Ink Completion

Laurent Denoue Patrick Chiu

FX Palo Alto Laboratory

Abstract

Ink completion is a technique for suggesting characters or words to users as they write with digital ink. With a simple gesture, users can accept a recommendation and continue writing. The technique improves the handwriting experience and is very suitable to support note-taking on tablet computers and PDAs.

Key words: ink completion, interaction technique, pen computing.

1 Introduction

Handwriting is becoming popular with new pen computers such as PDAs and TabletPCs. But for most people now, handwriting is too slow compared to typing on a keyboard, so it would be nice to improve the speed of handwriting. Ink completion accomplishes this by suggesting – in real-time – characters (Figure 1) or words (Figure 2) to users as they write.

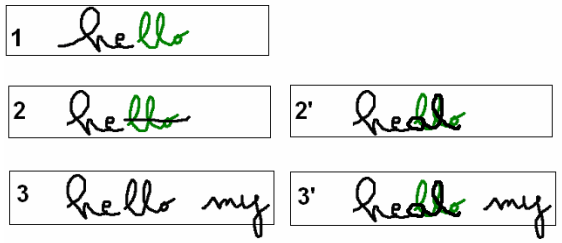

Figure 1. Ink Completion: 1) the user writes "he" and the system suggests "llo"; 2) if the user likes the suggestion, he/she draws a stroke across the suggestion to accept it and 3) moves on to writing more ("my"). If the user doesn't like the suggestion, he/she simply continues to write normally (2' and 3').

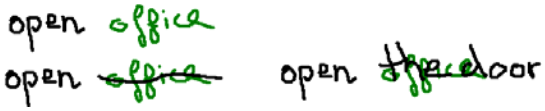

Figure 2. Ink completion at the word level.

The system can also suggest alternative words by displaying them below the current input line (Figure 3). Alternates are ranked by likelihood of being relevant to the current user's input.

Figure 3. Recommendations can also be displayed below the current line.

This input technique is especially suitable for ideographic languages like Chinese or Japanese. Using a keyboard to enter these languages is a chore; Japanese has an awkward keyboard system using a Western character set, and is even worse for Chinese which does not have a simple keyboard system. Pen input is much more effective for many types of applications.

In Chinese or Japanese, compound words are common. Hence, this technique can be useful in many cases and not just for specialized word sequences of the user. In particular, 2-word compounds are frequent, and 4-word compounds often occur. An example is shown in Figure 4.

Figure 4. Recommendations for compound words in Chinese.

2 Interaction technique

Users start writing normally. When they want a suggestion, they keep the stylus down for a short period of time. When this event is detected, the system displays recommendations.

The user can quickly accept a recommendation by gesturing over it from left to right (Figure 1.2). This interaction is fast and feels very natural: users feel as if they had written the word themselves. They stop the gesture after the end of the recommendation (Figure 1.2) and continue writing (Figure 2.3).

If none of the recommendations is satisfying, the user keeps writing (Figure 1.2' and 1.3').

3 Technical details

In order to suggest ink completions, the system matches the current user's input against a pool of previously handwritten words or available text-based words.

3.1 Matching ink to ink from the same user

When the user starts inking, strokes (defined by pen up/pen down) are segmented at local Y extremes and accumulated into an ink word (see Figure 5). The end of an ink word is either indicated by a pause or when the user starts inking far enough away (from the right boundary of the ink word or on the next line). A new ink word is then started.

The system matches the current ink word with existing ink words. In our implementation, the distance between two ink words is computed using the edit distance, similar to the algorithm presented in [1] and [4]. Other matching algorithms are available in the literature (see for example [2], [3], and [5]).

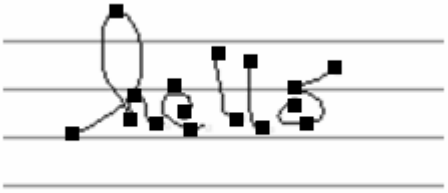

Figure 4. Ink strokes are segmented at local Y extremes and placed into an ink word.

3.2 Matching ink to text

It is also be useful to get recommendations from text words. Ink completion is currently used in the context of the Shared Text Input system [6] where words are extracted automatically from PowerPoint slides in real-time. These words make for a perfect set of candidates that users might find interesting to reuse, for example as they take notes during a lecture.

Instead of trying to convert ink to text through handwriting recognition, we have instead developed a technique that quickly matches ink words and text words. For each text word, we compute a signature that codes each letter depending on its category defined as follows:

- category 0: normal letter like a
- category 1: ascender like b
- category 2: descender like p
- category 3: both like f

For example, the word "ink" will be coded "001".

We also code an ink word similarly. For example, the ink word in Figure 4 is coded as "10110".

This technique works very well when few text words are possible for recommendation, which is the case in the Shared Text Input system where we only recommend words that are on the slide being shown during a lecture. We computed the signatures of words extracted from 2000 slides and found that only 2% of words in a given slide had identical signatures (like "feel" and "loud").

Instead of displaying the recommendation using typeset font, we display it using a handwriting font that was manually created by each user beforehand.

4 Conclusion and future work

We presented ink completion, an interaction technique that suggests handwritten characters or words to users. Users can choose a suggestion by drawing a line over it, minimally interfering with handwriting.

Two ink matching techniques were presented, one to match ink strokes against other ink strokes, and the other to match ink strokes to text-based words.

Informal testing was encouraging, and we will deploy this system in a new version of Shared Text Input where students will take handwritten notes on a PDA.

We are also very interested in matching ink words written by two or more users. Initial tests showed that the technique presented in section 3.1 is too sensitive to differences in handwriting styles. However, the ink to text matching technique (section 3.2) seems to work better and we will further investigate these possibilities.

Acknowledgements

We would like to thank Jim Baker and Candace Kamm for supporting this research.

References

[1] Daniel Lopresti, Andrew Tomkins. On the searchability of Electronic Ink, 1994

[2] Richard Hul, Dave Reynolds, Dipankar Gupta Scribble Matching, Hewlett Packard Technical Report, 1994

[3] Alex Poon, Karon Weber, Todd Cass. Scribbler: A tool for searching digital ink, CHI 1995

[4] Daniel Lopresti, Tomkins, A., Zhou J. Algorithms for matching hand-drawn sketches, 1996

[5] Shingo Uchihashi, and Lynn Wilcox. Automatic Index Creation for Handwritten Notes, In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (Phoenix, AZ), vol. 6, pp. 3453-3456, 1999., March 15, 1999

[6] Denoue L, Patrick Chiu. Shared Text Input for Note Taking on Handheld devices, CHI, 2002