A Comparison of Two Handwriting Recognizers for Pen-based Computers

Larry Chang & I. Scott MacKenzie
Dept. of Computing & Information Science, University of Guelph

Abstract

An experiment is described that compares two commercial handwriting recognizers with hand-printed characters. Each recognizer was tested at two levels of constraint, one using lowercase letters (which were the only symbols included in the input text) and the other using both uppercase and lowercase letters. Two factors – recognizer and constraint – with two levels each, resulted in four test conditions. A total of 16 subjects performed text-entry tasks for each condition. Recognition accuracy differed significantly among conditions. Furthermore, the accuracy observed was far below the walk-up accuracy claimed by the developers of the recognizers. Entry speed was affected not by recognition conditions but by users' adaptation to the idiosyncrasies of the recognizers. User satisfaction results showed that recognition accuracy greatly affects the impression of walk-up users.

Keywords: Pen-based computing, text entry, hand-printing, mobile computing, character recognition

1 Introduction

Pen-based computers have received considerable attention recently as products such as personal digital assistants (PDAs), personal organizers, and digital tablets enter the market place. They offer great advantages to people who work intensively with information and who work away from a desk (e.g., field service personnel, couriers, doctors).

Although several input methods appear attractive with pen-based computers, it is usually claimed that the primary mechanism for text entry remains handwritten characters. In fact, it is the rapidly maturing handwriting recognition technologies that have contributed to the increased availability and popularity of pen-based systems.

Although claims abound as to the effectiveness of handwriting recognizers, empirical data are lacking. Our research is motivated by the need for such data to guide designers of pen-based computing systems.

This paper reports the results of an experiment testing two commercial recognizers in a text-entry task. The input text contained lowercase letters only. We tested each recognizer under two levels of constraint: lowercase letters only (26 symbols) and uppercase-plus-lowercase letters (52 symbols). Recognition accuracy should decrease as the size of the symbol set increases; however, formal experiments on the effect of constraint have not been reported.

2 State of the Art

There is a substantial body of research on the use of a pen or stylus as a computer input device. Most is concerned with the capabilities of the pen for gestural input. This includes interaction techniques for creative drawing [2], editing text documents [5, 7, 14], or editing graphic objects [8, 15].

For text entry, a few researchers have

attempted to re-design the Roman alphabet with simplified strokes [6, 13]; however, the requirement that a technique must be learned is a serious drawback. Most researchers acknowledge that the most pervasive form of text entry is that which draws on existing handwriting skills [11, 14].

Researchers at IBM have found that printed characters have significantly higher recognition rates than cursive handwriting [14]. Gibbs [4], who surveys 13 handwriting recognizers, also reports that the majority of available products deal with printed handwriting translation and the newest products that deal with cursive handwriting are still in development and are available only as beta releases. Furthermore, Santos et al. [12] found that the highest recognition rate of printed characters on a grid display (96.8%) is the same as for human observers identifying isolated hand-printed characters. This suggests that current recognition engines for discrete hand-printed characters are almost as good as human interpretors. Recent work at IBM on user acceptance of handwriting recognition accuracy found a threshold around 97% [9]. That is, users are willing to accept error rates up to about 3%, before deeming the technology as too encumbering.

Several experiments have investigated how various interface characteristics affect recognition performance [12, 14]. Yet few have attempted to compare the recognition performance of available recognizers.

The experiment described in the next section evaluates two commercial recognizers: the Microsoft® character recognizer embedded in the *Pen for Windows®* operating system and *Handwriter™ 3.3* from Communications Intelligence Corp. (CIC). The input text consisted of lowercase words. Recognition constraint – lowercase letters (26 symbols) vs. uppercase-plus-lowercase letters (52 symbols) – was included as an additional factor. Constraint is felt to impact recognition performance in general [14], yet its impact on various recognizers may be different. Recognizer and constraint were two independent variables in the experiment. Three dependent variables were measured: entry speed, recognition accuracy, and user satisfaction.

3 Method

3.1 Subjects

Sixteen volunteer subjects participated in the experiment. They included seven females and nine males, 14 right-handed and two left-handed. Ages ranged from 19 to 55. Twelve subjects were university students and ten indicated they used computers on a daily basis.

3.2 Hardware/Software

Hardware for the experiment consisted of a 50 MHz 486 IBM-compatible PC with a Wacom® *PL-100V* tablet for pen entry. The *PL-100V* is both a digitizer for user entry and a 640×480 LCD gray-scale screen. Character entry was observed on a VGA monitor, which was tilted to prevent users from seeing it.

Software to run the experiment was developed in C using Microsoft® *Pen for Windows®*. The experiment simulated a typical pen-entry application.

Recognizers tested were the Microsoft® character recognizer included with *Pen for Windows®* and CIC's *Handwriter™ 3.3*.

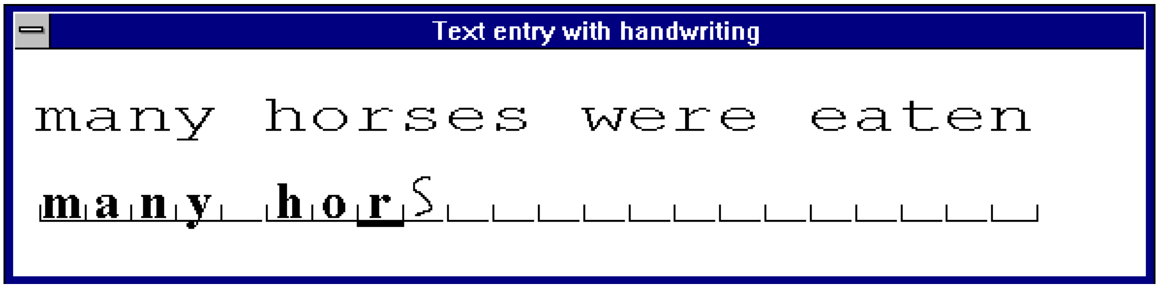
3.3 Procedure

The task consisted of entering characters provided by the software. Subjects printed in grids below the displayed characters (Figure 1).

Phrases containing 22 characters (four words and three blanks) were randomly presented in blocks of three. The single-letter frequency count table of Mayzner and Tresselt [10] assisting in creating a balanced phrase set such that each letter in the input text occurred with the same relative frequency as common English.

In a one hour session, subjects performed all four conditions, which were created by manipulating two factors. There was a 10-min break between session 2 and session 3. Conditions were counterbalanced using a Latin Square to minimize transfer effects related to the factors.

Execution of a condition consisted of a brief practice session of three phrases and then nine



Text entry with handwriting

many horses were eaten

many horS

Figure 1. Experiment screen.

blocks (three phrases each) of recorded entry. Auditory feedback was produced after each character to indicate whether the character was recognized correctly. To help motivate subjects, summary data for accuracy and speed were displayed at the end of each block.

Subjects were instructed to aim for both accuracy and speed. As well, they were told that if a mistake was made they should ignore it and continue with the rest of the sequence. The tablet was propped slightly at an angle as preferred by each subject.

3.4 Design

The two factors with two-levels each resulted in the following four conditions:

- Microsoft/Lowercase
- CIC/Lowercase
- Microsoft/Upper+lower
- CIC/Upper+lower

For each entry, the time from the completion of the previous character to the completion of the current character was recorded. The timing value for the first character in a sequence was meaningless as there was no start time to reference from (and thus the first character was not used in the statistical analysis).

Due to limitations in the experiment, user input was occasionally out-of-sync and generated erroneous timing values (negative values or values greater that 10 s). Those values were removed before the statistical analysis.

Text entry speed was expressed in words per minute (wpm) using the typist's definition of a word: 1 word = 5 characters (including spaces).

Subjects completed a pre-test questionnaire for demographic information and a post-test questionnaire for user satisfaction.

4 Results and Discussion

4.1 Condition Effects

The four conditions ranged in entry speed from 16.9 to 17.7 wpm and in accuracy from 86% to 95% (see Figure 2). Neither recognizer nor constraint had a significant effect on entry speed (for recognizer, $F_{1,15} = 2.85$, $p > .05$; for constraint, $F_{1,15} = 0.148$). However, both recognizer and constraint had a significant effect on recognition accuracy (for recognizer, $F_{1,15} = 37.3$, $p < .0001$; for constraint, $F_{1,15} = 29.1$, $p < .0001$). Regarding recognition accuracy, the Microsoft recognizer was also significantly more sensitive to constraint ($F_{1,15} = 15.4$, $p < .005$). This is evident in Figure 3, showing a dramatic reduction in accuracy for the Microsoft recognizer using the upper+lower symbol set.

The finding of "no significant effect" on entry speed was fully expected since entry speed is controlled more by the subject than by the interface. As long as the recognition latency is low enough, the actual entry speed should be user's writing speed. In Gibbs' [4] summary of 13 recognizers, the recognition speed is at least 4 characters per second, which translates into 48 wpm. This is well above typical human hand-printing speeds of 15 wpm [3]. Our mean of 17.4 wpm is slightly above this, so subjects were entering text at a rate they felt comfortable with. It is the accuracy observations that are most telling. None of our conditions yielded rates of 97%, the minimum rate for user acceptance found by LaLomia [9].

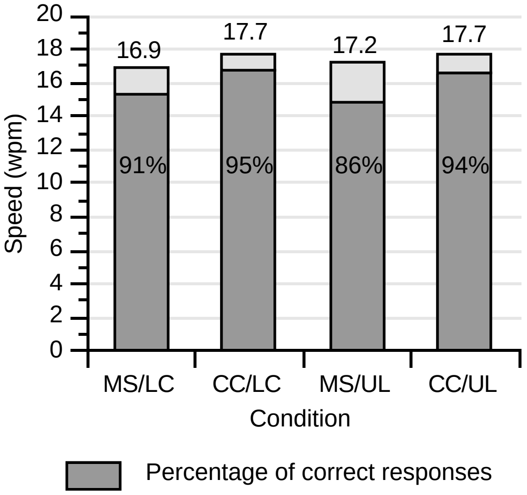


Figure 2. Comparison of the four conditions for entry speed and recognition accuracy (MS = Microsoft, CC = CIC, LC = Lowercase, UL = Upper+lower).

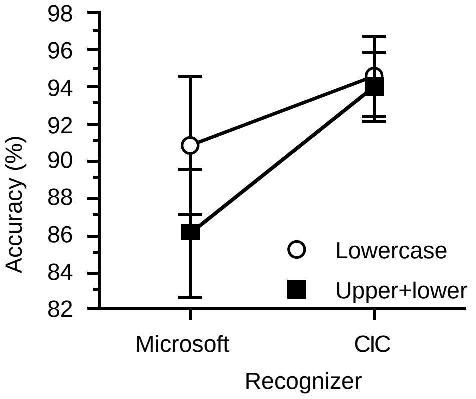


Figure 3. Interaction plot of recognizer vs. constraint for recognition accuracy.

4.2 Learning

Although our experiment did not test subjects for repeated sessions over a prolonged period of time, we did examine the learning effects over the four sessions administered. The four sessions had mean entry speeds of 15.8, 17.0, 17.9, and 18.7 wpm, and mean recognition accuracy of 91%, 91%, 92%, and 90%, respectively (Figure 4). The results showed that learning had a significant effect on entry speed ($F_{1,15} = 26.0$, $p < .0001$) but not on recognition accuracy ($F_{1,15} = 0.414$).

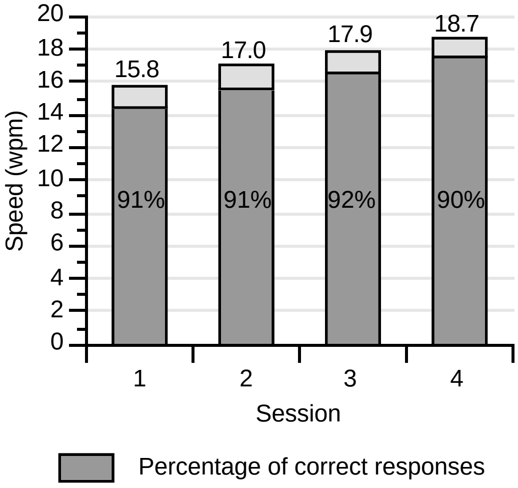


Figure 4. Comparison of the four sessions for entry speed and recognition accuracy.

Apparently subjects did not improve their accuracy with practice, but they did get faster. This is consistent with Bailey's [1] observation that “in activities where performance is primarily automatic, the proportion of errors will remain fairly constant, but the speed with which the activity is performed will increase with practice” (p. 101).

This suggests that our initial, somewhat low, observations on accuracy are not likely to improve with practice. Of course the limitation is primarily with the recognition software, and improvement in the recognition algorithms will, no doubt, yield improvements in accuracy.

4.3 Error Rates by Character

A detailed analysis for errors was undertaken by decomposing error data by character. For each condition, error rates were distributed as shown in Figure 5. The values show the contribution of each character to the total error rate, which effectively normalizes the data by the relative occurrence of each letter. All 26 values in one chart add to the mean error rate of that condition given in Figure 2. Notice that the letter “l” in the

Microsoft/Upper+lower condition is represented by a special "back bar" because the value is far higher than other values and does not fit in the chart. The actual value (4.9%) is shown beside the bar. The cause is explained later.

These results are useful for designers concerned with the overall performance rather than particular defects of the program. For example, we can easily see some letters (for example, "i", "n", and "v" in CIC/Lowercase condition) have a more significant impact on overall performance (their error contributions were all higher than 0.4%).

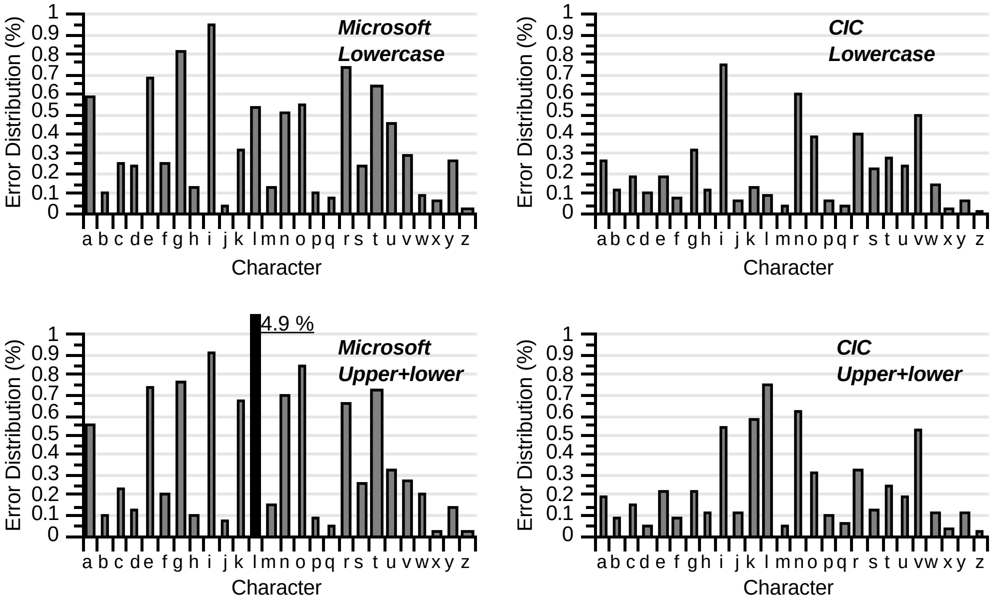


Figure 5. Error rates distributed along the alphabet.

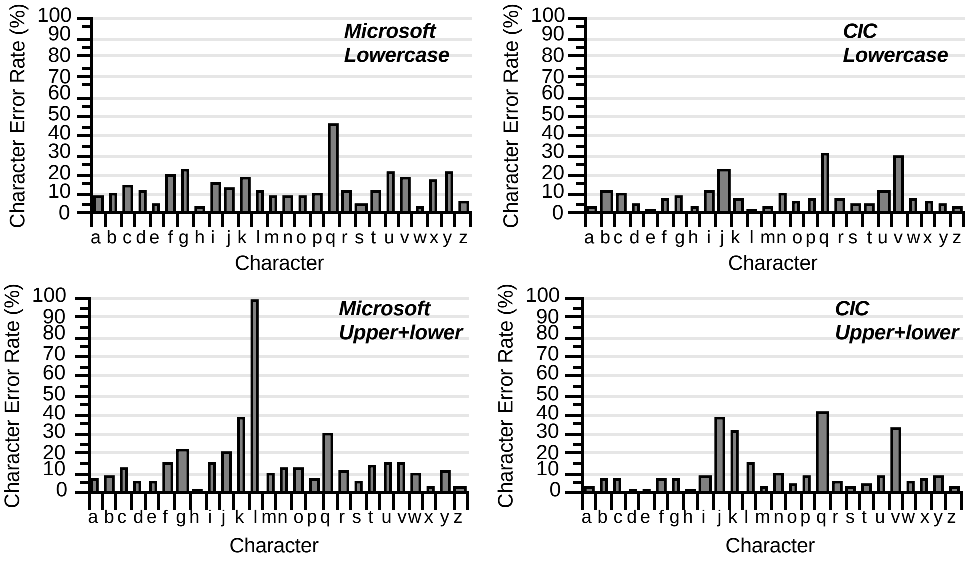


Figure 6. Recognition accuracy for individual character.

4.4 Misrecognition of Characters

For each condition, the misrecognition rate of each character was examined (see Figure 6). These results show how well a particular character was recognized. Those characters with high misrecognition rates may indicate certain defects of the recognition engine. For example, in the Microsoft/Upper+lower condition, the algorithm for the letters “l”, “k”, and “q” needs work because the misrecognition rates were all higher than 30%. In the worst case of the letter “l”, less than 5% of the letters were recognized. Notice that, although the letter “q” had a high misrecognition rate (Figure 4), it did not contribute too much to the overall error rate (Figure 5) because the occurrence of the letter “q” in English is low.

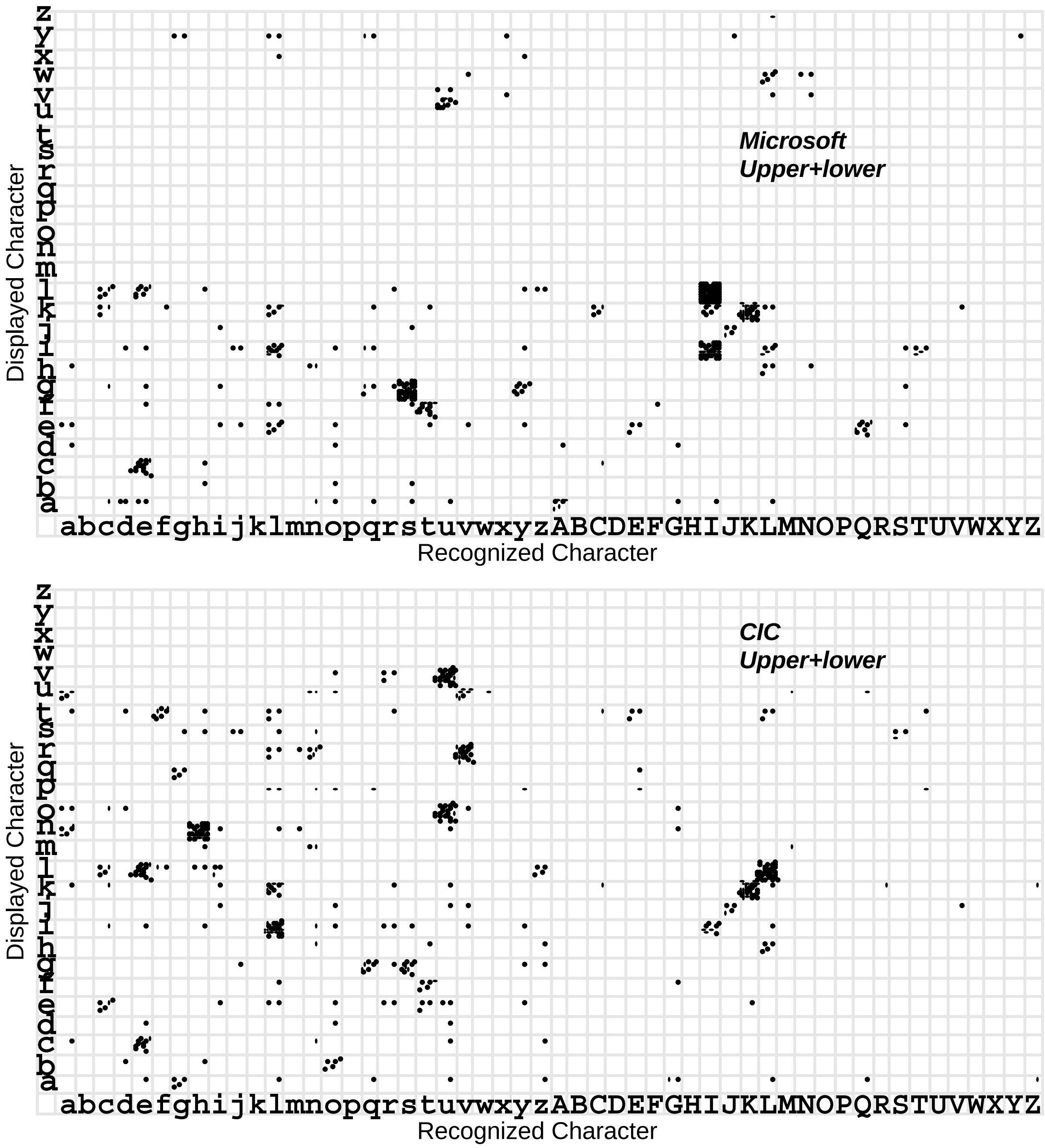


Figure 7. Misrecognition distribution map. Each dot represents three occurrences.

4.5 Misrecognition Distribution

For each condition, the distribution of misrecognition character pairs (displayed character vs. recognized character) was also examined. Only the distribution maps of the two upper+lower conditions are shown (Figure 7). These maps further identify possible defects in recognition engines. For instance, in Microsoft/Upper+lower condition, the letter “l” was frequently misrecognized as capital “I”, which explains why the letter “l” had a very high error rate. Other frequent misrecognition pairs were “g-s”, “i-I”, and “k-K” found with the Microsoft recognizer, and “i-l”, “k-K”, “l-L”, “n-h”, “o-u”, “r-v”, and “v-u” found with the CIC recognizer.

4.6 User Satisfaction

Subjects’ evaluations of the recognizers in the questionnaire matched the overall accuracy of four conditions. On average, subjects evaluated the CIC/Upper+lower condition as the best, CIC/Lowercase second, Microsoft/Lowercase third, and the Microsoft/Upper+lower condition as the worst. The results suggest that recognition accuracy is quite noticeable by users. Bad experiences, such as the “l-I” misrecognition pair in Microsoft/Upper+lower condition, can catch users’ attention and degrade walk-up impressions.

Subjects’ answers also suggested that hand-printing recognition is more suitable for simple text-entry input tasks such as form filling as opposed to input-intensive tasks such as creating a document from scratch.

The Wacom® tablet received numerous complaints on its surface texture. Subjects indicated that the texture was worse than paper.

5 Conclusion

With current hand-printing recognition technologies, text-entry speed depends mainly on the user’s printing speed. Recognition constraint has significant effect on recognition accuracy. The CIC *Handwriter*™ has a significantly lower error rates than the Microsoft® character recognizer. The Microsoft® character recognizer is also significantly more sensitive to recognition constraint.

To attract walk-up users, recognition systems must improve to reduce misrecognition patterns. Handwriting recognition technology can and will benefit from adaptive and context-sensitive algorithms now under development; however, these will not significantly affect the walk-up accuracy of systems that seek to entice novice users.

Acknowledgements

This research is supported by the University Research Incentive Fund (URIF) of the Province of Ontario, the Natural Sciences and Engineering Research Council (NSERC) of Canada, and Architel Systems Corp. of Toronto. We gratefully acknowledge this support without which this research would not have been possible. We would also like to thank Craig McQueen for writing the experiment software.

About the Authors

Larry Chang is an M.Sc. student in computer science at the University of Guelph. He holds a B.Sc. degree from Xidian University in China. Scott MacKenzie is Assistant Professor of Computing and Information Science at the University of Guelph. He holds a B.Mus. degree from Queen's University, and M.Ed. and Ph.D. degrees from the University of Toronto. His research interests include human performance measurement and modeling, and input devices and interaction techniques for human-computer interfaces. The authors can be reached through email: larry@snowhite.cis.uoguelph.ca, mac@snowhite.cis.uoguelph.ca.

References

[1] R. W. Bailey, *Human performance engineering* (2nd ed.). Englewood Cliffs, NJ: Prentice Hall, 1989.

[2] T. W. Bleser, J. L. Sibert, and J. P. McGee, "Charcoal sketching: Returning control to the artist," *ACM Transactions on Graphics*,

vol. 7, no. 1, pp. 76-81, 1988.

[3] D. B. Devoe, "Alternatives to handprinting in the manual entry of data," *IEEE Transactions on Human Factors in Electronics*, HFE-8, pp. 21-32, 1967.

[4] M. Gibbs, "Handwriting recognition: A comprehensive comparison," *Pen*, pp. 31-35, March/April 1993.

[5] D. Goldberg, and A. Goodisman, "Stylus user interfaces for manipulating text," *Proceedings of the ACM SIGGRAPH and SIGCHI Symposium on User Interface Software and Technology*, pp. 127-135. New York: ACM, 1991.

[6] D. Goldberg, and D. Richardson, "Touch-typing with a stylus," *Proceedings of the INTERCHI'93 Conference on Human Factors in Computing Systems*, pp. 80-87. New York: ACM, 1993.

[7] G. Hardock, "Design issues for line-driven text editing/annotation systems," *Proceedings of Graphics Interface '91*, pp. 77-84. Toronto: Canadian Information Processing Society, 1991.

[8] G. Kurtenbach, and B. Buxton, "GEdit: A testbed for editing by contiguous gestures," *SIGCHI Bulletin*, vol. 23, no. 2, pp. 22-26, 1991.

[9] M. J. LaLomia, "User acceptance of handwritten recognition accuracy," *Companion Proceedings of the CHI'94 Conference on Human Factors in Computing Systems*, p. 107. New York: ACM, 1994.

[10] M. S. Mayzner, and M. E. Tresselt, "Tables of single-letter and diagram frequency counts for various word-length letter-position combinations," *Psychonomic Monograph Supplements*, vol. 1, pp.13-32, 1965.

[11] C. McQueen, I. S. MacKenzie, B. Nonnecke, S. Riddersma, and M. Meltz, "A comparison of four methods of numeric entry on pen-based computers," *Proceedings of Graphics Interface '94*. Toronto: Canadian Information Processing Society, 1994.

[12] P. J. Santos, A. J. Baltzer, A. N. Badre, R. L. Henneman, and M. S. Miller, "On handwriting recognition system performance: Some experimental results," *Proceedings of the Human Factors Society 36th Annual Meeting*, pp. 283-287. Santa Monica, CA: Human Factors Society, 1992.

[13] D. Veniola, D., and F. Neiberg, "T-cube: A fast, self-disclosing pen-based alphabet" *Proceedings of the CHI'94 Conference on Human Factors in Computing Systems*, pp. 265-270. New York: ACM, 1994.

[14] C. G. Wolf, A. R. Glasser, and T. Fujisaki, "An evaluation of recognition accuracy for discrete and run-on writing," *Proceedings of the Human Factors Society 35th Annual Meeting*, 359-363. Santa Monica, CA: Human Factors Society, 1991.

[15] R. Zhao, R., "Incremental recognition in gesture-based and syntax directed diagram editor," *Proceedings of the INTERCHI'93 Conference on Human Factors in Computing Systems*, pp. 95-100. New York: ACM, 1993.