GeoFoto: Mobile Viewing of Geo-Referenced Photos

Jonathan Effrat
Stanford University
Stanford, CA
jje@cs.stanford.edu

Aditya Mandayam
Stanford University
Stanford, CA
adityams@stanford.edu

Karenina Susilo
Stanford University
Stanford, CA
ksusilo@stanford.edu

ABSTRACT
GeoFoto is a mobile phone application that provides a way for people to navigate photos based on a frequent desire: knowing which way to travel. By letting users explore photos based on directionality, we are supporting the discovery instinct of users who want to explore the hypothetical scenario of what lies ahead along various paths.

Keywords
Geo-reference, location-awareness, GPS, camera phone, photos, distributed

INTRODUCTION
GeoFoto supports impromptu exploration of unfamiliar areas by those who want to experience an area without being dependent on a guide or guidebook to tell them what is worth seeing. GeoFoto allows users to navigate through the geographically tagged photos which other users took (for example on their cameraphones) as a means of facilitating the decision of which way to head next. It can also be used simply for amusement purposes or when wanting to use one's current location as a starting point for browsing photos other people took.

The application was originally designed to detect the location of the user at all times using GPS technology (more on this later). Users can then navigate to any direction (North, South, West, East, etc.) relative to their current position to browse pictures taken by other people within the direction selected. As the photos added span the globe, the user will be able to choose to explore as far in one direction as s/he wishes to.

RELATED WORKS
Marc Davis's MMM project in University of California, Berkeley supports taking photos and tagging them with a georeference and time stamp, and uploading them to the server.

Mor Naaman's work at Stanford University involves visualizations of georeferenced photo collections on desktop PCs. This is not directly relevant, since handheld interactions are quite different, but we did use Naaman's database of geo-referenced photos of Stanford, so that we already have pictures to include in our system and provide for users on campus.

Dan Maynes-Aminzade of Stanford University did work with distorting maps to consolidate viewing areas with georeferenced photos on PCs.

Erik Blankinship has done work on viewing historical geo-referenced photos. We are similarly fascinated by the use of other metadata, such as time or season to add value to users in the photo viewing process.

USER STUDY
A series of user studies were conducted to answer the following questions:

- What are users' natural mental models in deciding where to go when they find themselves with some free time?
- Can users navigate and browse pictures of different directions successfully with the GeoFoto application?
- What are the advantages and disadvantages of using GeoFoto to decide where to go?

Participants
The user study was conducted with four participants. All of them were Stanford University students. In the interests of time we were unable to find anyone who was not somewhat familiar with Stanford University. We also felt that their familiarity will have minimal impact upon answering questions we had about GeoFoto's interaction design in terms of navigation and browsing of photos.

Methodology
Each study was conducted with a single participant. During the first part of the study, each participant was asked to tell what s/he usually would do to decide where to go when s/he had 30 minutes to spare. Then we asked what kind of interaction the participant expected to be able to do if they had a mobile phone application that would let him/her

browse pictures of places taken in a direction s/he chose relative to his/her current location.

In the second part of the interview, participant was shown a paper prototype that simulated a mobile phone with the GeoFoto application. The participant was asked to pretend that s/he was in front of Memorial Church and was curious how the area north of the tower looked. Perhaps s/he would find something interesting that would motivate him/her to explore that area. Using the paper prototype cell, the participant was asked to show how s/he would accomplish this task.

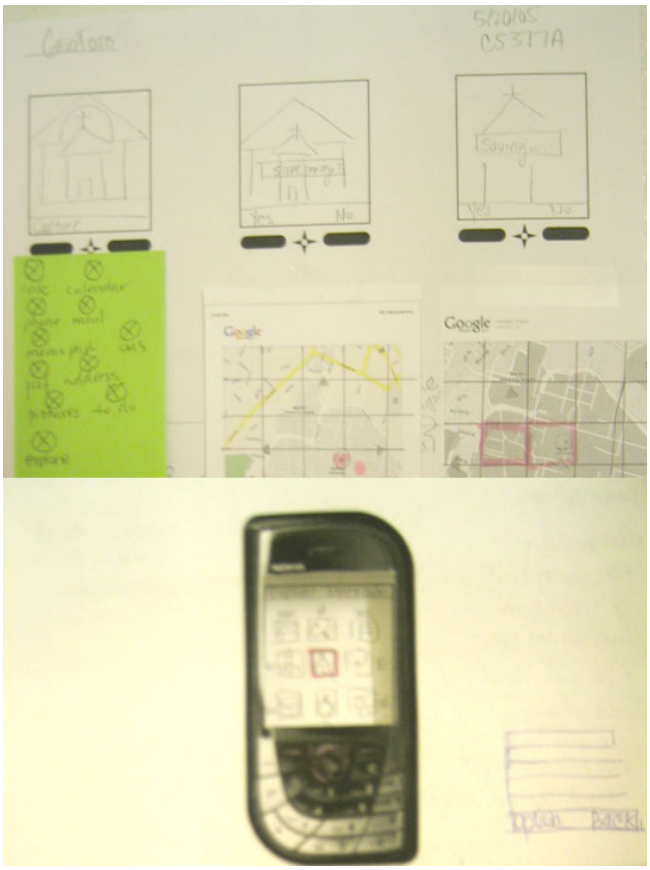

Figure 1. Paper prototype created for user study

Findings

All users said they would rely on finding a map or asking someone who is familiar with the vicinity to decide where to go in a limited time frame. Other strategies include going to an information center or a bookstore to get more information about the vicinity.

For a mobile application that would let them browse pictures taken in nearby areas, all participants wanted a GPS-enabled application that would give them recommendations of where to go based on how much time they have. Some participants wanted the application to give directions for how to get to those places.

The paper prototype was tested on all pariticipants. In the beginning we started a participant with a simple 3x3 grid. Each grid has a corresponding geographic direction (e.g. North, South, Northeast, etc.). However, it became clear that the direction needed some context that grounded the participant to where s/he was. We iterated on the navigation and decided to use a map with grids instead. A red dot indicated where the participant was on the map with the grid it was on highlighted. All participants understood how to navigate the grid by moving the highlighted area using the arrow keys.

All participants expected to see an enlargement of the selected grid, with several points of interest which they can then select in order to see the picture with some description of what it is. They also wanted to be given a direction for how to get to this place from there they currently were. This behavior was different from what we had expected.

We also tested another version of the navigation. This one shows thumbnails of pictures representing each direction. The thumbnails are in a 3x3 grid with the current location of the user in the middle grid. Participants did not expect to see multiple pictures of a certain direction when they selected a thumbnail. Instead they expected to see an enlargement of that thumbnail. One participant expressed her wish to be able to page through pictures one by one once she selected a direction.

IMPLEMENTATION

While frustrating challenges are common in ground-breaking software projects, we were surprised by how difficult developing for the mobile phone is compared with other platforms, such as desktops. We believe this has a lot to do with the currently balkanized nature of mobile device systems. We were developing for a class-provided Nokia 7610, which is a Series 60, MIDP 2.0, CLDC 1.0 camera-phone. Even though this is one of the more common platforms, we encountered a steep learning curve, and had usability difficulties with everything from the emulator to finding APIs. Without significant standardization, there was a shocking lack of good documentation, sample code, and help resources available to help us develop on the Nokia 7610.

Given the time and resource constraints of the class, we had to make many unfortunate tradeoffs in developing the application and related infrastructure. One of many examples of this is that because device configurations before CLDC 1.1 (the 7610 is CLDC 1.0) do not support doubles or floating point numbers, and we did not have time to finish integrating Servlets at the time of this paper, we were unable to calculate the Pythagorean Theorem formula needed to give a direct distance to photos in blocks diagonal from the current one.

Development Tools

The process of figuring out what software tools to use involved a lot of trial and error, particularly in trying to get

the Series 60 emulator (a desktop application meant to simulate how a phone would run and display your mobile application) to work. In the end, we used the J2ME Wireless Toolkit 2.2 on top of Java's JDK 1.5. We took advantage of Nokia Developer's Suite 2.2 for additional development information. We installed Nokia's Series 60 SDK in order to develop for the Nokia 7610 specifically. We used Eclipse 3.0.2 for code development, with the EclipseME 0.9.2 plug-in for linking up code development with verification of the code and running it on the emulator. We also used Nokia PC Suite 6.41.6 for transferring packaged applications from the desktop to the phone. Once we finally got everything installed and configured, develop was able to proceed much more smoothly.

Interaction Flow

From the low-fidelity user testing, we learned that it was very important to convey context to the user in the process of navigating these photos. At the same time, we knew that we also wanted to allow users to view a wide range of others' photos and let them view larger versions of the images given that thumbnail images on a cellphone can be difficult to decipher.

For this reason, we decided to develop a three-tiered architecture. When the user starts the application, s/he is taken into the first view, which displays the photos in a grid that can be navigated geographically. The interface indicates how far and in what direction the currently selected block is from the user's current location. Once the user selects a block to view more about, s/he is presented with a list of larger thumbnails from the same block. We layed out the photos in a vertical list, with the photo the user just selected listed first, and continue to indicate the location of the block relative to the user's current location in the title. This was designed to alleviate much of the confusion users in the user tests had with this change of metaphors. Lastly, once the user selects a photo from the block, s/he sees it maximized, thereby getting the best possible view afforded by the small, mobile display. This hierarchical approach allows for both large-scale geographic navigation, as well as the ultimate narrowing of choices and appreciation of the full-size image, all of which aids in the decision making process of where to visit next.

System Architecture

Downloading of the images was done in multiple threads to speed up execution time. Initially we had nine separate threads, but had to reduce these to five, since the phone seemed to have trouble managing nine separate threads and always failed to display one photo as a result.

Graphical elements can be displayed in J2ME using either high-level or low-level API classes. High-level classes, such as forms, allow for more automated layout, whereas low-level classes allow for more direct manipulation of the user interface. Because high-level components must be added to the UI in the order you want them to appear, this presented problems for downloading multiple photos simultaneously which might finish at different times, and low-level classes allowed us to be more explicit in our implementation, therefore we stuck with the low-level classes for most of the interface.

Figure 2. Top-level view allows geographic navigation

Not being able to use doubles and floats was a surprise, and kept us from being able to calculate a direct distance to the currently selected block. It took a long time to figure out that this was the problem, as the emulator gave no error, and it manifested itself on the phone simply by not letting us run the application at all.

For campus photos, we leveraged a database of georeferenced campus tour photos from Mor Naaman of Stanford University, and we are very grateful for his generous donation. As it turned out, the campus tour photos were rather limiting (and not necessarily representative of the average wandering tourist's photos) so we also took about 90 of our own georeferenced photos to include in the database. Thanks again to Mor for lending us the GPS unit which let us manually annotate these photos with latitude and longitude.

Despite many weeks trying to overcome various obstacles, we were unable to get GPS working on the Nokia 7610, the only phone to which we had access. We tried to leverage two different packages, both of which claim to work with Series 60 phones: Place Lab (http://www.placelab.org/), and sample code from Embedded Interaction (http://www.hcilab.org/documents/tutorials/BT_GPS/BT_G

PS.htm). Place Lab's code crashed on the phone, and according to the Place Lab researchers, this is due to a Bluetooth issue on the 7610. The Embedded Interaction code did not seem to be able to pick up the correct data strings from the service broadcast by the GPS unit, so this was equally unhelpful.

Another tradeoff we reluctantly made was that when we determined we did not have time to get Servlets fully functional, and instead created a script to rename the images using a naming scheme based on the x and y block number (converted from latitude and longitude), and the photo number within the block. We also created manual versions of photos in different sizes, so that the photos could be downloaded directly via HTTP requests, even though this would all probably be better accomplished dynamically using Servlets as an intermediate infrastructure. We struggled with this issue of basing block boundaries on latitude and longitude (a grid-based approach) versus districts or landmark areas. Because the landmark view is complicated by the fact that photos of landmarks may be taken from outside the block corresponding to that landmark, leading to user confusion, we ultimately we decided to go with the grid-based approach which simply aims to convey the views a user could expect to see if s/he travels to that block area.

CONCLUSION AND FUTURE WORK

There is much work remaining based on our original vision. Servlets (if they can be executed quickly) would be a good solution to several current problems. They would allow dynamic scaling and file type conversions, as well as show labels for the photos and take care of the decision of how to rank photos by relevance within a block (e.g. time of day or time of year similar to the present, or frequently viewed or sent photos are possible heuristics). They could also allow us to calculate how many photos exist in a block or in a certain direction, which could allow us to color-code the blocks to let the metadata convey the visual popularity of a region through ambient means.

A way of automating directionality (such as with a digital compass) would also be useful, albeit not an easy task. This could allow users to, for example, face a certain direction, and simply navigate forward to see photos taken further down that path.

Based on our user testing feedback, we also think it would be useful to provide directions for how to reach a location the user chooses they would like to visit.

The integration of geo-referenced photo taking, the photos from which are subsequently uploaded to the server and viewable by others is another long-term goal to complete the overall interaction loop, and we believe Marc Davis' Mobile Media Metadata (MMM) project is a good fit for this task.

We would certainly like to do more user tests, particularly with the deployed application, and especially look at how our application affects tourist behavior relative to the generic campus map or tour currently available. Our hypothesis is that it would allow a more distributed, personalized means of deciding what are the best sites for you as an individual or small group to visit. We believe this would consequently lead to visits to lesser-known but interesting sites that would have otherwise been overlooked, and a more tailored visitor experience overall.

While other research projects have looked at taking geo-referenced photos via camera-phones, or viewing such photos on the desktop, we believe this is ground-breaking work in the discovery of how best to let users take advantage of geo-referenced photos on mobile devices while out exploring. Technical hurdles remain before the interaction loop between mass, mobile photo taking and mass, mobile photo viewing is closed and refined to be simple and usable, but we believe we have gone one step closer toward reaching that goal.

ACKNOWLEDGEMENTS

There are many people to whom we are grateful for their help with this project. We would particularly like to thank Mor Naaman, Hartti Suomela, Professor Scott Klemmer and Brian Lee for their guidance and support throughout the project.

REFERENCES

1. Abowd, G.D., Atkeson, C.G., Hong, J., Long, S., Kooper, R., and Pinkerton, M. Cyberguide: A mobile context-aware tour guide. *Wireless Networks,* 3 (1997), 421-433.
2. Davis, M., King, S., Good, N., and Sarvas, R. From Context to Content: Leveraging Context to Infer Media Metadata. In *Proc. Twelfth ACM International Conference on Multimedia (ACM MM 2004)*, ACM Press (2004), 1-8.
3. LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., Schilit, B. Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Proc. Pervasive 2005*, Munich, Germany.
4. Naaman, M., Harada, S., Wang, Q., Paepcke, A. Adventures in Space and Time: Browsing Personal Collections of Geo-Referenced Digital Photographs. Stanford Technical Report, 2004.
5. Naaman, M., Paepcke, A., Garcia-Molina, H. From Where to What: Metadata Sharing for Digital Photographs with Geographic Coordinates. In *Proc. Tenth International Conference on Cooperative Information Systems (COOPIS 2003)*, November 2003, Catania, Sicily (Italy)