Retrospective vs. Prospective: A comparison of two approaches to mobile media capture and access

Arttu Perttula and Scott Carter
FX Palo Alto Laboratory, Inc.
3400 Hillview Ave., Bldg. 4
Palo Alto, CA 94304

Abstract

Mobile media applications need to balance user and group goals, attentional constraints, and limited screen real estate. In this paper, we describe the development and testing of two application sketches designed to explore these tradeoffs. The first is retrospective and time-based and the second is prospective and space-based. We found that attentional demands dominate and mobile media applications should therefore be lightweight and hands-free as much as possible.

1 Introduction

People are capturing increasing amounts of multimedia data with an increasing diversity of mobile devices. However, tools to organize and synthesize this data are scarce. In some cases synthesis is not as important and simple streams suffice (e.g., informal sharing via Flickr$^{\text{TM}}$). For many other tasks, though, it is vital to be able to structure or abstract media. Especially when data must be synthesized over not only a group of devices but also groups of users this can be so difficult that much media can go completely unaccessed [10].

In this work, we explore two different approaches of capturing and accessing media on mobile devices (see Table 1). One uses a time-based visualization to facilitate *retrospective* review of media captured by individuals and groups. The other uses a space-based visualization to use media to suggest *prospective* activities. Our method of exploration is to create proof-of-concept interactive sketches of each approach. As Greenburg and Buxton point out [8], "early designs illustrate the essence of an idea ... " and help "make vague ideas concrete, reflect on possible problems and uses, discover alternate new ideas and refine current ones." Importantly, though, these design sketches are not necessarily "suggestive of the finished product," but instead serve as a means of exploration. As Shrage writes [13], these early sketches "externalize thought and spark conversation." Our goal at this stage in our work is to use applications to help understand the challenges with mobile media capture and access tools.

In the rest of this paper, we describe our two systems: the retrospective system that links captured data to other media previously captured in the same place using a temporal representation; and the prospective system that uses media to guide users toward more entertaining or useful areas. We also describe case studies and pilot tests for each application. We then synthesize our findings, pointing out that users want to focus on the mobile device as little as possible, suggesting that a redesigned prospective system may provide the most value.

Table 1: Attributes of the retrospective and prospective capture systems

Retrospective	*Prospective*
Temporal representation	Spatial representation
Focus on media nearby user	Media where user might want to be
Media captured in the past	Media just captured or potential to be captured

2 Retrospective system

Our retrospective system – *Notelinker* – includes both a server and a mobile application. The mobile application can capture a wide variety of media (including video, image, audio) as well as meta-data (including Bluetooth proximity and device interaction history). The mobile application also includes a pannable interface to organize and annotate media (Figure 1). The server can receive data from the mobile application as well as a variety of different media formats uploaded through a web page. The server can also receive data from capture services (such as meeting capture systems) running in smart environments. The core novelty of the system is in the links it establishes between different media as well as between annotations of representations of digital documents and the original documents themselves.

The system structures recorded media by creating links based on proximity and content. To create proximity-based links, the mobile system continuously records audio as well as Bluetooth IDs of nearby devices. When a user makes a recording, this contextual metadata is saved on the server with the original recording. When media from other devices are synchronized with the server, the system automatically connects recordings with nearby Bluetooth IDs. The system also searches for similar audio clips in any video recordings and links to the appropriate segments.

To create content-based links, the system can use image-based features. For pictures, extracted OCR text is saved as meta-content and linked against other data uploaded to the server. In combination, these features allow users to connect seamlessly media captured by their device to media captured by non-enabled devices. Furthermore, the system allows users to collect, annotate, and organize representations of digital media that will be substituted with their original content when it becomes available.

Importantly, the system also includes mechanisms for organizing media captured by other nearby users, as well as proximate services, on-the-fly. The mobile application makes available representations of captures as they are recorded, including keyframes for videos, thumbnails of the most recent photo taken, and icons representing audio clips. The mobile application also automatically grabs these representations from nearby users and makes them available to the user in a staging area on the mobile interface. Also, a location resolution system on the mobile application continuously checks Bluetooth IDs recorded by the client against a capture service location database

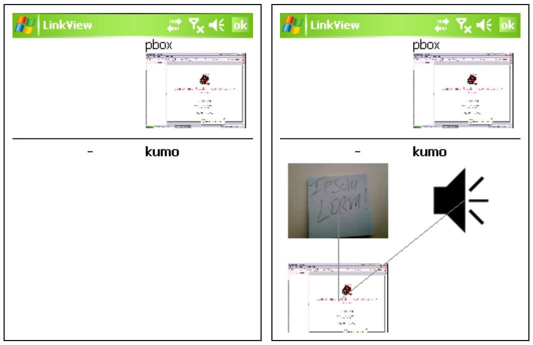

Figure 1: *Notelinker* in a smart environment. *(Left)* The system has determined its location ("kumo") automatically from Bluetooth tags in the environment and connected to a slide capture service available in the room ("pbox" in the upper right). *(Right)* The user has clicked on the slide from the capture service to create a copy in the active area (lower left). The user has also captured an image of notes she made (middle left) and an audio note (middle right), and linked both annotations to the slide.

on the server. When the application finds a nearby service, it grabs a representation of the latest capture and places it in the staging area. Once in the staging area, icons can be dragged into the main scene. When this occurs, the system automatically saves the original file to the user's profile that is available via the web interface (e.g., if the user selected a keyframe from a video, the video is saved). At this point, the user can select two icons to manually link content, and can add annotations to captured content 1). In this way, the system not only automatically links content, but also exposes content seamlessly that otherwise might go unnoticed.

2.1 Scenarios of use

The simplest scenario involves a single user capturing media and linking to media from capture services. Suppose that Bob, a user, wants to make a note during a presentation. If the presentation room as been tagged with a Bluetooth ID and includes a synchronous slide capture service, The system automatically processes the slide stream and makes a keyframe of the current slide available to Bob. He can drag this icon into the main scene and begin annotating it.

Now suppose Bob is in the field and wants to make a text annotation of a segment of video that his friend Marcia is recording on a standard Bluetooth-enabled digital video device. In this case, Bob will necessarily be near Marcia since he is commenting on something that she is recording. Bob can use the system to enter his comments. Behind-the-scenes, the system will automatically send with the comment a clip of 15-seconds of audio recorded before and after the comment as well as a snapshot of all of the nearby Bluetooth devices. Later, when Marcia uploads her recorded video, the system will use the audio and Bluetooth data to link Bob's comment to the correct device as well as the correct sequence of video that Bob was annotating. Note that links would have been created for any type of media (e.g., rather than making a text comment Bob could have taken a picture or recorded his own video).

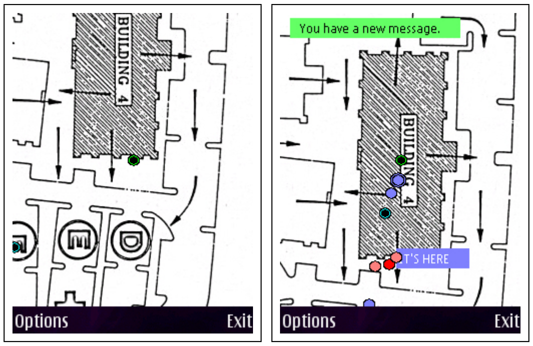

Figure 2: *Kartta* in a field setting. *(Left)* The application showing only the user's current location on a map. *(Right)* The user's current location as well as positive and negative votes, an incoming message notification, and a tag.

Bob could also take a photo of the same scene that of Marcia is recording just before or after he makes his comment. This action links Bob's comment to a particular keyframe in Marcia's video. Immediately after making the comment, Bob sees on his device the picture he took of the scene with the comment already linked. Later, when Marcia synchronizes her video, Bob's picture will become an active link into the source video. Bob can use this method to create collections of media on the fly that are combinations of original recordings he has made as well as pointers to recordings others have made. He can organize these clips on his own device immediately – all of the linking will occur post hoc.

Finally, Bob could also mark an interesting point-in-time with one button-press and, if Marcia is using an enabled device, automatically retrieve the latest keyframe from Marcia's recording and add that keyframe to his own collection.

2.2 Pilot

We asked six participants to diary important events that occurred during one day, making use of information exposed from a meeting capture service [9] and an informal public display service [2]. These services already include web APIs to expose data, making it easy to connect them to the database. We interviewed each participant about their experiences after the pilot. Our findings are preliminary, but one central issue is that each participant wanted the system to tie captures more directly not just to other captures, but also their work process ("I want to be able to make a note on a journal paper I've printed and have the system link in the PDF in the background"). The participants also all agreed that audio was the most useful annotation mechanism.

3 Prospective system

Based on a brainstorming session with 15 participants, we designed the prospective system – *Kartta* – to provide a guide for users in the field based on location-based feedback by other users. In this way, the system can be thought of as a Digg^{TM}for locations. The system includes a server and a mobile application built using Mobile Python. The mobile application uses a map-based interface to show users' current location as well as collaborative recommendations for places (Figure 2). The application includes a simple, one-button interface for voting a current location up or down. A positive vote is represented on the map as a red dot and a negative vote a blue dot. Over time, votes will create implicitly an interest-map of a place. Users can also launch media capture applications with another button. Media are automatically sent to a server along with location information. Media captured at a location automatically correspond to a positive vote for that place. Media can also be tagged with short text descriptions that appear as labels on the map. Users can access media captured by friends (configured a priori) by navigating to that area on a map. Thumbnails of photos taken nearby the cursor can be enlarged by pressing the select key.

The application currently senses location information via either embedded or attached GPS devices. While GPS currently affords only a gross estimate of location, we believe that it will be sufficient since our application depends on aggregated data (votes). Futhermore, we anticipate that users will add tags to disambiguate areas of interest.

3.1 Scenarios of use

Consider a group of researchers who are attending a conference with a few thousand attendees. Bob sees an interesting poster and wants to notify others. He takes a photo of it and tags the photo. After that others' devices download his photo automatically. Also, all of the group members can see a highly positive vote on the map indicating the place of the poster. Marcia zooms in and sees an interesting tag next to the vote. She presses one key and a photo pops up on the screen. She thinks that she must see that poster and checks out the map. She is already almost in the correct place but she cannot find the poster. So Marcia sends a message to Bob and asks about the poster. Bob can see Marcia on the map and he replies, "it's behind the corner next to the stairs." Marcia finds the poster and she thinks it is fascinating. She wants to give a positive vote to the poster and does it just pressing one key. A vote appears on the map and the rest of the group can see that there must be something interesting at that location since there are now two positive votes.

Later, Bob is sitting through an uninteresting talk. He gives it a negative vote, which is sent automatically to the others' maps. Bob also sends a message to everyone about the topic of the talk and writes that there is nothing new. Now everyone can look for something else. Marcia is at a different talk that is more interesting. She gives a positive vote to the talk and takes a photo, tagging it with the topic of the talk. Bob and others decide to migrate to that talk.

3.2 Pilot

Massimi et al. used a scavenger hunt for evaluating mobile collaborative systems [11]. This approach is useful because it blends the control of a laboratory study with the realism of a field experiment via a lightweight and well-understood game. We adapted this approach to test our system. In our experiment, four users were divided into two teams of two, and we gave every user a device running our system. The system came preloaded with a map of the campus near our building, which includes buildings, several flights of stairs, parking lots, a fountain, etc. The goal of the exercise was to find colored balls (30 in total) that had been scattered around the campus. We wanted to incentivize teams to split up to find items while still encouraging them to use the map to see what their teammates had found, and furthermore to communicate information about found items via messaging, taking a picture, or tagging a place. To that end, teams were given 10 points if either member found and took a picture of a ball, and another 6 if both members took a picture of the same ball. Two researchers followed participants during the study, taking notes and helping if users were completely stuck. Also, we held a focus group session with all participants after the study focusing on potential features.

Participants collectively captured 66 total targets and submitted 45 tags over one hour. The messaging feature went relatively unused. Ultimately, we found that the interface was not yet at a stage to judge its usability, but participants provided a host of recommendations in the focus group. Participants overall felt that the interface required too much attention, and requested vibratory and auditory alerts. They also suggested that the map may not be necessary at all, and that the visual interface might consist only of hotspots as well as paths to those spots (Figure 3). They also suggested audio tagging, list views of recorded tags, and orientation controls and views.

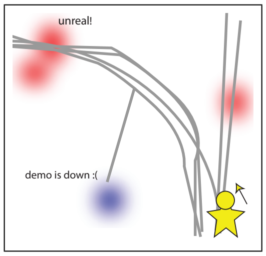

Figure 3: Potential *Kartta* redesign based on user study feedback. The application shows no map information, only spots with votes, GPS paths of other users, tags, and the user's current location and orientation.

Methodologically, we found that our attempts to encourage both individual search and group communication had mixed results. While participants clearly used the tagging feature to signal information about a target to the other group member, in other cases participants simply met up physically to discuss the location of targets, and in still other cases ignored the map altogether.

4 Comparisons of the two systems

Our retrospective system helped users put their media captures in context with other media captured nearby, while our prospective system used media to indicate areas of potential interest. Through our pilot testing we found that users are more likely to do post hoc organization than in situ organization, and that it is more important that the mobile application itself provide information but otherwise not require intense focus, or worse, get in the way of users' real-world tasks. This reinforces previous work that has found that most people would rather pay attention to the events around them than navigate through a phone's interface [12]. Futhermore, post-hoc organization could be accomplished using either system since the process of capturing meta-data is relatively streamlined. Also, given that mobile users want to focus on *now* and *next*, it is not useful to provide a full timeline on the mobile application.

Overall, then, we found that the prospective system provided both in situ and post hoc value, and we plan to focus on it in future work (or "get the design right" according to Greenburg and Buxton [8]).

5 Related work

Similar to *Notelinker*, Fono's and Counts' Sandboxes displays collaboratively captured multimedia on mobile phones [6]. However, this work does not address organization (i.e., it does not utilize context to structure captures). Furthermore, it does not provide any video recording or other video-related support, such as keyframe generation. Erol and Hull describe a system to index into a presentation using an image captured with a camera phone [3]. Their access interface displays the original captured slide and the video recording at the time it was presented (a similar system using scanned images appears in [1]). Fink et al. describe a system that senses TV audio to automatically recognize the program the user is currently watching [4]. They use this technology to support social viewing applications. However, these systems do not address collaboratively recorded media, they are not particularly designed for field settings, and they are designed primarily for retrieval rather than organization and synthesis.

Some map-based interfaces are used for retrospective purposes, and in many cases they can also be used for long-term planning. One example is EveryTrail (http://everytrail.com/) in which users upload GPS tracking information as well as GPS-tagged photos after a run, hike or bike ride – the system automatically plots that information on Google maps and offers a variety of social interactions around uploaded media (comments, ratings, etc.). However, this tool is not designed as a mobile app to help users decide where to go next in situ.

Fleck et al. [5] conducted an iterative deployment of a mobile capture tool to a museum setting. They quickly found that the capture application they had designed required an attentional shift away from the activity being captured that was unacceptable to users. In the end, they embedded capture technology into the environment itself and used the mobile device only to initiate an automated capture process (by swiping an RFID over a reader). In most mobile situations, though, it is not possible to instrument the environment in this way.

Finally, a variety of systems have linked digital and paper documents. Yeh et al. developed ButterflyNet, a mobile capture and access system that integrates paper notes with photos captured in the field and includes some automatic linking capabilities [14]. Data from different sensors are all linked temporally. Furthermore, a user can link photos to written text using a combination of gestures and temporal data, and can link photos and annotations using a visual tag. Graham and Hull developed Video Paper, a paper-based method for retrieving video segments [7]. A video's transcript is annotated with barcodes that jump to corresponding positions in the video. A remote control device reads the barcodes to control the replay of the video.

6 Conclusions and future work

In summary, our findings indicate that attentional demands dominate and mobile media applications should distract as little as possible and provide as much information as possible peripherally or using no visuals at all. To this end, we intend to extend our prospective system to support glancability and eyes-free notification. In particular, the phone should vibrate when a user is very close to an interesting area or when someone within walking distance votes up an area. We also plan to add audio notifications, such as the name of a user who just made a capture.

Finally, we encountered some difficulties testing these application sketches in pseudo-realistic environments. In particular, the applications were likely not yet robust enough for field deployment, and also our experimental conditions require iteration. After iterating the design, we intend to test our prospective system using an asynchronous approach. This would require each user to consult their device to find targets the previous user had already discovered, and to leave notes digitally for the next user. This would also more closely resemble a realistic environment as people rarely overlap exactly in time when exploring a new space.

References

[1] P. Chiu, J. Foote, A. Girgensohn, and J. Boreczky. Automatically linking multimedia meeting documents by image matching. In *HYPERTEXT '00*, pages 244–245.

[2] E. Churchill, L. Nelson, L. Denoue, and A. Girgensohn. The plasma poster network: Posting multimedia content in public places. In *INTERACT '03*, pages 599–606.

[3] B. Erol and J. J. Hull. Linking presentation documents using image analysis. In *Signals, Systems and Computers '03*, pages 97–101.

[4] M. Fink, M. Covell, and S. Baluja. Social- and interactive-television applications based on real-time ambient-audio identification. In *EuroITV '06*, pages 138–146.

[5] M. Fleck, M. Frid, T. Kindberg, E. O'Brien-Strain, R. Rajani, and M. Spasojevic. From informing to remembering: Ubiquitous systems in interactive museums. *IEEE Pervasive Computing*, 1(2):13–21, 2002.

[6] D. Fono and S. Counts. Sandboxes: supporting social play through collaborative multimedia composition on mobile phones. In *CSCW '06*, pages 163–166.

[7] J. Graham and J. J. Hull. Video paper: a paper-based interface for skimming and watching video. In *Consumer Electronics '02*, pages 214–215.

[8] S. Greenburg and B. Buxton. Usability evaluation considered harmful (some of the time). In *CHI '08*, pages 111–120.

[9] D. Hilbert, M. Cooper, L. Denoue, J. Adcock, and D. Bilsus. Seamless presentation capture, indexing, and management. In *SPIE Internet Multimedia Management Systems*, pages 305–313.

[10] S. Klemmer. A pervasive computing framework supporting collaboration in documentary history projects. In *DIS '02*.

[11] M. Massimi, C. H. Ganoe, and J. M. Carroll. Scavenger hunt: An empirical method for mobile collaborative problem-solving. *IEEE Pervasive Computing*, 6(1):81–87, 2007.

[12] A. Oulasvirta, S. Tamminen, V. Roto, and J. Kuorelahti. Interaction in 4-second bursts: The fragmented nature of attentional resources in mobile hci. In *CHI '05*, pages 919–928.

[13] M. Shrage. *Serious Play.* Harvard Business School Press, 2000.

[14] R. Yeh, C. Liao, S. Klemmer, F. Guimbretière, B. Lee, B. Kakaradov, J. Stamberger, and A. Paepcke. Butterflynet: a mobile capture and access system for field biology research. In *CHI '06*, pages 571–580.