

# **ONE-HANDED MOBILE TEXT ENTRY**

## *Evaluation of five-key text entry techniques*

Frode Eika Sandnes

*Faculty of Engineering, Oslo University College, P.O. Box 4, St. Olavs plass, N-0130 Oslo, Norway*

**Abstract:** This paper addresses text entry techniques for miniature or wearable devices. Five keys are used for interaction control – one key for each finger. Three approaches are examined: one dictionary-independent technique (multi-tap), a partially dictionary-dependent technique (tree-based) and a dictionary dependent technique (one-stroke). The study shows that users can type text at rates of up to 30 characters-per-minute after just 5 minutes of practice using the one-stroke approach. Although the proposed methods are slower than chord-based techniques they do not require the same amount of training that is required to acquire and maintain chording skills. Consequently, the proposed strategies are suitable for users that only type texts on a small device occasionally.

**Key words:** Mobile text entry; KSPC; chording; wearable computing; miniature devices.

## **1. MOBILE TEXT ENTRY**

The limited surface area offered by small devices only provides space for a few keys. There is also a trade-off between the number of keys and the size of the keys. Smaller keys are harder to use than larger keys and result in higher error rates as incorrect keys are more easily pressed accidentally.

Dictionary based text-entry techniques (King, 1995; Kreifeldt, 1989; Smith, 1971) have won wide commercial success due to their ease of use and fast text entry speeds. The market is dominated by the T9 system patented by Tegic, but other similar systems exist such as iTap and eZiText. Dictionary based text-entry allows text to be entered using close to one keystroke per character on phone keyboards.

Two and three key text entry systems were deployed on arcade game machines from the 70s. The user would pull the joystick to the left or right to cycle the alphabet (wheel-of-letters), or use rotator keys and press a select button to select the desired character when entering their names on high score lists. MacKenzie et al. (MacKenzie, 2002) has found that such text entry systems on average require 10.66 key-strokes per character for normal English text. In a different approach (Raghunath and Narayanaswami, 2002) the alphabet was split into two rings. The user could toggle between the two rings and then rotate the rings to obtain the desired character, i.e. one key is used to toggle between the two rings, one key is used to cycle forward in the rings and the final key is used to select a letter.

## 2. 5-KEY TEXT ENTRY

Five key text entry has been around for over 100 years. The first five key text entry techniques were chord based keyboards originally used for mail-sorting applications (see the excellent survey of chord based text entry by Noyes (Noyes, 1983). Several studies addressing various aspects of five-key chord keyboards exist (Lehikoinen and Roykee, 2001; Rosenberg and Slater, 1999; Seibel, 1962; Kirchenbaum et al., 1986) and commercial chord keyboards include the Microwriter. Five-keys have also been proposed for use in stenograph typewriters (Beddoes and Hu, 1994).

The five keys allow 31 unique keystroke-patterns, or chords, to be entered ( $2^5-1$ ) and thus allow the entire alphabet and more to be addressed. Although allowing for fast text entry rates, such systems have a high learning threshold, although (Gopher and Raj, 1985) claim that chording in principle is easier to learn than QWERTY touch-typing. The reason for this is probably the wide availability and familiarity users have with the QWERTY keyboard, and that most typists never have seen or tried a chord keyboard.

When Douglas Carl Engelbart invented the mouse in the 60'ies he originally visualized the mouse with five-keys allowing the users to type text using chords (Barnes, 1997). Although this particular detail of Engelbart's vision did not catch on, the idea of using five-keys is worthy of investigation due to the anatomy of the human body. Several strategies has been proposed for one-handed text-entry such as the ingenious half QWERTY keyboard (Matias et al., 1993). The principle of the half QWERTY keyboard is to transfer two-handed QWERTY typing skills using one hand on a special half keyboard. A totally different approach taken by (Isokoski and Raisamo, 2000) is to enter text in graphical patterns controlled by device independent north, south, east and west movements (i.e. four keys, mouse or joystick

directions) to mimic the graphical shape one would handwrite a character. Finally, a class of five-key text-entry systems can be found on computer game consoles, for example the Nintendo Advance. Characters are organised into a two-dimensional wraparound mesh. This mesh is navigated by using four navigation keys (left, right, up and down) and select. This class of text entry systems are easy to learn and can be used with very little training.

### 3. METHOD #1: MULTITAP

The multitap approach is probably the simplest five-key text entry technique and is analogous to the multitap technique of the mobile phones. The alphabet is split into four groups and each group is mapped to one key. For example: 1: ABCDEF, 2: GJKLM, 3: MNOPQR, 4: STUVWXYZ, 5: [break-key] To enter a letter the user selects the key with the respective letter assigned and repeatedly taps the key until the desired letter is selected. Then the next character is entered and so forth. If two consecutive letters are from the same group the first letter is chosen pressing the break-key. For example, the expression “car\_” is entered using 12 keystrokes. In general, the minimum number of keystrokes per character required to enter a character is  $K_{a,min}=1$ , the maximum number of keystrokes per character is  $K_{a,max} = 9$ , and the mean number of keystrokes per character for English  $K_a = 3.47$ . Note that the  $a$  subscript is used to denote the key entry component of the text entry procedure.

### 4. METHOD #2: TREE-BASED PARTIAL DICTIONARY TECHNIQUE

The tree-based technique is based on organising the characters of the alphabet into a five-way two-level tree, where characters are entered in two steps. In the first step the main category is selected and in the second step the desired character for the category is selected. The advantage of this technique is that usually  $K_a$  is constant, thus  $K_{a,min}=K_{a,max}=K_a=2$ . However, the problem is that such a tree only allows 25 leaf nodes ( $5^2$ ) and there are 26 characters in the English alphabet. In addition a leaf node is required for the space character. The solution is to cluster less frequently needed characters with other characters. For example, by clustering  $p$  and  $q$  to the same leaf node and  $y$  and  $z$  to another leaf node we end up with exactly 26 leaf nodes including space. And words containing the clustered groups are resolved by the means of a dictionary. For example, the word “zen” is entered as follows:

```

0 (5) [abcde fghij klmno pq-rstu vwx-yz_]
1 (4) [..... .. vwx-yz_]
2 (1) [abcde fghij klmno pq-rstu vwx-yz_]y
3 (5) [abcde ..... ..]y
4 (3) [abcde fghij klmno pq-rstu vwx-yz_]ye
5 (4) [..... .. klmno ..... ..]ye
6 (5) [abcde fghij klmno pq-rstu vwx-yz_]yen
7 (5) [..... .. vwx-yz_]yen
8 (?) [abcde fghij klmno pq-rstu vwx-yz_]zen_

```

At the first step the fifth group is selected by pressing the fifth key, then the yz-group is selected by pressing the fourth key (it is the fourth subgroup within the group). Then, the first group is selected and the e character is selected from the group by pressing the fifth key. At step four the third group is selected by pressing key tree, and the fourth character in the group is selected by pressing the fourth key. Finally, space is retrieved by selecting the fifth character from the fifth group. The resulting word is either "yen" or "zen", but "yen" is not in the dictionary and "zen" is selected. Thus, a total of 8 keystrokes are required to press the 4 character expression "zen" + space.

With the current configuration there will only be a small number of ambiguities for all valid sequences containing combined characters for English, i.e.  $K_b \approx 0$ . Note that the  $b$  subscript is used to denote the ambiguity resolution component of the text entry procedure. Another advantage of the approach is that only a small subset of the English dictionary is required, i.e. only the words containing the characters "p", "q", "y" or "z", or approximately 15,305 words (33.7% from a total of 45,392) which includes all grammatical variations.

## 5. METHOD #3: ONE-STROKE DICTIONARY BASED

The idea behind the one-stroke approach is to assign several characters to the same key, enter a character by pressing the assigned key and resolve ambiguities by employing a dictionary. By brute force search the optimal partitioning scheme reveals the following character to key assignment. Namely, 1: ABCDE, 2: FGHIJKLM, 3: NOPQR, 4: STUVWXYZ, 5: space character.

For example, assume the user wants to enter the word "approximately". The user first presses key one since the letter "a" is in the first group. The next four letters "ppro" are all in the third group and

the third group is pressed four times. Next the “x” is in the fourth group, selected with the fourth key and the “im” characters are both found in the second group. Next, the characters “ately” are in the first, fourth, first, second and fourth group respectively, activated with the corresponding keys. Finally, the space is retrieved by pressing the fifth key. The 15 character expression “approximately” + space is unambiguous and retrieved with just 15 keystrokes.

For the one-stroke approach  $K_b=0.52$  and  $K_{b,max}=1.33$  respectively. Other interesting statistics regarding this technique is that there are 3.7 ambiguities on average and a maximum of 32 ambiguities.

Table 1. Results of the typing experiment. ikd = inter keystroke delay

Subject	Measure	MultiTap	Tree-based	One-stroke
Subject #1	Mean ikd	1.0	2.3	2.1
	Median ikd	0.5	1.4	1.2
	Mean chars/min	22.5	13.0	28.5
Subject #2	Mean ikd	0.76	1.62	1.97
	Median ikd	0.52	1.02	1.94
	Mean chars/min	27.2	18.6	31.1
Subject #3	Mean ikd	0.73	3.93	1.44
	Median ikd	0.24	2.13	0.55
	Mean chars/min	26.5	7.7	26.2

## 6. EXPERIMENT

The three text entry techniques were implemented as part of a modular custom Java application framework. The applets were configured to accept key events from “a”, “s”, “d”, “f” and “\_” (space) keys representing the keys 1, 2, 3, 4 and 5 respectively. This configuration allows the user to touch type using the left hand.

Three subjects participated in the experiment. All the subjects were computer literate and capable of conventional QWERTY touch-typing at high rates.

Each participant was e-mailed a link to the experiment website. The tests were presented as a seven-step procedure. The entry page provided an overview of the experiment. The second page allowed the subject to practice the multi-tap experiment for five minutes, and the third page was dedicated to the typing experiment using the multi-tap technique with a duration of 15 minutes. A few lines of text to be typed were displayed at the bottom of the screen. Only a few lines are sufficient for such a short typing experiment as untrained typists are unable to type more. Pages four, five, six and seven were dedicated to practice and typing experiments for the tree-based

technique and the one-stroke approach and the final page comprised a short questionnaire allowing the subjects to explicitly voice their opinions regarding the three methods.

*Table 2.* Results of the questionnaire. MT = multi-tap, TB = tree-based, OS = one-stroke.

Subject	absolute			relative		
	MT	TB	OS	MT/TB	MT/OS	TB/OS
Subject #1	3	0	4	0	4	5
Subject #2	1	5	4	5	4	1
Subject #3	4	0	3	0	2	5

## 7. RESULTS

Tables 1 and 2 list the results of the experiment. Table 1 summarises the measurements of the typing experiment. The columns list the subject, the type of measurement and measurements obtained using the multi-tap, tree-based and one-stroke methods respectively. Three measurements are provided, the mean inter keystroke delay, i.e. the time between consecutive keystrokes, the median of inter keystroke delay (robust to outliers) and the characters typed per minute (including errors).

## 8. DISCUSSION

Clearly, Table 1 shows that the one-stroke method results in the highest typing rates of 28.5, 31.1 and 26.2 characters-per-minutes for the three subjects, followed by the multi-tap method and finally the tree-based strategy.

The multi-tap method requires the least cognitive processing and is thus the easiest to use and the tree-based method requires the most cognitive processing and is the hardest to use. This is evident from inspecting both the mean and median inter-keystroke delay. For subject 1 the median inter-keystroke delay is 0.5 seconds while it is 1.2 seconds for the one-stroke method and as much as 1.4 for the tree-based method. Similar patterns can be seen for subject and 3, which has inter-keystroke delays of 0.24, 0.55 and 2.13 for the multi-tap, one-stroke and tree-based methods respectively. The results for subject 2 varies slightly as the inter-keystroke delay for the tree based method is shorter (1.02) for the tree-based than the one-stroke method (1.94) – yet the multi-tap method yields the shortest inter-keystroke delay of 0.52.

The measurements are also relatively consistent with the questionnaire assessments which are summarized in Table 2. All the subjects rate the one-stroke method highly, while only subjects 1 and 3 rate the multi-tap method highly. Instead, subject 2 rate the tree based method highly. When comparing methods against each other then subjects 1 and 3 prefer multi-tap over the tree-based method, while subject 2 prefers the tree-based method to multi-tap. Subjects 1 and 2 prefer the one-stroke method over the multi-tap method while subject 3 prefer the multi-tap method. Finally, subjects 1 and 3 strongly prefer the one-stroke method to the tree-based method, while subject 2 prefers the tree-based method. Clearly, the preferences of subject 2 differ from those of subjects 1 and 2 and this is also visible in the measurement data in Table 1 where subject 2 has a much shorter inter-keystroke delay for the tree-based method than the other two subjects. Perception of preference is somewhat linked to productivity and success using the methods, and subject 2 had an unusual skill using the tree-based method despite its cognitive complexity.

One of the subjects commented in the response e-mail that the multi-tap method was easy to use, but that one loses patience quickly using it. Further, the subject reported that the one-stroke method feels fast, however effort is required to look through the alternatives when typing an ambiguous word.

Clearly, the experiments show that although the multi-tap strategy has the highest KSPC it is still better than the tree-based method, which has a much lower KSPC – the main reason being the huge difference in cognitive difference between the two methods. The one-stroke has the lowest KSPC and also yields the highest keystrokes per character, but is still more cognitively difficult than the multi-tap method.

Clearly, all the methods discussed in this paper are much slower than chording techniques such as the chording glove (Rosenberg and Slater, 1999). In comparison, the chording glove allows users to type approximately 50 characters-per-minute after 80 minutes of practice and 95 characters-per-minute after 10 hours of practice. However, (Rosenberg and Slater, 1999) reports that it takes approximately 45 minutes to memorise the chords. Research also shows that such typing skills quickly deteriorate without practice and chording is therefore most suitable for frequent users of the technology. However, two of the strategies discussed in this paper can be used with acceptable typing speeds just after a few minutes of practice. The proposed strategies are therefore suitable for occasional users.

## 9. SUMMARY

Techniques for dictionary enhanced text input on miniature mobile devices with five keys are proposed, allowing text to be entered using one hand. Three methods were investigated – multi-tap, tree-based and one-stroke. The one-stroke method resulted in the highest typing rates of approximately 30 characters-per-minute and a theoretical KSPC of 1.52. The multi-tap method was measured to require the least cognitive load, while the tree-based method was found to be the least productive typing strategy with the highest cognitive difficulty. All the techniques allowed the subjects to type text after just five minutes of training, which is much less training time than what is required for chord keyboards.

## REFERENCES

- Barnes, S. B., 1997, Douglas Carl Engelbart: Developing the underlying concepts for contemporary computing. *IEEE Annals of the History of Computing*, 19(3):16-26.
- Beddoes, M. P. and Hu, Z., 1994, A chord stenograph keyboard: a possible solution to the learning problem in stenography. *IEEE Trans. on Systems, Man and Cybernetics*, 24(7):953-960.
- Bellman, T. and MacKenzie, I. S., 1998, A probabilistic character layout strategy for mobile text entry. In *Proceedings of Graphics Interface '98*, pp. 168-176. Toronto: Canadian Information Processing Society.
- Gopher, E. and Raij, D., 1985, Typing with a two-hand chord keyboard: Will the qwerty become obsolete? *IEEE Trans. on Systems, Man and Cybernetics*, 18(4):601-609.
- Isokoski, P. and Raisamo, R., 2000, Device independent text input: A rationale and an example. In *Proceedings of the Working Conference on Advanced Visual Interfaces AVI2000*, pages 76-83, Palermo, Italy, ACM.
- King, M. T., 1995, Justtype. tm. – efficient communication with eight keys. In *Proceedings of the RESNA 95 Annual conference*, Vancouver, BC, Canada.
- Kirchenbaum, A., Friedman, Z., and Melnik, A., 1986, Performance of disable people on a chordic keyboard. *Human Factors*, 28(2):187-194.
- Kreifeldt, J. G., 1989, Reduced keyboard designs using disambiguation. In *Proceedings of the human factors society 33rd annual meeting*.
- Lehikoinen, J. and Roykee, M., 2001, N-fingers: A finger-based interaction technique for wearable computers. *Interacting with Computers*, 13:601-625.
- MacKenzie, I. S., 2002, Mobile text entry using three keys. In *ACM proceedings of NordCHI02*, pages pp.27-34.
- Matias, A. E., MacKenzie, I. S., and Buxton, W., 1993, Half-qwerty: a one-handed keyboard facilitating skill transfer from qwerty. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM press, pp.88 - 94.
- Noyes, J., 1983, Chord keyboards. *Appl. Ergonomics*, 14(1):55-69.
- Raghunath, M. T. and Narayanaswami, C., 2002, User interfaces for applications on a wrist watch. *Personal and Ubiquitous Computing*, 6:17-30.



- Rosenberg, R. and Slater, M., 1999, The chording glove: A glove-based text input device. *IEEE Trans. on Systems, Man and Cybernetics*, **29**(2):186-191.
- Seibel, R., 1962, Performance on a five-finger chord keyboard. *J. Appl. Psychol.*, **46**(3):165-169.
- Smith, S. L., 1971, Alphabetic data entry via the touch-tone pad: a comment. *Human factors*, **13**(2):189-190.