Pen-top Feedback for Paper-based Interfaces

Chunyuan Liao, François Guimbretière
Human-Computer Interaction Laboratory
Department of Computer Science
University of Maryland
College Park, MD 20770,USA
Tel: 1-301-405-2715
{liaomay, francois}@cs.umd.edu

Corinna E. Loeckenhoff
National Institute on Aging
Box #03, Gerontology Research Center
5600 Nathan Shock Drive
Baltimore, MD 21224, USA
Tel: 1-410-558-8196
{LoeckenhoffC}@grc.nia.nih.gov

ABSTRACT

Current paper-based interfaces such as PapierCraft, provide very little feedback and this limits the scope of possible interactions. So far, there has been little systematic exploration of the structure, constraints, and contingencies of feedback-mechanisms in paper-based interaction systems for paper-only environments. We identify three levels of feedback: discovery feedback (e.g., to aid with menu learning), status-indication feedback (e.g., for error detection), and task feedback (e.g., to aid in a search task). Using three modalities (visual, tactile, and auditory) which can be easily implemented on a pen-sized computer, we introduce a conceptual matrix to guide systematic research on pen-top feedback for paper-based interfaces. Using this matrix, we implemented a multimodal pen prototype demonstrating the potential of our approach. We conducted an experiment that confirmed the efficacy of our design in helping users discover a new interface and identify and correct their errors.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Experimentation, Human Factors.

Keywords: Paper-based interfaces, Pen interfaces, multimodal, feedback.

INTRODUCTION

The availability of new digital pens capable of capturing marks made on paper documents has created a renewed interest in paper-based interfaces such as the Xax [12] and PaperPDA [9]. The Anoto system [1], for example, combines a unique pattern printed on each page with a digital pen to capture strokes made on paper. Not only do digital pens alleviate the need for scanning annotated paper documents, they also provide accurate timing information for each stroke. This feature makes it possible to implement paper-based command systems. The Anoto command sys-
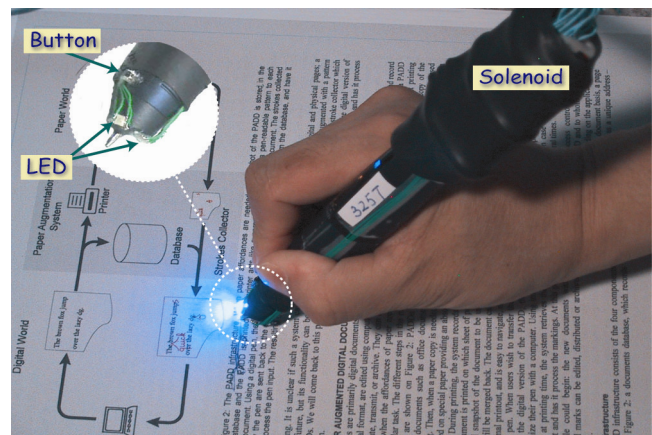
Figure 1: A prototype of the multimodal pen working on paper. It provides pen-top visual, tactile and auditory feedback through LED around the pen tip, solenoids at the top and an earphone (in this prototype connected to a PC for simulation) respectively. A button is used to switch between ink and gesture mode.

tem is based on small icons printed on paper (pidgets), but other approaches are also possible. PapierCraft [17], for example, proposed a command system which uses pen gestures on paper to support active reading. However, neither system offers any feedback beyond the ink left on paper. This severely limits the scope of features that can be offered. While it would be possible to rely on nearby computers [19, 30] to provide the necessary feedback, this approach constrains the flexibility of paper usage.

The Fly "pen-top" computer [15] introduced by Leap Frog offers an interesting compromise to this dilemma. Using its own version of the Anoto technology, the Fly pen system combines a voice interface, a set of printed templates, and the option to draw one's own interfaces, to offer a wide variety of games and educational activities. While the Fly pen interface is geared towards non-professional activity (e.g., its voice feedback is very slow), it shows that powerful feedback mechanisms can be integrated in a pen-sized device. This provides the potential for a rich variety of feedback supporting more and more complex interactions in a paper-only environment. However, the structural parameters, constraints, and contingencies of feedback mechanisms in paper-only environments have not been systematically explored.

In this paper, we propose a set of feedback mechanisms geared towards improving user experience in paper-only interactions. We identify three different levels of feedback: *discovery feedback* to assist users in discovering interface functionalities which may not be visible on paper, *status-indication feedback* to aid users in detecting possibly erroneous input, and *task feedback* to help users perform a given task such as searching a word inside a printed document. Focusing on three main sensory modalities, visual, tactile, and auditory, that can be easily implemented on pen-top computers we propose a set of guidelines for pen-top feedback design to support paper-only interactions.

We implemented the different feedback mechanisms for the PapierCraft command system by augmenting an off-the-shelf Logitech io$_2$ Bluetooth pen. Several LEDs around the pen tip provided visual status-indication feedback regarding the current status of the pen. Two quick response coils provided tactile status-indication feedback to alert users of potential errors and ambiguities. Finally, voice feedback (simulated by a host PC) provided higher-level information for the purposes of discovery feedback and task feedback. For our prototype, the pen simply "streams" the captured strokes to a nearby computer where they are processed to create the appropriate feedback. Because of the low computational complexity, we believe that such a system could be implemented as part of the Fly pen computer.

Using our prototype of a multimodal feedback pen, we conducted a user evaluation of the proposed discovery and status-indication feedback mechanisms in the context of a menu selection task. We compared our prototype to a tablet PC based implementation and a paper-based system using a simple reference card. Our results show that our discovery mechanism offers a good alternative for menu discovery. Further, compared to non-feedback settings, our error detection feedback significantly improves error detection and correction with only a small performance disadvantage.

RELATED WORK
Paper-Computer Integration
Several systems have proposed to augment paper-based interactions by establishing a tight connection with a nearby computer. Some systems such as the DigitalDesk [30], merge paper and computer interactions by projecting a computer display directly on paper. Others, such as A-Book [19], PaperLink [2], and ButterflyNet [32] offer a more relaxed coupling by considering computers as a support tool for paper interaction. For example, users can access the web page linked to a printed document by clicking on a given location on the paper. While very powerful, such systems sacrifice some paper affordances because they require close ties to a nearby computer. Other systems such as XaX [12], PaperPDA [9], Anoto [1], PADD [8] and recently PapierCraft [17] investigated paper-based interactions that achieve a similar level of flexibility as everyday use of paper. However, their approach of providing little feedback beyond the ink laid by the pen does not scale well, as the complexity of pen-based interfaces increases. In the present paper, we explore how to improve paper-based interactions using only the limited types of feedback that can be provided by the pen itself. In that respect, our work builds on the approach introduced by Leap Frog's Fly pen system [15]. However, whereas the Fly pen focuses on games as its application domain, we are investigating office worker's activities such as active reading that have a lower tolerance for errors or delays in processing.

Multimodal Feedback
In addition to the normal visual feedback, haptic and auditory feedbacks are two important information channels which are readily available on pen-top computers.

Haptic feedback
Our considerations focus on haptic feedback but not force feedback, since we do not expect the latter to become available for pen-top computers. Several previous systems have explored how haptic feedback may improve human-computer interaction, including Ambient Touch [24] and Haptic Touch screen [25]. In addition, previous work on a haptified GUI [21, 26] has used tactile channels to provide ambiguity warnings and to improve gesture accuracy. The haptic actuator used in the present system was inspired by the Haptic Pen [16], which provides a longitudinal solenoid at the upper end of a pen and uses solenoid hits to indicate the interaction states for buttons in the GUI.

Auditory feedback
Auditory feedback is especially good for mobile computing, as it is hands-free, can be implemented in a small form factor, and has relatively high bandwidth. Previous systems such as VoiceNotes [29], Nomadic Radio [27], and MOTILE [13] have extensively explored voice interfaces. Our use of speech in some parts of the interface was inspired directly by the voice-based interface of the Fly pen [15].

In comparison to voice feedback, non-speech sounds are particularly good for conveying simple and quick messages, such as confirmation or error indicators. ARKOLA [6] and Mercator [22] adopted everyday sounds as "auditory icons" and metaphors for GUI elements and operations. Other systems used structured abstract sounds to encode information as proposed in "earcons" [4]. Several of these systems, convey the changing properties of source events through parameterized auditory attributes like tone, pitch, register, and stereo positions [4, 7]. We borrowed such ideas for the purposes of pen gesture guidance and interface discovery.

FEEDBACK FOR A PAPER-BASED INTERFACE
Our main focus was to provide an interface which is adapted to the form factors of a pen and may support more real-time feedback for the PapierCraft command system. Of course, for situations where a nearby display is available, a richer set of features could be developed. However, this would limit the flexibility of usage. Using our prototype we hope to illustrate that it is possible to implement a wide range of digital affordances using only a pen-top computer and paper.

	Feedback type		
	Discovery	Status	Task
Visual	Command direction	Mode, Recognition	Page-level location
Tactile	Boundary warning	Ambiguity detection	
Sound	Boundary crossing		
Voice	Command name		Document-level location

Modality is the row label spanning the left side.

Table 1: Modality by feedback type matrix. The gray cells indicate unadvisable combinations. For example, voice feedback does not provide the speed and simplicity required for status-indication feedback. To limit mapping complexity, sounds are only used for discovery feedback at this point.

The PapierCraft command System

Using the PADD system [8] as a supporting infrastructure, PapierCraft, treats printouts as "proxies" of digital documents and allows users to manipulate the digital content printed on paper via a set of commands based on pen gestures (e.g., copying, pasting). The strokes captured on paper are cached in the pen and processed in batch upon synchronization.

PapierCraft uses structured gesture commands, which each consist of three parts: Scopes, delimiter, and command type. The system uses a scope set familiar to the user including underline, margin bar, and lasso. It adopts a "pigtail" gesture [11] as a delimiter between scope selection and menu selection and applies a marking menu system to select a given command. More details can be found in [17].

Feedback types

Our experience developing PapierCraft suggests that the following spectrum of feedback is needed for designing a pen-top interface:

Discovery feedback

As the complexity of systems like PapierCraft increases, it becomes more difficult for users to learn and remember all the available commands. For such systems to be practical, a discovery interface becomes necessary. Low technology solutions (such as a simple printed card) may help during early use of the system. However, as users become more confident, they may not carry such a card around and will need additional support to remember the location of an occasionally used command. For instance, like in the standard marking menu system [14], if a user does not remember a command, she should be able to explore possible commands by navigating around the menu center (in our system, the crossing point of the pigtail).

Status-indication feedback

In this category we include feedback that helps users to ascertain the current status of the system. Examples of feedback in this category include: indication of the current mode of the pen (e.g., stroke color, command vs. writing mode); indication that a given scope (e.g., a crop mark) has been recognized properly; and indication that the current command selection was recognized properly.

Such feedback needs to be easy to identify, yet as unobtrusive as possible to avoid limiting the performance of expert users. For example, it should be salient if a command may not be recognized, but there should be no distracting feedback if the same command was recognized correctly.

Task level feedback

The types of feedback in this category are application-specific and include interactions such as searching a given word in the current document, searching the definition of a word, or gathering additional information about the figures in a paper. It would seem an obvious solution to accomplish these tasks through the use of a nearby display such as in the Paper++ system [23]. However, we believe that it is useful to offer at least some of the simpler tasks in paper only environment to increase flexibility of use. Timing is less critical for such feedback since it corresponds to a task level time frame.
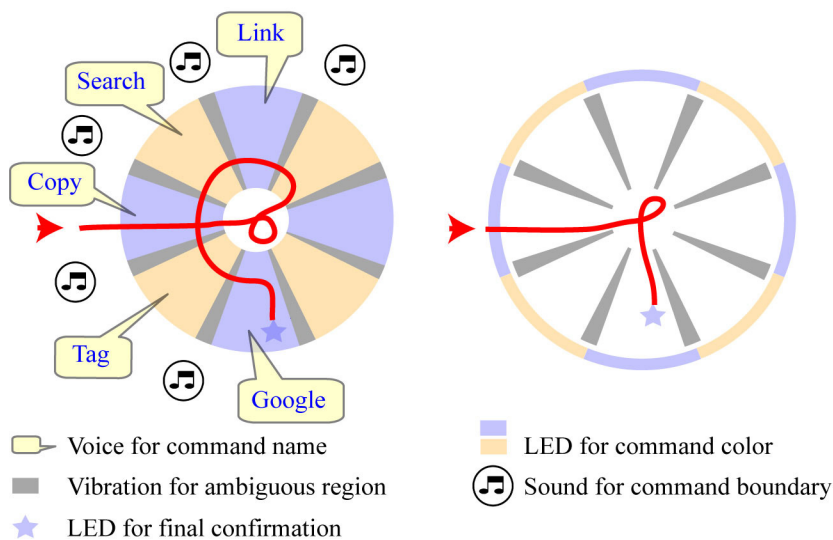
Feedback Mechanisms and Design Guidelines

Building on an examination of available feedback mechanisms, including haptic devices, LEDs, portable displays, speech, and sounds, we conducted several rounds of pilot tests with 2-3 participants in each round (total N = 12). For example, in one round, we studied different approaches for positive feedback using LEDs, sounds, and vibration. These pilot tests helped us to establish the following general design guidelines for pen-top feedback.

Tactile feedback

The current Logitech pens already use a small vibration motor to indicate error conditions. Inspired by this approach, we selected an implementation similar to Haptic Pen [16] to allow for better control of timing as well as a wider range of tactile sensations.

Our first finding with regard to tactile feedback was that it should be used *parsimoniously*. While powerful, such feedback can be a source of annoyance if it is used for any long period of time. This explains why we reserved tactile feedback for status-indication and discovery feedback (Table 1). Our second finding was that providing tactile feedback as positive feedback can actually *slow down* execution. For example, during pilot testing of a marking menu implementation, we found that when tactile feedback provides stronger vibrations as the pen gets closer to the optimal path, subject performance decreases. The same phenomenon holds for auditory or color coding positive feedback.

At first, this result may seem to contradict the work of Zhai et al. on Trackpoint [5]. However, the two tasks represent very different usage conditions. Zhai et al.'s experiment was focusing on a tracking task where positive feedback is very important. In contrast, optimal performance with marking menu is reached in "open loop" interactions where muscle memory plays a major role. In such contexts, add-

Figure 2: (Left) An illustration of a novice user receiving discovery feedback. The gesture stroke (red line) leads from left to right and the user receives feedback through voice, sound, vibration, and LEDs. The center blank region is a "silence" zone with no feedback. (Middle) An expert selects the same command, receiving feedback for final confirmation. (Right) Task level feedback during search. After a voice informs users which page contains the target word, they may scan this page from top to bottom. When lines containing a search hit are close by, the LED on the pen changes to high brightness (indicated by the thickness of the green line). For clarity, the target word "computer(s)" is underlined in this illustration.

ing feedback might be distracting as it may induce users to revert to a tracking strategy instead of an "open loop" interaction. Thus, we used tactile feedback exclusively to indicate erroneous or ambiguous areas. For example, our pen vibrates when the users are too close to the border between two marking menu octants thus creating the potential for ambiguity.

Low fidelity visual feedback

New multi-color LEDs are small, power-efficient, and provide a wide range of possible feedback. LEDs can be placed to shine directly on the paper (to show the currently selected color) or to be visible on top of the pen tip to show the status of the pen even when the pen is not near a piece of paper. For example, our system uses the top LED to provide feedback for command selection, since at the end of command selection, the pen is often already lifted into the air.

Our pilot studies showed that LEDs worked very well for modal feedback (e.g., current stroke color, confirmation of command selection). Not surprisingly, we also found that it is important to avoid blinking light as much as possible since it is very distracting for users. Further, for abstract mapping (e.g., the name of the currently selected command), it is important to limit the number of colors to a minimum. Otherwise, the mapping between color and meaning increases mental load. Another issue regarding color mapping are ambiguities encountered by color-blind users. To address this latter problem, it may be possible to use colors with different luminance

Unlike tactile feedback, visual feedback can be used for longer periods of time. However, like tactile feedback, it may slow down performance, especially if it is used as a visual guide. In some cases, this effect could be used to intentionally slow down users who are about to carry out a potentially problematic operation (e.g., "Delete").

High fidelity visual feedback

Some pen systems such as the WIZCOM translator pen [31], integrate an LCD display into the pen. It may even be possible to create small projection displays by installing a small low power laser raster [18] at the eraser end of the pen. Such displays would offer a high bandwidth channel. However, they will add to the bulk of the pen and might significantly impact power consumption. Therefore, we did not consider these options in our current design.

Auditory feedback

As shown by the Fly pen system [15], voice (and of course sound) can be used for pen-top interfaces. Like haptic feedback, auditory feedback has access to pre-attentive channels that can easily attract users' attention.

Because the presentation of information through spoken language is essentially serial, it may take up a lot of time. Therefore, voice feedback should be reserved for feedback which operates under more extended time frames, such as discovery and task level feedback. Nevertheless, short sounds can be used for status-indication feedback like status changes or action confirmations. The latter approach has been extensively implemented in many existing systems such as ARKOLA [6] and Mercator [22].

One potential problem is that auditory feedback may raise privacy issues. However, these could be easily alleviated by using headphones or a Bluetooth earpiece.

We present the cross-matrix of feedback mechanisms and feedback types in Table 1. The next section discusses the rationale behind our design approach.

DESIGN OF PEN-TOP FEEDBACK

We now detail the rationale behind specific design choices for our system. Figure 2 illustrates our pen-top feedback design for each of the types of feedback we explored.

Interface Discovery

Interface discovery feedback is intended to help novice users to discover the interface and intermediary users to remember commands that are not used very often. In pilot testing, we examined different mechanisms to convey command names including color coding, non-speech auditory coding, and vibration pattern coding. We found that voice feedback is the most natural way to provide information about command names. None of the other approaches worked well for a marking menu with eight or more items.

Figure 2 (left) illustrates our discovery feedback mechanism. Upon drawing the pigtail, the discovery mechanism is triggered after a timeout of 500ms to avoid distracting expert users. This delay is slightly longer than the typical 330 ms delay [11] because we discovered that users take slightly longer to mark a menu on paper. After the delay, the system speaks the name of the command triggered by the current octant. The top LED also indicates the current color of the octant. We first experimented with assigning one color for each possible direction, but discovered that this was quite confusing. Noting that what matters for users is to distinguish between two adjacent octants, we reduced our color coding to only two colors: blue for the primary direction (e.g., North vs. South, East vs. West) and yellow for the secondary direction (e.g., North-East vs. South-West). These two colors form a strong contrast (even for users with the common red-green color blindness) which makes them easy to distinguish.

As users start exploring available commands by circling around the center of the menu, tactile feedback is used to indicate menu boundaries and, more specifically, the areas for which command recognition might be ambiguous. To allow for quick exploration, the system speaks the name of a new octant as soon as the pen enters it (even if the previous name is not yet completed). This feature initially created some confusion since during rapid movements, command names were sometimes skipped altogether. For example, users would feel three vibrations but hear only two different spoken octant names. To better convey navigation status, a short "popping water bubble" sound was added whenever a boundary was crossed.

The desired command is selected upon pen lift. Simultaneously, the top LED shows a single pulse in the color of the selected octant and slowly decays. Note that to abort the selection, users can return the pen tip to the center of the menu (i.e., the crossing point of the pigtail) and lift the pen.

Status-indication feedback

Our system uses two LEDs to indicate the current pen status. The bottom LED indicates the current stroke color and the top LED indicates if the pen is inking or used in command mode.

Command selection feedback for expert users

Our design for experts focuses on alerting users about possibly ambiguous situations. To encourage skill transfer, the expert interface uses the same color coding and tactile feedback as the discovery interface, but does not use voice or color feedback during marking. If a mark is far away from any menu boundaries, we do not provide any feedback until the mark is finished. Then we provide a single, slowly decaying pulse of the color corresponding to the selected direction. If the mark enters one of the ambiguous areas between two menu octants, we use tactile feedback to warn users, and show the color of the recognized direction upon pen lift. This makes it easy for users to check if the system recognized the mark correctly.

When called up through a pigtail gesture [11], marking menu is often associated with an initially curly stroke that could trigger a transient vibration. To reinforce the mapping between the lack of tactile feedback and a successfully issued command, we added a "*silence*" zone in the center of the menu where no feedback is provided.

Task feedback

To explore an example of more complex feedback, we allowed users to search a given word in the printout of a document. For this feature, the pen needs to be uploaded with search information about the current text (e.g., during printing, through the PapierCraft infrastructure [17]). In the future, users will be able to provide their own written keywords, but in our present prototype, users select the keyword from printed text.

For this task, we identified two kinds of necessary feedback. First, users need to find the pages which contain the target word. Since it would be tedious to scan the whole document, we used voice feedback to provide the list of relevant pages. When a target page is reached, the user needs to identify the exact location of the word. For that sub-task, voice feedback would be time consuming and difficult to process (e.g., "left column, 3rd paragraph line 35"). Instead, we used visual feedback. Users simply swipe the pen along a page margin and observe the top LED on the pen which lights up when the pen passes a line that contains the word. If there is no word selected for the "search" command, the previous keyword, if any, will be used. Figure 2 (right) illustrates the process.

We considered using tactile feedback for the within-page location, but we decided against it because of the following concerns: first, narrow tactile "bumps" are difficult to reliably identify, especially if the user moves rapidly; second, wider bumps with a ramp are annoying for users since they create vibrations of long duration. Instead, we used tactile feedback to indicate errors (e.g., when searching on a page which does not contain the target word).

IMPLEMENTATION

The PapierCraft infrastructure and feedback creation were written in C++. A document management server was running on Linux, and all other parts were running on the MS Windows platform. We also used Acrobat SDK, Microsoft
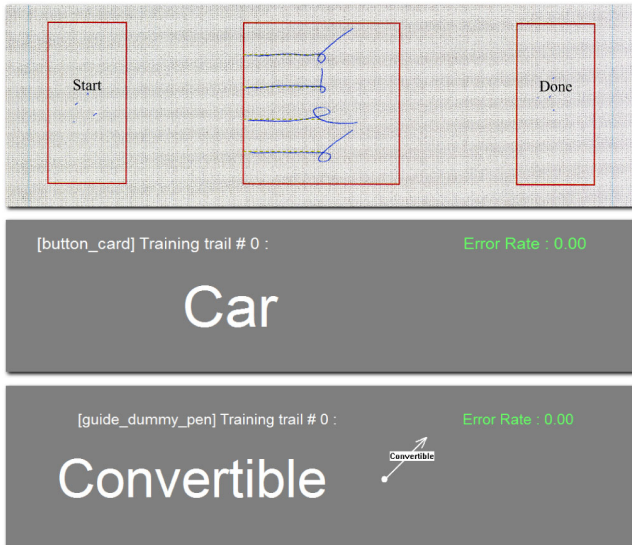
Figure 3: (Top) the experimental interface on paper. (Middle) A sample stimulus for the discovery setting with current average error rate at the top right corner (for the latest 20 trials). (Bottom) A sample stimulus for the expert setting. A labeled arrow in the middle illustrates the correct direction of the marking menu item (e.g., "convertible") for expert simulation.

Tablet PC Recognizer Pack, MS Text-To-Speech API, and Anoto SDK3.0 for the manipulation of paper documents. A PIC18F252 Microcontroller was used to control the LED and the valves from Lee Company in response to commands sent by a host PC.

EVALUATION

We conducted a controlled experiment to evaluate two important characteristics of our design. First, we examined if the discovery interface would help users rapidly improve their performance for an interface they are not familiar with. Second, we explored if our system would help expert users to reduce errors without reducing their performance too much. Due to time constraints, we did not evaluate task-level feedback (e.g., the search mechanism described above).

We compared three different techniques representing the range of currently available pen interfaces:

- **A simple PapierCraft** interface in which all interactions with the marking menu are performed on paper and the only immediate feedback is the ink itself. This technique represents the lower end of the spectrum of feedback fidelity. In accord with existing pen systems we also provided a quick reference card printed with the currently used marking menu commands. While users may not always carry such a card in real-world settings, we felt that not providing a reference card would be very frustrating for users and not tell us much about the value of our system.

- **A multimodal pen-top feedback system**, based on the implementation described above. In this condition, users were only allowed to see the quick reference

card during training. During experimental trials, they had to rely exclusively on our system to discover the menu or detect their errors. In this configuration, all interactions were performed on paper.

- **A tablet PC** with an implementation of the same marking menu task. This configuration represents the higher end of the feedback fidelity spectrum. The pigtail-enabled marking menu was implemented in a manner similar to Scriboli [11] with a 500 ms delay before the pie menu layout appeared. Upon selection, the selected octant was highlighted and appeared for 75 ms (without showing any other octants) for confirmation purposes.

Experimental task

To compare the three techniques, we picked a typical task in PapierCraft: use the "underline" scope selection to select a word, and then issue a given command. This task is easily tractable and forces users to always start their pigtail in the same direction. This is important since the performance of pigtail-based marking menus is affected by the starting direction.

To avoid the variance introduced by mode-switching actions and to focus more on feedback impact, we set the pen to gesture mode for all three techniques. Thus, participants did not need to press the gesture button. Accordingly, the top LED of the multimodal pen was not used for mode indication but for other types of feedback.

Figure 3 shows an example of the experimental interface which was either printed on paper or presented on the tablet PC screen. First, the users had to tick on the "start" button, then underline a dotted line in the central panel, then select the menu command presented on a screen in front of them (one out of eight possible choices), and finally tick the "done" button when they thought that their command selection was correct. Thus, users needed to first commit to their current selection before receiving feedback about its accuracy. Users could make as many corrections as they wished before clicking on the "done" button. After clicking this button, the computer played a sound reflecting success or failure. We provided error feedback to ensure that participants' error rate would stay below 12%. We chose a comparatively high threshold because one goal of the experiment was to examine how error rates differed across techniques. Thus, it was important not to impose too strict limitations on error rate. At the same time we needed to ensure that subjects did not rush through the experiment at the expense of data quality. We believe that the 12% level offers a good compromise between these two objectives

To evaluate both novice and expert performance, the task was run in two different configurations. In the discovery configuration, the stimulus screen provided only the name of the correct menu selection. Users could use any feedback available in the current setting to select the appropriate menu item. This configuration was intended to evaluate how interface discovery feedback could help users explore a new menu system. The discovery condition was always run first. In the expert setting, the stimulus screen provided

both the name and the direction of the correct selection as suggested in [3]. The goal for this configuration was to simulate expert performance and evaluate if status-indication feedback would help expert users to catch errors early.

Performance Measures

Task Completion Time and *Accuracy* were the two main dependent variables. Completion time is the interval between ticking the "start" and the "done" buttons. Accuracy refers to the rate of correctly submitted command selections. To examine error correction behavior, we also examined the *Correction Rate*, the proportion of trials in which an initially wrong menu selection was later corrected.

Procedure

We used a within-subject design for this experiment. Each participant completed the task using all three techniques in a fully counterbalanced order. For each technique, participants first completed the discovery setting and then the expert setting. For the discovery configuration, users performed 4 training blocks followed by a 12-block measurement phase. For the expert configuration, there were 2 training blocks and 10 measurement blocks (fewer blocks were used for this setting, since the task was simpler and showed less variance). There was a mandatory 4-second break after every 2 blocks. Each block corresponded to 8 selections (one for each octant of the menu) presented at random. To avoid learning effects across techniques, we used a different set of command names for each technique (e.g., names of vehicle types or fruit names). The association between sets of command names and techniques was counter-balanced as well. After each technique, participants completed a questionnaire about the technique. A general questionnaire was completed at the end of the experiment.

Experimental Setup

The experimental stimuli and input interface are illustrated in Figure 3. Visual stimuli were presented on a flat LCD display in front of the subjects. A pile of printed documents (with Anoto pattern) or a tablet PC was put in front of the display without fixed orientation or position, simulating the real-life scenario of reading and writing. Participants used a stylus on the tablet PC or our augmented digital pen on Anoto paper. The same pen was used for the simple pen and the multimodal pen settings but the feedback features were only enabled for the latter.

The experiment was controlled and data were logged by the same tablet PC for all techniques. Depending on the technique, the PC received stroke information from its own stylus or from the augmented pen (via Bluetooth). When feedback was enabled, it controlled the LED and the solenoid through a ribbon cable. Auditory feedback was rendered through a stereo headphone.

Participants

We recruited 12 participants, 3 female and 9 male. All of them were college students or had at least a college degree, and ranged in age from 18 to 35. All had used a desktop
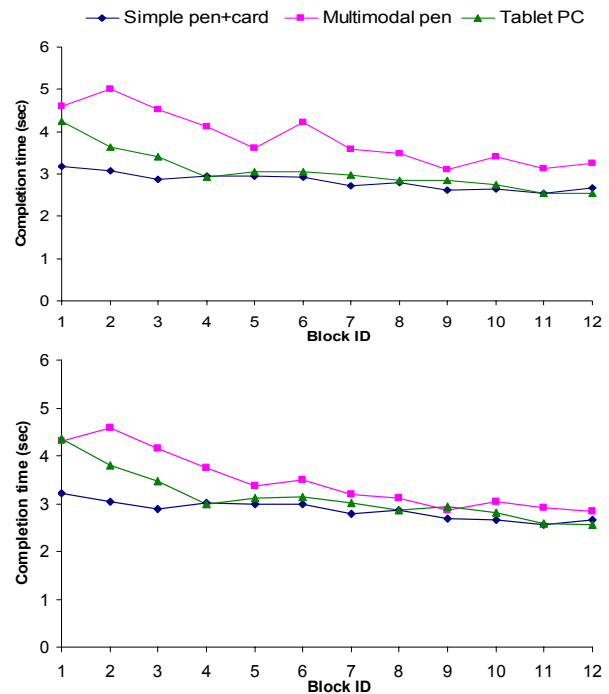

Figure 4: Mean completion time for the discovery setting. The top graph includes all participants while. The bottom graph excludes participants who did not follow instructions (see text).

computer for 7-18 years and 3-12 hours per day. Three participants had 1.5-5 years of experience with a PDA, the others had not used pen computing at all. None of them had ever used a digital pen. It took participants about 90 minutes to finish the experiment, including the training phases, experimental phases, and questionnaires. They received $20 for their participation.

RESULTS

In the results presented below, Greenhouse-Geisser correction was used to account for deviations from sphericity. All post-hoc tests used Bonferroni corrections.

Discovery setting

To control for outliers, we removed trials with completion times more than 3 SD above the mean within a given block and technique. This amounted to 1.59% of the data points.

Task completion time

We first studied how completion times for each of the techniques developed across subsequent blocks. A technique by block repeated measure ANOVA found a main effect of technique, $F(1.37, 15.1) = 5.35$, $p < .05$, partial $\eta^2 = 0.33$, suggesting that completion times (in milliseconds) in the multimodal pen condition (M = 3840, SD = 2330) were marginally higher than in simple pen condition (M = 2830, SD = 771, $p = .06$), but not significantly different from the tablet PC condition (M = 3070, SD = 1280, $p = .20$). We also found a main effect of block, $F(3.48, 38.2) = 20.0$, $p < .01$, partial $\eta^2 = 0.65$, suggesting that completion time decreased across blocks. However, these effects were qualified by a significant interaction suggesting that the gap
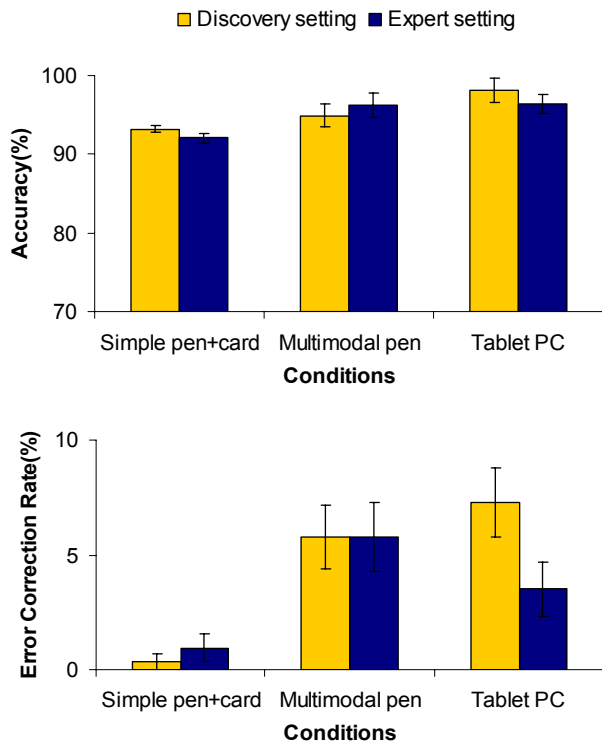
Figure 5: Mean accuracy (top) and correction rate (bottom), for both discovery and expert setting (error bars show 95% confidence interval).

between techniques diminished across blocks, $F_{(22, 242)} = 3.58$, $p < .01$, partial $\eta^2 = 0.25$, as shown in Figure 4 (top).

Because the variance in the multimodal pen condition was higher (SD = 2330 ms) than in the other two conditions (SD = 771 ms and 1280 ms, for simple pen and tablet PC respectively) and because the learning curve for the multimodal condition showed discontinuities (e.g., in Block 6, see Figure 4, top) we examined the log data and experiment records to identify any participants who might have had problems following instructions. We found that Participant 6 used the voice feedback instead of the LED for command type confirmation which created an artificial delay. Participant 8 focused on trying to recall the correct items instead of using the discovery interface, as indicated by an unusually long thinking time before starting to draw. To better understand the learning curves without this bias, we repeated our analyses excluding these two participants (Figure 4, bottom). The pattern of results was comparable showing a main effect of block, $F_{(11, 99)} = 16.5$, $p < .01$, partial $\eta^2 = 0.65$ and a strong technique by block interaction, $F_{(22, 198)} = 3.39$, $p < .01$, partial $\eta^2 = .27$. The effect of technique was only marginally significant, $F_{(2, 18)} = 2.61$, $p = .10$, partial $\eta^2 = 0.23$. However, note that the removal of two participants might invalidate the original counter-balancing among the techniques. Thus, a larger study will be needed to confirm these results.

Together, these results indicate that although participants' initially experienced a higher task completion time in the multimodal pen condition, they were able to master the interface after 12 blocks. The initially higher task comple-

tion time is not surprising since users needed to perform a sequential search to familiarize themselves with the menu items while in the other conditions, a simple glance at the reference card (in the simple pen condition) or the screen feedback (in the tablet PC condition) was sufficient. Also, as illustrated by the poor performance of participants 6 and 8, successful use of the multimodal system requires that users are thoroughly trained in the optimal strategies for this technique.

Accuracy and Error correction behavior
For accuracy, a block by technique ANOVA found a main effect of technique, $F_{(2, 22)} = 9.17$, $p < .01$, partial $\eta^2 = 0.45$, suggesting that tablet PC (M = .98, SD = .14) showed significantly higher accuracy than simple pen (M = .93, SD = .25 , $p < .01$) and multimodal pen (M = .95, SD = .22 , $p < .01$, see Figure 5, top, yellow bars). There was no main effect of block and no technique by block interaction.

Since it was one of our goals to help users detect their mistakes early, we also examined the correction rates. A block by technique ANOVA found a main effect of technique, $F_{(1.29, 14.2)} = 18.0$, $p < .01$, partial $\eta^2 = 0.62$, suggesting that the simple pen (M = .004, SD = .06) had significantly lower correction rates than the multimodal pen (M = .06, SD = .24, $p < .01$) and the tablet PC (M = .07, SD = .26, $p < .01$, see Figure 5, bottom, yellow bars). There also was a main effect of block, $F_{(11, 121)} = 2.29$, $p < .01$, partial $\eta^2 = 0.17$. Further examination revealed that correction rates increased across blocks, most likely because users became more familiar with the error correction mechanisms for each technique. There was no block by technique interaction.

These results suggest that while accuracy rates are comparable in the two paper-based interfaces, the multimodal pen is better than the simple pen and as good as the tablet PC in encouraging users to correct their mistakes.

Expert setting
As in the discovery setting, we removed trials with completion times more than 3 SD above the mean within a given block and technique. This amounted to 2.05% of the data points.

Task completion time
A block by technique ANOVA found a main effect of technique, $F_{(2, 22)} = 9.57$, $p < .01$, partial $\eta^2 = 0.47$, suggesting that the tablet PC (M = 1920, SD = 454) was significantly faster than the multimodal pen condition (M = 2270 , SD = 620, $p < .01$), while the difference between multimodal and simple pen (M = 2060, SD = 473) was not significant ($p = .17$). There also was a main effect of block, $F_{(9, 99)} = 3.55$, $p < .01$, partial $\eta^2 = 0.24$ reflecting a learning effect across blocks, but no block by technique interaction.

Accuracy and Error correction behavior
A block by technique ANOVA with accuracy rates as the dependent variable found a main effect of technique, $F_{(2, 22)} = 16.8$, $p < .01$, partial $\eta^2 = 0.61$, suggesting that both multimodal pen (M = .96, SD = .19) and tablet PC (M =

.96, SD = .19) had higher accuracy rates than the simple pen condition (M = .92, SD = .27 , p < .01 for both, see Figure 5, top, blue bars). There was no main effect of block and no significant block by technique interaction. This result reflects the effectiveness of the multimodal feedback for expert users.

For correction rates, a block by technique ANOVA found a significant main effect of technique, $F(2, 22) = 10.5$, $p < .01$, partial $\eta^2 = 0.49$, suggesting that the simple pen (M = .01, SD = .10) had significantly lower correction rates than the multimodal pen (M = .06, SD = .23, p < .01) and the tablet PC (M = .04, SD = .18, p < .05, see Figure 5, bottom, blue bars). There was no main effect of block and no block by technique interaction. This finding suggests that the stronger feedback in the multimodal pen as compared to the simple pen condition encourages error corrections.

Subjective evaluation
We included the NASA TLX using a 1-7 scale to evaluate user experience in each session. The results confirmed that the expert phase of the experiment had a lower task load. The questionnaire also reflected objective performance characteristics. A 3 (technique) by 2 (Discovery vs. Expert setting) ANOVA on TLX "Performance" scores, found a main effect of technique, $F(2, 22) = 4.28$, $p < .05$, partial $\eta^2 = 0.28$. Post-hoc tests showed that compared to the tablet PC condition (M = 5.71, SD = 1.23), user-perceived performance was significantly lower in the multimodal pen condition (M = 4.79, SD = 1.74, p < .05) and marginally lower in the simple pen condition (M = 4.79, SD = 1.41, p = .06). There was no difference between the discovery and expert setting and no setting by technique interaction.

Our questionnaire also asked participants to rate the usefulness of individual characteristics of the interface on a scale from 1-7. Users rated the LED color coding in the discovery setting (M = 6.17, SD = 1.59) and the opportunity to correct errors in general (M = 6.08, SD = 1.00) as the most important features of the multimodal pen interface. This suggests that our design was successful not only on the objective level (see above) but also on the subjective level. Another interesting finding was that the vibration feedback in the expert setting did not receive good scores (M = 2.67, SD = 1.56). A possible cause is that during rapid movement, the vibration may have been too short to be perceived reliably. Increasing the minimum vibration time may address this problem. The sound-based indicator for boundary crossing during discovery did not fare much better (M = 3.58, SD = 1.93). Future interfaces should fine-tune these features.

DISCUSSION AND FUTURE WORK
We believe that the present paper contributes to the development of advanced pen-top computing interfaces. Emerging pen-top computers, such as Fly pen [15], make it feasible to integrate all the software and hardware required for our prototype into a pen-shaped device. To deploy the system more widely we are planning to implement our design on such a pen-top computers.

While the present work focused on paper-only computing, pens are frequently used near other electronic devices (e.g., cell-phones, PDAs, and computers) which provide a bridge to the network infrastructure as well as a display. Access to external data resources such as those of the PADD database may enhance the level of task feedback available to users. For example, when working on a document printed elsewhere, the pen may fetch the digital version of the document to allow user to search its content. It may also offer different levels of service such as retrieving an audio version of the document.

As pointed out before, access to a nearby display will increase the potential of our system as well. For example, one's cell phone screen could present the definition of a highlighted word in a setting similar to the Paper++ system [23]. In our opinion, it will be a key element of future systems to allow users to flexibly create compound displays as they see fit, leveraging the ideas introduced in the Stitching system [10].

Our preliminary user study confirmed that our design provides support for interface discovery and helps users to detect (and correct) their errors early. To confirm the external validity of our results, we would like to test our system in a more complex setting. For example, we would like to explore how the PapierCraft system, augmented with our feedback mechanism, would compare to purely digital systems like XLibris [28] or OneNote [20].

CONCLUSION
In this paper, we discuss the design space for a pen-top interface supporting interactions limited to pen and paper as proposed in interactive paper systems. We identified the feedback information required for a pen-gesture based command system on paper like PapierCraft, and examined potential multimodal feedback rendering mechanisms, including LEDs for status feedback, vibrators for ambiguity feedback, as well as speech for discovery and task feedback. A controlled experiment showed that our pen-top interface can effectively help novice users to discover and learn a new interface in a paper-based setting. Expert users also benefit from better error correction mechanisms and stronger confidence when using a paper-only interface.

REFERENCES

1. Anoto, Development Guide for Service Enabled by Anoto Functionality. 2002, Anoto.

2. Arai, T., D. Aust, and S.E. Hudson. PaperLink: a technique for hyperlinking from real paper to electronic content. *Proceedings of CHI'97*, pp. 327 - 334.

3. Balakrishnan, R. and P. Patel. The PadMouse: facilitating selection and spatial positioning for the non-dominant hand. *Proceedings of CHI'98*, pp. 9 - 16.

4. Brewster, S.A., Using nonspeech sounds to provide navigation cues. ACM Transactions on Computer-Human Interaction, 1998. **5**(3): p. 224-259.

5. Campbell, C.S., S. Zhai, K.W. May, and P.P. Maglio. What You Feel Must Be What You See: Adding Tactile Feedback to the Trackpoint. *Proceedings of INTERACT'99*, pp. 383-390.

6. Gaver, W.W., R.B. Smith, and T. O'Shea. Effective sounds in complex systems: the ARKOLA simulation. *Proceedings of CHI'91*, pp. 85-90.

7. Gaver, W.W. Synthesizing auditory icons. *Proceedings of CHI'93*, pp. 228-235.

8. Guimbretiere, F. Paper Augmented Digital Documents. *Proceedings of UIST'03*, pp. 51 - 60.

9. Heiner, J.M., S.E. Hudson, and K. Tanaka. Linking and messaging from real paper in the Paper PDA. *Proceedings of UIST'99*, pp. 179 - 186.

10. Hinckley, K., G. Ramos, F. Guimbretiere, P. Baudisch, and M. Smith. Stitching: Pen Gestures that Span Multiple Displays. *Proceedings of AVI'04*, pp. 23 - 31.

11. Hinckley, K., P. Baudisch, G. Ramos, and F. Guimbretiere. Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. *Proceedings of CHI'05*, pp. 451-460.

12. Johnson, W., H. Jellinek, J. Leigh Klotz, R. Rao, and S.K. Card. Bridging the paper and electronic worlds: the paper user interface. *Proceedings of CHI'93*, pp. 507 - 512.

13. Kristoffersen, S. and F. Ljungberg. \"Making place\" to make IT work: empirical explorations of HCI for mobile CSCW. *Proceedings of ACM Group'99*, pp. 276-285.

14. Kurtenbach, G., The design and Evaluation of Marking Menus, PhD thesis, University of Toronto. 1993

15. LeapFrog, Fly Pen, http://www.leapfrog.com. 2005.

16. Lee, C.J., H.P. Dietz, D. Leigh, S.W. Yerazunis, and E.S. Hudson. Haptic pen: a tactile feedback stylus for touch screens. *Proceedings of UIST'04*, pp. 291-294.

17. Liao, C., F. Guimbretière, and K. Hinckley. Papier-Craft: a command system for interactive paper. *Proceedings of UIST05*, pp. 241-244.

18. LightBlueOptics, Portable laser projector, http://www.lightblueoptics.com/. 2005.

19. Mackay, W.E., G. Pothier, C. Letondal, K. Bøegh, and H.E. Sørensen. The missing link: augmenting biology laboratory notebooks. *Proceedings of UIST'02*, pp. 41 - 50.

20. Microsoft, http://www.microsoft.com. 2003.

21. Miller, T. and R. Zeleznik. An insidious Haptic invasion: adding force feedback to the X desktop. *Proceedings of UIST 98*, pp. 59-64.

22. Mynatt, E.D. and G. Weber. Nonvisual presentation of graphical user interfaces: contrasting two approaches. *Proceedings of CHI'94*, pp. 166-172.

23. Norrie, M.C. and B. Signer. Switching Over to Paper: A New Web Channel. *Proceedings of Web Information Systems Engineering'03*, pp. 209-218.

24. Poupyrev, I., S. Maruyama, and J. Rekimoto. Ambient touch: designing tactile interfaces for handheld devices. *Proceedings of UIST'02*, pp. 51 - 60.

25. Poupyrev, I., M. Okabe, and S. Maruyama. Haptic feedback for pen computing: directions and strategies. *Proceedings of CHI'04 (extended abstracts)*, pp. 1309 - 1312.

26. Ramstein, C., O. Martial, A. Dufresne, M. Carignan, P. Chass, and P. Mabilleau. Touching and hearing GUI's: design issues for the PC-Access system. *Proceedings of ASSETS'96*, pp. 2-9.

27. Sawhney, N. and C. Schmandt, Nomadic radio: speech and audio interaction for contextual messaging in nomadic environments. ACM Transactions on Computer-Human Interaction, 2000. **7**(3): p. 353-383.

28. Schilit, B.N., G. Golovchinsky, and M.N. Price. Beyond paper: supporting active reading with free form digital ink annotations. *Proceedings of CHI'98*, pp. 249 - 256.

29. Stifelman, L.J. A tool to support speech and non-speech audio feedback generation in audio interfaces. *Proceedings of UIST'95*, pp. 171-179.

30. Wellner, P., Interacting with paper on the DigitalDesk. Communications of the ACM, 1993. **36**(7): p. 87 - 96.

31. WIZCOM, http://www.wizcomtech.com/. 1995.

32. Yeh, R.B., C. Liao, S.R. Klemmer, F. Guimbretière, B. Lee, B. Kakaradov, J. Stamberger, and A. Paepcke. ButterflyNet: A Mobile Capture and Access System for Field Biology Research. *Proceedings of CHI'06*, pp. 571-580.