting him to name or classify a frequented coordinate.

comMotion keeps "to do" lists for each location, or class of locations (a user may shop at multiple "grocery stores", for example). These lists consist of text and voice entries. Other users may also send reminders to the user at some specific location, and (with consent) query the user's position. *comMotion* also provides mobile access to location-based information from the Web, such as locations of nearby banks, and delivers other information at specified times and days of the week.

A graphical user interface is most appropriate for display of map data and allows full control of administrative functions. But because *comMotion* is meant to be used while mobile, it also demands both alerting and a user interface which function while the user's hands and eyes are otherwise busy, such as while driving or cycling. To this end, core *comMotion* functionality is available through a non-visual speech and auditory user interface.

Modest user evaluation suggests that *comMotion*'s promise is attractive, but user interface design, including multiple modalities, is as important as size. For example, speech is well suited for driving, but at least initially some users find it embarrassing to call attention to themselves while speaking to their computer on foot in a public locale.

2 Related Work

Most previous location-aware applications have used predefined content and/or locations.

C-Map [1] is a tour guidance system which, based on location and individual interests, provides information to visitors at exhibitions. CyberGuide [2] is a collection of intelligent tour guides which provide information to tourists based on knowledge of their position and orientation. Metronaut [3] is an application developed for schedule management and guidance instructions for a visitor to a university campus. City Guide [4] enables a user to see his position on a map and request restaurant and hotel information. These applications have predefined content based on location and they are user-independent.

The Olivetti Active Badge [5] was used in several systems, for example to aid a telephone receptionist by dynamically updating the telephone extension a user was closest to. Augmentable Reality [6] allows users to dynamically attach digital information such as voice notes or photographs to the physical environment. Audio Aura [7] provides information via auditory cues based on people's physical actions in the workplace. These systems use predefined locations and are designed for users to find each other or objects in the environment.

The Forget-Me-Not [8] was a wearable device which recorded interactions with people and devices, and stored this information in a database for later query. The Remembrance Agent [9] provides text information relevant to the user's context, for example class notes when entering a specific classroom. These applications remind the user of past events associated with a location whereas *comMotion* associates events in the future.

comMotion can have predefined content associated to locations, however its main feature is user-defined content and the possibility to subscribe to Web content based on location. To the best of our knowledge no other system observes the user's mobility data to independently learn the frequented locations.

3 Overview

A user's interaction with *comMotion* begins with the location-learning agent. It observes the user's frequented locations over time and allows them to be labeled.

Once a location has been defined, a to-do list is associated with it. A to-do list is a set of text items or digital audio recordings; these may be ticked off once completed. When the user is in the relevant location, he will hear an auditory cue alerting him that he has items on the associated to-do list. In addition, other users can also send him reminders to his virtual locations. These reminders resemble the common 3M Posts-its™ and can be sent via regular e-mail.

The user can also subscribe to information services, such as headline news, weather reports and current movie listings; the subscription is per location and different schedules can be made for different days. For example, the user could request to receive a list of the movies showing at the local cinemas when leaving work on Fridays. In addition, *comMotion* can provide maps showing the user's current position together with neighbourhood locales, such as banks, movie theatres or grocery stores.

The data types and functionality of *comMotion* require a multi-modal user interface. Its alerting function reminds the user to, for example, buy milk when nearing a grocery store. Since we can rarely view a screen while travelling, this cue cannot be visual so an auditory alert is used. Map information, on the other hand, is best displayed visually; although we have in the past explored giving driving directions by voice [10], this requires a detailed and up-to-date street database, not just maps. *comMotion*'s reminders are either voice or text. Although its graphical interface is more extensive, *comMotion*'s core functions are accessible by speech input to allow mobile use.

comMotion, with the appropriate hardware, and with some modifications in the software, could accommodate different architectures. A number of scenarios can be envisioned, each adapted for different life-styles, or different modes of mobility: a wearable on-the-go architecture for the highly active, such as, cyclists; a car architecture for the more sedentary; a briefcase architecture for the mobile individual, for example, knowledge workers; or a stripped down kids architecture. These are all variations on the same *comMotion* system, tailored to different needs. What changes is the hardware, the user interface and the features included in the system, which range from full-fledged to stripped down variations. A first *comMotion* prototype was built in order to evaluate its feasibility and usefulness. In highly active situations, such as cycling, the components are put in a fanny pack or carried in a shoulder bag.

4 Architecture

The hardware includes a portable PC, a GPS receiver, a CDPD modem and a Jabra earphone speaker with a bone conductive microphone.

The human-computer interface, on the client side, is composed of both speech and graphical user interfaces. The former includes speech recognition and text-to-speech synthesis and was developed using AT&T's Watson SDK (software development kit). The Watson product is an integrated, Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) synthesis system which complies with the Microsoft Speech API (SAPI). The ASR engine uses phoneme-based sub-word analysis and, therefore, supports speaker-independence and continuous speech recognition. The *comMotion* speech server, developed with the Watson SDK, operates on the client device.

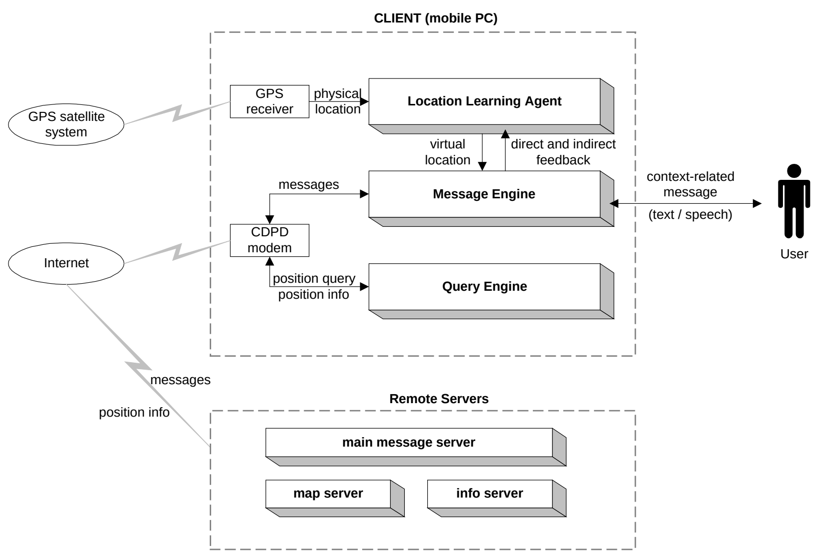

Figure 1: The architecture of comMotion showing the three main modules of the client application and its connection to the server

The client application communicates via TCP/IP sockets to all the different server processes, hence, these processes could easily be transported to the client device or to any other computer with Web-server capabilities. In the current setup, with only the speech server on the client device and all other servers on a remote station (Figure 1), even if connectivity were lost the user would still have full access to his to-do lists and any reminders which had previously been downloaded. Furthermore, since all position tracking and analysis are done on the client device, these would not suffer from lack of connectivity. Reminders sent from other users are immediately downloaded to the client device where they are stored until delivery time. If the server cannot access the client, these new reminders are saved until connectivity is re-established and they can be downloaded. Lack of Internet connectivity means information from on-line sources will not be accessible; likewise, no maps or related information can be downloaded.

5 Location and Learning

When designing location-aware applications it is vital to know where people are. *comMotion* focuses solely on outdoor tracking by means of a GPS (Global Positioning System) receiver; indoor tracking requires an additional location sensing system. GPS, until very recently, had an accuracy of less than 100 metres due to a deliberately induced error signal, preventing hostile military applications. Although position information is still inexact, it is often sufficient to know the vicinity (within 10 m) in which the user is located.

It is essential to design location-aware applications to take into account the accuracy and reliability of available location. Not only is GPS data not very accurate, but it is also not always available. GPS signal is lost when entering most buildings and the so-called concrete canyons in urban areas make reception difficult. The fact that most buildings are GPS opaque was exploited advantageously, permitting a simple learning mechanism for the locations of buildings.

Latitude and longitude coordinates are obtained via a GPS receiver connected to the client's serial port, using the NMEA-0183 protocol. All data is analysed for frequented locations, however, at present, only locations such as buildings can be identified, since the system recognizes locations where GPS signal is lost. After losing signal within a given radius on three different occasions the agent infers that this must be a building and marks it as a salient location. The user is prompted for a location name, which he can either designate or tag at a later stage by seeing the location on a map. Once tagged, the virtual location has a to-do list associated with it. However, if the location is of no interest, the user can indicate that it is to be ignored. For example, a frequented T-stop (metro) would be identified by the agent but the user would typically not want to have context-related information (to-do list items, electronic reminders, etc) delivered there.

To compute its position, a GPS unit must receive a signal from various satellites: three for a 2D fix (latitude and longitude) and four in the case of a 3D fix (latitude, longitude, and altitude). In urban areas, shadowing from tall buildings often occurs, and leads to long delays in position acquiring. Initially, locations were identified both upon user arrivals and departures. However, it was found that the GPS receiver often took several minutes to acquire its location when exiting a building and, therefore when the signal was regained, the user was no longer at the place he had left. Depending on his mode of transportation, this could be a considerable distance; hence false locations were identified.

The GPS receiver is polled for data every ten seconds but does not return a coordinate until it "sees" satellites. In the examples which follow, each line begins with the time of polling. If the data is returned immediately, the next line is ten seconds later. A longer delay indicates a poll waiting for the GPS data.

Date	Poll time	Latitude/longitude data	
03/30	09:43:53	4223.351,N,	07104.613,W
03/30	09:44:03	4223.153,N,	07104.625,W
03/30	09:44:13	4223.092,N,	07104.641,W
03/30	09:44:23	4221.867,N,	07103.786,W
03/30	09:54:14	4221.870,N,	07103.787,W
03/30	09:54:24	4221.872,N,	07103.789,W

Figure 2: GPS data showing how signal is lost when arriving at a building

As can be seen in the GPS data (Figure 2), at 9:44:13 the user arrived at location (4223.092N, 7104.641W). Ten seconds later the receiver was polled for data but a signal was received about 10 minutes later, locating him at (4221.867N, 7103.786W). If the GPS receiver loses signal two more times within a given radius of (4223.092N, 7104.641W), then this is understood to be a building and, consequently, a frequented location.

Date	Poll time	Latitude/longitude data	
03/29	16:34:09	4221.686,N,	07105.339,W
03/29	16:42:24	4221.687,N,	07105.341,W
03/29	16:42:34	4221.685,N,	07105.392,W

Figure 3: GPS data showing how signal was acquired only several minutes after departing from a building

In this second example (Figure 3), the user turned on the unit and left the office at 16:34:09, however, the GPS receiver took several minutes to acquire a position after exiting the building. Data was received ten seconds before the next entry, that is, at exactly 16:42:14. So in this case, data was received eight minutes and five seconds later –this includes the time it takes to descend three flights of stairs besides the acquiring time once outside the building. The user was identified at location (4221.686N, 7105.339W), several hundred metres from the building he exited.

The algorithm was modified to detect only arrivals. So from the data in the first example, (4223.092N 7104.641W) would be considered a location to analyse, however (4221.867N, 7103.786W) would be ignored. It takes the agent twice as long to identify a location but the false ones formerly recognized were eliminated. If previously your “home” location could be identified in two days, that is, after a sequence of arrive-depart-arrive or depart-arrive-depart, with the new algorithm it takes a sequence of three arrivals.

It will be necessary to further modify the learning algorithm since not all buildings were found to be GPS-opaque, that is, the GPS signal is sometimes received while within a building. Therefore, the data must also be analysed for stationary points. Even when a GPS receiver is static, the data indicates fluctuations of many metres; this requires compensation. The analysis to identify a stationary receiver is also necessary for the *comMotion* car architecture, where the GPS receiver will be permanently installed in the car, as opposed to in a mobile device with the user. When the car is stationary, such as in the grocery store’s parking lot, it may be in a salient location. Also, a parking lot may be large enough to cause confusion between multiple trips.

Any monitoring system raises many privacy issues, that is why in *comMotion* all the location tracking and analysis is done solely on the client device. Hence, the user's privacy is safe to the extent that the client device is safe.

6 Graphical User Interface

comMotion consists both of a GUI and a speech user interface which complement each other and are appropriate for different modalities.

Location-learning Module. When a frequented site has been identified, the user is prompted to tag it with a virtual location name. A speech dialogue is followed by the appearance of a visual interface through which the user specifies the location name and associates a to-do list with it.

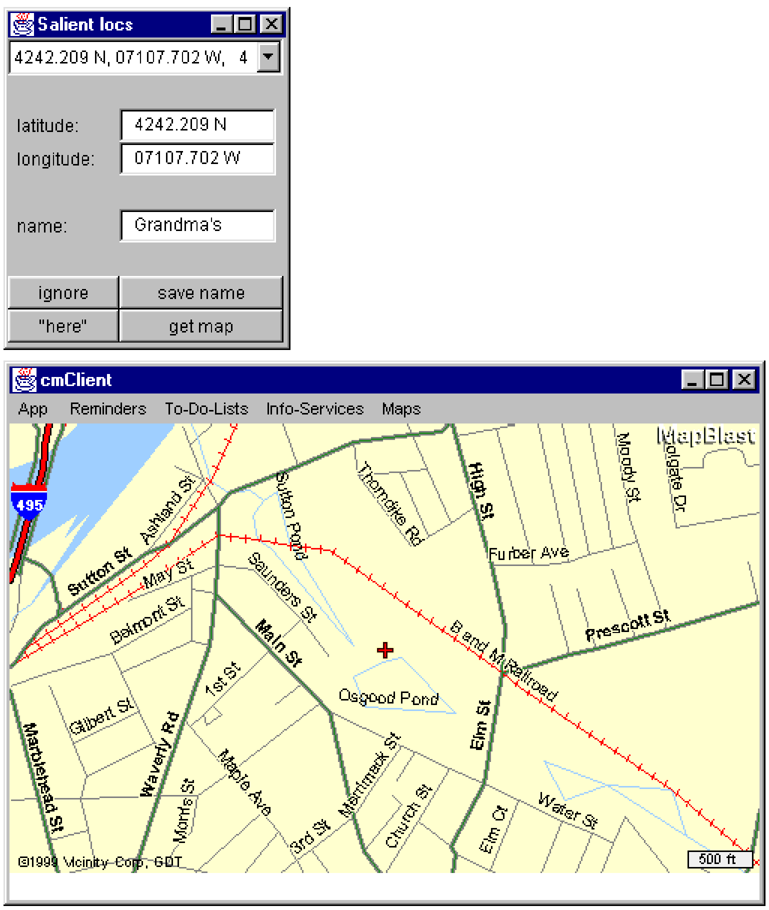

Figure 4: Visual interface used to see a salient location on a map and tag it with a virtual location name

A visual interface is essential since the location name must be text, and not audio. By default, the associated to-do list name is identical to the location name, however, an existing to-do list can also be chosen, since these lists can either be site specific or shared. A user who shops at three different grocery stores would probably only want one shared grocery list. On the other hand, places such as "home" and "work" would presumably have very different tasks associated with them, and, therefore, separate to-do lists would be preferable.

The speech dialogue occurs only when in situs however the tagging can also be done later; the user can see the exact location on a map (Figure 4). If the location is of no interest the user can indicate that it is to be ignored.

To-do Lists. A to-do list is associated with each defined virtual location. When the user is in the relevant location, he will hear an auditory cue ("psst") indicating that he has items on the associated to-do list and the visual interface will be displayed (Figure 5). Visually, the list resembles its real world paper parallel, on which items are simply listed and can be checked once they have been done. However, in *comMotion*, the to-do items can either be text, typed in using the graphical interface, or recorded audio.

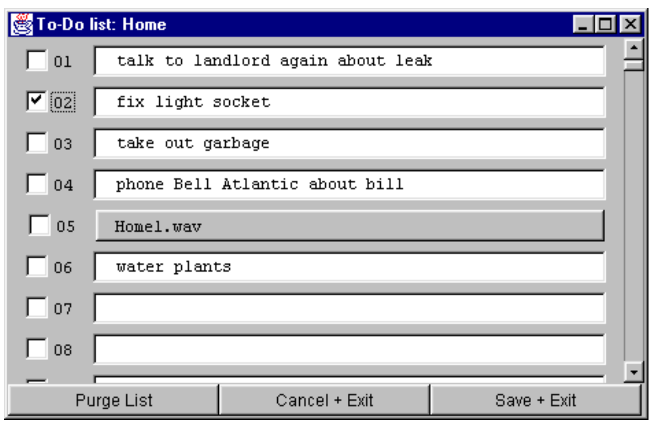

Figure 5: A to-do list with both text and audio items

The list items can be viewed and scanned using the visual interface or heard and scanned using the speech interface. Text items are synthesized to speech, whereas audio items are played. The recorded audio items are not transcribed to text, rather they appear as a button in the list, and can only be retrieved in the form of audio. Once the task has been completed, the item can be checked-off and the list purged.

The audio items can be recorded using either a visual interface or through speech commands. Although a file name is shown in the interface, the user simply specifies the list name. In the graphical interface, the user can start and stop the recording, and play back the audio until satisfied with the recording. Saving the file will automatically insert an audio item button in the specified to-do list, enabling future playback.

Reminders. The metaphor used for the reminders is the widespread 3M Post-its™; the visual component resembles them in shape, size and colour (Figure 6).

Context-aware reminders for *comMotion* mobile users can be sent via the regular e-mail system by specifying the location name in the subject line. This is important since the mobile user must be accessible to the rest of the world. Reminders, or electronic Post-it notes, can also be sent from another *comMotion* client, however, unlike the to-do items, they are text-only.

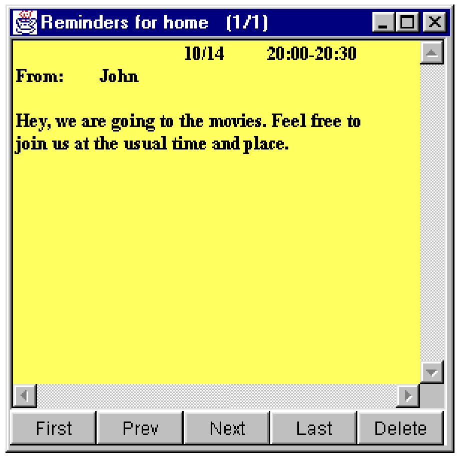

Figure 6: Visual interface component to view and scan reminders

The reminders are posted to a specific location and can be constrained to a certain date or date and time range. When the user is in the relevant context, he will hear an auditory cue indicating that he has some relevant reminders and a visual component resembling a Post-it note will appear on the display (Figure 6). These notes can then be viewed and scanned using the visual interface or heard and scanned as synthesized audio by means of the speech interface.

When creating a new reminder, the user can specify the frequency of repetition (daily, weekly, monthly, or none), the repeat increment, the day of the week, and the date range. For example, one could create a reminder for a certain event that takes place on Mondays, every other week, for the next three months. All these parameters can be defined whether sending the reminder from a *comMotion* client or via e-mail, however, the *comMotion* client provides a more convenient interface for this task. When sending a reminder via the *comMotion* interface, both the user and the location are chosen from a list, which is composed of all this particular user's locations, as well as all the locations of other *comMotion* users to which he has privilege. When sending the reminder through e-mail, if the format cannot be parsed out correctly, the e-mail sender will receive a reply indicating the problem and the correct format to be used.

Map Module. The user can view a map of his current location showing features such as banks, movie theatres, grocery stores, etc, acquired from MapBlast's [11] Web database. The display shows a map as well as the address and telephone number of the sites of interest. The input necessary for this module are latitude and longitude coordinates, hence it is not limited to the locations that *comMotion* has already learned. Undoubtedly it will be more useful when the user is in an unfamiliar neighbourhood.

Information Services. The message engine can also deliver content such as headline news, weather reports and current movie listings from information sources on the Web. As the requested information may vary depending on context, the user can subscribe to different information services based on location, and special schedules can be made for different days. For example, the user might request to receive a list of the movies showing at the local cinemas when leaving work on Fridays.

When the user is in the relevant context, he will hear an auditory cue indicating he has location-based information that he has subscribed to. The data will be displayed in the visual interface and may also be synthesized to speech.

Headline news is downloaded and parsed from the msnbc.com site; weather reports are taken from wunderground.com, and the movie listings are from the sidewalk.com site. The default city for the weather information is Boston but this can be modified by the user. Likewise, the default city for the movies (Cambridge) can be changed. The weather site supports most large cities in North America as well as several others around the world. However, the sidewalk.com site only supports a subset of the larger, or most important cities in the US. The data is parsed out and formatted for the visual display. Although the text can be synthesized to speech, its format was not specifically adapted for that purpose.

Query module. A user can be queried for his whereabouts via a *comMotion* client or through the regular e-mail system. In either case, the request goes to the main server where the priorities are checked. A *comMotion* user can establish priority levels per location and per user and can log by whom and when location information was requested. The log file includes: by whom he was queried, where the queried user was at the time, what information was released, and the date and time of the query. Querying from a *comMotion* client is done by choosing from a list of other *comMotion* user names. If the request was sent by an authorized person, a reply such as “Jane is at work” will be received; however, if queried by someone unauthorized, the response will be “Jane is incognito!”. A query request via e-mail is performed by sending a message, to the *comMotion* user who is being queried, with “cm query” as the subject.

7 Speech/Auditory Interface

comMotion is meant to be carried by its user at all times, and, almost by definition, location-based reminders will arrive while the user is busy doing something else – driving, walking, biking, etc. These factors suggest that the core memo management functions and alerting be speech input and auditory cues.

Speech is suitable for portable applications as speakers and microphones are small and require little power. Auditory input and output can be performed while the user's hands and eyes are busy, while driving or biking. More importantly, performing multiple tasks is more effective in terms of cognitive resources, if the tasks are performed using different sensory modalities [12].

The main downside of the auditory interface is that speech recognition is difficult, and its accuracy remains poor, especially in noisy environments. In addition, speech is transitory and serial in nature; it is difficult to take an auditory "glance" at a list to pick out the important item. Also, if the presentation is via text-to-speech synthesis, an even greater cognitive load is placed on the user [13].

Effective speech user interfaces must exploit its advantages while minimizing user frustration due to error and cognitive load. This influenced *comMotion* in several ways. Only a limited set of functions are made available by voice; these are related to hearing and modifying to-do lists and reminders. Limiting functionality allows for smaller speech recognizer vocabularies with resulting better recognition performance. Even if the less frequently used *comMotion* functions were speech accessible, users would most likely forget what to say to invoke them, increasing frustration with the recognizer. Actual reminders created by voice are simply saved as digitized audio; attempting dictation to text without visual feedback would be tedious at best. Although we have in the past explored giving spoken driving directions [10], *comMotion* limits map-based features to the graphical interface for simplicity.

Different auditory cues are associated with the various types of information (to-do lists, reminders and subscribed content information). Whenever the user has pending messages relevant to his context, the pertinent audio cue is heard. As the purpose of these cues is to alert the user without distracting him from his primary task, different ones can be chosen and in this way they may be as subtle as he wishes. The system default cues are currently a "psst" for the to-do list, a "ding" for the reminders and chimes for subscribed information. In the future new default cues may be chosen, based on the consensus of several users' preferences.

The user interacts with the system via speech commands, which are mostly two-word commands and should be intuitive. *comMotion* can be run in terse or verbose mode. When in verbose mode, the user will receive much more auditory feedback: after each speech command is recognized the system will repeat the command before performing the task. For example, if the user says "read item", the system will repeat "read item", before it synthesizes or plays back the relevant to-do list item. Once an auditory cue has been given, it will explain the reason for the cue. For instance, "You have five items on your grocery list". When choosing one of the locations from the list of names, after the user says "yes", the system will respond, for example, "bank".

Speech recognition is fragile; speech commands might not be recognized or they could be mis-recognized causing an unwanted action to be performed. Unlike in a visual interface, where the user can generally perceive what the system is doing, an audio interface is not transparent to the user. Therefore, explicit feedback is beneficial, especially for novice users. For example, when marking a to-do item as checked: in a visual interface the user will easily see if the wrong item was ticked, however in a speech interface he has no way of knowing unless the system gives explicit feedback, or he has enough experience with the system to trust it.

In *comMotion,* speech recognition is not continuously enabled since ambient conversation is often picked up and misinterpreted as direct input. However, since the objective of the speech interface is to enable system functionality in a hands and/or eyes busy situation, the user cannot be expected to manually activate the recognition by means of a push-to-talk button. Consequently, *comMotion* automatically activates the continuous recognition when it expects to receive speech commands. As long as the user interacts with the system, the continuous recognition remains enabled. For example, if the system alerts the user (with an auditory cue) regarding pending messages, the continuous speech recognition is activated, giving the user the opportunity to ask for the messages to be read. As long as the user gives speech commands, the recognition stays on. However, unanticipated speech interaction requires manually initiating the recognition. Once activated, it will remain enabled as long as it picks up audio.

Location-learning Module. When a new salient site has been identified, the user is prompted for a virtual location name. The system initiates the speech dialogue with: “This is a frequented location. Would you like to name it?". A negative answer (speech command: “no” or “nope”), or no answer whatsoever, terminates the dialogue, whereas a positive answer (speech command: “yes”, “yep”, “sure” or “okay”) from the user causes a visual component to appear, through which the user specifies the location name and associates a to-do list with it. The user can indicate that the location is to be ignored by saying “Ignore this location”, when prompted for a location name.

To-do Lists. The speech commands used to navigate within a specific list are: “first item”, “last item”, “next item”, “previous item” and “read item”. If operating in verbose-mode, after each command the system repeats the command before actually performing it. The command “check item” is used to toggle the check mark next to an item. If operating in verbose-mode, after performing the command, the system says, for example, “Item: fix light socket, is now checked”. The verbose feedback is useful for the user to gain confidence in the system and to understand the system’s behaviour in the case of mis-recognitions when a command is misunderstood for another. If the command is not recognized, the system answers: “Say that again”.

The user can initiate the dialogue to record an audio item by saying: “Choose list”. The system then recites the names of the lists, one at a time, until the user says “yes” to one of them. If the user does not want a specific list, he can say “no”, or simply say nothing –if no response is heard after five seconds, the next list name is recited. When recording an audio item via the speech commands, it is not possible to stop the recording once it has started. The system automatically records during 20 seconds. An auditory cue, a beep, is heard just before the recording begins, and once again after the 20 seconds timeout.

More sophisticated list management and audio browsing, inspired by VoiceNotes [14], will be implemented in future prototypes.

Reminders. The speech commands used to navigate between the reminders are: "first reminder", "last reminder", "next reminder", "previous reminder", "read reminder" and "delete reminder". If operating in verbose-mode, after each command the system repeats it before actually performing the command.

Speech Levels. The speed of the audio output can be modified with the commands "speak faster" and "speak slower" and the volume level with "speak louder" and "speak softer". The system defaults can be reset by "reset speech levels".

8 EVALUATION

For *comMotion* to be most effective, it should be used on a regular basis over a period of time. The system must first learn the user's frequented locations as only after these virtual sites have been established can location-related information be delivered. Therefore, two different parts of the system must be evaluated: the location learning feature and the delivery of information.

8.1 Location learning

GPS data was collected over a couple of months by the authors. This data was used to evaluate and make iterative improvements to the location learning algorithm. For instance, *comMotion* initially identified locations based on when the user arrived and when he departed; however as GPS can take several minutes to regain its location after leaving a building the user often "reappeared" at quite some distance. This gave rise to problems so the algorithm was modified and now locations are only identified when arrived at. The problems encountered and how they were resolved are described in detail in the section on learning. At present, the *comMotion* learning algorithm effectively identifies GPS opaque buildings after they have been visited three times.

8.2 Information delivery

Preliminary tests were carried out to evaluate the effectiveness of information delivery and to refine the initial interface design.

The *comMotion* system was taught three virtual locations: a local post office, a bookstore and a bank; all are within the same two blocks. The users were told that they had pending items on their to-do lists and reminders specific to each location. They were asked to walk in the general area of the different locales, choosing whichever route they preferred.

Feedback was received from 4 different people –two of whom are non-technical with very little experience with mobile computing devices, and two members of the Speech Group who have a deep understanding of the concepts used. Their comments can be divided into different categories:

Hardware. The hardware must become smaller and lighter, to be truly mobile, and the earphone must be wireless. One person noted that even a wireless speaker is unacceptable since it is uncomfortable and most of the time the system is not "speaking"; loud-speakers could be integrated into clothing or the frame of eye-glasses. The device should not be larger than the size of a PalmPilot and should perhaps be integrated into a device which already is carried around, such as a cellular phone.

Speech Input. Speech as a form of input is beneficial when in a hands and/or eyes busy situation. However, using speech commands can raise both social and privacy issues. People do not want to appear to be talking to themselves and may not necessarily want others to know what they are doing. One person refused to use the speech commands while walking around with the system. Speech recognition is still problematic and one of the users with a non-American accent had great trouble being understood. Ambient noise levels also proved to be a problem. In a car situation, both ambient noise and self-conciousness would be much less of a problem.

Precision and Alert Timing. GPS data is intentionally imprecise –when the user evaluation was done, accuracy was within 100 metres. For this application, exact position information is not required. When two different virtual locations are physically within metres of each other, however, due to the inaccuracy of the position data, one location is identified and not the other –that is, location shadowing. This can be solved by clustering the virtual locations and providing alerts for all the locations within the cluster. The lack of precision of position data also strongly affects the alert timing and auditory cues were sometimes given too late. Loss of GPS signal due to shadowing by tall buildings was also experienced.

Alerts and Data. Users indicated that the auditory alerts, when given at the right time, were useful since it enabled them to be doing something else, such as walking down the street and talking, and still draw their attention without losing their thread of thought. Since certain users didn't want to use the speech interface to retrieve the location-specific information, upon receiving an alert, they pulled out the visual display and intuitively navigated amongst the reminders. A couple of people suggested a tactile vibra-alert as an alternative to the audio one. It was also suggested that to-do list items could be prioritized and a deadline indicated.

The results are non-conclusive and more extensive evaluation must be done, however it seems clear that there is place and desire for such a system.

9 FUTURE WORK

The current prototype allowed us to develop location learning algorithms and do preliminary user evaluations. Based on these, the next version will include a more sophisticated speech UI on more portable hardware, allowing longer term evaluation of *comMotion* utility in daily life. We will also explore ways to increase location awareness of a community and the sharing of information, for example, between the members of a family.

References

1. Sumi, Y., T. Etani, S. Fels, N. Simone, K. Kobayashi and K. Mase. "C-MAP: Building a Context-Aware Mobile Assistant for Exhibition Tours". *Social Interaction and Communityware*, Japan, June 1998.
2. Long, S., D. Aust, G. Abowd, C. Atkeson. "Cyberguide: Prototyping Context-Aware Mobile Applications". *Proceedings of the conference on Human Factors in Computing Systems, CHI'96.*
3. Smailagic, A. and R. Martin. "Metronaut: A Wearable Computer with Sensing and Global Communication Capabilities". *Proceedings of the International Symposium on Wearable Computing*, IEEE, 1997.
4. Kreller, B., D. Carrega, J. Shankar, P. Salmon, S. Bottger, and T. Kassing. "A Mobile-Aware City Guide Application". *ACTS Mobile Communication Summit*, Rhodos, Greece, 1998.
5. Want, R., A. Hopper, V. Falcao, and J. Gibbons. "The active badge location system". *ACM Transactions on Information Systems*, 10(1): pp. 91-102, January 1992.
6. Rekimoto, J, Y. Ayatsuka and K. Hayashi. "Augmenta-able Reality: Situated Communication through Physical and Digitial Spaces". *Proceedings of the International Symposium on Wearable Computing*, IEEE, 1998.
7. Mynatt E., M. Back, R. Want. "Designing Audio Aura". *Proceedings of the conference on Human Factors in Computing Systems, CHI'99.*
8. Lamming, Mik and Mike Flynn. "Forget-me-not: Intimate computing in support of human memory". *FRIEND21: International Symposium on Next Generation Human Interface*, pp. 125-128, 1994
9. Rhodes, B. "The Wearable Remembrance Agent: a system for augmented memory." *Proceedings of the International Symposium on Wearable Computing*, IEEE, 1997.
10. Davis J.R. and C. Schmandt. "The Back Seat Driver: Real Time Spoken Driving Instructions". *Vehicle Navigation and Information Systems*, 1989.
11. Mapblast. http://www.mapblast.com/mapblast/start.hm
12. Allport, D., B. Antonis and P. Reynolds. "On the Division of Attention: a Disproof of the Single Channel Hypothesis". *Quarterly Journal of Experimental Psychology*, 24:225-235, 1972.
13. Luce, P. A., T. C. Feustel, and D. B. Pisoni. "Capacity Demands in Short-Term Memory for Synthetic and Natural Speech". *Human Factor*, 25(1):17-32, 1983.
14. Stifelman, L., B. Arons, C. Schmandt and E. Hulteen. "VoiceNotes: A Speech Interface for a Hand-Held Voice Notetaker. *Proceedings of INTERCHI '93*