Controlled Experiment Comparing Text Input Methods for Smartwatches

Antoine Viscardi

March 6, 2019

1   Introduction

Keyboards are one of the most widespread text input devices. However, with the advent of touch screen and ever smaller devices, the need for alternative text input methods as grown. The following report presents an experiment that had for goal to compare two text input methods for such small wearable devices. The first section goes over the experimental design and the different software tools used. The next section presents the hypotheses. The results are then presented and analysed. Finally, before concluding, we discuss how the results were affected by the experimental design and how they relate to the hypotheses.

2   Methodology

2.1   Experimental Design

In order to compare the two text entry techniques, an experiment was designed to collect data that was then analysed to compare different metrics and draw relevant conclusions. This section goes over the different characteristics of the experiment while justifying the different design decisions.

The first step when designing the study was to identify the targeted participants. They were selected as to cover a wide demographic range in order to capture the numerous market segments attracted to wearable devices[6]. Particularly, they were characterized by their technology literacy and their age ranging from 19 to 54 years old. Due to time and resource constraints, only four participants were selected from my personal social circle. The implications of this limitation are discussed in the discussion section.

The first independent variable is the text entry method. It has two levels, the normal keyboard or the zooming keyboard. The second independent variable is the visual feedback which also has two levels, the presence and the absence of visual feedback. This variable was chosen because we want to investigate the relevance of such a feature: does it help and, if so, does it equally help for both text entry techniques and to what extent?

The studied task is to write a sentence. This task was chosen over writing single words as it provides a more realistic experience. Indeed, users very seldom type a single word on a device when browsing the web or texting for example. Thereby, every trial consisted in typing a sentence selected at random from a corpus of 500 sentences published by MacKenzie and Soukoreff [4]. They do not contain punctuation marks nor upper case letters.

Upon starting the experiment, participants were briefed about the purpose of the study and were asked to read and sign the consent form (Appendix J and L). In order to minimize the effect of environmental variables, sessions were conducted in a quiet room and participants adopted the same posture: they were seated and the device was placed as to mimic the position of a

wearable attached to the wrist (see appendix M). Before running the experiment, the participants were explained that they had to type as they would normally type a text message and they were demonstrated how to operate the two keyboards. Errors were corrected at their discretion. There were no practice session and five minutes breaks were allowed between each condition. At the end of the experiment, they were asked to complete an exit survey (see appendix G).

It was empirically determined that it requires approximately 150 seconds to complete three trials. Given that 4 conditions were tested, participants performed 9 trials for each condition. Factoring in short breaks between conditions and the possibility that some participants are slower, the experiment would run for 45 minutes at most.

Given the two independent variables, the study tested four conditions. The first condition (A) featured the traditional keyboard and no visual feedback. The second condition (B) also features the the traditional keyboard but with visual feedback. The third condition (C) featured the zooming keyboard without visual feedback. The fourth condition (D) also featured the zooming keyboard but with visual feedback.

Due to the small number of participants, the experiment followed a within-subject design. Each participant were therefore exposed to all four conditions. In order to mitigate learning transfer effect, partial counter balancing in the form of a Latin square was used. That is, the first, second, third and fourth participants were administered the conditions in the following order respectively: ABCD, BCDA, CDAB and DABC. It is important to note that designing a fully counter-balanced experiment was not possible since it would have required 32 (4!) participants.

2.2   Software

A software application was developed based on the starting code provided by the teaching assistants. The final version of the software features significant changes and is publicly accessible[1]. The application is built using the REACT framework. During the experiment, the application would run locally on a Linux environment connected to the network via WiFi. The web interface was then accessed via an Android smartphone's Chrome browser. Animations of the interface for the four different conditions can be found in the public repository[2].

Every input was recorded by the application along with a Unix timestamp and, at the end of every trial, the target sentence and the transcribed text. After the last trial of a condition, a JSON-formatted file containing all the recorded data was downloaded locally on the smartphone. An example of the resulting log file is included in Appendix A. The raw data was then preprocessed with a Python script (see Appendix B) in order to extract time intervals between keystrokes and the completion time for every trial based on the recorded Unix timestamps. The script also extracted different values used to computed the analyzed metrics, or dependant variables, namely the number of correct keystrokes, the number of fixes, the number of incorrect keystrokes not fixed and the number of incorrect keystrokes and fixed. They allowed to compute *the minimum string distance error rate* (MSD), the *key strokes per character* (KSPC), the *participant conscientiousness*, the *utilized bandwidth* and the *total error rate*. An in-depth description of those metrics can be found in the paper on metrics for text entry research published by Soukoreff and MacKenzie[5]. Also, an overview of how those values are extracted, how the different metrics are derived from them and what they represent can be found in Appendix N. Furthermore, a sample of the data returned by the preprocessing script is included in Appendix C. Finally, descriptive statistics were extracted from the data with the help of an OpenOffice spreadsheet[8] and RM-ANOVA tests were performed for each metric measured with the help of the GoStats software[7] (see Appendix F).

3   Hypotheses

The goal of this study is to determine which of the two studied method of text input is the most efficient for small devices such as smart watches. The question we therefore need to ask ourselves is how each method affects speed, accuracy and user satisfaction.

Although selecting a target using the zooming keyboard requires two taps instead of one, Fitts' law states that increasing the size of a target decrease the time required to select it [3]. The first Hypothesis is that the effect predicted by Fitts' law will outweigh the overhead that represents tapping twice. This is even more true considering that the first tap does not need to be precise. Effectively, this translates into the zooming keyboard allowing for faster text input, better accuracy and increased user satisfaction compared to the normal keyboard.

The second hypothesis is that the visual feedback will allow for faster text input. Indeed, it minimizes users' effort by sparing them from looking at the text field between each input to confirm that it was correct. Indeed, taking the eyes off the keyboard between each input could represent a significant overhead.

4   Results and Analysis

Before analysing the results, outliers were identified using the in quartile range method on four features, namely average time between keystrokes, average task completion time, key strokes per character and total error rate. The standard deviation method was used to identify outliers based on the minimum string distance error rate because its value is zero most of the time making the quartile method irrelevant. Also, due to the small sample size, a distance of only two standard deviation from the mean was used instead of three. Specifics on how the data was filtered is available in the original spreadsheet [8].

Once the outliers were removed, simple descriptive statistics were computed for the studied metrics. Appendix E contains graphs with bars representing the mean and error bars representing the standard deviation. Additionally, the tables from Appendix D show the mean value and the standard deviation of the metrics for all four conditions. One fact that stands out when looking at those results is that the standard deviation is high for all of the metrics. This is probably due to the small sample size. Indeed, only four participants contributed nine sentences each for every condition. This makes it hard to infer statistical conclusions simply by looking at those descriptive statistics. However, we can already see from the plots that the presence of visual feedback seemed to have had only a marginal effect on those metrics.

Repeated measures ANOVA (RM-ANOVA) tests were conducted to analyse the statistical effects of the independent variables on the different computed metrics. The results are included in Appendix F. They show that the presence or not of visual feedback had no statistically significant effect on any of the metrics. Indeed, for the average time between inputs, the task completion time, the KSPC, the MSD, the participant conscientiousness, the utilized bandwidth and the total error rate, $F_{1,3}$ is equal to 0.014, 0.175, 0.244, 6.857 (with $p > 0.05$), 0.483, 0.038 and 0.034 respectively. Also, the F value was either smaller than 1 or the p value was greater than 0.05 across all metrics when looking at interaction effects between the input method and the presence of visual feedback indicating the absence of any such effect.

When it comes to the input method, the RM-ANOVA tests showed that it had no statistically significant effect on the MSD nor the participant conscientiousness with $F_{1,3} = 0.01$ and $F_{1,3} = 0.007$ respectively. However, the input method was shown to have a statistically significant effect on the other metrics.

First, the mean time between inputs was 1402ms for the zooming keyboard which is more than 75% slower than the observed mean time between inputs with the normal keyboard (800ms). The difference was statistically significant since $F_{1,3} = 112.036$ and $p < 0.005$. This shows that when looking at individual keystrokes, typing is faster on the normal keyboard than on the zooming keyboard. This was to be expected since the zooming keyboard requires two tapping gestures for every input compared to only one for the normal keyboard.

Second, the observed KSPC for the zooming keyboard was about 23% less than for the normal keyboard with mean values of 1.48 and 1.14 respectively. The difference was also statistically significant since $F_{1,3} = 47.898$ and $p < 0.01$. This result shows that using the zooming keyboard allows for a better typing efficiency by a considerable margin over the normal keyboard. That is, a single keystroke is more "useful" on the zooming keyboard than on the normal keyboard.

Third, the observed mean utilized bandwidth for the zooming keyboard was more than 28% higher than with the normal keyboard with mean values of 0.897 and 0.698 respectively. The difference was statistically significant with $F_{1,3} = 123.656$ and $p < 0.005$. When considering the usefulness of every keystroke, this result confirms that the zooming keyboard is more efficient in terms of accuracy than the normal keyboard. That is, if we look at typing as information transfer, the zooming keyboard allows for a better utilization of the bandwidth.

Fourth, the mean total error rate for the zooming keyboard was 0.061 compared to 0.187 for the normal keyboard. Effectively, the error rate was reduced by about 67% when using the zooming keyboard over the normal keyboard. The difference was statistically significant with $F_{1,3} = 71.517$ and $p < 0.005$. This shows that typing on the zooming keyboard is more accurate than typing on the normal keyboard as it results in an important reduction of mistakes.

Finally, the mean task completion time was 42,2s for the zooming keyboard which is about 35% slower than the 32.7s mean task completion time with the normal keyboard. The difference was statically significant since $F_{1,3} = 163.927$ and $p < 0.005$. When considering the total time it takes to complete the task, this result shows that the zooming keyboard is considerably slower than the normal keyboard. This result is surprising as we could have expected the zooming keyboard, despite its slower per-keystroke speed, to allow faster sentence typing given the gain in efficiency.

On another note, the results from the exit surveys show that no consensus was reached regarding user satisfaction (see Appendix H). Indeed, opinions are divided with two participants advocating for each method. The relevance of the visual feedback shows similar mixed results with two participants in each camp. Therefore, we cannot draw any conclusions about which method offered the most satisfying experience or if the visual feedback is relevant for user satisfaction.

In short, we saw from the results that the normal keyboard is faster when looking at individual keystrokes. Additionally, we concluded that the zooming keyboard allows for considerably more accurate typing. However, we also saw that this does not make up for the slower typing speed. That is, the less accurate normal keyboard is still faster when typing whole sentences.

5 Discussion

The first hypothesis was that the zooming keyboard would allow for faster typing, better accuracy and increased user satisfaction. We saw from the results that although the zooming keyboard is more accurate by a considerable margin, the normal keyboard is still faster when typing full sentences. This is due to the fact that the observed gain in accuracy does not outweigh the overhead introduced by tapping twice instead of once. However, it is important to note that this overhead gets diluted as longer text gets typed. This is indeed observed in the results: the margin by which the zooming keyboard is slower is considerably less important when looking at whole sentences

(35%) versus when looking at single keystrokes (75%). It is therefore reasonable to think that over longer texts such as entire paragraphs, the zooming keyboard could actually be faster. However, this would not only require further experiments to show but it is also a much less reasonable and realistic task for a wearable device.

The last part of the first hypothesis was that the zooming keyboard would provide a better user experience. However, the results from the exit survey are mixed. Overall, the first hypothesis is invalidated by the observed results.

It is important to note that the small number of participants forced the experiment to use a within-subjects design with partial counter balancing which is prone to learning transfer effects. Having 24 participants would have allowed to fully counter balance the four conditions. Ideally, however, a between-subject design would have allowed to study the effect of training to a higher level of skills. Indeed, it is possible that after practice participants become considerably faster when using the zooming keyboard over the normal keyboard. Running such a study would require approximately 30 participants for each condition which can be a limiting factor, but would allow to get much clearer insight regarding the efficiency of both input methods.

The second hypothesis was that the presence of visual feedback would allow for faster text input. However, it was observed that visual feedback not only had no statistically significant effect on typing speed, but also had no effect on any of the metrics nor the users' satisfaction. The second hypothesis is therefore invalidated. However, the small sample size might explain why no statistical effect was observed. It would be beneficial to conduct further experiments featuring more participants in order to confirm the irrelevance of visual feedback.

In short, although the results seems to clearly show that the normal keyboard remains faster than the zooming keyboard despite the lower accuracy, the experiment did not study the effect of skill gain. Indeed, the zooming keyboard might actually be faster, but only after a certain learning period. Further study would be required to confirm or invalidate that claim. Also, the results show that the presence or not of visual feedback has no effect on the efficiency nor the user satisfaction. Again, further study with a larger sample size would be beneficial to confirm this claim.

6   Conclusion

In conclusion, the experiment allowed to draw relevant conclusions regarding which of the studied input method was the most efficient for small devices and whether or not the visual feedback is a relevant feature. The results showed that the zooming keyboard, although it allowed to significantly reduce the error rate, was slower than the normal keyboard. Also, participants' opinions were mixed regarding which method offered the most satisfactory experience. We therefore concluded that the normal keyboard appears to be the most efficient typing method for small devices. However, an important caveat is that the experiment did not allow to compare the effect of skill development on the efficiency of the two methods. Indeed, it is possible that the zooming keyboard is faster when users are trained to a higher skill level, but further study is needed to investigate this possibility.

On another note, the results showed that visual feedback did not have any statistically significant effect on efficiency nor user satisfaction. This feature does not seems relevant.

Lastly, it would be relevant to study how different variations of the zooming keyboard affect efficiency. For example, the zooming animation's speed or the scale factor could both be adjusted. Also, it would be interesting to investigate a hybrid method where the initial touch of the screen triggers the zooming, but the input is registered only upon releasing, allowing the user to move his finger while touching the screen to readjust his position. This would have the potential to significantly reduce the double tapping overhead.

References

[1] Antoine Viscardi (2019) https://github.com/antoineviscardi/ZoomKeyboardStudy

[2] Antoine Viscardi (2019) https://github.com/antoineviscardi/ZoomKeyboardStudy/blob/ master/conditions.md

[3] MacKenzie I.S. & Buxton W. (1992) Extending Fitt's law to two-dimensional tasks. *Proceedings of the ACM Conference on Human Factors in Computing Systems* CHI '92, pp. 219-226. New York, ACM. Retrieved from https://www.yorku.ca/mack/CHI92.html

[4] MacKenzie I.S. & Soukoreff W.R. (2003) Phrase sets for evaluating text entry techniques. *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems - CHI 2003*, pp. 754-755. New York: ACM. Retrieved from https://www.yorku.ca/mack/chi03b.html

[5] Soukoreff W. R. & MacKenzie I.S. (2003) Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric. *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI 2003*, pp. 113-120. New York, ACM. Retrieved from https://www.yorku.ca/mack/chi03.pdf

[6] Statista (2019) *Wearable user penetration rate in the United States, in 2017, by age.* Retrieved from https://www.statista.com/statistics/739398/us-wearable-penetration-by-age/

[7] http://www.yorku.ca/mack/GoStats/index.html?GoStats.html

[8] https://github.com/antoineviscardi/ZoomKeyboardStudy/blob/master/data/clean/clean-data.ods

A Sample log file

```json
{
  "type": "normal",
  "size": "38mm",
  "feedback": "false",
  "trials": [
    {
      "targetPhrase": "prepare for the exam in advance",
      "inputPhrase": "prepare for the exam in advance",
      "inputs": [
        {
          "input": "p",
          "timestamp": 1550192669241
        },
        {
          "input": "r",
          "timestamp": 1550192670229
        },

        ...

        {
          "input": "delete",
          "timestamp": 1550192679462
        },
        {
          "input": "x",
          "timestamp": 1550192680109
        },

        ...

        {
          "input": "c",
          "timestamp": 1550192685625
        },
        {
          "input": "e",
          "timestamp": 1550192685747
        }
      ]
    },
    ...
```

B Preprocessing script

```python
import json
import csv
import itertools
from Levenshtein import editops, matching_blocks, distance


def compute_taxonomy_values(target, transcribed, stream):
    """ Function that computes and returns the different taxonomy values as
    described by Soukoreff and MacKenzie (https://www.yorku.ca/mack/chi03)
    """

    # Compute the Levenshtein distance (Incorrect Not Fixed (INF))
    ops = editops(target, transcribed)
    inf = len(ops)

    # The number of correct keystrokes (C) is the difference between
    # the length of the transcribed text and INF
    blocks = matching_blocks(ops, target, transcribed)
    c = sum([block[2] for block in blocks])

    # The number of fixes (F) is equal to the number of 'delete'
    # present in the input stream
    f = [x for x in stream].count('delete')

    # The number of incorrect input that were fixed (IF) is the number of
    # non-delete characters present in the input stream but not in the
    # transcribed text
    _if = [x for x in stream if x != 'delete']
    _if = distance(''.join(_if), transcribed)

    return (inf, c, f, _if)


def test_compute_taxonomy_values():
    """ Simple test based on the example provided by Soukoreff and MacKenzie in
    https://www.yorku.ca/mack/chi03
    """

    inf, c, f, _if = compute_taxonomy_values(
        'the quick brown',
        'th quick brpown',
        ['t', 'h', ' ', 'q', 'u', 'i', 'x', 'x', 'delete',
         'delete', 'c', 'k', ' ', 'b', 'r', 'p', 'o', 'w', 'n'])

    assert inf == 2, 'inf was {} was expecting 2'.format(inf)
    assert c == 14, 'c was {} was expecting 14'.format(c)
```

```python
        assert f == 2, 'f was {} was expecting 1'.format(f)
        assert _if == 1, '_if was {} was expecting 1'.format(_if)


def compute_average_time(stream):
    deltas = [a - b for (a, b) in zip(stream[1:], stream[:-1])]
    return round(sum(deltas) / len(deltas))


def compute_task_completion_time(stream):
    return (stream[-1] - stream[0]) // 1000


if __name__ == '__main__':
    """ Script that reads the data from JSON files and outputs a CSV file with
    relevant computed values ready for analysis
    """

    # Open the output file
    output_path = 'data/clean/clean-data.csv'
    output_file = open(output_path, 'w')
    fields = ['uid', 'participant', 'type', 'feedback', 'trial',
              'avg_time(ms)', 'task_time(s)', 'INF', 'C', 'F', 'IF']
    writer = csv.DictWriter(output_file, fields, lineterminator='\n')
    writer.writeheader()

    # Loop over all result files
    uid = 1
    results_path = 'data/clean/results-participant-{}/results-{}-38mm-{}.json'
    iterable = itertools.product(
        [1, 2, 3, 4],
        ['normal', 'zoom'],
        ['false', 'true']
    )
    for participant, kb_type, feedback in iterable:

        # Load results into dictionaries
        with open(results_path.format(participant, kb_type, feedback), 'r') as file:
            results = json.load(file)

        # Loop over all trials
        for (i, trial) in enumerate(results['trials']):
            inf, c, f, _if = compute_taxonomy_values(
                trial['targetPhrase'],
                trial['inputPhrase'],
                [x['input'] for x in trial['inputs']])
```

```python
        timestamp_stream = [x['timestamp'] for x in trial['inputs']]
        avg_time = compute_average_time(timestamp_stream)
        task_time = compute_task_completion_time(timestamp_stream)

        # Write results to file
        results = [
            uid,
            participant,
            kb_type,
            feedback,
            i+1,
            avg_time,
            task_time,
            inf,
            c,
            f,
            _if
        ]
        _ = writer.writerow(dict(zip(fields, results)))
        uid += 1

# Close the output file
output_file.close()
```

C Sample preprocessed data

```
uid , participant , type , feedback , trial , avg_time (ms) ,INF,C,F, IF
1 ,1 , normal , false ,1 ,458 ,0 ,31 ,3 ,3
2 ,1 , normal , false ,2 ,549 ,0 ,30 ,21 ,21
3 ,1 , normal , false ,3 ,610 ,0 ,27 ,21 ,21
4 ,1 , normal , false ,4 ,460 ,0 ,31 ,7 ,7
5 ,1 , normal , false ,5 ,547 ,0 ,19 ,8 ,8
6 ,1 , normal , false ,6 ,432 ,0 ,25 ,19 ,19
7 ,1 , normal , false ,7 ,540 ,1 ,26 ,3 ,3
8 ,1 , normal , false ,8 ,545 ,0 ,25 ,8 ,8
9 ,1 , normal , false ,9 ,517 ,0 ,24 ,12 ,12
10 ,1 , normal , true ,1 ,538 ,0 ,30 ,10 ,10
11 ,1 , normal , true ,2 ,494 ,2 ,32 ,12 ,12
12 ,1 , normal , true ,3 ,570 ,0 ,25 ,7 ,7
13 ,1 , normal , true ,4 ,529 ,0 ,19 ,0 ,0
14 ,1 , normal , true ,5 ,610 ,0 ,34 ,7 ,7
15 ,1 , normal , true ,6 ,521 ,0 ,29 ,9 ,9
16 ,1 , normal , true ,7 ,499 ,1 ,20 ,13 ,13
17 ,1 , normal , true ,8 ,569 ,0 ,30 ,7 ,7
18 ,1 , normal , true ,9 ,566 ,1 ,23 ,5 ,5
19 ,1 , zoom , false ,1 ,1232 ,0 ,36 ,3 ,3
20 ,1 , zoom , false ,2 ,931 ,0 ,26 ,1 ,1
21 ,1 , zoom , false ,3 ,1046 ,0 ,32 ,3 ,3
22 ,1 , zoom , false ,4 ,827 ,0 ,21 ,1 ,1
23 ,1 , zoom , false ,5 ,1023 ,0 ,27 ,2 ,2
24 ,1 , zoom , false ,6 ,1450 ,0 ,30 ,0 ,0
25 ,1 , zoom , false ,7 ,1189 ,0 ,36 ,1 ,1
26 ,1 , zoom , false ,8 ,1120 ,0 ,25 ,2 ,2
27 ,1 , zoom , false ,9 ,1022 ,0 ,31 ,1 ,1
28 ,1 , zoom , true ,1 ,959 ,0 ,24 ,5 ,5
29 ,1 , zoom , true ,2 ,917 ,0 ,29 ,3 ,3
30 ,1 , zoom , true ,3 ,923 ,0 ,32 ,2 ,2
31 ,1 , zoom , true ,4 ,882 ,0 ,32 ,4 ,4
32 ,1 , zoom , true ,5 ,948 ,0 ,29 ,0 ,0
33 ,1 , zoom , true ,6 ,941 ,0 ,28 ,2 ,2
34 ,1 , zoom , true ,7 ,828 ,0 ,23 ,1 ,1
35 ,1 , zoom , true ,8 ,839 ,0 ,27 ,2 ,2
36 ,1 , zoom , true ,9 ,819 ,0 ,30 ,6 ,6
37 ,2 , normal , false ,1 ,701 ,0 ,20 ,11 ,11
38 ,2 , normal , false ,2 ,781 ,0 ,39 ,13 ,13
39 ,2 , normal , false ,3 ,542 ,0 ,27 ,1 ,1
40 ,2 , normal , false ,4 ,662 ,0 ,21 ,8 ,8
```

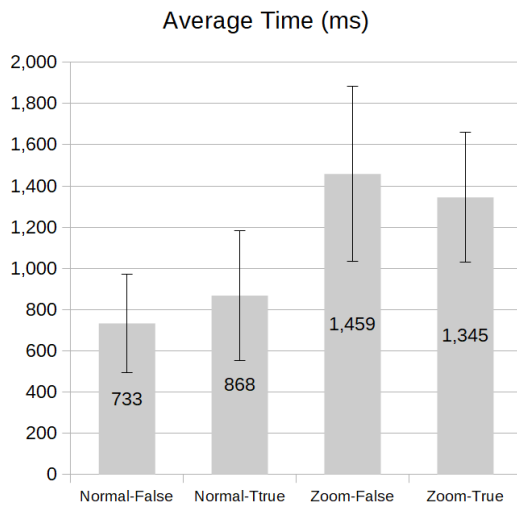# D   Metrics' descriptive statistics tables

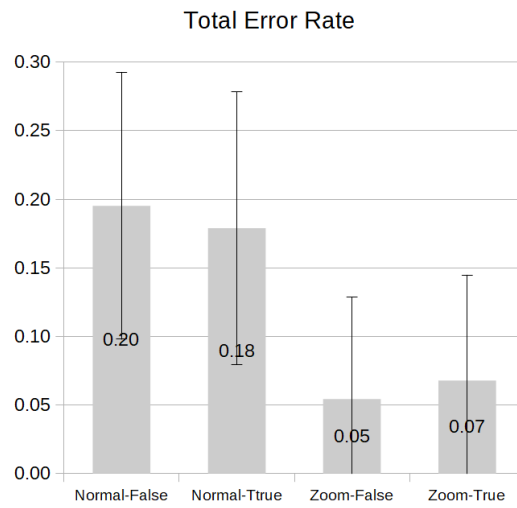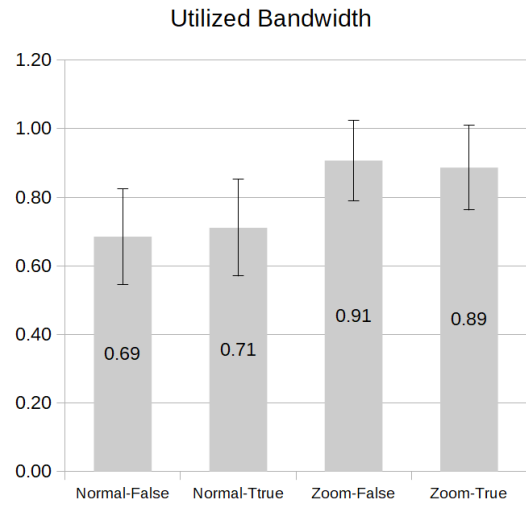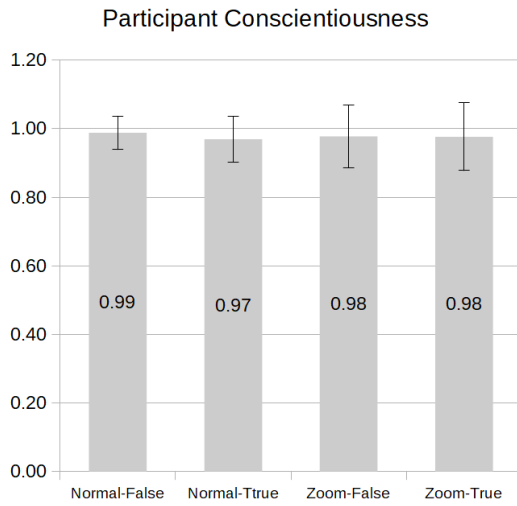| Condition	Input time (ms)	Task time (s)	MSD	KSPC	Consci-entious-ness	Utilized Band-width	Total Error Rate
Normal-false	733	30	0.0023	1.52	0.99	0.69	0.1953
Normal-true	868	35	0.0082	1.45	0.97	0.71	0.179
Zoom-false	1,459	46	0.0036	1.12	0.98	0.91	0.054
Zoom-true	1,345	43	0.0059	1.15	0.98	0.89	0.068

Table 1: Mean value of the different metrics for every condition

Condition	Input time (ms)	Task time (s)	MSD	KSPC	Consci-entious-ness	Utilized Band-width	Total Error Rate
Normal-false	238	15	0.0091	0.31	0.049	0.14	0.097
Normal-true	314	17	0.0176	0.30	0.067	0.14	0.099
Zoom-false	425	17	0.0160	0.19	0.092	0.12	0.074
Zoom-true	315	16	0.0137	0.22	0.099	0.12	0.077

Table 2: Standard deviation value of the different metrics for every condition

E   Metrics' descriptive statistics graphs

Average Time (ms)

Task Completion Time (s)

Minimum String Distance Error Rate

Key Strokes Per Character

Participant Conscientiousness

Utilized Bandwidth

Total Error Rate
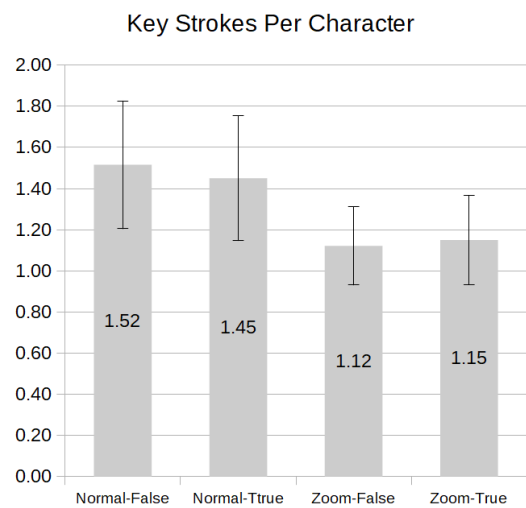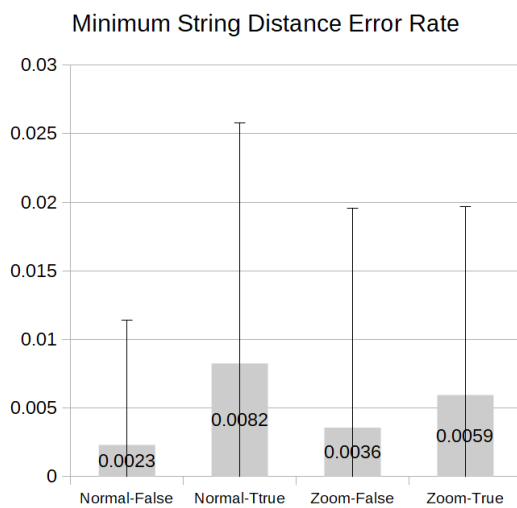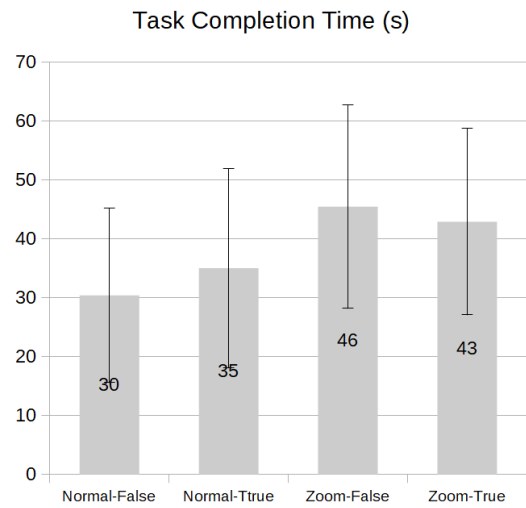
F RM-ANOVA test outputs

ANOVA_table_for_Average time (ms)

Effect	df	SS	MS	F	p
Participant	3	1116088.820	372029.607		
Input method	1	1516016.872	1516016.872	112.036	0.0018
Input method_x_Par	3	40594.588	13531.529		
Visual feedback	1	113.873	113.873	0.014	0.9133
Visual feedback_x_Par	3	24422.411	8140.804		
Input method_x_Visual feedba	1	76784.245	76784.245	1.079	0.3752
Input method_x_Visual feedba	3	213428.237	71142.746		

Data_file: anova−avg−time−ms.txt

Summary statements:
The effect of input method on average time was statistically significant (F(1, 3) = 112.036, p < .005).
The effect of visual feedback on average time was not statistically significant (F(1, 3) = 0.014, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 1.079, p > .05).

ANOVA_table_for_Task Completion Time (s)

Effect	df	SS	MS	F	p
Participant	3	2045.827	681.942		
Input method	1	562.297	562.297	163.927	0.0010
Input method_x_Par	3	10.291	3.430		
Visual feedback	1	4.122	4.122	0.175	0.7038
Visual feedback_x_Par	3	70.656	23.552		
Input method_x_Visual feedba	1	62.965	62.965	0.756	0.4486
Input method_x_Visual feedba	3	249.950	83.317		

Data_file: anova−task−time−s.txt

Summary statements:
The effect of input method on task completion time was statistically significant (F(1, 3) = 163.927, p < .005).
The effect of visual feedback on task completion time was not statistically significant (F(1, 3) = 0.175, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 0.756, ns).

ANOVA_table_for_Key Strokes Per Character

Effect	df	SS	MS	F	p
Participant	3	0.148	0.049		
Input method	1	0.501	0.501	47.898	0.0062
Input method_x_Par	3	0.031	0.010		
Visual feedback	1	0.002	0.002	0.244	0.6554
Visual feedback_x_Par	3	0.023	0.008		
Input method_x_Visual feedba	1	0.011	0.011	1.441	0.3161
Input method_x_Visual feedba	3	0.023	0.008		

Data_file: anova−key−strokes−per−character.txt

Summary statements:
The effect of input method on key strokes per character was statistically significant (F(1, 3) = 47.898, p < .01).
The effect of visual feedback on key strokes per character was not statistically significant (F(1, 3) = 0.244, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 1.441, p > .05).

ANOVA_table_for_Minimum String Distance Error Rate

Effect	df	SS	MS	F	p
Participant	3	0.000	0.000		
Input method	1	0.000	0.000	0.013	0.9158
Input method_x_Par	3	0.000	0.000		
Visual feedback	1	0.000	0.000	6.857	0.0791
Visual feedback_x_Par	3	0.000	0.000		
Input method_x_Visual feedba	1	0.000	0.000	0.649	0.4793
Input method_x_Visual feedba	3	0.000	0.000		

Data_file: anova−minimum−string−distance.txt

Summary statements:
The effect of input method on minimum string distance error rate was not statistically significant (F(1, 3) = 0.013, ns).
The effect of visual feedback on minimum string distance error rate was not statistically significant (F(1, 3) = 6.857, p > .05).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 0.649, ns).

ANOVA_table_for_Participant Conscientiousness

Effect	df	SS	MS	F	p
Participant	3	0.001	0.000		
Input method	1	0.003	0.003	31.221	0.0113
Input method_x_Par	3	0.000	0.000		
Visual feedback	1	0.001	0.001	0.483	0.5371
Visual feedback_x_Par	3	0.005	0.002		
Input method_x_Visual feedba	1	0.001	0.001	2.372	0.2212
Input method_x_Visual feedba	3	0.001	0.000		

Data_file: anova−participant−conscientiousness.txt

Summary statements:
The effect of input method on participant conscientiousness was statistically significant (F(1, 3) = 31.221, p < .05).
The effect of visual feedback on participant conscientiousness was not statistically significant (F(1, 3) = 0.483, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 2.372, p > .05).

ANOVA_table_for_Utilized Bandwidth

Effect	df	SS	MS	F	p
Participant	3	0.040	0.013		
Input method	1	0.163	0.163	123.656	0.0016
Input method_x_Par	3	0.004	0.001		
Visual feedback	1	0.000	0.000	0.038	0.8577
Visual feedback_x_Par	3	0.006	0.002		
Input method_x_Visual feedba	1	0.003	0.003	1.818	0.2703
Input method_x_Visual feedba	3	0.005	0.002		

Data_file: anova−utilized−bandwidth.txt

Summary statements:
The effect of input method on utilized bandwidth was statistically significant (F(1, 3) = 123.656, p < .005).
The effect of visual feedback on utilized bandwidth was not statistically significant (F(1, 3) = 0.038, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 1.818, p > .05).

ANOVA_table_for_Total Error Rate

Effect	df	SS	MS	F	p
Participant	3	0.018	0.006		
Input method	1	0.066	0.066	71.517	0.0035
Input method_x_Par	3	0.003	0.001		
Visual feedback	1	0.000	0.000	0.034	0.8648
Visual feedback_x_Par	3	0.003	0.001		
Input method_x_Visual feedba	1	0.001	0.001	1.385	0.3241
Input method_x_Visual feedba	3	0.002	0.001		

Data_file: anova−total−error−rate.txt

Summary statements:
The effect of input method on total error rate was statistically significant (F(1, 3) = 71.517, p < .005).
The effect of visual feedback on total error rate was not statistically significant (F(1, 3) = 0.034, ns).
The input method_x_visual feedback interaction effect was not statistically significant (F(1, 3) = 1.385, p > .05).

G   Exit survey

Exit Survey

What did you think of the zooming keyboard?

Your answer

Can you think of any improvement you would make to the zooming keyboard?

Your answer

What did you think of the normal keyboard?

Your answer

Can you think of any improvement you would make to the normal keyboard?

Your answer

Did you feel the visual feedback helped? Why?

Your answer

SUBMIT

H   Exit survey results

What did you think of the zooming keyboard?

4 responses

I think it is was really good because the screen resolution seems to be the limiting factor behind the non-zooming keyboard. Keystrokes seem to be registered more correctly than non-zooming even if I was basically double tapping the whole time

Less frustrating than the normal one because easier to selct the right letter. However, it's really slow

I find it tedious to have to tap twice to enter a letter.

i find confusing and it made my eyes hurt

Can you think of any improvement you would make to the zooming keyboard?

4 responses

Zoom faster, make the letters thicker, haptic vibration feedback

No I can't

Yes have it zoomed by default and it work like a magnifier where you can move around a magnified keyboard

being able to move the zoom if you clicked on the wrong part of the keyboard. Or having an option to lock the zoom and move around with your finger.

What did you think of the normal keyboard?

4 responses

Good but needs more ML: I would like it to auto-correct and also to understand that when I delete something it shouldn't output the same letter again. I think it was a strain to look at too (same goes for non-zooming)

It's too hard to select the right letters also hard to clic on the space bar because it's too close to the letters

It is too small

fast and easy to use

Can you think of any improvement you would make to the normal keyboard?

4 responses

Enlarge the keys once pressing like is done on iPhone keyboards! helps if you type slow as you can adjust in mid-type by holding and moving.

I would separate the space bar more from the rest of the letters

Make it bigger

no

Did you feel the visual feedback helped? Why?

4 responses

No it was more of a hindrance than anything because i got my visual feedback from the line above rather than from the feedback. It was useful for delete though, i would keep that feedback.

Yes it did. It helped me go faster because it reassured me that I was choosing the right letters.

Yes it helped because you know immediately if you make a mistake.

no, my finger was blocking the feedback on the screen so it made it even more confusing. I also think it made my typing slower because my eyes did not know where to look between the actual phrase (normal reflex) or the keyboard.

I

Study Protocol

Project Title: Controlled Experiment Comparing Text Input Methods for Smartwatches

Investigators: Antoine Viscardi, avis@cs.toronto.edu

Background and purpose of Research: The purpose of our study is to understand current or potential wearable users to help us identify the most efficient text input method for a small form factor device intended to be used by such users. More particularly, the research will compare two methods, namely a traditional keyboard, where users enter text by tapping on the keys, and a zooming keyboard, where users first tap on the keyboard to zoom into a section and then tap on the desired key.

Participant selection and eligibility: Four participants will be chosen from my personal social circle. They will be identified via a deliberate process and selected as to cover a wide demographic range in order to capture the numerous market segments attracted to wearable devices. In general they will be characterized by a decent level of technology literacy and their age will range from 19 to 55 years old.

Procedure: The participants will be briefed about the purpose of the study, explain the attached consent form to them, and ensure that they consent to participate and sign the consent form. The participants will then be engaged in a 45-minute long quantitative experimental evaluation. The evaluation will take place in a quiet room as to avoid any distractions that could influence the results. Also, with the permission of the participants, the device's inputs will be recorded throughout the session.

Voluntary Participation & Early Withdrawal: The participation in this study is entirely voluntary, and participants are free to cease participation at any time, for any reason, without the need to give any explanation. At their request, we will delete any of their data and it will not be used in our analysis or any subsequent reports or presentations.

Relationships: The participants are friends or acquaintances.

Risk and benefit: There are no anticipated risks associated with participation in this study, beyond those associated with everyday use of computer (e.g. participants may feel that they have wasted their time). The only benefit will be to contribute to the education of the investigators.

Compensation: Participants will receive no compensation.

Information sought: The information to be sought is described in the attached task description sheet.

Privacy and confidentiality: Information will be kept confidential by the investigators. Names or other identifying or identified information will not be kept with the data. The only other use will be to include excerpts or copies in the assignment submitted, but names and other identifying or identified information will not be submitted.

J

Consent Form

Consent Form: Controlled Experiment Comparing Text Input Methods for Smartwatches

I hereby consent to participate in a study conducted by Antoine Viscardi for an assignment in University of Toronto Computer Science 428, Human-Computer Interaction.

I agree to participate in this study the purpose of which is to identify the most efficient text input method for a small form factor device, intended to be used by current or potential wearable users.

I understand that

- the procedure to be used is a 30-minute long quantitative experimental evaluation.

- I will receive no compensation for my participation.

- I am free to withdraw before or any time during the study without the need to give any explanation.

- all materials and results will be kept confidential, and, in particular, that my name and any identifying or identified information will not be associated with the data.

_____

Participant's Printed Name

_____

Participant's Signature

_____

Date

_____

Experimenter Name

_____

Experimenter's Signature

K

TASK DESCRIPTION SHEET

The task consist in typing a sentence displayed above the keyboard. This simple task will be repeated nine times for every condition.

There is a total of four conditions. The first condition features the traditional keyboard without visual feedback. The second condition features the traditional keyboard with visual feedback. That is, the tapped letter is swiftly displayed on top of the keyboard. The third condition features the zooming keyboard without visual feedback. The fourth condition features the zooming keyboard with visual feedback. The participants will not necessarily undergo the conditions in that order.

Sentences will not include uppercase letters nor punctuation marks. The sentences will be randomly selected from a set of 500 sentences published by MacKenzie and Soukoreff[1].

For each trial, the target sentence and the transcribed sentence will be recorded. The stream of inputs and its associated timestamp will also be recorded by the software for each trial. This will allow to compute metrics as described by Soukoreff and MacKenzie and their paper on metrics for text entry research[2]. Moreover, the timestamps will allow to compute time deltas between every input and, ultimately, to perform statistical analysis regarding the text entry speed.

1   MacKenzie, I. S., & Soukoreff, R. W. (2003). Phrase sets for evaluating text entry techniques. *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems - CHI 2003*, pp. 754-755. New York: ACM.

2   Soukoreff, R. W., & MacKenzie, I. S. (2003). Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric. *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI 2003*, pp. 113-120. New York: ACM.

L   Consent forms

CONSENT FORM

Consent Form: Controlled Experiment Comparing Text Input Methods for Smartwatches

I hereby consent to participate in a study conducted by Antoine Viscardi for an assignment in University of Toronto Computer Science 428, Human-Computer Interaction.

I agree to participate in this study the purpose of which is to identify the most efficient text input method for a small form factor device, intended to be used by current or potential wearable users.

I understand that

- the procedure to be used is a 30-minute long quantitative experimental evaluation.

- I will receive no compensation for my participation.

- I am free to withdraw before or any time during the study without the need to give any explanation.

- all materials and results will be kept confidential, and, in particular, that my name and any identifying or identified information will not be associated with the data.

THOMAS HOLLIS
_____
Participant's Printed Name

_____
Participant's Signature

15 Feb 2019
_____
Date

Antoine Viscardi
_____
Experimenter Name

_____
Experimenter's Signature

Consent Form

Consent Form: Controlled Experiment Comparing Text Input Methods for Smartwatches

I hereby consent to participate in a study conducted by Antoine Viscardi for an assignment in University of Toronto Computer Science 428, Human-Computer Interaction.

I agree to participate in this study the purpose of which is to identify the most efficient text input method for a small form factor device, intended to be used by current or potential wearable users.

I understand that

- the procedure to be used is a 30-minute long quantitative experimental evaluation.

- I will receive no compensation for my participation.

- I am free to withdraw before or any time during the study without the need to give any explanation.

- all materials and results will be kept confidential, and, in particular, that my name and any identifying or identified information will not be associated with the data.

Tony Viscardi
Participant's Printed Name

Participant's Signature

17 FEV 2019
Date

Antoine Viscardi
Experimenter Name

Experimenter's Signature

CONSENT FORM

Consent Form: Controlled Experiment Comparing Text Input Methods for Smartwatches

I hereby consent to participate in a study conducted by Antoine Viscardi for an assignment in University of Toronto Computer Science 428, Human-Computer Interaction.
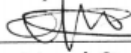
I agree to participate in this study the purpose of which is to identify the most efficient text input method for a small form factor device, intended to be used by current or potential wearable users.

I understand that

- the procedure to be used is a 30-minute long quantitative experimental evaluation.

- I will receive no compensation for my participation.

- I am free to withdraw before or any time during the study without the need to give any explanation.

- all materials and results will be kept confidential, and, in particular, that my name and any identifying or identified information will not be associated with the data.

CLÉMENTINE MATTESCO
Participant's Printed Name

Participant's Signature

17-02- 2019
Date

Antoine Viscardi
Experimenter Name

Experimenter's Signature

CONSENT FORM

Consent Form: Controlled Experiment Comparing Text Input Methods for Smartwatches

I hereby consent to participate in a study conducted by Antoine Viscardi for an assignment in University of Toronto Computer Science 428, Human-Computer Interaction.

I agree to participate in this study the purpose of which is to identify the most efficient text input method for a small form factor device, intended to be used by current or potential wearable users.

I understand that

- the procedure to be used is a 30-minute long quantitative experimental evaluation.

- I will receive no compensation for my participation.

- I am free to withdraw before or any time during the study without the need to give any explanation.

- all materials and results will be kept confidential, and, in particular, that my name and any identifying or identified information will not be associated with the data.

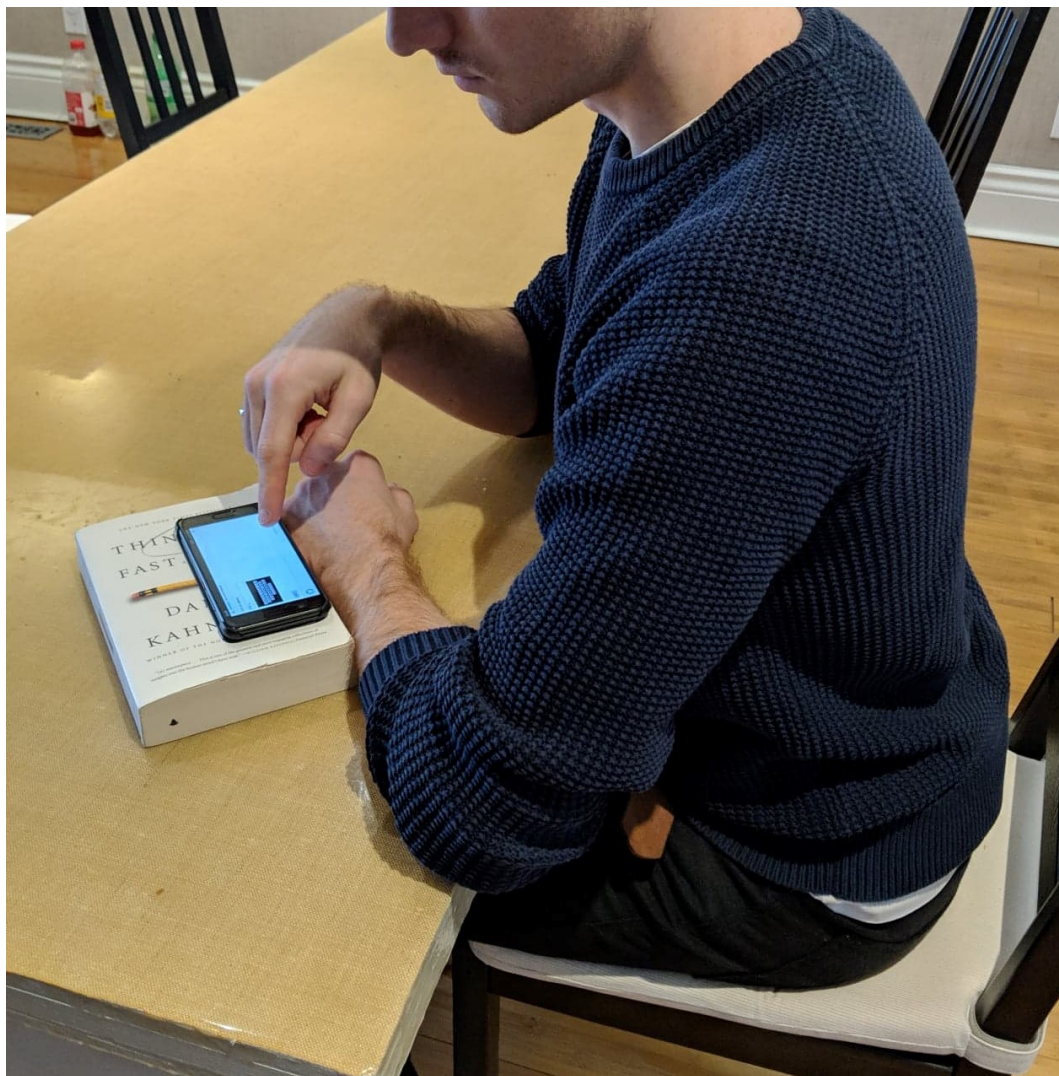_Gabrielle Viscardi_
Participant's Printed Name

Gabrielle Viscardi
Participant's Signature

Feb 19, 2019
Date

Antoine Viscardi
Experimenter Name

_A. V._
Experimenter's Signature

M   Experimental setup

N Text entry values and metrics

The values and metrics presented here were originally defined by Soukoreff and MacKenzie and their paper on metrics for text entry research[5].

From the recorded target sentence, input stream and final transcribed text different values can be extracted to help us derive relevant metrics for the evaluation of text entry methods. Only the metrics used in this study are presented below.

The extracted values are the number of correct keystrokes (C), the number of fixes (F), the number of incorrect keystrokes and not fixed (INF) and the number of incorrect keystrokes and fixed (IF). We can extract C and INF directly from the transcribed text by computing the Levenshtein distance between the transcribed text and the target. INF corresponds to that distance while C corresponds to the number of characters in the transcribed text that match characters in the target sentence. We can extract F simply by counting the number of editing keystrokes present in the input stream. In our case we only have one such editing function, the backspace keystroke. We can extracted IF by counting the number of keystrokes (in the input stream (excluding editing functions) not present in the transcribed text. Concretely, this can be done by computing the Levenshtein distance between the concatenated input stream excluding the editing functions and the transcribed text.

From the four values described above we can compute different relevant metrics for the evaluation or comparisons of text entry methods. These metrics and their computation are summarized in the following table.

Metric	Computation	Role
Minimum String Distance Error Rate (MSD)	$$\frac{INF}{C + INF} \times 100\%$$	A higher MSD means that users using a text entry method are more prone to making errors.
Key Strokes per Character (KSPC)	$$\frac{C + INF + IF + F}{C + INF}$$	Indication of how efficient the text entry method is
Participant Conscientiousness	$$\frac{IF}{IF + INF}$$	Measures how efficient participants are at catching their mistakes when using the text entry method
Utilized Bandwidth	$$\frac{C}{C + INF + IF + F}$$	Good overall measure of the efficiency of the text entry method.
Total Error Rate	$$\frac{INF + IF}{C + INF + IF} \times 100\%$$	General metric to evaluate how error prone users are while using the text entry method.