PROCEEDINGS of the HUMAN FACTORS SOCIETY 33rd ANNUAL MEETING—1989

REDUCED KEYBOARD DESIGNS USING DISAMBIGUATION

J. G. Kreifeldt, Ph.D.
S. L. Levine, Ph.D.
Department of Engineering Design
Tufts University
Medford, MA 02155

C. Iyengar, M.Des., M.S.
Pitney Bowes
Stamford, CT

ABSTRACT

There are a number of important applications where the accuracy of typed copy is less important than other criteria, such as the speed of entry, ease of learning, reduced keyboard size, provision for innovative keyboard designs and work methods, etc. Some of these applications include communication devices for the motor handicapped and rough drafting. The applications in question still provide single finger typing as opposed to multifinger entries for chord keyboards.

One approach to keyboard design for such applications involves using a multicharacter key set. Because there is more than one character on a key, the number of keys can be reduced to a point limited only by the required accuracy of the text produced which is itself limited by the ability of "disambiguation" algorithms to decode the ambiguous text strings. Reducing the number of keys increases text entry rates and reduces learning time.

Experiments with keysets of 10, 12, 14, and 16 keys have produced keystroke rates of 3.3 to 1.7 characters per second and a 6% keying error while the disambiguation programs can produce an accuracy of 85 to 95%. Learning time in the experiments was about 9 to 12 minutes.

The experiments demonstrate the feasibility of increasing text entry rates and freeing the keyboard from architectural constraints.

INTRODUCTION

As personal computers become increasingly common, more users are becoming typists who otherwise might hardly have had so much exposure to keyboards. Many of these will not become or even desire to become "expert typists" or touch typists and will always be in a state of "learning" or will reach a rather low proficiency level that will not materially change. Likewise there is often the need for less than perfect copy as when rough drafts are being made, or when the material will later be retyped by a professional. There are also special applications for populations such as the motor handicapped for which "special" keyboard designs are indicated.

However, there are a number of important criteria in typing such as typing speed, learning time, fatigue, etc., which are pertinent almost regardless of the application or user. Consequently, there have been many studies made of typing and keyboard designs to improve one or more of these criteria. A brief review of some of these findings is summarized here.

Character Assignments to Keys

Computer keyboards are directly adapted from the typewriter keyboard invented by the Sholes brothers in 1873 which has, after modifications, come to be called the QWERTY keyboard because of its letter arrangement on the first row. Because of mechanical limitations of the original mechanisms, the letters were assigned to the keys so as to prevent adjacent strikers from jamming together by locating frequently occurring letter pairs (digrams) as far apart on the keyboard as possible. Thus the QWERTY design was a deliberate attempt to slow the typist down.

Even after typing machines no longer relied on the early mechanical mechanisms, the QWERTY persisted because of the system investment already made in it even though other layouts such as the DVORAK (1936) have been studied as improvements particularly for speed. The DVORAK has been found to permit a 5 to 10% faster typing speed (Gentner, 1984). Alphabetical layouts have also been proposed on the hypothesis that they "make more sense" so should be faster to learn and permit higher speed as well. However, Norman and Fisher (1982) found that people do not know the alphabet as well as it was assumed and the alphabetical layout involves a mental as well as a visual search compared to the purely visual search for "random" arrangements such as QWERTY or DVORAK layouts.

A recent attempt to optimize the assignment of single characters to single keys is reported by Noel & McDonald (1989). They claim a 11% improvement over the Sholes layout and a 1.2% improvement over the Dvorak layout. Interestingly, they used an optimization technique called "anealing" which appears similar to the programming technique used for this paper.

Architectures

Duncan and Ferguson (1974) found that fatigue and pain are essentially due to the combined and interfering effects of the static muscular work required to hold the upper and lower arms and palms pronated close to their anatomical limits and the dynamic muscular work required to move the fingers. Klockenberg (1926), relying on biomechanical studies, proposed a major change in the keyboard architecture by splitting it into two identical halves so as to distribute the keys equally between hands. Kroemer (1972) developed this idea further (the "K - Keyboard"). In this design the two halves are separated and turned inward at an angle mimicking the natural inward posture of the forearm at typing. Although no significant differences compared to standard designs were found, the K-keyboard permitted a longer period of repetitive typing before fatigue set in. Nakaseko et al (1985) designed other split keyboard architectures based on anatomical studies, user preferences and self reports.

Much keyboard research has been addressed to the single-key single-character (Ks€s) concept and has attempted to optimize certain criteria via key location, size, pressure, geometry, etc.

Multikey - Single Character (KmCs) Keyboards

Another approach to increasing transcription speed by reducing the number of keys is to assign a combination of keys (a chord) to a single character or symbolic unit. Chord keyboards in which multiple keys must be simultaneously depressed for a character or other symbolic representations have been in common use by court stenographers to enter a type of shorthand rather than simply spelling. Conrad & Longman (1965) compared a particular chord keyboard for single character (alphabetic) typing with a standard one and found the chord keyboard to support a slightly higher learning rate but also a somewhat higher error rate. Rochester et al (1978) at IBM designed and studied a one-handed, four finger (the little finger is excluded), alphabetic chord keyboard for single character typing. In this design the finger(s) pushes either one key or pushes on the border of two or more keys (the chord) to transcribe a character. Their studies indicate that this keyboard has the same learning time as the standard one for typing speeds up to about 40 wpm. However, it offers the advantages of one handed typing, small size, ability to handle large alphabets and ease of use by the handicapped.

Military and aerospace requirements for limited panel space, one handed operation and low error tolerance have also spurred innovative designs. In these cases, the key chords represent various special purpose codes including the alphabet and numerals. (Butterbaugh and Rockwell, 1982).

Single Key - Multicharacter (KsCm) Keyboards

In addition to single key - single character and multikey - single character, the single key-multicharacter is a concept which has been somewhat explored and which is the basis for the studies reported here. The telephone keyboard is a common example of this. Such keyboards remain "one finger at a time" designs. (The multicharacter key differs from the multi-function key such as upper/lower case, etc.)

The basic idea behind such assignments is to reduce the total number of keys, thus reducing finger travel distances (and thus travel time) but without requiring the user to learn codes as in a chord keyboard. Because the total keying time can be considered as the sum of the decision time (T1) + travel time (T2) + strike time (T3) a keyboard in which a finger never needs to move from its position effectively eliminates T2 thus shortening the total time. Because there are fewer keys, the size of the keyboard can also be reduced and or each key can be made larger. Multicharacter keys have the obvious requirement for somehow "unscrambling" the entered text since it is ambiguous.

Levine et al (1986) have made extensive studies of multi-character keys as a means to increase text entry rates for disabled users. The resulting ambiguous text is disambiguated by computer programs. A number of "disambiguation" techniques are possible based upon common statistical characteristics of the English language. One technique involves two parallel approaches to producing readable text from the keystroke sequences: (1) dictionary search for matching possible words from the entries, (2) An analysis of trigram (sequences of three characters) frequencies and selection of sequences with the highest probability of occurrence.

Disambiguation can be considerably enhanced if the inherent ambiguity is reduced by decreasing the number of characters/key and/or through an "optimal" assignment of characters to keys. One such assignment involves allocating one key to just the "space."

The work reported here was directed towards developing a multicharacter keyboard and based on disambiguation programs for the frequent yet untrained typist.

RESEARCH

A human factors engineering approach was adopted consisting of: (1) determining the effect of the number of keys on text entry rate, (2) determining which characters should be assigned to which keys, and (3) determining the resulting key to finger assignment. The number of keys actually used must be an optimization between the text entry rate (the fewer the keys, the faster the rate) and the disambiguation accuracy (the more keys the higher the accuracy). The third consideration requires making the assignment of key (characters) to finger based on finger tapping rates and strength.

Keys and Entry Rates

Assuming that the keys are stroked in a more or less random fashion, Hick's law suggests that the stroke time per key will be proportional to the entropy of the key set and the entropy cannot be larger than the logarithm of the number of keys. Overall, this hypothesizes that the fewer the keys, the faster the entry rate will be. This is intuitively obvious considering that if all the characters are on one key, the "entry" time will be much faster than if there are (26) keys. The difference in time would be largely due to the difference in decision time per key/finger.

In order to verify this hypothesis and determine the actual entry rate values, an empirical study was made on a simulated 10, 12, 14, and 16 key keyboard. The keyboards were made by functionally masking off all but the test number of keys on a standard QWERTY computer keyboard and programing the CRT to show which key should be pressed by a position mimic of the keys on the screen.

Sixteen males and females, ranging from very slow hunt and peck to touch typists were subjects. Key sequences of 2, 3, or 4 keys to be typed would light up randomly out of the whole keyset and the subject typed the same keys in the same sequence. The sequences were made to simulate the proportion of 2, 3, and 4 letter words in English. The choice of each displayed key was randomly determined and the same key could occur more than once in a sequence simulating the multi-character nature of the keys. Each subject typed about 100 keystrokes with each keyboard. Subjects practiced with each keyboard until they felt confident and comfortable with it.

RESULTS

A number of measures were recorded including: (1) Motor Time defined as the temporal difference between the first and last keystroke of a sequence. This represents actual typing time; (2) Decision time defined as the temporal difference between the start of the display and the first keystroke; (3) Total time defined as the temporal difference between the start of the display and the last keystroke. This is the sum of the decision and motor times. Since the keys were programmed to conform to the natural position of the fingers, motor time as expected was negligible, which is part of the rationale for reduced key keyboards; (4) Errors for each key.

Time and Errors

Figure 1 shows the general increase in time taken per keystroke as the number of keys in the keyboard increased from 10 to 16.

All times increase as the number of keys in the keyboard increases and the decision time constitutes the major portion of the total time. However, it appears that the decision time is slowly increasing while the motor time appears to be somewhat accelerating so that for a 26 key keyboard (or a dedicated character keyboard) the motor time may become an increasingly larger portion of the total. The general trends and differences are statistically significant at the 0.05

level. There was a 5 to 8% error for each keyboard without any statistically supported trend although the errors appear to increase with the number of keys. (A large error variability reflected the wide differences in subject skill level.) Only the 12 key keyboard showed a statistically significant difference in key errors. Otherwise all keys showed essentially the same error rate for the other 3 keyboards.

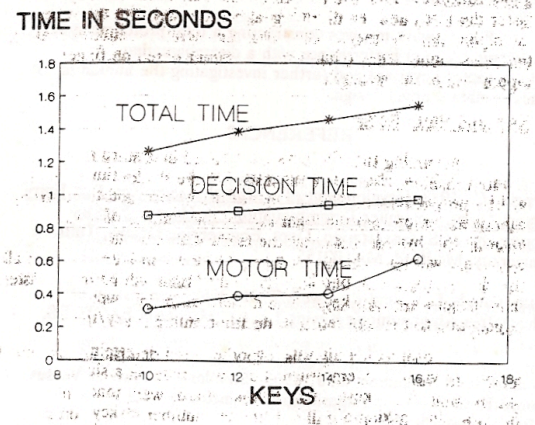

Figure 1 Increasing Keys Increases Basic Times per Keystroke

Although more keys were not used, it appears that the total time per key increases as more keys are added. Thus a multicharacter keyboard should permit faster entry than the standard keyboard once it is learned. Although learning studies were not made, it can be conjectured that this presents no particular problem for the following reasons. On a standard keyboard, the user must learn the particular character on a key and its position on the board and in the touch method must also learn the keys for each finger. In a multicharacter keyboard there are fewer keys per finger (only one key per finger in the ten key board) and fewer key locations to learn. The subjects in this experiment took on the order of 9-12 minutes to feel ready to begin the test. Also, it is known that proficiency in touch typing is achievable with practice.

DISAMBIGUATION AND OPTIMIZING THE LAYOUT

Thus, it is clear that decreasing the number of keys considerably increases the entry rate. However, key reduction also increases the characters on a key and hence increases the difficulty of disambiguation as shown in Figure 2. Disambiguation programs attempt to assign characters to keys so as to produce the highest disambiguation accuracy. But once a number of keys is determined, there remains the problem of making the key (character groups) assignment to the fingers in some optimal fashion.

A Disambiguated Keyboard

We used evolutionary programming as a type of optimization technique for assigning character to keys to produce the most accurate "decoding" or disambiguation of the ambiguous text produced by striking the keys. The programs assume that standard English is being entered and take advantage of the statistical frequencies of individual letters and letter pairs (digrams). For the resulting character assignments, the accuracy of "translation" of keystrokes is about 85% for a ten keyboard and close to 95% for 16 keys. (An interesting unexplored question is what level of accuracy is acceptable in most common typing tasks and what the actual levels are. It is a basic presumption here that as in most common tasks, the common typist will accept or not notice some accuracy level less than 100%).

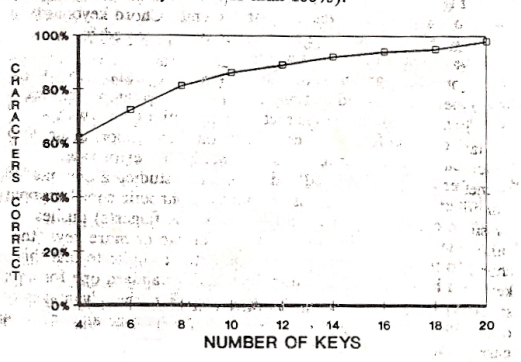

Figure 2 Disambiguation Accuracy Increases as the Number of Keys Increase

Once a keyboard and character assignment is made, it is necessary to optimize the assignment of keys to fingers in order to produce the fastest, least fatiguing text entry. Haaland (1963) and Dvorak (1936) have shown that finger strengths and finger tapping rates essentially follow the natural ordering of the digits (little finger weakest) and the dominant hand is slightly superior to the nondominant one. These findings are among the major variables affecting keying performance and multicharacter keyboards permit optimization of key to finger.

Based on fatigue studies it is reasonable to assign the most frequently used keys to the strongest, least fatigue prone finger. This is illustrated for the case of a 12 key keyboard which was chosen as a representative sample. The "optimal" assignment of characters to key (maximum of 3) is:

KEY	CHARACTER	KEY	CHARACTER
1.	(Space)	7.	. G L
2.	U D X	8.	S
3.	J T	9.	I V
4.	M Z A	10.	B R K
5.	N H	11.	O C
6.	E W Q	12.	Y P F

When keys on this keyboard are struck, the disambiguation program "decodes" the character string into the most likely English equivalent using dictionary search and linguistic statistics.

Once a key set is determined it can be optimized by determining how often each key would be struck in common English and assigning the most frequent keys to the "best" fingers.

Layout Optimization

A measure of finger efficiency was made as the product (Tapping Rate x Pushing Force) based on studies by Dvorak and Haaland cited above. Thus each finger gets an "efficiency" rating (the higher the rating the better the finger) and each key gets a frequency rating. Keys and fingers are matched until the summed products of (FINGER EFFICIENCY x KEY FREQUENCY) is as high as possible. The result is given below in Figure 3 for character assignment of the 12 key keyboard above.

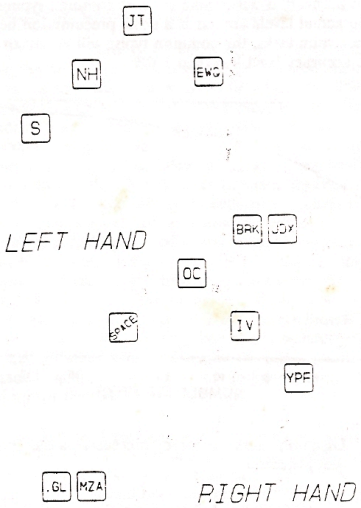

Figure 3 An Optimized Assignment of Characters to Keys and Keys to Fingers

The assignment can now be geometrically configured following Kromer or Nakaseko for the best anatomical positions of the hand/wrist/arm, etc.

DISCUSSION

These studies have demonstrated the procedure for producing an optimized multicharacter keyboard predicted on the ability of computer programs to disambiguate or decode the ambiguous resulting text to some acceptable level of accuracy less than 100%. The purpose was to produce a keyboard design which would allow a considerably higher entry rate than the QWERTY by reducing the number of keys and which would also be ergonomically designed.

Because disambiguation accuracy declines while entry rate increases in this approach, there must obviously be some global system performance measure which determines how many keys to use on the keyboard even though it may be strongly weighted in favor of accuracy.

However, in many applications, 100% accuracy is not required because the overall redundancy of English permits a reader to ---- in missing words or to corect uncorrected ones. Such text would be called "readable." It remains to investigate acceptable and actual levels of accuracy in common typing tasks. Acceptable levels may also be strongly conditioned by the necessity to produce "readable text" rapidly (get it to the finish typist as soon as possible) or with the minimum amount of effort as perhaps important to the motor impaired.

Further development can also be made in disambiguation programs which (e.g.) use trigram (sequences of three characters) and higher n-gram frequencies in English in addition to the diagrams as used here and other linguistic statistics. In fact there is extensive literature addressed to cryptanalysis which has hardly been explored for multicharacter key use and which can be expected to improve disambiguation accuracy considerably while faster computers and computational techniques can also reduce the disambiguation time.

Of course typing with a multicharacter system would be somewhat different visually as well since some previously typed characters would change as new ones were entered (disambiguation at work) or the display could be delayed until a word was entered and disambiguated. Many typists only look occasionally at the screen while typing. An intriguing design concept for a 10 key keyboard is to attach each key to each finger like an electrical thimble so that the hands are free to be placed wherever desired or comfortable. Simply tapping the fingers would constitute the typing.

Future work involves constructing a total prototype system to determine actual performance with a disambiguating multicharacter keyboard and further investigating the human factor issues in such a novel design.

REFERENCES

Iyengar, C., 1988, Development of a Multicharacter Key Text Entry System Using Computer Disambiguation: A Human Factors Approach. M.S. Dissertation, Dept. Engineering Design, Tufts University, Medford, MA, 02155

Butterbaugh, L. C. and Rockwell, T. H., 1982, "Evaluation of Alternative Alphanumeric Keying Logics," HUMAN FACTORS, 24,5.

Conrad, R. & Longman, D. J., 1965, "Standard Typewriter Versus Chord Keyboard - An Experimental Comparison," ERGONOMICS, 8,1, 77 - 88.

Duncan, J. & Ferguson, D., 1974, "Keyboard Design and Operating Posture,", ERGONOMICS, 17,6, 731-744.

Duncan, J. & Ferguson, D., 1974, "Keyboard Operating Posture and Symptoms in Operating," ERGONOMICS, 17,5 651-662.

Dvorak, A., Merrick, N. L., Dealey, W. L., Ford, J. C., 1936, Typewriting Behavior, New York, American Book Co.

Gentner, D. R., Norman, D. A., 1984, "The Typists Touch," PSYCHOLOGY TODAY, 4, 66-72.

Grandjean, E., Nakaseko, M., Hunting, W., & Gierer, R. M., 1985, "Studies on Ergonomically Designed Alphanumeric Keyboards," HUMAN FACTORS, 27, 2, 175-187.

Haarland, J., Wingert, J., & Olson, B. A., 1963, Force Required to Actuate Switches, Maximum Finger Pushing Force, and Coefficient of Friction of Mercury Gloves, Honeywell Memorandum Feb. 22.

Kroemer, K. H. E., 1972, "Human Engineering the Keyboard," HUMAN FACTORS, 14, 1, 51-63.

Levine, S. H., Minneman, S. L., Getschow, C. D., & Goodenough - Trepanier, C., 1986, "Computer Disambiguation of Multicharacter Key Text Entry: An Adaptive Design Approach," IEEE Proceedings, 298-301.

Norman, D. A., & Fischer, D., 1982, "Why Alphabetic Keyboards Are Not Easy to Use: Keyboard Layout Doesn't Matter Much," HUMAN FACTORS, 24,5,509-519.

Noel, R. W., McDonald, J. E., 1989, "Automating the Search for Good Designs: About the Use of Simulated Anealing and User Models," Proc. INTERFACE 89, Human Factors Society, 241-246.

Rochester, N., Bequaert, F. C., & Sharp, E. M., 1978, "The Chord Key-board," IEEE proceedings, (Computer), 57-63.