
Learning Shape Writing by Game Playing

Per Ola Kristensson

Department of Computer and
Information Science
Linköpings universitet, Sweden
perkr@ida.liu.se

Shumin Zhai

IBM Almaden Research Center
San Jose, California, USA
zhai@almaden.ibm.com

Abstract

We present a computer game designed to efficiently and playfully teach users shape writing – a new text entry method for pen-based devices.

Keywords

Text entry, mobile devices, shorthand, gesture, computer games

ACM Classification Keywords

H5.2. Information interfaces and presentation: User interfaces – input devices and strategies.

Introduction

Text entry such as instant messaging, emails, and short text-messages, is one of the most frequent tasks on mobile devices. To address the small form-factor requirement on these devices researchers have invented, modeled and evaluated a plethora of text entry methods (see MacKenzie and Soukoreff [6] for an extensive review). A challenge when developing an effective text entry method is that the method needs to simultaneously, 1) be fast and accurate, 2) have a small form-factor, 3) accommodate both short-term and long-term performance, 4) be easy to learn, and 5) fun to use.

Copyright is held by the author/owner(s).
CHI 2007, April 28–May 3, 2007, San Jose, California, USA.
ACM 978-1-59593-642-4/07/0004.

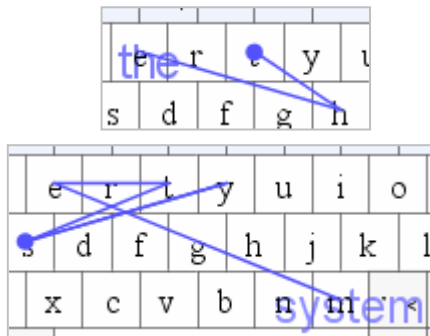


figure 1. Ideal word shapes for the words *the* and *system*. Starting point indicated with a dot.

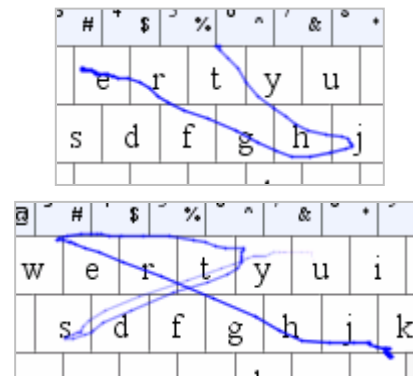


figure 2. Actual pen-gestures for the words *the* and *system*. The pen-trace for *system* is slightly faded out in the beginning since ShapeWriter attempts to minimize visual clutter by gradually fading away the tail of long pen-strokes.

In an attempt to satisfy the requirements of all these five dimensions we have created a new promising text entry method called *shape writing* [2,10] (previously known as SHARK shorthand). In this paper we focus on designing a computer game to enable users learn shape writing in an efficient and pleasurable fashion. In the next section we briefly describe how and why shape writing works. We refer the reader to Zhai and Kristensson [10] and Kristensson and Zhai [2] for the in-length descriptions on how the technology works, and the motivating principles that guided its design.

Shape writing

In ShapeWriter words are represented as ideal pen-traces that consecutively intersects all letter keys in a word. For example, figure 1 shows the ideal traces for the words *the* and *system* on a QWERTY layout.

To shape write a word the user draws a pen-gesture that resembles the shape of the ideal pen trace for the intended word. Figure 2 shows a user's pen-gestures for the words *the* and *system*.

To "shape write" a word a *ShapeWriter* is used. The ShapeWriter displays a keyboard layout, records pen-motion, performs pattern recognition and outputs the most likely word to the user. The use of a pattern recognizer makes it possible to write "sloppy". As long as the pen-gesture matches more closely the ideal word shape of the intended word than those of other words the intended word will be identified correctly. Words with double-letters are written by treating them as one, for instance, to write the word *maroon* the user writes *maron*. To disambiguate words such as *to* and *too* the user opens an alternative words list, see the next subsection.

The ShapeWriter software we have developed uses a multi-channel pattern recognition system, described in Kristensson and Zhai [2]. The first version could search 20,000 patterns in less than 40 ms on a standard 1GHz tablet PC. Since publishing the original recognition paper [2] we have been optimizing the algorithms used, and as a result the latest version can search 57,000 words in less than 20 ms on a standard 1GHz tablet PC.

Editing and error correction

As in all text input methods, the user can write too fast and "sloppy" for the system to be able to correctly identify the intended word. In ShapeWriter words are displayed in an *edit buffer* area on top of the keyboard (figure 3). The user can reveal a list of alternative word candidates for the word displayed by selecting the word with the pen. The caret (shown at the very right edge of the edit buffer in figure 3) can be moved by crossing words from above. Words can be deleted from the edit buffer by crossing them from any direction except from directly above (since that action moves the caret). By crossing the left edge the user can also switch between the edit buffer and un-buffered direct output (where recognized text is immediately sent to the active application on the desktop).

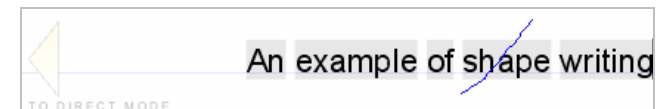


figure 3. Words appear from right to left inside the edit buffer as they are written. When words leave the left edit buffer edge they are sent to the active application.

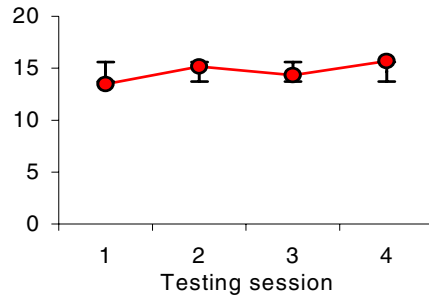


figure 4. The number of words learned during each training session. Adapted from Zhai and Kristensson [10].

Advantages with shape writing

Efficiency

ShapeWriter encodes words in a compact manner [3]. In unistroke pen-gesture alphabets such as *Graffiti*, *Jot* or the stability-enhanced *EdgeWrite* [8], a pen-gesture results in a single letter. In contrast, with shape writing a single pen-gesture results in a complete word. Since shape writing uses error-tolerant pattern recognition, expert users do not need to look at the keys much to produce recognizable pen-gestures. Our previous research has shown that users can learn to write about 15 new word shapes without looking at the graphical keyboard per 45 minute training session (figure 4). In Kristensson and Zhai [2] “expert estimates” show that the common pangram “the quick brown fox jumps over the lazy dog” can be written in 69-70 words per minute (wpm) using ShapeWriter.

Transition from novice to expert

Similar to marking menus [4] ShapeWriter features a seamless transition from novice to expert behavior. A novice user traces out the letter keys of a word on the keyboard using visual guidance. Over time the action of gesturing the word shape builds up in the user’s motor memory. For an expert user, the pen-gesture actions for frequently used words are completely memorized and can be quickly articulated by memory recall instead of tracing by visual guidance. Between the novice and expert modes the learning experience is modeless. Since the graphical keyboard is always displayed as a visual mnemonic map, users can use both motor and visual components to gesture words.

Fun

An important and perhaps underestimated property of successful text entry is how engaging the method is

perceived by the users. In a study presented in Zhai and Kristensson [10] users were positive to the method and considered it fun. The “fun” factor in a text entry method leads us to the task of designing a playful and engaging practice game for ShapeWriter.

Balloon practice game

A challenge for new text entry products is the need for an easy and fun tutorial or game on how to effectively write text using the new method. A multitude of typing tutors have been developed for the desktop keyboard. One of the highly polished typing tutors is *Typing of the Dead* by Sega Entertainment where the user kills zombies by typing text shown in front of them. A variant of that game has recently been researched in the context of learning how to write Japanese characters [7].

To teach shape writing we opted for a more neutral environment suitable to all audiences. We decided to create a balloon practice game (figure 6). The basic game concept is a variant of the traditional typing tutor game in which the user must respond to letters, or words, on the screen as quickly as possible. In our version balloons are floating upwards on the screen. In each balloon a word is displayed. To pop the balloon the user writes the word inside the balloon. To get better score and accuracy the user must pop as many balloons as possible while avoiding writing incorrect words.

Game design

The design goals of the game were threefold:

1. *Efficiency*. It should teach users to shape write the most common words in a short amount of practice time
2. *Fun*. The game should be engaging, or at least not repetitive and boring.
3. *Challenging*. The game should challenge users to write faster.

EFFICIENCY

Not all words occur at equal frequency. In fact, word distribution is heavily skewed and can be modeled with Zipf's law: $P(r) \sim 1 / r^a$, where $P(r)$ is the probability of occurrence of the word, r is the rank of the word, and a is close to unity. An example of Zipf's law is the distribution of English words. In the British National Corpus¹ (BNC) 46% of the vocabulary is composed of the 100 most frequently used English words. Therefore even practicing the top 100 words will greatly benefit the user. This is also the reasoning behind the practice game displaying words in the order of their frequency of occurrence (in a large corpus). For instance, the first words displayed in the practice game for U.S. English are: *the, of, and*, etc.

Second, not all words are equally difficult, and users think it is frustrating to be forced to re-write words they are already good at [9]. On the other hand, until a word shape is completely learned by the user, repetition is necessary in order to push the user to expert mode. For this reason, words are scheduled to appear according to an expanding rehearsal interval

(ERI) algorithm [5]. Briefly, ERI schedules a word to appear within t seconds. If the user writes the word correctly, t is doubled. If the user fails to write the word, t is left unchanged. If there is no word to currently schedule, a new word is picked from the set of unpracticed words. ERI has previously been used successfully in stylus keyboard research by Zhai, Alison and Accot [9], and Zhai and Kristensson [10]. To our knowledge, the use of ERI in a typing tutor is a novel concept.

Third, we decided to introduce yet another concept from psychology in order to make the game more effective. By tapping on the *Auto* button the game demonstrates for the user how to write the ideal word shape for the currently displayed word. Psychomotor research has shown the benefit of "observational practice" [1]. The demonstration mode can also eliminate unnecessary habits, such as trying to arc around (as opposed to directly cross) letter keys. These unnecessary habits only make recognition more difficult. They have been observed among some users in informal user tests.

FUN

To make the game feel less predictable the balloons do not float up to the sky using a direct vertical route. Instead we create a unique movement path for each balloon by generating a spline whose two control points vary in the x and y directions within pre-set intervals (figure 5). The acceleration of a balloon along the path also has a random component that enables a balloon to temporarily "catch wind" and float a little faster along its movement path.

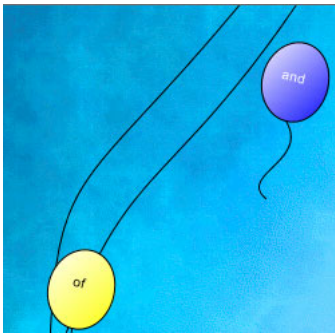


figure 5. The solid lines show the splines that define the movement paths for the two balloons. A balloon's bounding box's upper corner is always positioned on the spline. (These splines are only revealed when the software is in debug mode.)

¹ <http://www.natcorp.ox.ac.uk>

CHALLENGE

To motivate users to write fast, the balloons do not “stick” at the top of the screen. Instead balloons disappear and are counted as a “miss”, when they float above the view port. Therefore the user must quickly write the prompted words inside the balloons before they disappear. The number of balloons displayed simultaneously on the screen and the speed in which the balloons rise in the air increase slightly as the game progresses to keep the game interesting when the user improves. Another variation we are currently considering allows novice users to “freeze” the currently displayed balloons and repeatedly practice those words. Freezing the screen allows users who do not currently understand how shape writing works to practice words in a playful, yet relaxed setting. To avoid user frustration in having to start the game from the beginning (as in [9]), and to keep players interested in going back to the practice game, the current player state is automatically saved and restored every time the user exits or starts the game.

Conclusions

In this paper we briefly reviewed the shape writing text entry method for pen-based computers. To give users an effective and enjoyable introduction and to practice the method in a playful manner we developed a balloon game with three design goals: it should be efficient, fun, and challenging. We have been relying on informal user feedback when we developed the game, and the comments received so far have been encouraging. When demonstrating shape writing using the game it has also been obvious that users immediately understand the game concept and objective.

Acknowledgements

This work was in part sponsored by the Santa Anna IT Research Institute (SICS Linköping) and IBM Research.

References

- [1] Kohl, R. and Shea, C.H. Pew (1966) revisited: acquisition of hierarchical control as a function of observational practice. *Journal of Motor Behavior* 24, 3, 247-260, 1992.
- [2] Kristensson, P.O. and Zhai, S. SHARK²: a large vocabulary shorthand system for pen-based computers. *Proc. UIST 2004*, 43-52.
- [3] Kristensson, P.O. Breaking the laws of action in the user interface. *Ext. Abstracts CHI 2005*, 1120-1121.
- [4] Kurtenbach, G., Sellen, A. and Buxton, W. An empirical evaluation of some articulatory and cognitive aspects of “marking menus”. *Human-Computer Interaction* 8, 1, 1993, 1-23.
- [5] Landauer, T.K. and Bjork, R.A. Optimum rehearsal patterns and name learning. In *Practical aspects of memory*, 1978, 625-632.
- [6] MacKenzie, I.S. and Soukoreff, R.W. Text entry for mobile computing: models and methods, theory and practice. *Human-Computer Interaction* 17, 147-198.
- [7] Stubbs, K. Kana no senshi (kana warrior): a new interface for learning Japanese characters. *Ext. Abstracts CHI 2003*, 894-895.
- [8] Wobbrock, J.O., Myers, B.A. and Kembel, J.A. EdgeWrite: a stylus-based text entry method designed for high accuracy and stability of motion. *Proc. UIST 2003*, 61-70.
- [9] Zhai, S., Alison, S. and Accot, J. Movement model, hits distribution and learning in virtual keyboarding. *Proc. CHI 2002*, 17-24.
- [10] Zhai, S. and Kristensson, P.O. Shorthand writing on stylus keyboard. *Proc. CHI 2003*, 97-104.

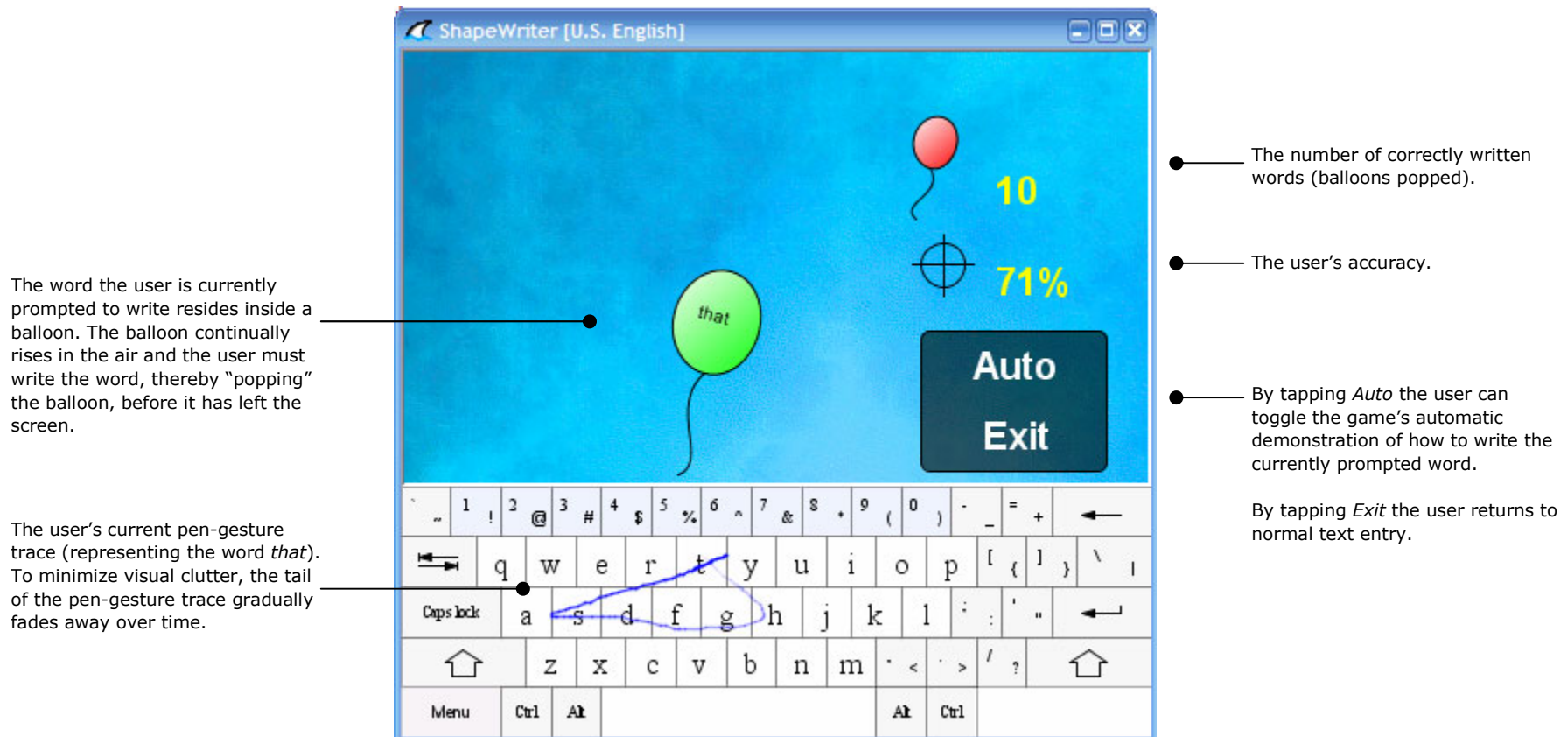


figure 6. A balloon with the word *that* inside is presented to the user. Balloons float from the bottom of the screen and rises in the air towards the top of the screen until they are no longer visible. The user pops the balloons by quickly shape writing the prompted words on the keyboard. As the game progresses more balloons are floating on the screen at the same time, and they can also drift away faster.