

Feature Stroke: A Text Entry Method Using Joystick

Zhenyu Gu, Cheng Chu, Xingya Xu, and Zhanxun Dong

Abstract—This paper presents a new text entry method, Feature Stroke (FS). FS is a predictive letter-group-based method using joysticks. 26 letters are grouped and associated with 8 primitive gestures on joysticks. FS has a novel mnemonic strategy for visual feedback of uncertainty in the group-based inputting. The design of FS is aimed to improve efficiency and user experience of text entry in some special circumstances using joysticks. Multi-session test of the prototype shows that the learning curve rises quickly. And the participants' subjective evaluation is highly positive. In the non-predictive mode, pairwise test shows that skilled users' performance of FS is much better than that of the onscreen keyboard selection, the most prevalent method for joysticks so far.

Index Terms—Joystick, text entry, mnemonic, ergonomics, GUI design.

I. INTRODUCTION

Joysticks have been widely accepted as default input peripherals in many circumstances, such as smart TV, game box and some in-car infotainment systems. The devices are simple, compact and universally accessible for all, even those with motor impairments [1]. Smart TV and Video games always require extensive text entry for search, registration, identification and system configuration. As general-purpose devices, however, joysticks are not optimized for text entry.

Text entry methods of these systems commonly adopt on-screen selection keyboards. A selection halo is moved up, down, left or right using joysticks or other multi directional devices. Every time the user presses a joystick button, the currently highlighted key is “pressed.” The methods are intuitive without any learning process. Selection-based methods, however, are slow, requiring many movements to select one letter [2]. Users quickly become bored. Another drawback is the methods mostly require screen real estate to display options, and have to allocate the user's attention to both the keyboard and the input lines of text.

In this paper, we present a new text entry method with joysticks, which we called Feature Stroke (FS). It uses a number of “primitive gestures” of joysticks to signal “keypresses”. It has a novel animated mnemonic strategy for visual feedback of the uncertainty existing in the group-based input. The aim of FS is to improve text input efficiency under the constraints of joystick-like devices without too much cognitive load, errors and learning cost.

Manuscript received December 14, 2014; revised April 24, 2015. This work was supported in part by Humanities and Social Sciences Foundation of the Ministry of Education (11YJCZH44 & 12YJCZH031)

The authors are with the School of Media and Design, Shanghai Jiao Tong University, Shanghai, China (e-mail: {zygu, cc060705, fireon7, dongzx}@sjtu.edu.cn).

II. DESIGN OF FEATURE STROKE

A prototype system of FS has been implemented as shown in Fig. 1.

The approach tries to combine the group-based input with a number of primitive gestures of writing using joystick like devices. FS has a predicting mechanism for the group-based text entry.

The FS interface is composed of three parts: an input box, an instructional panel and a prediction box. Input box display inputted words and the uncertain state of a word inputting in 8 gestures. In Fig. 1 you can see four strange symbols in the input box, which are mnemonics of letter sequence “DESI” that we expected. But so far, the system still has several options with its word level prediction, until the next gestural input of “G”. The word “DESIGN” will emerge out.

The instruction panel is designed for users' learning. It gives effect of direct manipulations in response to gestures of the joystick.

Feedback and feed forward animations are very integral parts of text entry methods. Former researches found visual design quality has significant influence not only on usability but emotional experience. Input methods rely heavily on small details that make big differences [3].

When the predictive mode is on, words predicted are listed in the prediction box. The prediction uses the Corpus of Contemporary American English (COCA) and sorts the words according to their frequency in the corpus. The number of candidate words will reduce as more keystrokes are entered. Users can scroll to a wanted word in the predictive box any time.

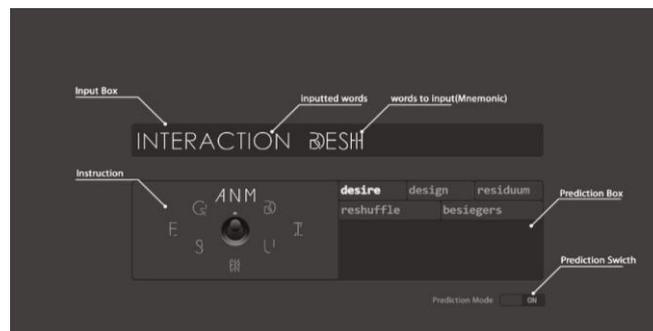


Fig. 1. The GUI of feature stroke.

A. Ergonomic Concern and Primitive Gestures Final Stage

It is not so easy to “write” the letters with joystick. It has been testified that joystick performed much worse than touchpad [1].

So we prefer a set of primitive gestures obviously distinct from each other (see Fig. 2, we use 4 directions instead of 8, to avoid possible mistyping). Such simple and symmetrical

gestures are not only easier for users to remember and also stabilize the machine recognition. The number of gestures users can remember would seem to be seven, plus or minus two.

The Simple directional movements together with bouncing force helping joystick self-centering enables users to control joysticks with less efforts and energy. The physical round edge provides tangible feedback in movement and results in greater speed and stability of motion.

B. Letter Groups Based on Visual Features

FS is designed to group the letters based on their visual characteristic strokes, which could be associated with the primitive gestures. For instance, the gesture of group CGOQ is drawing in a counter-clockwise arc from top, and 'I' 'H' and 'K' share a downward vertical line, which is similar to their common feature stroke.

Using a reduced set of gestures has some ergonomic and cognitive advantages. Smaller the number of gestures, simpler the gestures are. However, the disadvantage is that KSPC (Key Strokes per Character) [4] may go up. KSPC is the number of key strokes, on average, to generate each character using a given text entry method. FS is a compromise between number of keys/gestures and KSPC. When the number of keys is reduced to 8, KSPC does go up 50%. The KSPC of Base 8 without prediction is $\log_8 27$, about 1.5. Considering the different frequencies of the letters, KSPC of Base 8 in Huffman coding is about 1.463, if high frequency letters are coded in different groups as default ones.

When word level prediction is applied, KSPC of 8 keys could approximate the limit of "one stroke per character", about 1.1, which is the record of T9.

Different grouping schemes have different efficiencies of input. Gong and Tarasewich [5] proposed a metric, Disambiguation Accuracy (DA), for evaluating the goodness of a particular keypad grouping. DA is the probability that, if a sequence of keys is pressed, the "correct" word appears. That means we must maximize information gain of each keystroke. As total entropy of English on 26 characters is certain, if we use a reduced set of keys for English text entry, it is better to evenly allocate the entropy to each key (each group), and minimize the mutual information between each group pair. The number of grouping schemes, however, is astronomical, to find the optimum is almost incomputable.

Gong and Tarasewich [6] used genetic algorithm to search for the best assortment of letters on mobile phone keypad. The results show that optimum assortments do improve efficiency somehow of the conventional T9 keypad. The DA of the conventional T9 keypad is 95.41% when that of the best unconstrained keypad is 98.5%. We also have measured the DAs of several 8-key grouping schemes by computational traversal on the English corpus COCA. We found that there are little differences among the different grouping schemes, if the overall letter frequencies are evenly allocated to the groups. The DA of the grouping scheme in Fig. 3 is around 94.22%, slightly lower than T9's 95.41%.

The most efficient coding scheme for machine, however, may not be so user-friendly. MacKenzie, Zhang [2] verified irregular letter assortment increases the difficulty of learning.

A better solution must compromise both coding efficiency and usability, and somehow the usability is more weighted. The grouping in T9 maintains the alphabetic sequence of the letters, so that greatly reduces users' learning cost. In FS, we try to group letters based on their intuitive writing strokes for the same reason.



Fig. 2. The 12 primitive gestures we sorted out. The first and second rows are adopted as feature strokes representing 8 letter groups. The rest four gestures are reserved for other usages, for instance, the "back space", or the switch of the prediction mode.

AMN	
HIX	
EF	
TZ	
CGOQ	
BDPR	
SJ	
LUVWY	

Fig. 3. The 8 groups of capital letters based on 8 selected primitive gestures.

C. Animated Ambiguous Mnemonics

The mnemonics we designed are enlightened by ambiguous images, a psychological phenomenon. Some ambiguous images such as Rubin's vase (also known as Rubin's face) and J.Jastrow's Rabbit-Duck are quite well known. The concept is that perception is somehow affected by current expectation in mind. Johnson [7] has discussed the phenomenon of ambiguous perceptions in text reading. He demonstrated two principles: one is "you see what you expected", another is "perception biased by the context". He

demonstrated that a character is perceived as different letters depending on the surrounding letters.

The graphic mnemonics of FS are ambiguous symbols mixed with shapes of all the possible letters in each group, as shown in Fig. 4.



Fig. 4. The graphic mnemonics of the 8 letter groups drawn by designers utilize the principle of “you see what you expected”, if you have any a letter of 26 in mind, you may see it among the 8 mnemonics, ambiguously.

In the predictive mode, the mnemonics in the input box could be pruned by more keystrokes entered. The mnemonics may seem complicated at the beginning and will be gradually simplified while keystroking. Fig. 5 shows the morphing process of the word VISUAL and COMPUTING in the predictive mode.

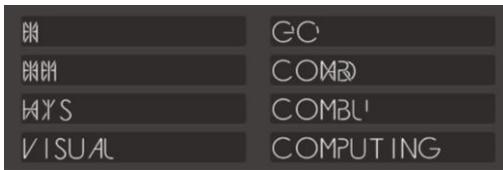


Fig. 5. The morphing process of the word visual & computing.

D. Non-predictive Mode

Non-predictive mode is optional in FS. For a single gesture input, the mnemonic of it is prompted at the cursor in input box, the default letter will be highlighted over the mnemonic. While the rest letters in the group are still visible.

E. Context of Application and Hardware

FS is developed as a universal text entry method for joystick and joystick like devices. We testified FS's compatibility with a typical joystick, an Xbox game controller, a multi directional knob for in car system, and a variant form of joystick: the ring button on remote control for TV Box.

The in car knob and TV remote control are extremely simple devices, both having only 4 digital sampling points in four directions around the stick, same as a 2X2 matrix keyboard. After some technical refit, 8 different gestures of FS can be detected fluently (with four unary keys and four dual key combinations processed with an Arduino board). Multi-directional knobs are popular in in-vehicle system as central control units, such as BMW iDrive, Benz COMAND and Audi MMI. Most of them can be slid in all directions as joysticks. Infotainment systems in those cars use interfaces like time stamp selection to input addresses and contacts.

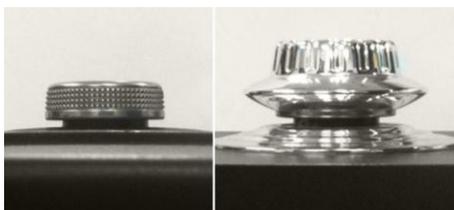


Fig. 6. Left is original multi directional knob in car. Right is a refitted one easier to shake. This device is selected in following usability test.

The self-centering bouncing force of most in car joysticks is relatively stronger than normal joysticks. This takes a bit more effort to complete the gestures on them. To make up for this, we raised the height of the knob by adding a higher cap on it to reduce the damping of the self-centering bouncing force (see Fig. 6).

III. EXPERIMENT AND EVALUATION

The apparatus in the experiments is the joystick specially refitted for in car system shown in Fig. 6. The stick can be comfortably moved in all direction to input the gestures. Twisting and pressing the knob on the stick can efficiently move the screen cursor and make selections. A GUI is developed using Adobe Flash, with a socket server running in background, transmitting the signals from the device. The system feedback has been tested without perceivable time delay.



Fig. 7. We prepared a place for participants sitting comfortably like drivers. The joystick is placed at their right hands. The GUI is presented on a 17 inch led display and about 3 feet away from the seat.

To examine the input efficiency and learnability of FS, we conducted a multi-session test on the FS predictive mode, and a pairwise test between the FS non-predictive mode and the onscreen keyboard selection. An onscreen keyboard was developed as testing benchmark for comparison.

18 participants (16 novices, 2 experts) were recruited for the test on the FS predictive mode. Expert users are those who well understand FS and have had adequate practices.

In the test in the predictive mode, participants were required to take five continuous sessions of test using FS. After the first and the fifth session, participants were asked to fill out a subjective evaluation questionnaire. The phrases are randomly selected from a collection of 500 phrases for evaluations of text entry methods published by Soukoreff and MacKenzie [8]. In the pairwise test between the FS non-predictive mode and the onscreen keyboard selection, participants were required to successively use these two text entry methods for test. The two entry methods were assigned to participants in a fully counterbalanced order to neutralize learning effects and fatigue. Analyses of variance (ANOVA) for test order show no significant differences.

We use WPM (Word per Minute) to describe the input speed. We find that the session of test has a significant influence on the input speed ($F_{4,90}=8.966, p<0.001$). Novice users' input speed has been promoted, the average WPM of the first session is 5.68, and that of the fifth session is 9.39,

almost twice that of the first session. Expert users' input speed has been more than twice the novice users' input speed, the average WPM is 25.

We calculated the average corrected error rate of the first two sessions and the last two sessions, and found that the sessions had a significant effect on corrected error rate ($F_{1,32}=4.629, p<0.05$). Since the novice users' uncorrected error rate is near 0, so the sessions also had a significant effect on the total error rate. We find that the error rate of the novice users reduced after a period of practicing. By contrast, the error rate of expert users is lower and the curve is flatter. It implies the limit that novice users could possibly reach after more practicing.

After the first and the fifth sessions, the participants were asked to fill out a subjective evaluation questionnaire. We hope to observe the changes of users' feelings after some practice. Fig. 8 shows that after using FS longer, most indicators of the subjective evaluation promoted, which is consistent with the improvement of the input speed and the decline of error rate we mentioned above. The decline of the score on enjoyment might be caused by taking the input tasks too long.

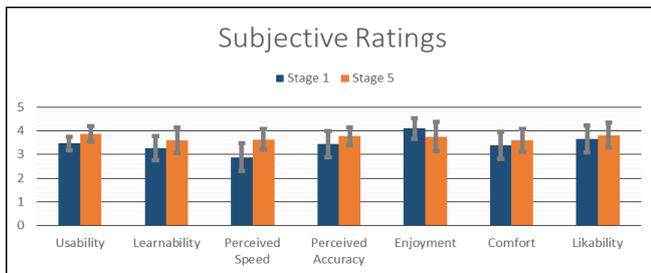


Fig. 8. The subjective evaluation of the first and the fifth session.

We conducted a pairwise test between the FS non-predictive mode and the onscreen keyboard selection. Since the non-predictive mode and the onscreen keyboard selection are both based on letters as their input unit, we use Characters per Minute (CPM) to describe users' input speed. We found that there's significant correlation between the same participant's input speed of the FS non-predictive mode and the onscreen keyboard selection ($\text{Sig.}<0.005$). Besides, we found that though the mean input speed of the FS non-predictive mode is 5.34 CPM, slower than that of the onscreen keyboard selection, there's no significant difference between them ($\text{Sig.}>0.05$).

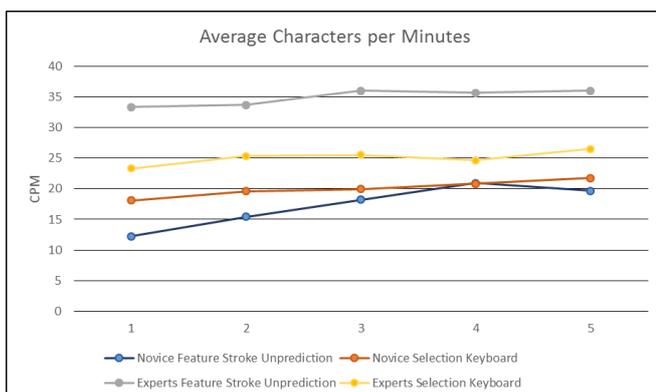


Fig. 9. The mean input speed of the non-predictive mode of Feature Stroke and the onscreen keyboard selection.

The uncorrected errors of the two methods are almost zero. So the total error rate can be approximately replaced by the corrected error rate. Novice users' total error rate of the FS non-predictive mode kept declining at the first three sessions, and remained relatively stable after that. The total error rate of the onscreen keyboard selection showed little change over sessions, remaining around 3%. The expert users' total error rate of the non-predictive mode is relatively stable, much lower than the novice users' at the last three sessions. As for the onscreen keyboard selection, the total error rate of the novice and expert users are both on a stable level, and there's little numerical difference between them.

Learning curve is customary in studies of human performance and learning to fit a regression curve in the form of the power law of learning [2]. As Fig. 10 shows, though the input speed of the FS non-predictive mode is lower at the beginning, it increases fast and overtook the onscreen keyboard selection at the session 5 or 6. The predictive mode has a great advantage in the novice learning. The non-predictive mode has slower speed than the onscreen keyboard selection at the beginning, but its learning speed is faster and its input speed overtook the latter's at session 5 or 6.

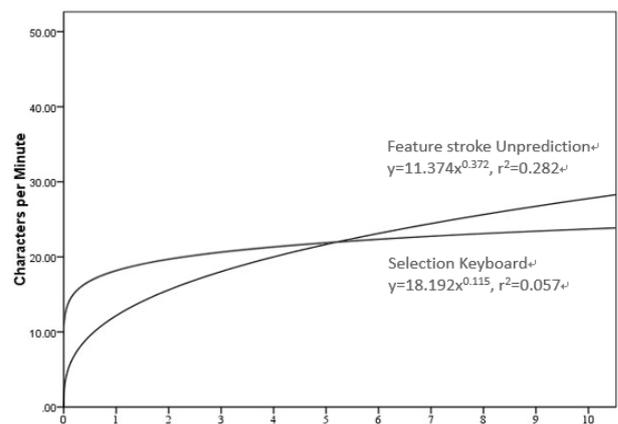


Fig. 10. The learning curves of the non-predictive mode of feature stroke and the onscreen keyboard selection.

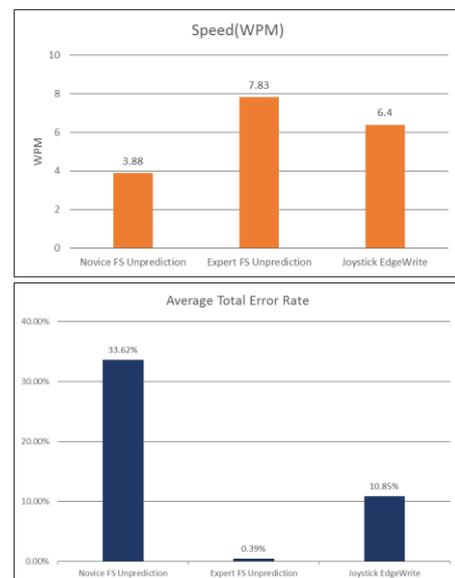


Fig. 11. The comparison between the input speed and total error rate of the FS predictive mode and edgewise for joystick.

IV. DISCUSSION

Fig. 11 shows that in the non-predictive mode, the input speed of Edge Write (6.4) is faster than the novice input speed of FS (3.88), but slower than the expert input speed (7.83). The novice error rate of FS (33.62%) is much higher than the error rate of Edge Write (10.85%), while the expert error rate of FS is almost 0 (0.39%).

As shown in Fig. 12, the predictive mode has a great advantage in the novice learning. The non-predictive mode has slower speed than the onscreen keyboard selection at the beginning, but its learning speed is faster and its input speed overtook the latter's at session 5 or 6.

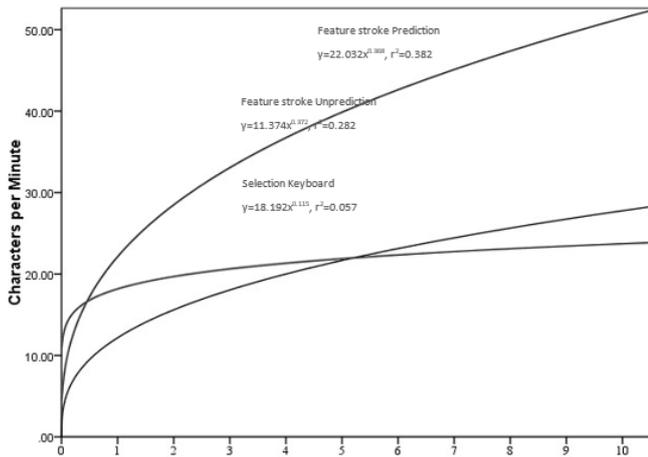


Fig. 12. The learning curve of the FS predictive mode, the FS non-predictive mode and the onscreen keyboard selection, on the scale of CPM.

V. CONCLUSION AND FUTURE WORKS

We have presented Feature Stroke, a new approach to group based text entry using joysticks. The 26 characters are associated with 8 primitive gestures of joysticks, according to their visual features of intuitive writing strokes. Another contribution of this paper is that we introduced a strategy of ambiguous mnemonics, to visually present the uncertain states while typing in letter groups.

The usability tests have shown that FS generally improves the input performance. The learning curve of FS goes up steeply. Questionnaires show that the participants' satisfaction after the fifth session of test apparently increases

compared with that of the first session. More future user tests may reveal what and how the many factors affect users' performance and experience.

ACKNOWLEDGMENT

We thank all the undergraduates participating in the user test.

REFERENCES

- [1] J. O. Wobbrock *et al.*, "Integrated text entry from power wheelchairs," *Behaviour and Information Technology*, vol. 24, no. 3, pp. 187-203, 2005.
- [2] I. S. MacKenzie, S. X. Zhang, and R. W. Soukoreff, "Text entry using soft keyboards," *Behaviour and Information Technology*, vol. 18, no. 4, pp. 235-244, 1999.
- [3] K. Ehrlich, "A conversation with Ted Selker," *Interactions*, vol. 4, no. 5, p. 34-47, Sept.-Oct., 1997.
- [4] I. S. MacKenzie, "KSPC (keystrokes per character) as a characteristic of text entry techniques," in *Human Computer Interaction with Mobile Devices*, Springer, pp. 195-210, 2002.
- [5] J. Gong and P. Tarasewich, "Testing predictive text entry methods with constrained keypad designs," in *Proc. Human Computer Interfaces International (HCII 05)*, 2005.
- [6] J. Gong and P. Tarasewich, "Alphabetically constrained keypad designs for text entry on mobile devices," in *Proc. the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2005.
- [7] J. Johnson, "We perceive what we expect," in *Designing with the Mind in Mind*, J. Johnson, Ed., Morgan Kaufmann, Boston, ch.1, pp. 1-9, 2010.
- [8] R. W. Soukoreff and I. S. MacKenzie, "Metrics for text entry research: an evaluation of MSD and KSPC, and a new unified error metric," in *Proc. the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2003.



Zhenyu Gu is an associate professor, and the director of the Interaction Design Lab in the School of Media & Design of the Shanghai Jiao Tong University. He received B.E in industrial design in 1994 from the Southern Yangtze University in the People's Republic of China. In 2006 he obtained his Ph.D from the Department of Design in the Hong Kong Polytechnic University. From 1994 to 2008 he worked as a lecturer teaching design sketch and Computer Aided Industrial Design in the Southern Yangtze University. From 2000 to 2006 he worked as a researcher in the area of developing interactive and intelligent design computing systems. In 2008 he joined the School of Media & Design of the Shanghai Jiao Tong University. His research interests include interface design, knowledge discovering, generative and evolutionary design, data driven aesthetic computing. Dr. Gu is the executive director of the Information and Interaction Design Education Committee of the Design Society of China.