LEDBoard: Using visual feedback to support text entry with physical keyboards

Andreas Komninos[1][0000−0001−8331−1645], Vassilios Kavvathas[1], and Ioulia Simou[1]

Computer Engineering and Informatics Dept., University of Patras, Greece
email{akomninos,vkavvathas,simo}@ceid.upatras.gr

Abstract. Physical keyboards persist as one of the most common input devices for personal computers. Low familiarity with the keyboard translates to frustration during text entry, as the user constantly shifts their attention between the keyboard and the screen, in order to locate the next key to press, and to inspect input for errors. To decrease the need for attention shifts, we present physical keyboard typists with visual feedback, within their field of view, using an RGB LED strip to indicate spelling errors in different colours. We conducted a user experiment with 36 participants. Users' performance was evaluated a) without visual feedback, b) showing feedback with a LED strip on the keyboard, and c) showing feedback with a LED strip at the bottom of the screen. We find that our prototype improves corrective action behaviour for slow typists and reduces screen-glancing behaviour for fast typists.

Keywords: Text entry · Intelligent keyboards · Physical keyboards · Visual feedback

1 Introduction

Mobile devices are overtaking personal computers (PCs) as the device most likely to be owned and operated by the general population. However, PCs remain indispensable for a range of creative tasks which require long text entry sessions (e.g. document editing, coding, etc.). A key enabler for such tasks is the physical keyboard, which may also, at times, be connected to mobile devices such as tablet computers. Physical keyboards allow superlative text entry rates compared to mobile virtual keyboards, and the keyboard-mouse combination affords accurate and fast error-correction. Due to the physicality of keyboards, users can become proficient to the extent that they can touch-type, i.e. enter text without looking at the keyboard at all. However, all users have to start off as novices, and even those who attain touch-typing skills, often do not carry them forward in everyday use [22]. Further, with PCs increasingly used by users of older age, and users with vision problems, text entry with physical keyboards still deserves the attention of researchers, and might benefit from integration with some intelligence to support text entry, just as mobile keyboards have benefited in the past.

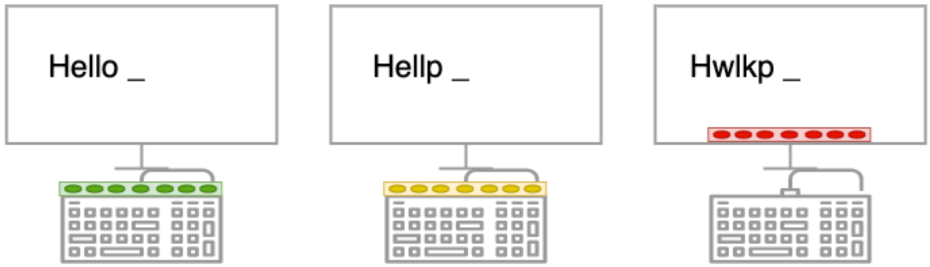

Fig. 1. LEDBoard operating principles. A spellcheck is performed after a word has been typed. If there are no errors, the strip lights up green (left). Errors cause the strip to light up yellow (middle) if they are slight, or red (right) if the word is unintelligible. The strip can be positioned either on the keyboard itself, or on the computer monitor.

In this paper, we discuss LEDBoard, an augmentation of the physical keyboard with a visual feedback cue mechanism, to provide text entry error awareness to users. Our aim is to support novice users in improving their performance with physical keyboards, hypothesising that error-checking behaviour (i.e. looking up at the screen to check recently input text) and corrective actions can be better supported if the user does not have to visually scan for errors during typing. Our prototype (Fig. 1) provides visual feedback on the keyboard itself, or at the bottom of the computer's monitor (within the peripheral vision of a user). We evaluate the prototype with a cohort of young computer science students whose text entry skills vary and find that LEDBoard improves corrective action for slow typists and reduces screen-glancing behaviour in fast typists.

2 Related Work

Mobile keyboards have received much attention in terms of support for text entry, since the difficulty of entry imposed by the device size constraints, as well as the opportunities afforded by the virtual nature of soft keyboards, have acted as drivers for innovative approaches to support text entry. For example, autocorrection and next-word prediction [4, 19], gestural input [13], error highlighting [1, 12], even totally invisible keyboards [24] have been successfully demonstrated either as research prototypes, or even commercially implemented.

In contrast, text entry in physical desktop keyboards has not received as much attention in the last two decades. Studies such as [5, 8, 16, 20] focus on the observation and analysis of user behaviour during text entry, including attention, motor and cognitive performance during various text entry tasks. Previous efforts to augment physical keyboards with novel abilities, include implementing methods to recognize sounds while using the keyboard [10, 14], creating pressure sensitive keyboards [7], recognizing objects interacting on physical areas [11], enhancing physical keyboards to act as pointing devices [17] or detecting gestures performed on the keyboard [23]. Further, the integration of physical keyboards in virtual or mixed reality has recently become a topic of interest, focusing mostly on reconfiguration or augmentation of information displayed on the keyboard (e.g. [15, 18]). All these efforts aimed at reducing input time or user' effort and improve user performance, but we have not been able to source previous literature relating to the implementation of intelligent error-correction, or error-prevention methods for physical keyboards. Commercially, Apple's post 2016 MacBook Pro models with a touch bar, allowed the implementation of predictive text for some applications.

Given the proliferation of computers in society, the requirement to support text entry for novice users is urgent. Physical keyboards augmented with a level of intelligence to support text entry could help the learning process for novices, or support users with limited abilities. Our aim for this paper is to evaluate an error support system for physical keyboards, in the form of a visual feedback indicator that detects and informs users about text entry errors as they type.

3 Materials and Method

3.1 Materials

To implement LEDBoard, we used a Raspberry Pi 4 microcontroller (RPi) as a proxy between a physical keyboard and the host computer operated by the user. The physical keyboard is connected to the RPi, which forwards incoming characters to the host computer, while keeping a sentence-level buffer, to support cases where the user corrects a mistake earlier than at the end of the input stream. At the same time, the RPi checks user input for spelling mistakes after each word has been typed, and controls an RGB LED strip to provide visual feedback when it detects them.

The RPi was configured using Python 3 to be recognized as a keyboard device once connected to a computer via USB. The keyboard used by the participants during the experiment was connected to the RPi directly and the device provided the computer with the characters pressed by the user and simultaneously presented its output on a second screen, which participants could not see.

For spell-checking the Python library *PyEnchant* was used. Every time a full word was typed by the users, a spellchecking function ran, and using the *Levenshtein* distance between the user entered text and the first suggestion provided by the PyEnchant library, the program classified the entry as *non-erroneous* (e.g. the k^{th} word input by the user $I_k = Hello$, the 1^{st} suggestion being $S_1 = Hello$), a *slight error* (e.g. $I_k = Hellp, S_1 = Hello$), or a *serious error* (e.g. $I_k = Hwllp, S_1 = Hello$).

To provide users with visual feedback, a strip of non-addressable RGB LEDs was used. The materials used were 1) a LED strip with three terminals for RGB and one terminal for 12V DC power supply, 2) a breadboard, 3) three N-channel MOSFETS, 4) a suitable external power supply of the LED strip (12V DC 2A), 5) a power jack to separate the power supply plug into positive and negative, 6) cables with terminals for the wiring. The wiring scheme between the RPi and the LED strip is shown in Fig. 2. The LED strip was placed in an aluminium

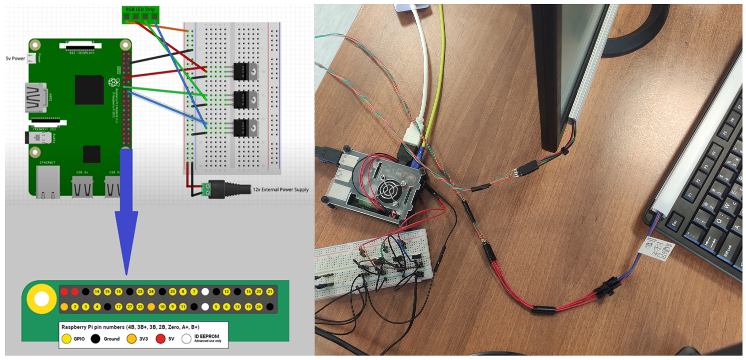

Fig. 2. Wiring scheme between the Raspberry Pi 4 and the LED strip

casing with an opaque cover, to diffuse the light and improve visibility. Full implementation details and code are openly available (see Section 6).

3.2 Participants

We recruited 38 first year undergraduate CS students from our university, to take part in the experiment without compensation. We selected this sample to represent users with a clear need to frequently use physical keyboards, but with moderate experience in their use. Two of them were later excluded, as they reported inability to distinguish between the colours used, due to colour blindness. The average age of remaining participants was 19.81 ($\sigma = 2.82$), out of which 31 were male and 5 were female. 71.1% of them had an advanced or higher certificate of proficiency in English.

3.3 Procedure

Our experiment employed the typical phrase transcription task used in most text entry research protocols. WebTem [2] was used to record and gather text entry metrics during the experiment. The phrases presented to users were selected from the "200 Memorable English Phrases" phrase set [21] and were all presented in lower case. WebTem records a range of metrics, but for this paper we focus on Words per Minute (WPM), Total Error Rate (TER), Corrected Error Rate (CER), and BackSpace Count (BC). We also analyse interkey intervals (IKI).

The experiment began with participants filling out a form regarding their demographic characteristics and answering to some questions on the frequency of use of physical keyboards and on how familiar with using physical keyboards they are. After completing the form, operation of the LEDBoard was explained and demonstrated to the participants, including the semantics of the colour feedback, before starting. The process was divided into 3 input sessions of 10 phrases each, per condition (*Baseline*: no visual feedback; *Keyboard*: visual feedback on the keyboard; *Screen*: visual feedback at the bottom of the computer monitor, Fig.

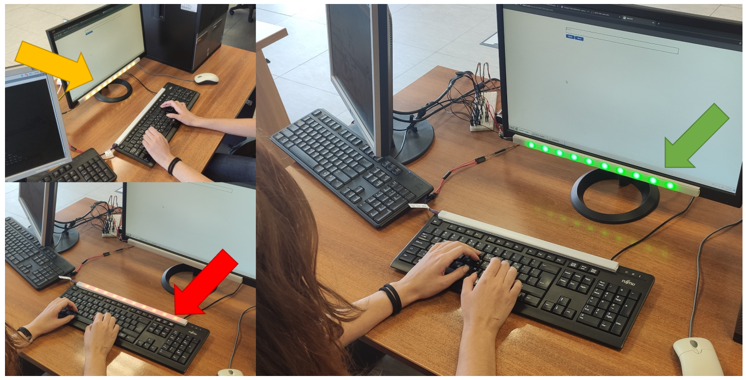

Fig. 3. Position of LED strips

3), resulting in 90 phrases per participant in total. Participants were asked to type freely, looking at either the screen or the keyboard while typing, and to correct any spelling errors using only the backspace key. Having completed all three sessions in a condition, participants were administered a NASA Task Load Index (NASA – TLX [9]) instrument and a few minutes' break followed.

4 Results

We divided participants in two groups. Those with a self described familiarity with QWERTY (≤ 3 on a 5-point Likert scale) or use of single digit typing were assigned to the "Slow" group (18 participants). The "Fast" group participants had an average WPM of 62.852 ($\sigma = 19.652$) while the "Slow" group has an average WPM of 34.333 ($\sigma = 6.334$). This split aligns reasonably well with findings in [6] where the average is 51.56WPM ($\sigma = 20.20$) Subsequent analyses were made using statistical hypothesis testing. Tests were chosen after examining the data and its fulfilment of assumptions for parametric and non-parametric test use. Data was analysed using Python 3.8 and SPSS v27.

4.1 Typing Speed and Total Error Rate

First, we examine the impact of the device in participant typing speed (words-per-minute) and total error rate, as measures of participant experience. We hypothesize that experienced participants will demonstrate a better typing speed without making more mistakes during text entry, compared to participants in the inexperienced group, at least for the baseline condition.

With regard to input speed, fast participants exhibit a higher WPM measure across all conditions (Fig. 4 left). The difference with the slow group is statistically significant, across conditions (Mann-Whitney U, Baseline $Z = -5.125, p < 0.001$; Keyboard $Z = -4.714, p =< 0.001$; Screen $Z = -4.746, p =< 0.001$). We also note that Friedman tests across the conditions in each group also show no statistically significant differences (Fast $\chi^2 = 1.444, p = 0.486$; Inexperienced $\chi^2 = 1.333, p = 0.513$), demonstrating that the presence of the device did not affect either group's text entry speed.

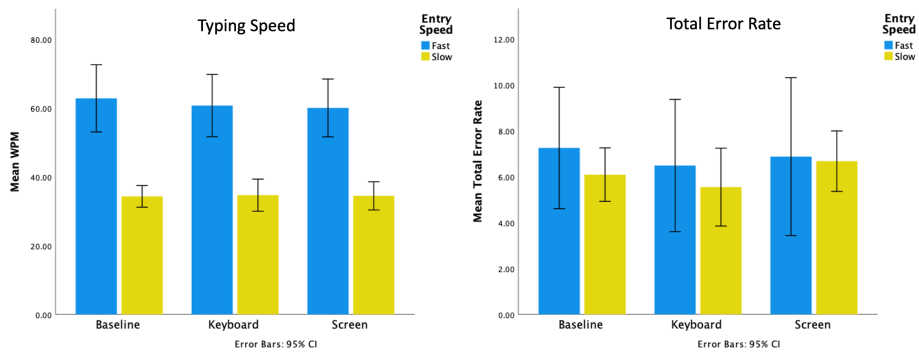

Fig. 4. Words per Minute and Total Error Rate per condition

Another indicator of participant expertise is total error rate (errors made during typing regardless of being fixed), see Fig. 4 right. For this metric, differences across conditions in each group did not exhibit statistical significance (Mann-Whitney U Baseline $Z = -0.664, p = 0.506$; Keyboard $Z = -0.285, p = 0.776$; Screen $Z = -1.266, p = 0.206$). Friedman tests to compare across conditions in each group again show no statistically significant differences (Fast $\chi^2 = 0.444, p = 0.801$; Slow $\chi^2 = 1.778, p = 0.411$). While we might have expected that slower, less experienced typists might make more errors during typing, it is known from literature that users will trade speed for accuracy to prevent errors, therefore the finding is in accordance with the lower WPM discovered for the inexperienced group [3]. Paired with the findings from text entry speed, we find here evidence that the grouping of our participants is sensible.

4.2 Error-correcting behaviour

We hypothesise that the device will make typing errors more obvious, thus resulting in a higher corrected error rate. There were statistically significant differences in the corrected error rate between the groups and across conditions (Mann-Whitney U Baseline $Z = -2.088, p = 0.037$; Keyboard $Z = -3.132, p = 0.002$; Screen $Z = -3.417, p \leq 0.001$). A Friedman test showed that differences across conditions in the Fast group were not statistically significant, but the reverse was found for the Slow group (Fast $\chi^2 = 1.000, p = 0.607$; Slow $\chi^2 = 8.111, p = 0.017$). Post-hoc Bonferroni corrected pairwise tests for the Slow group reveal a statistically significant difference between the Baseline and Screen conditions only (Wilcoxon $Z = -2.983, p = 0.003$). The results are shown in Fig. 5 (left).

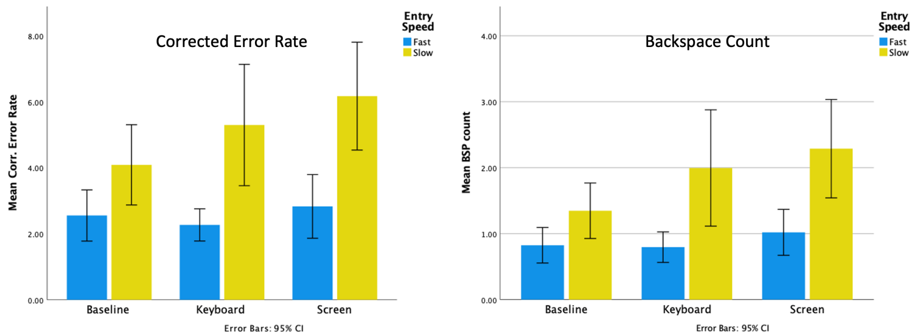

Fig. 5. Corrected Error Rate and Backspace Count per condition

A further related metric is use of backspaces to correct errors. In fact, Spearman correlations in all conditions between corrected error rate and backspace use are all statistically significant at the $p < 0.001$ level, for both the Fast and Slow groups. As with the corrected error metric, differences are observable in Fig. 5 (right) and we found statistically significant differences between groups and across conditions (Mann-Whitney U Baseline $Z = -2.137, p = 0.033$; Keyboard $Z = -2.960, p = 0.003$; Screen $Z = -3.039, p = 0.002$). Comparing conditions within the groups, we find no statistically significant differences in the Fast group ($\chi^2 = 1.211, p = 0.546$) but the Slow group demonstrates statistical significance ($\chi^2 = 9.155, p = 0.010$). Again this difference is statistically significant after pairwise post-hoc Bonferroni corrected tests only between the Screen and Baseline condition (Wilcoxon $Z = -2.809, p = 0.005$).

4.3 Glancing behaviour

Typists have to look at the screen in order to identify mistakes during text entry. For touch-typists, this is not so much a problem since their attention is already on the screen, however, non-expert typists need to frequently shift their attention between the keyboard and the screen to check for entry errors. Our hypothesis was that our device might reduce the number of shifts, therefore explored glancing behaviour with and without LEDBoard. To identify glancing episodes, we rely on the interkey interval (IKI) (i.e. time elapsed between consecutive keypresses). A glancing episode is characterised by a high IKI, compared to the IKI observed during typing. This can be illustrated when plotting the keyboard events during an entire session as a signal, with an amplitude according to the IKI of each event (Fig. 6). Since every participant's behaviour is unique, to identify these "peaks" in the signal we employ a signal processing method to discover local maxima in the signal, setting a minimum height for identified peaks equal to the mean IKI of the entire session plus a multiple of the session IKI standard deviation ($\bar{x}_{IKI} + n \times \sigma_{IKI}$).

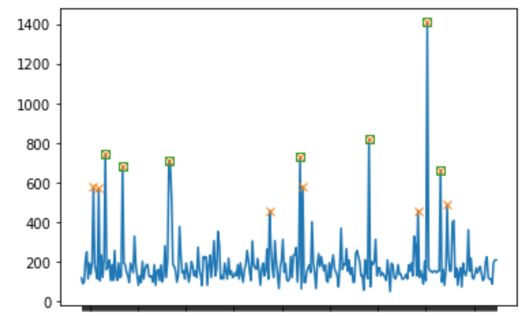

Fig. 6. Example of IKI plotting of one session from a single participant. Peaks are identified with x-markers at $\bar{x} + 2\sigma$, and square outlined markers at $\bar{x} + 3\sigma$

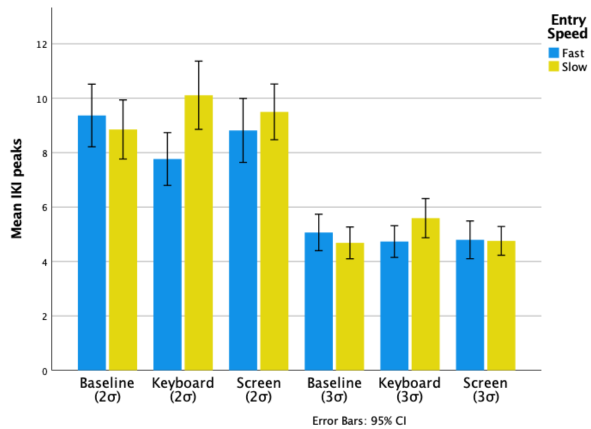

Fig. 7. Interkey intervals per condition, at $\bar{x} + 2\sigma$ and $\bar{x} + 3\sigma$

As observed in Fig. 7, there are some observable differences in the peaks at 2σ and 3σ, however, these are only statistically significantly different between groups for the 2σ mean in the Keyboard condition (Mann U, $Z - 2.830, p = 0.005$). Further, for each group, there are no statistically significant differences across the conditions, both for peaks at 2σ and 3σ. Measuring the number of users who benefited from fewer glances in each group, we find that it is actually the Fast typists who benefit most, as 13/18 participants saw a reduction in glances (at 2σ) in the Keyboard condition (max reduction 57%, min reduction 4%, $barx = 31.91\%, \sigma = 14.06\%$). In contrast, only 4/18 of the slow typists benefited from the device in the keyboard condition (max reduction 47%, min reduction 19%, $barx = 34.44\%, \sigma = 12.44\%$).

4.4 Subjective Feedback

As discussed earlier, participants undertook a NASA-TLX test after each condition. Overall we found no statistically significant difference except for Physical Demand in the baseline condition between the groups (Mann U $Z = -2.716, p = 0.007$) where the Slow group indicated a higher physical demand ($\bar{x} = 30.00, \sigma = 4.240$) compared to the Fast group ($\bar{x} = 15.56, \sigma = 1.847$). Comparing within the groups and across conditions, Friedman tests showed statistically significant differences within the Fast group for physical demand ($\chi^2 = 13.579, p = 0.001$) and temporal demand ($\chi^2 = 8,464, p = 0.015$). Wilcoxon post hoc tests with Bonferroni correction show that the Fast group perceived higher physical demand against baseline with the keyboard condition ($Z = -2.740, p = 0.006$) and screen condition ($Z = -2.993, p = 0.003$), while for temporal demands there were no statistically significant results in the pairwise tests. In the Slow group, only performance showed a statistically significant difference ($\chi^2 = 8.400, p = 0.015$) but Wilcoxon post hoc tests with Bonferroni correction did not uncover further statistically significant differences. Overall, although differences in participant performance were discovered in the quantitative analysis, these differences were imperceptible from a subjective point of view to participants, demonstrating that the device is able to subtly and unobtrusively deliver likely benefits.

5 Discussion

From our findings, we summarise LEDBoard did not have an impact in participant text entry speed or error rates committed during input. This is understandable - the experience and motor skills of participants cannot be gravely affected in the course of a short experiment. On the other hand, we noticed that LEDBoard was successful in helping slow typists become more aware of their entry errors and to correct more of these, compared to fast typists. Despite our hopes that the device would generate confidence in the slow typists and reduce their need to frequently glance at the screen in order to detect mistakes, we did not find any evidence to support this hypothesis. In fact, the device seems to increase glancing in slow typists, while fast typists were able to take advantage of its presence to further reduce their error-checking behaviour.

As with any experiment, the generalisability of our findings is constrained by the selection of our sample population. We recruited solely 1st year computer science students, thus their level of familiarity with physical keyboard entry may not necessarily be the same as the general population. In the future, we would like to repeat our experiment with a population from other backgrounds and to observe the prototype's effect on users of older age, or with vision problems.

Improvements to our prototype could include error-detection at the bigram or n-gram level (e.g. alerting the user that an unlikely character has just been entered, instead of at the end of a word), as well as syntax and grammar support. Different feedback display strategies could also be investigated, e.g. blinking, animations, or gradual fading of the lighting, as text entry progresses.

Another limitation is the need for the microcontroller to preserve a sentence-level buffer in case the user wants to correct previously entered words. Even with this implementation, the system would not work in case the user tried to edit text written in a previous paragraph or earlier, as the microcontroller would have no way to obtain the context of the text being edited (cursor position and text). To allow smart keyboards such as LEDBoard full integration with applications, significant changes to the operating system and keyboard drivers would be required in order to allow the keyboard microcontroller to obtain the edited text context. A "workaround" might be possible by using the accessibility features built into various operating systems, and by setting up a server on the host computer to feed this contextual information back to the microcontroller.

To summarise, we discover reasonable evidence to support the integration of methods for intelligent text entry support in physical keyboards. Such support may provide various types of benefits to users, according to their current skill and expertise. More work is needed to better understand the likely benefits across various user categories (e.g. novices, children, elderly, disabled), and to determine optimal feedback strategies to maximise these benefits to users.

6 Code and Data availability

The project implementation details, source code and experiment data are openly available at https://github.com/komis1/LEDBoard.

References

1. Alharbi, O., Arif, A.S., Stuerzlinger, W., Dunlop, M.D., Komninos, A.: WiseType: A tablet keyboard with color-coded visualization and various editing options for error correction. In: Proceedings of graphics interface 2019. GI 2019, Canadian Information Processing Society, Kingston, Ontario (2019). https://doi.org/10.20380/GI2019.04, iSSN: 0713-5424 Number of pages: 10
2. Arif, A.S., Mazalek, A.: WebTEM: A Web Application to Record Text Entry Metrics. In: Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces. pp. 415–420. ISS '16, Association for Computing Machinery, New York, NY, USA (Aug 2016). https://doi.org/10.1145/2992154.2996791, https://doi.org/10.1145/2992154.2996791
3. Banovic, N., Rao, V., Saravanan, A., Dey, A.K., Mankoff, J.: Quantifying aversion to costly typing errors in expert mobile text entry. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. p. 4229–4241. CHI '17, ACM (2017). https://doi.org/10.1145/3025453.3025695, http://doi.acm.org/10.1145/3025453.3025695
4. Cerney, M.M., Mila, B.D., Hill, L.C.: Comparison of Mobile Text Entry Methods. Proceedings of the Human Factors and Ergonomics Society Annual Meeting **48**(5), 778–782 (Sep 2004). https://doi.org/10.1177/154193120404800508, https://doi.org/10.1177/154193120404800508, publisher: SAGE Publications Inc
5. Dhakal, V., Feit, A.M., Kristensson, P.O., Oulasvirta, A.: Observations on Typing from 136 Million Keystrokes. In: Proceedings of

the 2018 CHI Conference on Human Factors in Computing Systems. pp. 1–12. CHI '18, Association for Computing Machinery, New York, NY, USA (Apr 2018). https://doi.org/10.1145/3173574.3174220, https://doi.org/10.1145/3173574.3174220
6. Dhakal, V., Feit, A.M., Kristensson, P.O., Oulasvirta, A.: Observations on typing from 136 million keystrokes. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. p. 1–12. CHI '18, Association for Computing Machinery, New York, NY, USA (Apr 2018). https://doi.org/10.1145/3173574.3174220, https://doi.org/10.1145/3173574.3174220
7. Dietz, P., Eidelson, B., Westhues, J., Bathiche, S.: A practical pressure sensitive computer keyboard. pp. 55–58 (Jan 2009). https://doi.org/10.1145/1622176.1622187
8. Feit, A.M., Weir, D., Oulasvirta, A.: How We Type: Movement Strategies and Performance in Everyday Typing. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. pp. 4262–4273. CHI '16, Association for Computing Machinery, San Jose, California, USA (May 2016). https://doi.org/10.1145/2858036.2858233, https://doi.org/10.1145/2858036.2858233
9. Hart, S., Staveland, L.: Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research (1988). https://doi.org/10.1016/S0166-4115(08)62386-9
10. Kato, J., Sakamoto, D., Igarashi, T.: Surfboard: Keyboard with microphone as a low-cost interactive surface (Oct 2010). https://doi.org/10.1145/1866218.1866233
11. Kim, D., Izadi, S., Dostal, J., Rhemann, C., Keskin, C., Zach, C., Shotton, J., Large, T., Bathiche, S., Nießner, M., Butler, D.A., Fanello, S., Pradeep, V.: RetroDepth: 3D silhouette sensing for high-precision input on and above physical surfaces. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1377–1386. CHI '14, Association for Computing Machinery, New York, NY, USA (Dec 2014). https://doi.org/10.1145/2556288.2557336, https://doi.org/10.1145/2556288.2557336
12. Komninos, A., Nicol, E., Dunlop, M.D.: Designed with Older Adults to SupportBetter Error Correction in SmartPhone Text Entry: The MaxieKeyboard. In: Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct. pp. 797–802. MobileHCI '15, Association for Computing Machinery, New York, NY, USA (Aug 2015). https://doi.org/10.1145/2786567.2793703, https://doi.org/10.1145/2786567.2793703
13. Kristensson, P.O., Zhai, S.: SHARK^2: a large vocabulary shorthand writing system for pen-based computers. In: Proceedings of the 17th annual ACM symposium on User interface software and technology. pp. 43–52. UIST '04, Association for Computing Machinery, New York, NY, USA (Oct 2004). https://doi.org/10.1145/1029632.1029640, https://doi.org/10.1145/1029632.1029640
14. Kurosawa, T., Shizuki, B., Tanaka, J.: Keyboard Clawing: Input Method by Clawing Key Tops. In: Kurosu, M. (ed.) Human-Computer Interaction. Interaction Modalities and Techniques. pp. 272–280. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39330-3_29

15. McGill, M., Brewster, S., De Sa Medeiros, D.P., Bovet, S., Gutierrez, M., Kehoe, A.: Creating and Augmenting Keyboards for Extended Reality with the Keyboard Augmentation Toolkit. ACM Transactions on Computer-Human Interaction **29**(2), 15:1–15:39 (Jan 2022). https://doi.org/10.1145/3490495, https://doi.org/10.1145/3490495
16. Papoutsaki, A., Gokaslan, A., Tompkin, J., He, Y., Huang, J.: The eye of the typer: a benchmark and analysis of gaze behavior during typing. In: Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications. pp. 1–9. ETRA '18, Association for Computing Machinery, Warsaw, Poland (Jun 2018). https://doi.org/10.1145/3204493.3204552, https://doi.org/10.1145/3204493.3204552
17. Ramos, J., Li, Z., Rosas, J., Banovic, N., Mankoff, J., Dey, A.: Keyboard Surface Interaction: Making the keyboard into a pointing device. arXiv:1601.04029 [cs] (Jan 2016), http://arxiv.org/abs/1601.04029, arXiv: 1601.04029
18. Schneider, D., Otte, A., Gesslein, T., Gagel, P., Kuth, B., Damlakhi, M.S., Dietz, O., Ofek, E., Pahud, M., Kristensson, P.O., Müller, J., Grubert, J.: ReconViguRation: Reconfiguring Physical Keyboards in Virtual Reality. IEEE Transactions on Visualization and Computer Graphics **25**(11), 3190–3201 (Nov 2019). https://doi.org/10.1109/TVCG.2019.2932239, https://www.computer.org/csdl/journal/tg/2019/11/08794572/1dXEHv0aKMo
19. Silfverberg, M., MacKenzie, I.S., Korhonen, P.: Predicting text entry speed on mobile phones. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. pp. 9–16. CHI '00, Association for Computing Machinery, New York, NY, USA (Apr 2000). https://doi.org/10.1145/332040.332044, https://doi.org/10.1145/332040.332044
20. Soukoreff, R.W., MacKenzie, I.S.: Metrics for Text Entry Research: An Evaluation of MSD and KSPC, and a New Unified Error Metric. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 113–120. CHI '03, ACM, New York, NY, USA (2003). https://doi.org/10.1145/642611.642632, http://doi.acm.org/10.1145/642611.642632
21. Vertanen, K., Kristensson, P.O.: A versatile dataset for text entry evaluations based on genuine mobile emails. In: Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services. pp. 295–298. MobileHCI '11, Association for Computing Machinery, New York, NY, USA (May 2011). https://doi.org/10.1145/2037373.2037418, https://doi.org/10.1145/2037373.2037418
22. Yechiam, E., Erev, I., Yehene, V., Gopher, D.: Melioration and the transition from touch-typing training to everyday use. Human Factors **45**(4), 671–684 (Dec 2003). https://doi.org/10.1518/hfes.45.4.671.27085
23. Zhang, H., Li, Y.: GestKeyboard: enabling gesture-based interaction on ordinary physical keyboard. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1675–1684. CHI '14, Association for Computing Machinery, New York, NY, USA (Dec 2014). https://doi.org/10.1145/2556288.2557362, https://doi.org/10.1145/2556288.2557362
24. Zhu, S., Luo, T., Bi, X., Zhai, S.: Typing on an Invisible Keyboard. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. pp. 1–13. CHI '18, Association for Computing Machinery, Montreal QC, Canada (Apr 2018). https://doi.org/10.1145/3173574.3174013, https://doi.org/10.1145/3173574.3174013