

Mobile Phones and Paper Documents: Evaluating A New Approach for Capturing Microfinance Data in Rural India

Tapan Parikh, Paul Javid
Department of CSE
Univ. of Washington
tapan,pjavid@cs.washington.edu

Sasikumar K.
ekgaon technologies
Madurai, India
sasi@ekgaon.com

Kaushik Ghosh
Human Factors India
Mumbai, India
kaushik@humanfactors.com

ABSTRACT

CAM is a user interface toolkit that allows a camera-equipped mobile phone to interact with paper documents. It is designed to automate inefficient, paper-intensive information processes in the developing world. In this paper we present a usability evaluation of an application built using CAM for collecting data from microfinance groups in rural India. This application serves an important and immediate need in the microfinance industry. Our quantitative results show that the user interface is efficient, accurate and can quickly be learned by rural users. The results were competitive with an equivalent PC-based UI. Qualitatively, the interface was found easy to use by almost all users. This shows that, with a properly designed user interface, mobile phones can be a preferred platform for many rural computing applications. Voice feedback and numeric data entry were particularly well-received by users. We are conducting a pilot of this application with 400 microfinance groups in India.

Author Keywords

paper user interface, visual codes, document processing, mobile phone, microfinance, rural development, ICT

ACM Classification Keywords

H5.2. Information interfaces and presentation (e.g., HCI): User Interfaces.

INTRODUCTION

Microfinance is the provision of financial services to poor, disadvantaged and under-served communities. Including services such as loans, savings and insurance, microfinance is an effective tool for enabling sustainable local socio-economic development. One of the biggest challenges facing microfinance service providers, particularly in rural areas of the developing world, is implementing a Management Information System (MIS) that can interface with a large number of clients across a region with unreliable physical infrastructure (communication, power, transport, etc.) [14].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2006, April 22-27, 2006, Montréal, Québec, Canada.

Copyright 2006 ACM 1-59593-178-3/06/0004...\$5.00.

Mobile phones have been cited as the most likely modern digital tool to support economic development in under-developed regions [18]. As shown in the example of Grameen Phone [2], if a phone is shared by a group of people, it can be afforded by even the poorest communities. For rural computing applications, a mobile phone has inherent advantages over a PC in terms of cost, portability and familiarity to users.

Previously, we conducted a design experiment with microfinance groups in rural India [17]. We observed that paper plays an important role in group record-keeping. We demonstrated that a user interface based on graphical icons, voice feedback in the local language, and numeric data entry was well-suited for rural users that were new to computing and who may be functionally illiterate. We have implemented these guidelines in CAM, a mobile phone user interface toolkit designed to automate inefficient, paper-intensive information processes in India and elsewhere in the rural developing world [15].

In this paper we present an instantiation and usability evaluation of CAM for a specific application — collecting data from rural microfinance groups in India. Mobile user interfaces are frequently cited as less usable than PC interfaces due to their limited screen size and input capabilities [1]. Our quantitative results show that a CAM-based user interface is efficient, accurate, can quickly be learned by rural users, and is competitive with an equivalent web-based UI on a PC. Qualitatively, CAM was described as easy to use by almost all users. This justifies our design and shows that, with a properly designed user interface, mobile phones can be a viable platform for many rural computing applications. Based on this successful evaluation, we are planning a pilot implementation of CAM with 400 microfinance groups near Madurai, Tamil Nadu, India.

THE CAM INTERFACE

CAM is a user interface toolkit that allows a camera-equipped mobile phone to interact with paper documents containing visual codes — two-dimensional data glyphs that can be recognized and decoded from a camera image [21]. Visual codes serve as references to interactive multimedia functions — allowing the user to enter data, review values and communicate with a remote database. CAM is designed to link inefficient, paper-intensive processes to online information systems.

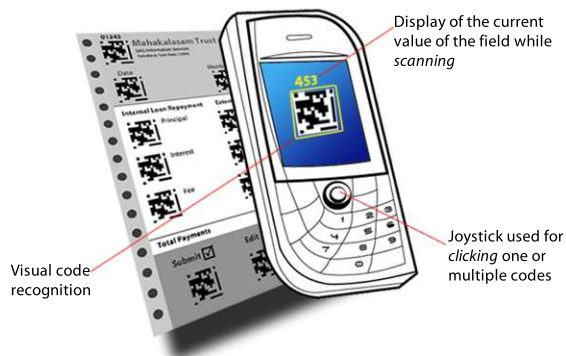


Figure 1. The CamBrowser application. The current value for the field is displayed beside the visual code.

CamBrowser

The CAM client application is called the CamBrowser, and has been implemented for phones based on Nokia's Series 60 platform. CamBrowser acts like a virtual window and one-button mouse for paper documents. A phone's physical interface is limited to a small screen, a 12-button numeric keypad and some *soft-keys*. By recognizing visual codes using the phone's camera, this interface can be extended to a nearly infinite number of widgets situated in the real world. For code recognition we are using the software developed by Rohs and Gfeller [21]. We have re-packaged the recognition functionality and compiled it as a stand-alone library.

Other researchers have presented a rich set of interactions made possible by tilting, rotating and translating the camera relative to an individual visual code [22]. However, it also has been observed that most of these interactions were not accessible to novice users [25]. In CAM, we have simplified the interaction to two primitives. The user *scans* codes by moving the viewfinder over visual codes in real time, or *clicks* codes by taking a high-resolution image using the joystick button. Our aim was to design a simple and intuitive interaction model with well-defined affordances between actions.

CamForms

CamForms are paper documents containing visual codes. An example CamForm is shown in Figure 2. CamForms are organized in the following canonical sections:

- **Header** — The user begins interacting with a form by clicking on the header (located at the top of Figure 2). The header visual codes identify the form type and instance to the CamBrowser. Depending on the form, the browser may then automatically display a sequence of dialogs for transcribing the hand-written values using the keypad (as in the shortcut code discussed later in this paper).
- **Input Codes** — Input codes are like HTML input tags. They are co-located with form fields (located in the middle of the form in Figure 2). When the user scans an input code, the current value is shown on the screen. This can either be what the user had entered, or the result of some calculation. To edit the field, the user clicks on the

associated code, which brings up a dialog window. When a set of input codes is clicked together, their respective dialogs are displayed in sequence.

- **Buttons** — Once the user has transcribed all of the form fields, the user can submit the data to the server. To do this the user clicks on the *Submit* button, seen at the bottom of Figure 2. The second button is for refreshing the displayed data from the server, and the third is for deleting the database record corresponding to this form instance.

CamShell

CamShell is the API used to specify interaction with visual codes. Each visual code invokes a distinct callback function when it is *clicked* or *scanned*. These functions are programmed in a XML-based scripting language that includes support for function calls, control flow, arithmetic and basic datatypes [24]. Each callback contains a number of sequential actions, some of which may be executed conditionally. This can be used for data validation. For a scan callback, the value returned by the function is displayed on the screen. This is used to display the value entered for a field, the result of some calculation or the name of the form or button. An example callback function is shown below:

```
<function name="u_click" params="param1">
  seq = input_int("Please input Form ID");
  if (!seq) return false;
  uri = "http://abc.com/reload.php?"
    . "seq=" . seq;
  return http_get(uri);
</function>
```

CamShell provides an API for accessing the mobile phone's features. Example API functions include launching a user dialog, recording an audio clip, saving an image, making an HTTP request or sending an SMS message. Every action can be accompanied by voice and graphical prompts. By using audio, images and numeric data entry, CAM applications can be localized even if the phone's operating system does not support the target language.

APPLICATION BACKGROUND

In this section we introduce the target application — collecting data from *Self-help Groups* (SHGs). SHGs are a kind of autonomous microfinance group prevalent in India. There are estimated to be more than 1 million SHGs in India, with a total membership of more than 15 million (90% of whom are women). This constitutes one of the largest and fastest growing microfinance activities in the world.

SHGs are supported by *Self-Help Group Promoting Institutions* (SHPIs). While most SHPIs are non-profit Non-Governmental Organizations (NGOs), some are government agencies, commercial banks, farmers cooperatives, or even private individuals. SHPIs employ *field staff* to form and train groups. Field staff are recruited from villages and rural areas near the districts where they will work. They usually have at least a partial high school education and are paid a small salary or commission per group.

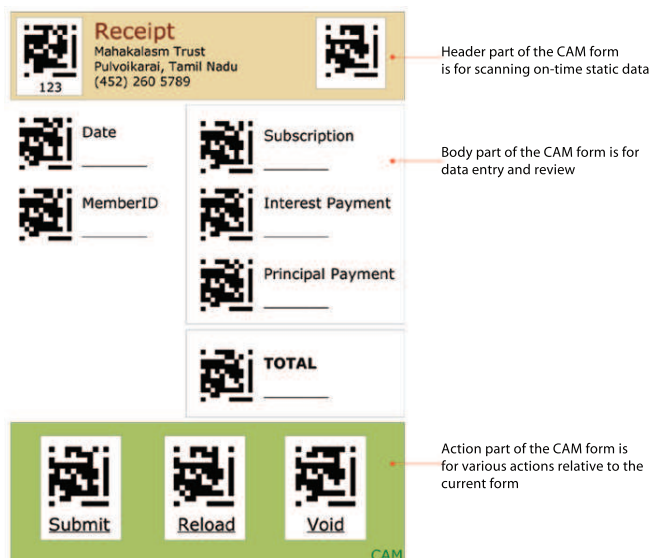


Figure 2. An example CamForm, showing the three sections the user interacts with.

An SHG consists of between 10-20 members. While the education level and affluence of members varies by group, most are very poor and minimally educated or literate (if at all). SHGs accumulate pools of money through their own savings. Eventually they can also access loans from banks and other commercial institutions. This money is redistributed within the group as loans, to be repaid with interest at regularly scheduled *group meetings*.

Mature SHGs are expected to document their own meetings and manage their internal accounts and lending. If none of the members is capable, a local educated person or school-going child is enlisted as the SHG *accountant*. However, most SHGs are still not able to produce financial statements and lending reports. They have inadequate accounting controls and systems for monitoring risk in the loan portfolio. Embezzlement, loan delinquency and mis-management are common. Therefore banks and other lenders find it difficult to assess SHGs' stability and financial history when considering loan applications.

Even for field staff with a basic education, the current accounting and documentation processes are onerous. For the past two years we have been working with a NGO based in Tamil Nadu, India to make the SHGs' paper-based accounting processes simpler and easier to learn. Even using this simplified system, it has taken over six months to train field staff to produce basic reports and statements. These are still inconsistently prepared, and errors are frequent.

If we handle accounting and report preparation automatically, it would dramatically reduce the workload of field staff, and improve the efficiency and transparency of SHGs. Currently, very few SHPIs have succeeded in computerizing at the SHG level. We have developed an application using CAM for collecting data from SHGs. We are planning a pilot of this application in partnership with the same NGO

in Tamil Nadu. Field staff will be equipped with a mobile phone to record member-level transactions. CamForms will be printed in bulk and physically distributed to field staff and groups. Transactions will be entered on the phone using the CamBrowser application, at the time of the meeting, or afterward from the paper records. This data will be posted to an on-line server via an SMS message. The resulting reports and financial statements will be printed from a secure access point, or faxed from the NGO head office to the field office, from where they will be picked up by the group or delivered by the staff. These reports will be used by the group for monitoring their portfolio and applying for loans and other services. Less common documentation requiring textual data entry (such as registering a new group or submitting a group loan application) will be faxed to the head office, where the data will be entered by a trained local language data entry operator. If the image resolution is sufficient, we can use the mobile phone camera to transfer these documents.

USABILITY TESTING

In this section we present a usability evaluation of CAM as a mobile user interface for capturing SHG records. The results will help us determine whether CAM meets the performance, accuracy and accessibility requirements for this application. We also compare the CAM interface to web-based data entry on a PC. While we do not expect CAM to be more efficient, we can determine if it is a viable alternative given the phone's other advantages in cost, portability and familiarity.

The quantitative experiment consisted of a set of timed user trials measuring the speed and accuracy of data entry using CAM. Ease of learning and recall were assessed by repeating the experiment over a period of four days, followed by another test five to twelve days later. Qualitative data was collected on the basis of self-report questionnaires, spot polls, informal feedback, and direct observation.

User Group

The subjects in our user study were the staff from two NGO field offices that are participating in the pilot. All of the subjects were educated rural women from the Indian state of Tamil Nadu. In the Pulvoikarai field office, the education ranged from 8th to 12th grade completion, with an average of 9.7 years of education. In the Natham field office, the women averaged 11.9 years of education. A group of Computer Science graduate students were tested in Seattle, WA as a benchmark for experienced technology users.

| Group | Users | Avg Age | Avg Educ. | M | F |
|-------------|-------|---------|-----------|---|---|
| Pulvoikarai | 6 | 27.8 | 9.7 | 0 | 6 |
| Natham | 8 | 22.4 | 11.9 | 0 | 8 |
| Seattle | 9 | 26.6 | 18+ | 5 | 4 |

Table 1. Basic demographics for each of the user groups.

While all of the Tamil Nadu users were literate and could perform arithmetic, they had very little technology exposure. Some had participated in our earlier experiments [17]. In Natham, a few had been using drawing applications on a

recently purchased office computer. This had conferred the important skill of using the mouse. In Pulvoikarai only one of the women had previous PC experience. All had used phones before, either mobile or landline. All of the staff used calculators for their current accounting tasks.

Task Description

The task we evaluated was recording the payments made by one group member during a meeting. Following accounting convention, this transaction was recorded as a group *receipt*. The design of the receipt is shown in Figure 2 (translated to English from Tamil). The receipt includes input codes for the Date, Member ID, Savings, Interest and Principal Repayment amounts. The Total is auto-calculated and displayed upon scanning. Buttons are included for submitting or reloading the data from the database or for voiding the transaction.

We measured the execution time and error rate for entering and submitting four receipts in sequence. We used a Nokia model 6600 phone running the CamBrowser application. The paper receipts were pre-populated with written values for each field. The user had to capture an image of the header plus each of the input codes — either together or in sequence. Each input code brought up a prompt to enter the value of the field. The prompts were displayed in Tamil (as a bitmap since the phone did not have a Tamil font installed), accompanied by Tamil audio indicating the name of the field. The data entry was all numeric (member identity was recorded as a numeric ID). After entering all the values, the user captured the ‘Submit’ code to post the transaction to the web server. For the purpose of this experiment, the web server was running on a laptop connected to the phone via a bluetooth connection.

The use of the application was explained and demonstrated to each of the users on the first day, after which they were given up to ten minutes to practice. On subsequent days the practice period was limited to 2-3 practice trials. After the task, the data on the server was reviewed to determine if there had been any errors. Each database value different from what was written on the paper form was considered one error. Execution time was measured by an observer with a stopwatch, from when the user focused the phone camera on the form until when the data for the fourth receipt had been received by the server.

To compare these results to PC-based data entry, each user performed the same task using a similarly designed web form. This test was conducted using a Sony Vaio laptop with an attached USB mouse. The same receipts and written values were used. The user had to enter one more three-digit value — the sequence number of the particular receipt instance (embedded within the visual code in the header for the CAM version). When the user submitted the form, the data was posted to a server running on the same laptop. The order of the two variations was counter-balanced and randomized. The test was repeated each day for four days (three for the Seattle users).

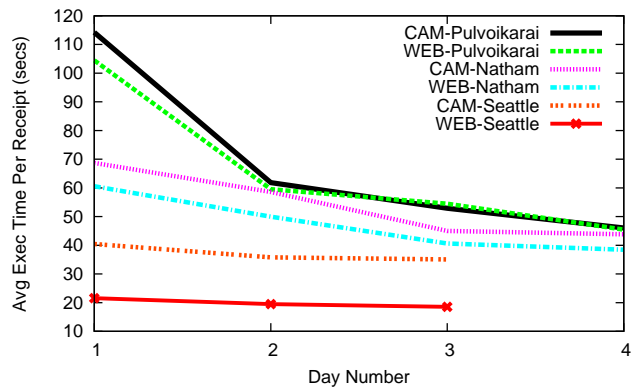


Figure 3. Graph showing average execution time (per receipt) on each day for the basic variations. CAM performance is comparable to the WEB for Tamil Nadu users. The standard deviation was high on the first day, especially in Pulvoikarai (73 sec. for CAM, 72 for WEB), but then stabilized (7-11 sec. by the last day for all conditions).

Efficiency and Learnability

Figure 3 shows that CAM was learned effectively by all user groups within a three day period. In Pulvoikarai, CAM performed equally well to the Web alternative. In Natham, CAM performed only slightly worse (averaging 10% to 12% slower). In neither case was the last-day difference statistically significant. In contrast, the Seattle users performed the task much faster with the Web variation ($p < 0.001$). Partly this was because they used the Tab key to switch between form fields and pressed Return to activate the ‘Submit’ button. The Tamil Nadu users were not aware of these shortcuts.

This was significant, as some of them had difficulty using the mouse and the mobile phone camera as targeting devices. They often had trouble quickly accessing the desired field. Shortcuts like the Tab key can speed up data entry when the random access functionality of a general purpose interface is not needed. We introduced a similar shortcut to the CAM receipt, in the form of a visual code button linked to a callback function that iterated through all of the required data entry prompts and actions (this code can be seen in the top right of Figure 2). This optimized for the common case when the user wants to enter the data for the form and submit as fast as possible.

We developed another variation where the phone was placed in a cradle on top of a wooden box, with the camera looking through a hole at the document inside (see Figure 4). This was a prototype of a kiosk or ATM version of the application, where the mobile phone is part of a larger installation with the capacity to store receipts and money. From a usability perspective, this further reduced the motor control problem, by placing the paper at a fixed distance from the camera, and removing the phone from the user’s potentially unsteady hand. A holder for a flashlight was included inside the box to improve the clarity of the captured image.

With the Natham user group, we tested all three of the new variations (Web, after telling users to use the TAB key;



Figure 4. The BOX variation was a major improvement for users that had trouble with fine motor control. It fixed the paper document at the correct distance from the camera and stabilized the phone.

CAM, after introduction of the shortcut CODE; and CAM, using both the shortcut code and the BOX), at the same time as the earlier protocol, counter-balanced and randomized with the earlier variations. The results are in Figure 5.

All of the shortcuts led to improvements in average execution time - both first and last day performance were significantly improved for the CODE and BOX variations ($p < 0.005$). The variance was also reduced. The first day standard deviation was 4.6 seconds for the BOX variation, 6.9 seconds for the CODE variation and 14.9 seconds for the regular CAM variation. By the fourth day these were reduced to 2.4 seconds for both the CODE and BOX, and 7.2 seconds for CAM. The BOX variation was slightly faster than CODE on the first day, but the difference was not significant. The last day average was between 32 and 33 seconds per receipt for all three new variations.

To show that these improvements were applicable to the less-educated Pulvoikarai group, we tested these variations on the fourth day of their protocol. They averaged 36 seconds for the CODE variation, 39 seconds for the BOX and 42 seconds for the web using the TAB key. Again, the shortcut code was a significant improvement over the basic CAM variation ($p < 0.05$).

Error Rate

On each receipt a total of five values were entered — the Date, the Member ID, the Savings amount and the Interest and Principal repayments. The error rate was defined as the percentage of these values stored in the database that did not match the value written on the receipt. As Table 2 shows, the observed error rates were 1% or below for each of the conditions. The two Tamil Nadu groups made more errors than their Seattle counterparts.

Memorizability

To evaluate the memorizability of this interface, a follow-up test was conducted five to twelve days later without intervening use of either application. Table 3 shows that except for

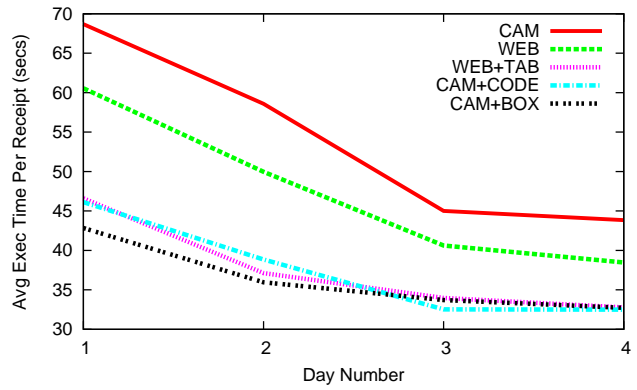


Figure 5. Graph showing average execution time (per receipt) for all variations conducted with the Natham group. The new variations improved efficiency and speed of learning. Standard deviations were reduced for the BOX and CODE conditions, starting at 5-7 sec. and reaching 2 sec. by the last day.

the Pulvoikarai group, the difference between the last test in the first set of trials (day 3 or 4) and the repetition test (5-12 days later) was less than 5% for all conditions. In Pulvoikarai the difference could have been more due to the longer time interval, but even there the difference was not extreme.

Qualitative Survey

On the last day of the protocol users were asked to complete a short survey asking them to rate on a 1-5 Likert scale (5 being very easy) how easy it was to perform various actions using each interface. From the results in Table 4, it is clear that both the CAM and WEB interfaces were considered easy to master by almost all users. None found either interface difficult or very difficult.

| Group | Variation | Error Rate |
|-------|-----------|------------|
| Pul | CAM | 0.5% |
| Pul | WEB | 0.6% |
| Nat | CAM | 1.0% |
| Nat | WEB | 0.6% |
| Nat | WEB-TAB | 0.8% |
| Nat | CAM-CODE | 0.6% |
| Nat | CAM-BOX | 0.6% |
| Sea | CAM | 0.0% |
| Sea | WEB | 0.1% |

Table 2. Error rate of the variations.

| Group | Variation | Last | Repetition | Interval |
|-------|-----------|---------|------------|----------|
| Pul | CAM | 46 sec. | 49 sec. | 12 days |
| Pul | WEB | 45 sec. | 52 sec. | 12 days |
| Nat | CAM | 44 sec. | 42 sec. | 8 days |
| Nat | WEB | 38 sec. | 38 sec. | 8 days |
| Nat | WEB-TAB | 33 sec. | 33 sec. | 8 days |
| Nat | CAM-CODE | 33 sec. | 34 sec. | 8 days |
| Nat | CAM-BOX | 33 sec. | 34 sec. | 8 days |
| Sea | CAM | 35 sec. | 37 sec. | 5 days |
| Sea | WEB | 19 sec. | 18 sec. | 5 days |

Table 3. Memorizability of the variations.

| Group | Variation | Very Easy | Easy | Medium |
|-------------|-----------|-----------|------|--------|
| Pulvoikarai | CAM | 6 | 0 | 0 |
| Natham | CAM | 2 | 6 | 0 |
| Seattle | CAM | 5 | 3 | 1 |
| TOTAL | | 13 | 9 | 1 |
| Pulvoikarai | WEB | 6 | 0 | 0 |
| Natham | WEB | 5 | 3 | 0 |
| Seattle | WEB | 9 | 0 | 0 |
| TOTAL | | 20 | 3 | 0 |

Table 4. Number of respondents rating ease of use of each interface on a 1-5 Likert scale. None found either interface difficult or very difficult.

Other Tests

We conducted the following experiments after completion of the basic protocols described thus far.

Audio

To assess the importance of the voice prompts, we tested versions of the CAM interface without audio (text-only), without text (audio-only), and with both text and audio (normal). This experiment was conducted with 13 users from the Pulvoikarai and Natham groups. The order of the variations was counter-balanced and randomized. The shortcut code was included.

The text-only variation was slightly faster on average, maybe because users did not always wait for the voice prompt before entering a value, but the difference was not statistically significant. In fact, users unanimously preferred the audio accompaniment — some users even mentioned that the text was not necessary. The removal of audio was disconcerting, particularly after five days of using CAM with audio. This is reflected in the higher error rate for the text-only condition.

| Variation | Exec. Time | Error Rate |
|----------------|------------|------------|
| Audio-only | 32 sec. | 0.0% |
| Text-only | 29 sec. | 2.2% |
| Audio and Text | 32 sec. | 0.0% |

Table 5. Execution times and error rates for the audio variations.

CAM without the Camera

After introducing the CODE variation, we decided it would be possible to develop a version of CAM that did not require the camera. We replaced the shortcut code with a prompt that asked the user to enter the form ID and visual code ID directly. This would trigger the callback and start the correct interaction sequence for each form.

We tested this variation with the Pulvoikarai users. The users were asked to enter a 2-digit value for the form ID, and then a 6-digit value including the visual code ID and the sequence number of the particular form instance (embedded earlier in the header). These values were printed on the receipt for ready reference. The results are shown in Table 6.

The additional data entry had a significant impact on the execution time ($p < .001$). The impact would have been greater if we had asked users to enter an IP address or

file name to access the correct form instance. As most of the programmable smart phones that run the CamBrowser application have a built-in camera, at present there is no reason to consider this alternative. However, in the future, this technique could be used for smart phones that do not have a camera. By printing the form and visual code IDs directly on the forms, we can maintain a strong linkage between the paper document and the digital process. Entering a numeric ID is more convenient on a phone keypad than entering a textual URL or traversing a menu structure. Designers at Nokia have cited inflation of menu items as a prime usability concern when adding features to mobile devices [12].

| Variation | Avg. Exec. Time | Error Rate |
|----------------|-----------------|------------|
| CAM w/ Camera | 35 sec. | 0.0% |
| CAM w/o Camera | 49 sec. | 0.0% |

Table 6. Comparing use of CAM without the camera.

Mobile Web Browser

With the Seattle group we tested the same task by asking users to fill out a web form using the Opera browser on the mobile phone. This mobile browser implementation had several significant usability defects that we will not detail here. Users averaged 61 seconds per receipt, almost twice the average time for the CAM variation and four times the average for the PC browser. Due to this poor performance and negative user feedback we decided that it would not be fruitful to test this variation with the Tamil Nadu users.

Other Observations

Motor Skills

Fine motor control was a consistent problem for the Tamil Nadu users, particularly for the older and less educated. This problem manifested in several ways. Using the mouse, when users had positioned the pointer they often changed their hand position to deliberately press the button, in the process shifting the mouse out of the intended focus area. The same effect was observed in the CAM variation — it was difficult to hold the camera steady while pressing the joystick button, leading to the image being captured out of focus. We also noticed users' hands shaking when using the mouse or phone. We attributed this to their nervousness at being tested and in using a device that they considered prohibitively expensive.

We note the benefits gained by removing actions requiring extra motor coordination and/or explicit decision points. By *guiding* the user through the interaction sequence, accompanied by voice feedback, the task becomes a dialog between the user and the machine that is natural to execute.

Using the camera was a completely new experience for this user group. It took them time to understand the correlation between shifting the camera position and the change in the scene observed through the viewfinder. Finding the correct focal distance between the camera and the paper document was also a problem, even for some Seattle users.

For those users that had extreme difficulty in using the camera and keeping their hand steady, the BOX variation was

a major improvement. The box fixed the paper document at the correct distance from the camera, and allowed the user to keep the phone in a stable location when pressing the button. After using the box, one of the Tamil Nadu users exclaimed “The box is GREAT! Now I can use the system with confidence!”. We believe this increase in confidence led to improved performance overall.

Phone Buttons

Users had difficulty pressing the buttons on the phone interface. The small size and physical closeness of the buttons led to incorrect key presses. Identifying the purpose of some buttons was also a problem. All test subjects had previously used a phone or a calculator so they could easily recognize the 12 keys on the numeric keypad. However, the six soft keys and the four-way directional joystick were unintelligible. Users had difficulty looking at the device screen and correlating a label listed there with a button on the phone body. In some cases this label conflicted with the labels printed on other buttons (for example, when the two soft-keys to the right and left of the joystick were labeled with ‘OK’ and ‘Cancel’, they conflicted with the green and red keys adjacent). It was confusing for users to decide which key to press. We reinforced that they needed to be concerned with only three soft-keys - the joystick button to take an image, and the OK and Cancel buttons to confirm or cancel an entry. Eventually some users learned that the OK key could also be pressed to capture an image, meaning that they could perform the task using only one soft-key.

Lighting

The system worked best in natural or diffuse lighting conditions. When the light came from an overhead point source, the mobile phone cast a shadow that made it difficult to decode visual codes from the captured image. This could be rectified by adding a flash. When using the BOX variation at night we attached a flashlight inside the box to ensure proper lighting of the document.

Group vs. Individual Use

All of the tests described thus far were conducted in the NGO field office. This is far from a typical laboratory setting. Other field staff, SHG members and passers-by were commonly present. Unrequested guidance, cajoling and sometimes even intervention were difficult to avoid. In an earlier experiment we found that users performed significantly better when allowed to complete tasks in relative isolation [16].

Because in this experiment we were interested in the performance of CAM in a realistic application context, we decided not to control for this variable. In a village it is impossible that this task could be performed without significant observation and interference. However, this is a research question that we plan to address again in the future.

IN SITU TESTING

Following the success of our laboratory study, we decided to simulate CAM data entry in situ. We traveled with NGO field staff to five separate SHG meetings. Four out of the

five staff we accompanied were users that had participated in the laboratory experiment. After the meeting was completed we measured the time it took to capture the meeting records using CAM, and the number of errors that were made. We used either the CODE or BOX variation of CAM, as per the user’s preference.

All the meetings consisted of between 13 to 21 members, and took between 6 to 8 minutes to capture using CAM. Given that the average length of a meeting is between 60 to 90 minutes, this was only a small percentage of the overall time. Moreover, several of the conventional records and ledgers could be automatically generated, further shortening the length of the meeting. One field staff estimated that by using CAM the average length of an SHG meeting could be shortened from 90 to 45 minutes. Amazingly, all five meetings were also recorded without error. Even considering that the in situ experiments were conducted with some of the better CAM users, this gave us confidence that the system was ready for a real deployment.

ACCEPTANCE BY SHG MEMBERS

Testing CAM in situ gave us an opportunity to evaluate CAM for another important class of user — SHG members. During the pilot SHG members will not capture their own transactions, but are *secondary users* in the documentation process [16]. Secondary users do not directly interact with the interface, but it is on their behalf that the task is being executed. Such *intermediated information tasks* are common in the developing world, where access to technology is limited due to cost and educational constraints. We wanted to understand the members’ appreciation of CAM interaction, and whether they would accept this new technology.

First we explained the purpose of the new process, and what would be the resulting benefits for the group. Then, after the CAM data collection, we conducted an informal qualitative interview with the group as a whole. Most aspects of the interaction were not intelligible to the SHG members as they could not see the screen or keypad. They repeatedly mentioned the audible prompts as their only indication about what was happening. Several members requested that their names and the amounts of each transaction also be audibly repeated so that they could confirm the correct data was entered for each member.

In general, members were confident their data had been captured correctly due to their familiarity with the field staff. Still, they were adamant that their individual paper records (the member passbooks and receipts) be retained in any future system. Initially, some group members insisted that the group ledger be retained also. It was only after we showed them an example of a printed report that they agreed the paper ledger was no longer necessary. This paper report was an important artifact from the group perspective that clarified the operation of the new system. Once they understood the time they would save in accounting and documentation, the group was very enthusiastic about the new system.

All of the groups we visited unanimously voted for immediate introduction of the system. It was not apparent whether this was because they understood the benefits, or that they wanted to be associated with some modern technology. It is likely that a combination of these factors contributed to their decision. It is equally clear from their insistence that the individual paper records be retained, and their happiness with the paper report, that they did not trust or understand the new technology entirely. They required a local record that could be referenced independently.

DISCUSSION

In this section we discuss the results of the quantitative and qualitative evaluation, and whether our implementation meets the requirements of the tested application.

Efficiency

In microfinance where the cash value of individual transactions is very small, the only way to be profitable is to serve many clients efficiently. This is an important measure of efficiency in the industry. If we can make an interface faster to use, it would allow each staff to serve more clients. In our experiment an optimized CAM user interface performed as efficiently as an equivalent PC interface. In fact, on average, novice rural users were able to complete the task using the CAM interface only 44% slower than experienced technology users on a PC. It is unlikely that any mobile interface could provide significantly better performance.

Accuracy

Obviously, in an application dealing with financial transactions, accuracy is of the utmost importance. The observed error rates of 0.6-1.0% were good but not completely satisfactory. In all the variations conducted after the main protocol (audio, without camera), and during the in situ experiments, users did not make a single error (except for the text-only case that we have already noted). This leads us to believe in the main protocol some users were either bored or so concerned with speed that they were not fully careful. When we asked them to complete the task in a new circumstance they became more attentive and the error rate again verged on zero. Still, in the pilot it will be important to be vigilant about errors and audit the paper records regularly.

Accessibility

NGO field staff are entry-level employees recruited from rural areas. The lower the bar for this position, the more potential participants. In this experiment we have only tested users with a base level of literacy and education. Currently this level can be assumed for the field staff and SHG accountants in charge of documentation for SHGs. While it had taken six months to train staff to prepare financial statements manually, the same staff learned to use CAM within four days. By preparing the reports and financial statements automatically we are well-positioned to deliver immediate benefits.

We believe that much of the CAM interface is still appropriate for the semi-literate demographic. We have already discussed the SHG members' reaction to CAM as secondary

users. Eventually we hope that SHGs will purchase their own phones and avail CAM services directly. But first we must demonstrate the feasibility and benefits of the system so that such an investment would be justified. It was a conscious choice to conduct the initial evaluation with younger, more educated subjects that will likely be the first adopters of this technology.

DESIGN GUIDELINES

In this section we discuss three key elements of the CAM system: paper, voice feedback and mobile phones; and how each contributes to the design of this application.

Linkage to Paper Processes

Even if groups purchase their own phones, they will still be shared between members. The paper receipt will be the only record that is immediately accessible to every member. Each CAM receipt will contain a persistent link to the corresponding database record. By reloading the data from the server, entered values can be confirmed using any CAM-equipped phone, not just the one belonging to the group or to a particular staff.

As CAM tasks are triggered by paper documents, the user does not have to enter a URL, scroll for a link or traverse a menu to find the application corresponding to a particular process. This may not be an issue while there are still a small number of applications, but it will become important when more paper-linked applications are developed. A URL is a good example of the power of an accessible, expressive indexing mechanism for applications and content. For CAM, we have discussed how either a visual code or a numeric identifier could be used as this index. For example, a phone number-like numeric string could be printed on the form that when entered starts the right interaction sequence.

Voice Feedback

Voice feedback ended up being more important than even we imagined. Earlier we found that audio helped functionally illiterate users interact with a user interface [17]. In this experiment we found local-language audio helped build a rapport even with literate users. We often found users (and ourselves) singing or joking along with common phrases uttered by the application ("Success! Your transaction has been recorded."). While the attitude was jovial at the time, the deep attachment created with users was a serious success.

This was particularly apparent for secondary users (SHG members), who otherwise had very little interaction with the system. Gaining the trust and confidence of SHG members will be an important requirement for the long-term acceptance of the system. When group members asked that their transaction amounts also be audibly repeated, we were concerned about the individual members' privacy. When we mentioned this, the users only laughed. In the current structure of a SHG, privacy is a non-issue. The payments and balances of each member are already known to everyone in the group and staff. In the future, if we introduce financial products for individuals, privacy may become more of a concern.

Mobile Phones

A mobile phone is a much more useful and familiar device in the rural context than a PC. Its communications facilities offer an immediate utility to any user. Solid state memory, extended battery-life and a compact, rugged form factor are all great design choices for the village environment. From a UI perspective, a camera-equipped mobile phone has the ability to support both voice feedback and linkage with paper processes. The numeric keypad's long history makes it comfortable for billions of users.

However, numeric data entry also constrains the data that is collected. This can be seen as a benefit. We have observed that it is difficult to maintain quality in free-form data entry with this user population, particularly for content in the local language. Inconsistent spelling and other mistakes are common. By restricting input to a discrete set of choices, consistency can be maintained. We have included image capture and audio recording as two ways to collect free-form data. This data can be processed further upstream (if at all).

Mobile phones also offer an obvious cost benefit. Most institutions that have computerized at the SHG-level have PCs at the field office used for entering data. From a price-performance perspective, CAM provides a clear advantage. The current market cost of the mobile phone used in our experiment is about half that of an entry-level Celeron PC. By reducing staff trips to the field office, travel costs can also be reduced, and field staff can spend more time in the field, allowing them to service more clients. There would also be less infrastructure required at the field office. Field offices might not be required at all — commercial banks without a rural branch infrastructure are already paying agents on a commission basis to form and manage SHGs. Using CAM this agent can operate using only a mobile phone and a printer or fax.

Still, even mobile phones are a large investment in the rural context, and only time will tell whether they are justified. We asked 14 staff members whether they would prefer the CAM documentation system, a PC-based system or the current paper-based system, considering all of the mitigating factors (usability, efficiency, travel, cost, etc.). 13 preferred CAM, one preferred keeping the paper notebooks and none preferred the PC-based solution.

RELATED WORK

The two main areas of related work are other methods of electronic data collection in microfinance, and other paper user interfaces (PUIs). There is no published work in the intersection of these domains. In fact there is little prior work evaluating the usability of any interface for the rural developing world [3, 17]. Of these, this is the first description of a quantitative test conducted in a controlled setting.

Electronic Data Collection in Microfinance

Several microfinance institutions have pilot-tested electronic data collection using PDAs or handheld computers [7]. Other institutions have implemented low-cost rural ATMs. Hewlett-Packard has developed a Rural Transaction System

(RTS) for capturing microfinance transactions using a mobile phone and a smart card reader. Three of the main problems observed with this system were 1) the system was not flexible for adapting to different business processes, 2) the time for a meeting was increased by up to an hour, and 3) the system did not work well with organizations' existing back-office solutions [9]. Being a programmable system linked to paper formats, our system is meant to complement existing processes, not replace them. Some organizations have issued smartcards to clients as a personal record of transactions. Paper receipts are a more accessible client record that can directly be referenced.

Paper User Interfaces

Researchers have commented on the importance of paper in workplace settings [13, 23], and sought to improve coordination between paper and digital media [4, 5, 8, 11, 19, 26]. Companies have extended these efforts to develop document processing systems that recognize structure and text from scanned paper forms and integrate with back end data sources [6]. None of these systems have used a mobile phone as the primary image capture and data entry device, and none provide a general purpose network-aware programming toolkit for specifying user interaction.

The Mobile Server Toolkit [25] is a visual-tag based system that allows mobile devices to access site-specific information services. The MST transmits individual UI events to an application running on a bluetooth-connected server. CAM handles all user interaction on the phone itself — messages to the server contain data elements only. CAM can also work offline and send messages asynchronously using standard protocols. These features make it more suitable for accessing remote, disconnected information services.

The Cooltown project [10] uses RFID, barcodes and other sensing technologies to retrieve specific HTML documents. The visual code widget library developed by Rohs [20] includes a declarative UI specification that links visual codes with a rich set of data entry widgets. Neither of these systems provide a scripted programming environment supporting voice feedback or sequences of actions.

CONCLUSION

In this paper we have quantitatively and qualitatively evaluated an example application built using the CAM toolkit for collecting data from rural microfinance groups in India. Our results show that this design is viable for this application, and that it is competitive with an equivalent PC-based UI. We are starting a pilot of this application in rural Tamil Nadu, India, covering 400 SHGs, 24 staff and over 6000 SHG members. This application eventually has the potential to impact up to 15 million SHG members in India. If we include similar groups in other regions, the number of possible beneficiaries increases to more than 100 million.

Our results demonstrate that given a suitable user interface, mobile phones can be a viable platform for rural computing applications. For such applications, a mobile phone has inherent advantages over a PC in terms of cost, portability

and familiarity to users. We have demonstrated that the CAM toolkit can be used to develop such an application. Specifically, we found that tight integration with paper processes, voice feedback and number-only data entry were successful for this application. CAM is the first toolkit that supports these features using a simple scripted programming model allowing for sequences of actions. Mobile UI designers have recognized that sequential execution is best suited for devices with limited screen space [12]. The success of the shortcut code in our experiment demonstrates this advantage.

There are other important applications in the rural developing world having similar user interface requirements. To name just two, *primary health data collection* and *multi-level marketing* of retail products both feature decentralized operation, a similar user demographic and the same dependence on paper processes [3]. We believe that CAM will be a good choice for automating these and other important information processes in the rural developing world.

ACKNOWLEDGMENTS

We would like to thank the Mahakalasm SHG Federations and the Covenant Centre for Development (CCD) for being supportive partners throughout this project. Thanks to David Notkin, Ed Lazowska, James Landay, Richard Anderson, Greg Wolff and Apala Lahiri Chavan for providing encouragement and valuable feedback. CCD and ekaon technologies provided the facilities and logistics required for the usability testing. Mr. Olaganathan of CCD was extremely helpful as a discussion leader. Thanks to our test subjects in Madurai and Seattle for contributing their time and energy. Thanks to the anonymous reviewers for their valuable feedback. Finally, thanks to the International Development Research Centre (IDRC), Microsoft Research India and the Marc Lindenberg Center for supporting this research.

ADDITIONAL AUTHORS

Kentaro Toyama, email: kentoy@microsoft.com.

REFERENCES

1. S. Brewster. Overcoming the lack of screen space on mobile computers. In *Personal Ubiquitous Comput.*, Springer-Verlag (2002), 6(3):188–205.
2. Grameen Phone home page. <http://www.grameenphone.com>, March 2005.
3. S. Grisedale, M. Graves, and A. Grunsteidl. Designing a graphical user interface for healthcare workers in rural India. In *Proc. CHI 1997*, ACM Press (1997), 471–478.
4. F. Guimbretière. Paper augmented digital documents. In *Proc. UIST 2003*, ACM Press (2003), 51–60.
5. J. M. Heiner, S. E. Hudson, and K. Tanaka. Linking and messaging from real paper in the paper PDA. In *Proc. UIST 1999*, ACM Press (1999), 179–186.
6. IFP success stories, June 2003. http://www2.clearlake.ibm.com/GOV/ifp/success_stories.html.
7. CGAP IT innovation series: PDAs in microfinance, June 2004. http://www.cgap.org/docs/IT_pda.html.
8. W. Johnson, H. Jellinek, J. Leigh Klotz, R. Rao, and S. K. Card. Bridging the paper and electronic worlds: the paper user interface. In *Proc. CHI 1993*, ACM Press (1993), 507–512.
9. M. Kam and T. Tran. Lessons from deploying the remote transaction system with three microfinance institutions in Uganda. In *Proc. of the Berkeley-UNIDO Bridging the Divide Conference*, April 2005.
10. T. Kindberg. Implementing physical hyperlinks using ubiquitous identifier resolution. In *Proc. WWW 2002*, ACM Press (2002), 191–199.
11. S. R. Klemmer, J. Graham, G. J. Wolff, and J. A. Landay. Books with voices: paper transcripts as a physical interface to oral histories. In *Proc. CHI 2003*, ACM Press (2003), 89–96.
12. C. Lindholm, T. Keinonen, and H. Kiljander. *Mobile Usability: How Nokia Changed the Face of the Mobile Phone*. McGraw-Hill, NY, NY, 2003.
13. W. E. MacKay. Is paper safer? the role of paper flight strips in air traffic control. In *ACM Trans. Comput.-Hum. Interact.*, ACM Press (1999), 6(4):311–340.
14. T. S. Parikh. Rural microfinance service delivery: Gaps, inefficiencies and emerging solutions. Submitted to ICTD 2006, May 2006.
15. T. S. Parikh. Using mobile phones for secure, distributed document processing in the developing world. In *IEEE Perv. Comp. Magazine*, 4(2):74–81, April 2005.
16. T. S. Parikh and K. Ghosh. Designing for intermediated information tasks in the developing world. Submitted to IEEE Pervasive Computing, April 2006.
17. T. S. Parikh, K. Ghosh, and A. Chavan. Design studies for a financial management system for micro-credit groups in rural India. In *Proc. CUU 2003*, ACM Press (2003), 15–22.
18. The real digital divide. *The Economist*, Mar. 2005.
19. J. Rekimoto and K. Nagao. The world through the computer: Computer augmented interaction with real world environments. In *Proc. UIST 1995*, ACM Press (1995), 29–36.
20. M. Rohs. Visual code widgets for marker-based interaction. In *Proc. IWSAWC'05: ICDCS 2005 Workshops*, June 2005.
21. M. Rohs and B. Gfeller. Using camera-equipped mobile phones for interacting with real-world objects. In *Advances in Pervasive Computing*, Austrian Computing Society (2004), 265–271.
22. M. Rohs and P. Zweifel. A conceptual framework for camera phone-based interaction techniques. In *Proc. Pervasive 2005*, Springer-Verlag(2005).
23. A. J. Sellen and R. H. Harper. *The Myth of the Paperless Office*. MIT Press, Cambridge, MA, 2003.
24. Simkin language, May 2005. <http://www.simkin.co.uk/>.
25. E. Toye, R. Sharp, A. Madhavapeddy, and D. Scott. Using smart phones to access site-specific services. *IEEE Perv. Comp. Magazine*, 4(2):60–66, April 2005.
26. P. Wellner. Interacting with paper on the DigitalDesk. *Communications of the ACM*, 36(7):86–97, 1993.