# *MobShare*: Controlled and Immediate Sharing of Mobile Images

Risto Sarvas[1], Mikko Viikari[2], Juha Pesonen[2], and Hanno Nevanlinna[2]

[1]Helsinki Institute for Information Technology (HIIT)
P.O.Box 9800, 02015 HUT, Finland
risto.sarvas@hiit.fi

[2]Futurice
Merimiehenkatu 29, 00150 Helsinki, Finland
{mikko.viikari, juha.pesonen,
hanno.nevanlinna}@futurice.fi

## ABSTRACT

In this paper we describe the design and implementation of a mobile phone picture sharing system *MobShare* that enables immediate, controlled, and organized sharing of mobile pictures, and the browsing, combining, and discussion of the shared pictures. The design combines research on photography, personal image management, mobile phone camera use, mobile picture publishing, and an interview study we conducted on mobile phone camera users. The system is based on a client-server architecture and uses current mobile phone and web technology. The implementation describes novel solutions in immediate sharing of mobile images to an organized web album, and in providing full control over with whom the images are shared. Also, we describe new ways of promoting discussion in sharing images and enabling the combination and comparison of personal and shared pictures. The system proves that the designed solutions can be implemented with current technology and provides novel approaches to general issues in sharing digital images.

## Categories and Subject Descriptors

H.5.1 [**Information interfaces and presentation** (*e.g.*, HCI)]: Multimedia; H.4.3 [**Information systems applications**]: Communications Applications.

## General Terms

Design, Human Factors

## Keywords

Camera Phones, Digital Image Management, Wireless Multimedia Applications, Multimedia Tools.

## 1. INTRODUCTION

People share their digital pictures by email, web publishing, paper prints, slideshows, and often simply by showing from the camera's LCD screen. Several current consumer products take into account sharing, managing, and archiving of personal media. Software like ACDSee [1], Adobe Photoshop Album [2], or

Apple iPhoto [3] facilitate searching, browsing, and archiving of pictures, and sharing them with other people. Also, there has been significant academic research done on personal photo collection management, where the focus has been on automatic organization of image collections [7, 12, 14, 24], and innovative user interface designs for browsing and searching the collection [5, 11, 13, 19].

Mobile phones with a built-in digital camera are at the moment outselling digital cameras [20] and therefore challenging digital cameras' position as the number one consumer media-recording device. However, the dominant consumer and research applications mentioned above take little into account the special characteristics of mobile phone cameras, mainly, that mobile phone cameras are primarily a communication device.

New picture publishing and sharing systems are emerging which take advantage of the special characteristics of camera phones [6, 10, 22, 23, 25, 29, 32]. In contrast to regular digital cameras, camera phones' technical characteristics include several network connections that enable sharing images directly from the phone (*e.g.*, GSM/GPRS, Bluetooth, and infrared), computing resources accessible by standardized programming interfaces (*e.g.*, J2ME, BREW, and Symbian), and access to contextual and social information on the phone (*e.g.*, calendar, location data, time and date, address book, as well as call and messaging logs). Camera phones also have the ubiquitous characteristics of mobile phones in general: people carry them everywhere and they become an indispensable accessory in people's lives.

In this paper we present a mobile picture sharing system *MobShare* that focuses on immediate and controlled sharing of pictures within a circle of acquaintances. Our system design is based on related literature and an interview study we conducted on mobile camera phone users. The main contributions of *MobShare* to mobile picture sharing are the following:

- Immediately sharing captured mobile images from the phone to an organized and manageable web album.

- Providing full control over whom the images are shared with by leveraging the social contact information stored on the phone's address book.

- Integrating discussion into the sharing and viewing of mobile images.

- Enabling the combination and comparison of pictures both taken by the user and shared by other users.

In the following subsections we discuss related work, our research methodology, and present primary use cases for *MobShare*. Section 2 discusses the novel solutions implemented in sharing

mobile pictures. Section 3 discusses the novel solutions implemented in browsing, combining, and discussing the shared mobile pictures. Section 4 gives a system architecture overview, Section 5 presents an evaluation of the system, Section 6 discusses future research, and Section 7 concludes.

## 1.1  Related Work
The design of *MobShare* was based on related work and literature on traditional and digital photography, personal image management, and mobile camera phone use as well as on analysis of existing mobile imaging systems.

### 1.1.1  Related Literature
Research in cultural studies has studied what people photograph and for what purposes (see *e.g.*, [15, 30]). Studies by [4, 31] have looked into what purposes do photographs and other memorabilia serve in people's lives, and [9, 26] have studied how people manage and share their digital pictures.

The problem of managing the rapidly growing personal image collections has been researched from the user interface point of view, focusing on the organization, browsing and searching of pictures (see *e.g.*, [5, 11, 13, 19]). Also, the event-based automatic categorization of images based on their timestamp metadata has been studied (see *e.g.*, [7, 12, 14, 24]), as well as location information in images [33].

However, personal image management and people's picture sharing habits in the domain of mobile pictures has not been as widely studied as in the general digital imaging domain. Currently picture taking and sharing using mobile phones has been studied in [16, 17, 18, 21]. Our previous research in [8, 27, 34] has looked into image annotation and automatic metadata generation on mobile phones.

### 1.1.2  Related Systems
Related systems [25, 29, 32] enable mobile picture publishing on the Internet (*i.e.*, "photo blogging" or "moblogging") but take personal image management issues little into account. They focus on publishing images immediately from the phone to the Internet and having discussion functionality. The main difference to *MobShare* is that in these "blogging" systems controlling the sharing of images and combination of shared images is not supported, also, picture organization features are limited.

Other related systems [6, 10, 23] have more specific features for personal image management, for example, editing, organizing into folders, and choosing images for publishing. Also, these systems enable sharing of pictures to specific people. The contribution of *MobShare* into these systems is enabling the sharing directly from the phone itself and using the phone's address book in the process. In addition, the shared picture folders can be easily combined and compared as described later.

Lifeblog [22] combines the media and messages on the phone into a media diary on a PC. Like *MobShare*, Lifeblog takes advantage of the time and date information in organizing and visualizing the pictures on a timeline. However, unlike in *MobShare*, there is no sharing or publishing feature.

## 1.2  Methodology
Based on related work we conducted an interview study. Ten mobile phone camera users in the Helsinki area were interviewed about their use of film, digital, and mobile phone cameras and

pictures. The interviewed were men and women age 19 to 32 who had had their camera phones for at least two months. Each interview covered picture-taking habits and situations, picture sharing, and going through the interviewee's pictures.

The results from the literature and interview studies showed that sharing of mobile pictures has the same motives and uses as traditional sharing and discussion of pictures (*e.g.*, telling about travels, reminiscent of shared events), in addition to new ways and uses enabled by the technology (*e.g.*, documenting everyday life, improvised pictures). However, current mobile picture sharing applications support only partly the whole mobile photo capture and sharing process: some support controlled sharing, some support web publishing, and some support discussion. From these results we designed and implemented our prototype system *MobShare*. The objective of the system was to wholly support the mobile picture taking, sharing, and discussion process. Especially, the goal was to learn how the issues and requirements identified from the studies could be implemented with current and widely adopted technology. The technology is discussed in more detail in Section 4.

## 1.3  Primary Use Cases
We present two use cases to give an overview of the system. The first use case describes the setting for Section 2: capturing and sharing of images to the remote server using the client application. The second use case describes the setting for Section 3: viewing the user's own images and the images shared by other people using the desktop web browser component.

### 1.3.1  Capturing & Sharing Using the Client
The user captures a picture using a camera program integrated to the *MobShare* client. After taking a picture the user selects the new image or any set of images on the phone to be posted. By *posting* we mean the user experience of uploading the images to the remote server on the Internet (see Figure 1 for screenshots).

After selecting the images to be posted, the user chooses a folder where the images will be located. The folder can be an existing folder he has created before, or he can create a new folder. If he creates a new folder, he names the folder and selects the people



**Figure 1. The posting process where a) the images are selected, b) posting initiated, c) new folder created, d) named, e) the people selected, and f) finally the images are uploaded.**

who he wants to share the folder with. The sharing procedure is similar to sending a text message or making a phone call: he selects the people he wants to share the images with from the phone's address book. The phone transfers the pictures to the remote *MobShare* server over the phone network.

If the people the user shared the images with have not used *MobShare* before they will receive an SMS notification including a password and instructions for viewing the images. Old users also have the option to receive SMS notifications of new shared pictures.
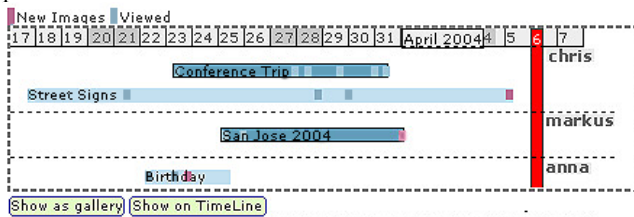


**Figure 2. The Horizontal Timeline View of the user's (Chris) own folders and folders shared by others (Markus and Anna). The two selected folders are in darker color.**

### 1.3.2 Viewing Images Using the Desktop Component

After logging into the system the user is presented with his own page of images. On the top of the page on a horizontal timeline are his own folders of images and the folders that others have shared with him (see Figure 2). For each folder the user can see if there are new pictures that he has not viewed yet.

The user can select one or several of the folders (see Figure 2, where Chris has selected two folders: his own "Conference Trip" folder and Markus's "San Jose 2004"). The selected folders can be viewed as a gallery of thumbnails (see Figure 3). The gallery view shows the thumbnails one after another in temporal order as one large collection. The selected folders can also be viewed on a vertical timeline of thumbnails (see Figure 4). The timeline view shows the thumbnails on synchronized vertical timelines. Both the gallery view and the timeline view also show who else can see the selected folders (*e.g.*, Markus and Anna for Chris's folder "Conference Trip"). If the folder is the user's own, then the user interface shows who has already visited (*i.e.*, viewed) the folder. The user can also modify the viewing rights of his own folders.

On the right side of the gallery view is a conversation board that shows small thumbnails of the pictures that have been commented on and the respective comments (see Figure 3). The user can click on any thumbnail to open a pop-up window. The small window contains the picture in its full size, together with the capture time and date information, description of the image, and a possibility to discuss the image. If the image is user's own, he can also edit the description, rotate or delete the image, and delete comments.

## 2. IMMEDIATE SHARING & CONTROL

Sharing pictures with other people is probably the most popular activity with photographs. Paper photos are shared with relatives and friends by showing them personally, or including photos in letters [9]. Digital pictures are often shared by email, burned on a CD, published on a web page, or shared from a computer, television, or camera screen [4, 9]. In addition, people often print digital pictures and share them like traditional paper photos.

Pictures taken on a mobile phone camera can be shared in the same ways as any digital pictures, but the inherent network connectivity supports sharing the images immediately after capture. For example, using MMS (Multimedia Messaging Service), pictures can be sent directly from one phone to other phones and email addresses. Images can also be sent to a web service, which publishes the images on a web page. As mentioned before, this kind of web publishing of images from a mobile phone is often called *photo blogging,* or *moblogging*.

Compared to traditional sharing of digital images, sharing images directly from the phone has one step less for the user to go through: with camera phones there is no need to transfer the images to a networked PC to be able to share them. Mobile phone images can be shared immediately after capture from the actual capture device, and the images can be shared over the network to distant locations. This kind of combination of immediacy and remote sharing has not been possible with film or traditional digital cameras.

As mentioned previously, current mobile image "blogging" and photo gallery software (see, *e.g.*, [6, 10, 23, 25, 29, 32]) enable immediate sharing of images on a webpage. After capturing, the images are sent one at a time to a specific address. Another way of sharing images from the phone is the standardized MMS messaging, which enables direct sharing of multimedia to email addresses and MMS compatible phones. Nevertheless, these software take very little account of the organization and handling of images, which are the key features in personal image management applications. Also, the mobile image "blogging"



**Figure 3. Gallery view and discussions. The folder-level discussion is on the left above the gallery view. The visibility and visitor information of each folder. The summary of picture-level discussions are on the right.**

**Figure 4. Vertical timeline view of pictures. Here are the same two selected folders as in Figure 4, but on separate timelines according to the owner of the folder.**

software are designed for publishing images on the Internet, and hardly take into account the need for privacy and control over who can view the images.

In the following subsections we discuss how we implemented the organization of shared and received images, enabled full control over who the images are shared with, and the benefits of implementing this on a dedicated client-side program.

## 2.1 Organizing Shared and Received Images

Due to the always-present nature of mobile phones, people have the opportunity to take images at any moment, and send them immediately to other people. This will increase the number of images people capture, send, and receive, which makes the need to organize the images more acute.

Organizing and managing large collections of images is difficult on the mobile phone itself because of its small screen size and limited input keypad. To tackle this problem images are often simply deleted or uploaded to a computer where they are stored on the operating system's folder hierarchy, or they are organized, managed, edited, and shared using specialized image management software.

However, images received from other people are often not managed at all. Before they can be managed or even browsed efficiently, the received images have to be moved or saved from the email or MMS inbox, CD, or web page. In practice, they often remain in the inbox or web page without any organization, and are hardly ever managed with the same image management application as the user's own images.

Another problem in organizing images is the lack of user motivation in going through images and annotating them with metadata or keywords. Even simple moving of images into folders with informative names can be tedious work. Therefore, the work is often postponed into the future, thus both forgetting information about the pictures (*e.g.*, where it was taken, who are the people in the picture) and eventually creating a monumental task of organizing old pictures [9].

In the *MobShare* system we take advantage of the users' motivation to share images to make the organization work less tedious and more rewarding. As the user shares the images, the client program's user interface requires him to categorize the shared images into a new folder or an existing folder. Our previous work in [34] has shown that people are not motivated to thoroughly annotate their pictures, especially on a mobile phone.

Combining the time and date metadata in the pictures with the user-managed categorization into folders we were able to visualize the folders of images on a simple calendar view (*i.e.*, the horizontal timeline view of image galleries in Figure 2). The benefits of these organized galleries are discussed later in Section 3 where we describe the image browsing implemented in *MobShare*.

## 2.2 Controlled Sharing

One of the concerns about sharing images online is the question of privacy, that is, who has access to the images. When paper photos are shared restricting access is easy, as well as with digital photographs as long as they are not published on a public webpage in the Internet.

According to our interviews people share pictures almost solely with their circle of acquaintances (*i.e.*, intimates like family and close friends). The interviewed also told that within this circle of five to fifteen known people privacy issues are not a major concern and they feel more comfortable sharing pictures inside this group. One of the interviewed explained some of his pictures: "These images were taken while everyone was drunk. No one would like to put these kind of images into the web." Another interviewed had even gone through the task of creating a group website on Microsoft MSN[1] to share images with his friends and control the images' visibility. Camera phone studies [16, 17] also support the users' requirement to be able to control the visibility of shared images.

Controlling who can see shared digital pictures is often done by sending the pictures as email attachments. In email the user has control over who the recipients are, and the recipient can often see who else has received the message and attached pictures. Another popular way of limiting the viewing of digital images is to have web page that is either password protected or unlinked. However, search engines can find unlinked pages.

Mobile phones enable a novel way of controlling image sharing. The phone's address book contains all the names and phone numbers of the user's circle of acquaintances. In the *MobShare* system, we take advantage of this information: the user can select from the phone's address book the people with whom he wants to share the images. Using the address book for defining viewing rights provides the user with an intuitive list of his acquaintances, and it provides the system with relatively unique user identification numbers (*i.e.*, each person's phone number, that is, the MSISDN). Also, as mentioned previously, the same address book user interface is familiar to the user from making phone calls, emailing, and text and multimedia messaging.

Using phone numbers for user identification has clear benefits. There is no need to create any system specific identifiers for users, phone numbers are more accurate identifiers than names

---

[1] http://groups.msn.com/

that vary (*e.g.*, the same person can be called by his first name, surname, or nickname), and phone numbers are easily available because the salient people, with whom images are most often shared, are in the user's phone's address book.

The disadvantages of phone numbers are that one person may have several phone numbers or people's phone numbers may change. The latter issue is taken into account by allowing users to change their phone numbers in the system. The issue of several phone numbers has to be tested in user test to understand better the practical implications.

## 2.3 Dedicated Client Application
Several of the people we interviewed mentioned the ease of taking pictures with their camera phone. The effort of taking a picture was considered low, because the camera was ready for use, always available and reachable. One of the interviewed commented: "[The camera phone] is always with you, and ... it's so small and you can take it to any place, and you take pictures wherever you happen to be." However, the easiness ended when people wanted to share the pictures. One of the problems was that not everyone within the users' circle of acquaintances had a phone capable of receiving images, or the phone was not correctly configured.

In the *MobShare* system we implemented image capture and sharing in the same client program. This way the user had one contiguous process from taking the picture to sharing it, and we were able to control the usability and visual image of the user interface for the whole process. Also, having both capture and sharing under one application enabled more control over the gathering of contextual metadata at the time of capture. For example, currently for every image we store on the server capture time, date, and GSM network cellID information (the cellID can be used to infer location, however, it is not used in the current version of *MobShare*). A dedicated client application can also access other information on the phone, for example, the user's calendar events or call and message logs. This information can be used to infer more descriptive metadata about the context of image capture (*e.g.*, who, where, when) [8, 27, 34].

## 3. BROWSING, COMBINING, AND DISCUSSING
Browsing, combining, and discussing the pictures is more convenient using a desktop computer with a large screen, a mouse, and a keyboard rather than using a mobile phone with a small screen and a limited input keypad. Therefore, browsing, combining, and discussing was designed and implemented for a web browser that connects to the web server where the posted images are located.

As mentioned in Sections 2, the images on the server are categorized into folders as they are shared from the phone, and therefore, the recipients of the shared image galleries have an organized view of the images. In this section we present and discuss how the picture collections were visualized in the system, how we enabled combination and comparison of shared galleries, and how discussion about the images was promoted.

## 3.1 Visualization of Pictures
People value simple browsing capabilities in going through digital pictures more than highly advanced techniques like audio annotation or image-based queries [26]. Therefore, our focus on the implementation of the picture visualization was to keep it simple, and to have shared images organized also from the recipients' point of view.

### 3.1.1 Horizontal Timeline of Folders
The visualization and browsing of pictures in *MobShare* is based on a horizontal timeline of folders. Both the user's own folders and the folders shared with him are displayed on the same timeline. The user creates the folders either using the mobile phone (see Figure 1) or the web browser. The position and the width of the folder on the timeline reflect the capture dates of the photos in that folder (see Figure 2). This approach combines the familiar folder metaphor with an intuitive calendar timeline. Also, it is easily implemented because the required time and date metadata is practically always available in mobile pictures. From the horizontal timeline the user can select one or several folders to be presented as galleries of thumbnails.

### 3.1.2 Gallery View and Vertical Timeline View
Once the user has chosen the folders to view from the horizontal timeline, he can view the images either as a gallery of thumbnails in chronological order (see Figure 3), or as a set of vertical timelines (see Figure 4). In the gallery view all the images from the selected folders are displayed in a chronological order regardless of whose images they are. The user can click on a thumbnail to open a pop-up window where the image is shown in its actual size, and he can discuss the image. In the vertical timeline view the thumbnails of each selected gallery are on separate vertical timelines side by side. The view can be scrolled if not all the thumbnails fit the screen. Like in the gallery view, the thumbnails can be clicked to see the full size image and discuss it.

### 3.1.3 Notification of New Pictures
As we discussed in Section 2, mobile images can be shared immediately after capture. For example, a user traveling abroad can post images for others to see as the trip progresses. To support this kind of immediacy we implemented a visual notifier in the picture folders to show if there are new pictures (see Figure 2). Similarly to email programs, as the user logs into *MobShare* he can quickly see by the color of the picture markers if someone has posted new pictures for him to see, and in which folder.

If the user with whom the pictures are shared is not a user of *MobShare*, he gets an SMS notification which includes a password and instructions for becoming a user. All users also have the option to get SMS or email notifications when new pictures are shared with them.

## 3.2 Combining and Comparing Pictures
In browsing pictures, the user can combine his own images and other people's images into one collection. As emphasized previously, *MobShare* is designed for people who know each other in advance (*i.e.*, close friends, relatives, or colleagues). Therefore, it can be assumed that these groups occasionally attend the same events and are often the subject of each other's pictures. For that reason one of the main functionalities of the *MobShare* system is the possibility to combine and compare different people's picture folders.

For example, in the screenshots in Figures 1 through 4, two friends (Chris and Markus) who attend the same conference take pictures with their mobile phone cameras and share them with

each other and their mutual friend (Anna). Using the browser component of *MobShare*, any one of the three can select the two friends' folders of the conference trip and view them as one gallery (as in Figure 3). This kind of sharing enables people to collaboratively document an event with several cameras and easily combine the pictures into one gallery.

On the other hand, using the vertical timeline view the user can distinguish between the selected folders and compare the pictures on the same timeline. For example, Anna who did not attend the conference wants to compare Chris's and Markus's individual recordings side by side (see Figure 4).

Combining and comparing folders is intuitive when the pictures viewed are based on an event. However, combination and comparison of non-event pictures can also be meaningful for the user. For example, combining two users' folders of everyday life can be entertaining, or comparing folders of pet animal pictures.

As discussed in Section 2, having the user organize his pictures for sharing (*i.e.*, categorizing the pictures into folders) removes the tedious work of organizing the pictures later on. And by sharing the pictures in organized folders the recipients of those pictures can (re)use the same organization. Without this kind of sharing and organization, combining and comparing one's own images to the images received from other people would require a lot of effort.

## 3.3  Discussing Images
People take pictures to discuss them. Whether it be to tell about a vacation trip, to reminisce a shared experience, or to joke about a comic image [4, 9, 18]. Discussion about pictures is present in sharing images with email, showing paper photos, writing captions, and image MMS messaging. Often pictures are taken only to initiate conversation or contact [16]. In other words, discussion or communication in one form or another is an integral part of sharing images, perhaps even the primary reason [18].

In the *MobShare* system we integrated discussion functionality into picture browsing, and promoted it by making the possibility to discuss visible all the time. Having the discussions in the server database also stores and archives the discussions along with the pictures. Therefore, the discussions are not lost like they easily are in MMS or email messaging.

### 3.3.1  Presenting Several Discussion Threads
Discussion about pictures can be about individual images or the whole image folder (*e.g.*, discussion about a picture of the Eiffel Tower or discussion about a whole trip to Paris). In the *MobShare* system the discussion about individual pictures is implemented in the pop-up window where the image is shown in full size. The folder-level discussion is implemented in the gallery view.

However, having a discussion for each individual image results in several discussion threads. To make the viewing of these several discussion threads easier, each picture-level discussion is summarized on the gallery-level (see Figure 3). This enables the user to quickly skim through the latest discussions on picture-level without opening each image. By clicking the small thumbnail image next to the discussion in the summary, the picture pop-up window is opened, and the user can discuss the image. In other words, the pictures can be viewed also by the discussions they have.

### 3.3.2  Seeing Who Can Read the Comments
An issue in discussing images is to know who else can see the conversation and take part in it. For example, when images are shared in email attachments the recipients of the email can usually see who else has received the images, and reply to the email accordingly.

In the *MobShare* system the users see who else can view the selected folders. In other words, the list of people selected by the user from his phonebook is visible to the people in the list (see Figure 3 and 4). Therefore, each user is aware of the people who can read the discussions. Also, the people whom the folder is shared with probably know each other, and showing the names of these people creates a feeling of togetherness and familiarity.

## 4.  SYSTEM OVERVIEW
The *MobShare* system consists of a mobile phone client application and a web application server that is accessed using a regular web browser. Sharing pictures from phone to phone requires special phones, which are less common than web access, and importantly, accessing pictures using a web browser enables design for a larger screen and better input devices. See Figure 5 for a system architecture diagram.
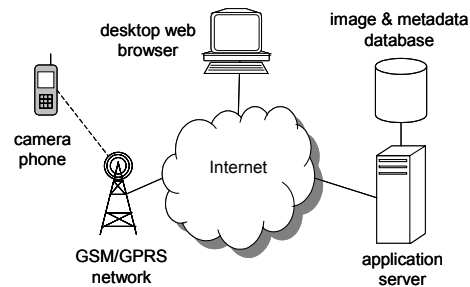


**Figure 5. Architecture overview.**

### 4.1.1  Client
The *MobShare* client-side software is a Symbian C++ application for Series 60 terminals. We selected the Symbian development platform instead of widely supported Java 2 Mobile Edition (J2ME), because J2ME does not currently provide sufficient access to low-level resources like stored image files or the phone's address book. The client-side software uses the Series 60 Phone Book Engine [28] to access the native phonebook application and contact information. Also, the integrated camera uses functionality provided by the platform.

### 4.1.2  Transfer Protocol
The communication between the client program and the web server is built on the HTTP protocol. The underlying data transfer network used in building the system was GSM/GPRS. However, the implemented communication solution is not dependent on the underlying network as such, and therefore it is compatible with any HTTP providing network. In the used GPRS network the client-side software uses a so-called Internet Access Point (*i.e.*, direct connection) instead of a WAP access point. The advantage is that with a direct Internet Access Point connection there is no unnecessary WAP specific optimization. The client program logs in on the web server using the phone number (MSISDN) as the authentication identifier, and uploads the images and related metadata to the server.

### 4.1.3 Server

The server used is a J2EE web application that runs on Tomcat servlet container[2]. The uploaded pictures, including the server-generated thumbnails, are stored as JPEG image files in the file system. Picture metadata (time and date of capture and posting, name of folder, owner of picture, comments, and picture description) and user data (usernames, MSISDNs, and who has viewed what image folders and when) is stored in a relational database. The database used is PostgreSQL[3].

When a registered user shares photos with a friend (*i.e.*, selects the friend's name from the address book) who has not used *MobShare* before, the server creates a new user account for the friend. The visible username for the new user is the name retrieved from the phone book, the user identifier is the friend's phone number (MSISDN), and a password is generated automatically. An SMS invitation sent to the new user contains information on who wants to share pictures with him, at what web address they can be viewed, and how to login to the system. This way no additional registration phase is required.

Using the MSISDN for authentication, both in uploading the pictures from the phone and in logging in to the web interface, simplifies the system design. It enables a globally unique identifier (*i.e.*, no two phone numbers are the same), and it is intuitive and well known to the users even before they register to the system.

The web user interface was implemented using DHTML (*i.e.*, Dynamic HTML, which is a combination of HTML, CSS, and JavaScript), and it was targeted for the Internet Explorer browser. The server side implementation is based on Struts Web Application Framework[4] and Java Server Pages.

## 5. EVALUATION OF IMPLEMENTATION

The GPRS network transfer speed was known to be limited and our tests showed that uploading one image (640x480 pixels) took from 19 to 30 seconds on average depending on the network carrier. Therefore, posting images was a relatively slow task and the phone was put aside during the process. Uploading the image with MMS would take approximately the same time because both use the GPRS network. Alternative methods would have been Bluetooth, infrared, or a cable connected directly to a computer, but these methods would disable the mobility aspect of sharing images.

The problem of slow upload speed might be solved in faster future phone networks, but at the same time, the resolution, and hence the size of the captured images increases (*e.g.*, latest mobile phones have two to three megapixel cameras). Another solution is to burden the user less and implement the image upload as a background process that is executed only when the phone is idle.

However, the main issue found in the implementation of *MobShare* was the reliability of the image transfer from the phone to the server. The slow upload speed means that if the image transfer is not reliable, in other words, the pictures have to be uploaded several times, posting images to the server becomes an overwhelming bottleneck. Therefore, we implemented special confirmation and acknowledgement into the image transfer process. By this we were able to inform and keep the user aware of the successfulness of the upload.

The other main issue was designing the database for both images and metadata. Combining the pictures and metadata of multiple users required advanced queries, and the database schema and caching had to be optimized for best performance.

Overall the straightforward client-server architecture supported the main design principle: pictures taken with the phone are shared on a web server. Especially, the dedicated client application enabled full control over the capture and sharing process, and the web server provided a centralized and easily accessible way to browse, discuss, and combine the shared images. *MobShare* also proved that the design of the system could be implemented with widely adopted current mobile phone and server-side technology.

## 6. FUTURE WORK

Our future work will focus on how people's picture taking habits and the subjects of photography change as mobile phone cameras are more widely accepted and adopted. In addition, we will conduct user studies to get feedback on the usability of the system, and user insight into implementing new features to the system (*e.g.*, printing and archiving of images, searching functionality, and time zone shifting). Also, the system already gathers network cellID information during image capture and transfers it along with other metadata to the server. Future work will take advantage of this contextual information in generating and inferring content metadata for the images, thus combining our previous work on mobile media metadata [8, 27, 34] with picture sharing.

One of the broader future research issues is photography as a collaborative group activity by enabling the combination and comparison of other people's pictures. *MobShare* makes it very easy to record and document events with several cameras into one common gallery, thus blurring the boundaries of ownership and authorship in images and image collections.

## 7. CONCLUSIONS

We have presented the design and implementation of a picture sharing system *MobShare* that enables immediate, controlled, and organized sharing of mobile pictures, and the browsing, discussing, and combining of the shared pictures. The design is based on related literature on consumer photography, personal digital image management, camera phone use, and current mobile picture publishing systems. The literature study was supported by interviews we conducted on camera phone users.

The implemented *MobShare* system is a client-server architecture using current and widely adopted technology. The client is a Series 60 compatible phone with a camera, and the server is a web server with a database. The communication protocol used is HTTP over the GPRS network, and the web user interface is designed for a desktop web browser.

In the implementation of the system we described novel solutions in both sharing and viewing the mobile phone pictures. We implemented a contiguous process from image capture to sharing of images into an organized web repository. We also presented our solution to enabling full user control with whom the pictures

---

[2] http://jakarta.apache.org/tomcat/

[3] http://www.postgresql.org/

[4] http://jakarta.apache.org/struts/

are shared by leveraging the phone's address book. In viewing the images on the server we presented an implementation that promotes and manages discussions of individual pictures and picture galleries. We also implemented a new way for users to combine and compare both their own and other people's galleries.

Mobile phone cameras are still a novelty for most of the people, and how people will adopt the device remains to be seen. However, we believe that the *MobShare* system is a novel approach to future mobile picture sharing and management. It shows that with a simple architectural design the special characteristics and uses of mobile phone cameras can be taken into account and applied to well-known personal image management principles. We believe that as *MobShare* is extensively tested on users we will gain more understanding and identify new research problems in the domain of mobile imaging.

## 8. ACKNOWLEDGEMENTS

## 9. REFERENCES

[1] ACDSee Systems. ACDSee. http://www.acdsystems.com/

[2] Adobe. Photoshop Album 2.0. http://www.adobe.com/

[3] Apple Computer. iPhoto. http://www.apple.com/ilife/iphoto/

[4] Balabanovic, M., Chu, L.L., and Wolff, G.J. Storytelling with digital photographs. In Proc. of CHI 2000. ACM Press, 2000, 564–571.

[5] Bederson, B. PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and BubbleMaps. Proc. Of UIST 2001. ACM Press, 2001, 71-80.

[6] Cognima Ltd. Cognima Snap. http://www.cognima.com/

[7] Cooper, M.D., Foote, J., Girgensohn, A., and Wilcox, L. Temporal Event Clustering for Digital Photo Collections. In Proc. of Multimedia 2003. ACM Press, 2003, 364-373.

[8] Davis, M., and Sarvas, R. Mobile Media Metadata for Mobile Imaging. In Proc. of ICME04. IEEE Society Press, 2004.

[9] Frohlich, D., Kuchinsky, A., Pering, C., Don, A., and Ariss, S. Requirements for Photoware. In Proc. of CSCW '02. ACM Press, 2002, 166-175.

[10] Futurice. PhotoBlog. http://www.futurice.fi/

[11] Gargi, U., Deng, Y., Tretter, D.R. Managing and Searching Personal Photo Collections. HP Labs Technical Report HPL-2002-67, 2002.

[12] Gargi, U. Modelling and Clustering of Photo Capture Streams. In Proc. of Multimedia 2003. ACM Press, 2003, 47-54.

[13] Girgensohn, A., Adcock, J., Cooper, M., Foote, J., and Wilcox, L. Simplifying the Management of Large Photo Collections. In Proc. of INTERACT'03, IOS Press, 2003, 196-203.

[14] Graham, A., Garcia-Molina, H., Paepcke, A., and Winograd, T. Time as Essence for Photo Browsing Through Personal Digital Libraries, In Proc. of JCDL'02, ACM Press, 2002, 326-335.

[15] Hirsch, M. *Family Frames: Photography, Narrative and Postmemory*. Harvard University Press, 1997.

[16] Ito, M. and Okabe, D. Camera phones changing the definition of picture-worthy. Japan Media Review (2003). http://www.ojr.org/japan/wireless/1062208524.php

[17] IPSe Marketing Inc. The Mobile Phone Morphs into "Camera-equipped Email Terminal". Online report, 2003. http://www.ipse-m.com/company/release/release_02_e.htm

[18] Koskinen, I., Kurvinen, E. and Lehtonen, T.-K. *Mobile Image*. IT Press, Helsinki, 2002.

[19] Kuchinsky, A., Pering, C., Creech, M.L., Freeze, D., Serra, B., and Gwizdka, J. FotoFile: A Consumer Multimedia Organization and Retrieval System. In Proc. of CHI '99. ACM Press, 1999, 496-503.

[20] Mawston, N. Camera Phones Outsell Digital Still Cameras in H1 2003 and Beyond. In Wireless Devices Strategies, Strategy Analytics, Newton Centre, MA, 2003.

[21] Mäkelä, A., Giller, V., Tscheligi, M., and Sefelin, R. Joking, Storytelling, Artsharing, Expressing Affection: A Field Trial of How Children and their Social Network Communicate with Digital Images in Leisure Time. In Proc. of CHI'00. ACM Press, 2000, 548-555.

[22] Nokia. LifeBlog. http://www.nokia.com/lifeblog

[23] Ofoto Inc. Kodak Mobile. http://www.kodakmobile.com/

[24] Platt, J.C., Czerwinski, M., and Field, B.A. PhotoTOC: Automatic clustering for browsing personal photographs. Microsoft Research Technical Report MSR-TR-2002-17.

[25] Pyra Labs. Blogger. http://www.blogger.com/

[26] Rodden, K. and Wood, K.R. How Do People Manage Their Digital Photographs? In Proc. of CHI03. ACM Press, 2003, 409-416.

[27] Sarvas, R., Herrarte, E., Wilhelm, A., and Davis, M., Metadata Creation System for Mobile Images. In Proc. of MobiSys04, ACM Press, 2004 36-48.

[28] Series 60 Developer Platform 1.0/2.0: Using the Phone Book Engine, Version 1.0, Nokia, 2004.

[29] Six Apart. Typepad. http://www.typepad.com/

[30] Spence J. & Holland P. *Family snaps: The meanings of domestic photography*. Virago Press, 1991.

[31] Stevens, M.M., Abowd, G.D., Truong, K.N., Vollmer, F. Getting *into* the Living Memory Box: Family Archives & Holistic Design, *Personal Ubiquitous Computing (7)*, 2003, 210-216.

[32] Textamerica. Textamerica. http://www.textamerica.com/

[33] Toyama, K., Logan, R., Roseway, A., Anandan, P. Geographic Location Tags on Digital Images. In Proc. of Multimedia 2003. ACM Press, 2003, 156-166.

[34] Wilhelm, A., Takhteyev, Y., Sarvas, R., Van House, N., and Davis, M. Photo Annotation on a Camera Phone. In Proc. of CHI2004, ACM Press, 2004, 1403-1406.