Effects of Text Input Latency on Performance and Task Load

Andreas Schmid
University of Regensburg
Regensburg, Germany
andreas.schmid@ur.de

Michael Bierschneider
University of Regensburg
Regensburg, Germany
michael.bierschneider@stud.uni-regensburg.de

Johannes Hoffmann
University of Regensburg
Regensburg, Germany
johannes-maximilian.hoffmann@stud.uni-regensburg.de

Yu Liu
University of Regensburg
Regensburg, Germany
yu.liu@stud.uni-regensburg.de

Raphael Wimmer
University of Regensburg
Regensburg, Germany
raphael.wimmer@ur.de

ABSTRACT

The latency of a text editor describes how long it takes from pressing a key to the corresponding letter appearing on the screen. It is well known that high latency affects how quickly authors can write and edit texts. In order to quantify the effects of latency on users' performance and task load, we conducted a study in which 31 participants had to re-type or correct provided texts with a physical keyboard. Each participant completed each task once with a low latency of 20 ms and once with a high latency of 200 ms. We found that latency had no significant effect on users' performance during the copy task, but correcting texts was affected significantly by high latency. Additionally, our results suggest that fast typers are more likely to notice latency than slow typers. Our findings regarding the effects of text input latency on users' performance contributes to the existing body of research on latency in interactive systems.

CCS CONCEPTS

• **Human-centered computing** → *Empirical studies in HCI*; *Keyboards*; **Text input**.

KEYWORDS

latency, text input, typing

ACM Reference Format:
Andreas Schmid, Michael Bierschneider, Johannes Hoffmann, Yu Liu, and Raphael Wimmer. 2023. Effects of Text Input Latency on Performance and Task Load. In *International Conference on Mobile and Ubiquitous Multimedia (MUM '23), December 03–06, 2023, Vienna, Austria.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3626705.3627784

1 INTRODUCTION

When interacting with computers, users are constantly affected by latency, the time delay between user input and system response. Due to processing times and polling cycles, this end-to-end latency is comprised of several partial latencies: input latency caused by input device and USB polling rate [36], latency contributed by operating system and application frameworks [32], and applications such as text editors [13]. The effect of latency is particularly noticeable in real-time applications such as video games [3, 18, 24]. As the Model Human Processor [6] illustrates, humans need to perform multiple operations to complete tasks. After every operation, users wait for the system's response in order to adapt their behavior and plan the next steps. High latency disrupts this cycle, delays users' actions, and therefore leads to poor user experience and performance [26].

Even though early work by Shneiderman et al. [34] recommends an end-to-end latency of 50 to 150 ms for simple tasks such as typing, significantly smaller amounts of latency can be perceived by users in certail situations. Ng et al. [28] found that latency below 20 ms can be noticed on touch screens and Jota et al. [20] found that high latency leads to lower user performance in dragging tasks. Accordingly, Attig et al. [2] concluded that previous guidelines recommending a maximum latency of 100 ms are outdated. To gain an advantage and reduce end-to-end latency, the industry offers hardware peripherals with lower latency or higher refresh rate on monitors [22, 24].

Even though there is a vast amount of research on the influence of latency on users in different tasks, such as target selection or playing video games, to our knowledge, there have been no user studies investigating the influence of latency on text input. We consider this as a significant research gap, as typing and editing text is an elemental building block for many more abstract tasks. Even though text input can in theory be broken down to individual keystrokes and therefore described by theoretical models, in practice, this is only applicable for inexperienced *hunt-and-peck typists* [10]. For experienced typers using the 10 finger system, typing becomes second nature and individual keystrokes merge to a more complex task. Additionally, as most typers use their own, idiosyncratic typing style [29], theoretical models might not be applicable to all users.

Therefore, we conducted a user study investigating the influence of text input latency on users' performance when copying and correcting texts with a physical keyboard. Our results suggest that high latency affects users more during the correction task, where we could find significant effects in terms of task completion time and perceived task load. For the copy task, we could not find a significant

This work is licensed under a Creative Commons Attribution International 4.0 License.

MUM '23, December 03–06, 2023, Vienna, Austria
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0921-0/23/12.
https://doi.org/10.1145/3626705.3627784

effect of latency on users' typing speed, however, frustration and effort were significantly higher when typing with latency. Lastly, an exploratory analysis of our data indicates that fast typers are more likely to notice latency than slow typers.

2 RELATED WORK

In this section, we first describe different typing studies and the importance of visual feedback during typing. Additionally, we discuss the effect of latency in interactive systems. Finally, we analyze different methods for measuring a system's end-to-end latency.

2.1 Typing Studies

Studies show that typing with a keyboard is a complex process that consumes working memory resources [1, 19, 21]. The visual attention and sensorimotor action are not uniform and continuous in time and space. The typing process consists of a *higher-level writing process* such as planning or editing the typed content and a *lower-level writing process* such as spelling words and pressing keys on a keyboard [19].

Those sub-processes are also classified as the outer and inner loop of typing. The *outer loop* is responsible for transforming text and thoughts to words and checking the accuracy of the keystrokes entered by the inner loop. Meanwhile, the *inner loop* controls the hands and is responsible for converting intended words into keystrokes [25]. It ensures that the typed text is correct and makes any necessary corrections as needed. During typing, the outer loop monitors the keyboard input on the screen and requires the inner loop to ensure that the text is correctly entered [25].

Dahm and Rieger [10] identified two different categories of typists: *ten-finger typists* use all their fingers and type without looking at the keyboard because they have a fixed assignment of finger to key. *Hunt-and-peck typists* often do not have a fixed assignment of the finger and do not type with all their fingers [10]. Therefore, their typing behavior is mainly dictated by the outer loop. In contrast, Hunt-and-peck typists are unable to type without looking at the keyboard and their visual attention to the keyboard distracts them from the screen. Accordingly, they pay less attention to the screen than ten-finger typists do [10]. Their typing behavior is therefore relying more on the inner loop. However, most people fall somewhere between the two extremes of ten-finger typing and hunt-and-peck typing [19].

Visual feedback is important for typing text, as typists use it to determine if written content is correct [13]. Skilled typists require a smaller amount of working memory resources to handle the lower-level writing process, as they presumably execute this process automatically [1, 19]. They will focus their visual attention primarily on the screen, whereas unskilled typists look at the keyboard more often because they require more resources to perform the lower-level writing process [27]. Compared to free typing, copy typing generally requires more attention to the text [29].

As latency disrupts the feedback loop of user input and system response [6, 26, 36], it can slow down typing by affecting the inner loop [13].

2.2 The Impact of Latency on Interactive Systems

In their seminal work, MacKenzie and Ware [26] have shown that latency directly affects users' performance in pointing tasks, reducing task completion time and throughput. The effect of latency is even more apparent with direct input devices such as touch screens. Numerous studies focusing on touchscreen technology have highlighted the significant impact of latency on both user experience and task performance. Notably, Jota et al. [20] established a clear connection between latency and user performance. Their research demonstrated a consistent decline in user performance as latency increased. Specifically, they observed that participants were able to perceive feedback latency within the range of 20 to 100 milliseconds, with a noticeable drop in user performance occurring when latency exceeded 25 ms in dragging tasks. Ng et al. [28] further investigated dragging tasks and discovered that participants could detect dragging delays of less than 20 ms, with the perception threshold being as low as 2 ms in certain situations. This finding further underscores the sensitivity of users to latency, particularly in scenarios involving dragging interactions. Deber et al. [12] contributed to existing research on touch screen latency by taking into consideration both dragging and tapping, utilizing both direct and indirect input methods. Their investigation revealed that users were less likely to perceive tapping latency as acutely as dragging latency when using direct input. By synthesizing these studies, it becomes evident that latency is a crucial factor influencing user interactions with touchscreen devices, with the specific effects varying based on the type of task, input method, and latency magnitude.

There are also several studies on the impact of latency in real-time games: Beigbeder et al. [3] investigated the effects of packet loss and latency on user performance for the popular First-Person Shooter (FPS) game *Unreal Tournament 2003*. Their findings indicate a slight decrease in player performance as latency increased. Players consciously switch to weapons requiring less precise aiming when experiencing high latency. As for player experience, players do not notice the delay when the connection latency is as little as 75 ms. They also do not consider that the game is less entertaining when the latency exceeds 100 ms. A study by Ivkovic et al. [18] investigated the effect of local latency on player performance when aiming at targets. They also found the player performance for both, aiming and tracking, degrades significantly when the local latency increases. Moreover, local latency as little as 41 ms causes significant degradation of targeting performance. Players need more time to acquire more challenging targets when the local latency is over 41 ms and demand more time to achieve less challenging targets when the local latency is over 11 ms. Additionally, Ivkovic et al. [18] investigated the end-to-end latencies of typical gaming setups. Local latency of those systems ranged from 23 to 243 ms.

Lastly, Tolia et al. [35] studied the effect of bandwidth and simulated network latency on the time between user actions in remotely controlled applications. They found that for highly interactive applications, such as image manipulation software or tracking changes in a word processor, high latency can have a strong effect on users' behavior. However, Tolia et al. did not conduct a controlled user study but based their experiments on pre-recorded interaction logs.

2.3 Latency Measurement

In order to reduce the latency of interactive systems, it is important to identify bottlenecks by measuring the partial latencies contributed by individual components of a system in addition to the systems's end-to-end latency [36]. Depending on input and output modalities, different methods for measuring latency are required. A straight-forward method for measuring a system's end-to-end latency is using a high speed camera to record an interaction process and then determine start and end timestamps by analyzing the resulting slow motion video [14]. Ng et al. [28] conducted informal tests with high-speed cameras on various iOS, Android, and Windows devices. They found that the typical latency in commercial devices is between 50 and 200 ms. They also used a high-speed camera in their study to measure the touchscreen dragging latency. They placed a ruler on top of the device and moved the finger across the screen along the edge of the ruler. The movement of the finger and the graphical feedback was captured by the high-speed camera. The frames were used to directly observe the dragging latency, which is correlated to the distance between the finger and the on-screen feedback. Yet, it is worth noticing, that the accuracy is only 4 ms, even with a high-speed video camera running at 250 fps [4]. Therefore, automated latency measuring methods based on fast electrical components can provide more accurate results [31]. A similar measuring method was introduced by NVIDIA. Their *NVIDIA Reflex Analyzer* detects mouse clicks measuring the time between a mouse click and the resulting change on the screen [33].

2.4 Summary

In summary, previous research on users' typing behavior has shown that the process and involves different sensorimotor processes, as well as working memory resources. Visual feedback is an essential component of typing, allowing users to continuously check the result of their input. However, latency can disrupt this visual feedback cycle, leading to interference with user input [13, 36]. While many experiments have investigated the effects of latency on touchscreens and gaming performance, there is no published research on the effects of text input latency on performance and task load when typing on a physical keyboard.

3 METHOD

In our study, we investigate the effects of text input latency on performance and task load. To ensure valid results, we measured the end-to-end latency of our apparatus. We define this end-to-end latency as the time that elapses between pressing a key on the keyboard and a character being displayed on the monitor. We also took our computer hardware [36] and text editor [13] into consideration, as they also influence the end-to-end latency.

Our study consists of two tasks: copying texts and correcting texts with error characters. Participants used a physical keyboard with the QUERTZ layout to absolve those tasks. We standardized texts for our study by selecting similar paragraphs from Wikipedia[1]. We used the readability index (LIX) [5] to ensure we had selected texts with similar length and difficulty.

[1]https://de.wikipedia.org/wiki/Wikipedia:Lesenswerte_Artikel

3.1 Apparatus

For our study, we used an *HP Pavilion Gaming 790*[2] desktop PC running Debian Buster 5.10 with proprietary *Nvidia* graphics drivers (version 470.103.01). In terms of periphery, we used an *ASUS ROG Strix XG248Q* at 1920 × 1080 pixels with 240 Hz and 1 ms response time (Gray-to-Gray), a *Logitech G15* gaming mouse[3], and a *Logitech G213* gaming keyboard[4]. We set the USB polling rate on the operating system to 1000 Hz to avoid a bimodal distribution of input device latency [36]. We picked the *Logitech G213* gaming keyboard because it has a low average latency and standard deviation (Fig. 2) compared to keyboards measured by Wimmer et al. [36]. By using Mech-Dome keys, the G213 keyboard provides a tactile response similar to that of a mechanical keyboard[5].

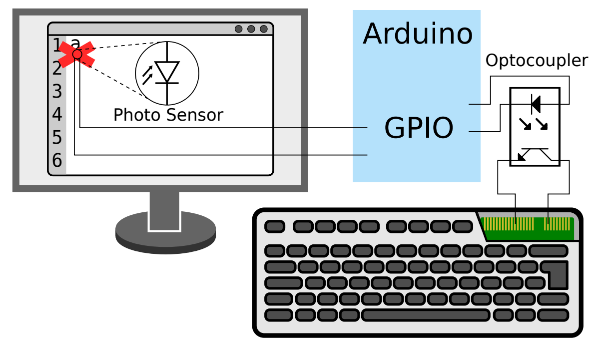

Figure 1: Setup for latency measurements to validate the apparatus. An Arduino microcontroller triggers key presses on a keyboard via an optocoupler connected to the keyboard's ASIC. Once a key press is triggered, a character appears in the text editor displayed on the monitor. Using a photo diode attached to the monitor, a microcontroller senses a brightness change once a character appears. This way, the microcontroller can measure the time between a key press and a character appearing on the monitor.

In a preliminary study, we compared different text editors as they affect the end-to-end latency according to Fatin [13]. We found that *gedit*[6] shows the best performance with our setup.

As illustrated in Figure 1, we measured the end-to-end latency of our system following the procedure explained by Schmid and Wimmer [31]: We opened the gedit editor on our system and set the font size to the maximum so that the photo sensor could easily detect a character. We then taped the photo sensor to the monitor, where characters generated from the microcontroller are displayed. We connected the microcontroller to the keyboard controller so that it could trigger key events. After also connecting the microcontroller to the computer, we started the measurement by physically turning on the microcontroller. Results in Figure 2 show an average end-to-end latency of 19.97 ms (SD=2.59 ms) for 75 measurements. Therefore, we set the base latency for our study to 20 ms.

[2]Intel i7-8700 (3.2 GHz), Nvidia GTX 1080, 16 GB DDR4 RAM
[3]input device latency: 2.17 ms (SD: 0.3 ms) as reported by Wimmer et al. [36]
[4]input device latency: 2.55 ms (SD: 0.34 ms)
[5]https://www.logitechg.com/en-gb/products/gaming-keyboards/g213-rgb-gaming-keyboard.html
[6]https://help.gnome.org/users/gedit/stable/

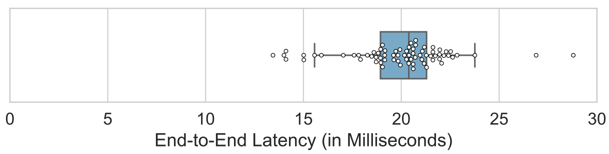

Figure 2: End-to-end latency of our apparatus. Latency was measured from an electrically triggered key press until a character appears on the screen.

3.2 Text Paragraphs

For the template texts used in our study, we selected paragraphs from articles in the German Wikipedia. We collected twenty texts with similar difficulty based on the LIX score (min=50, max=59). Word count of our texts ranged from 70 to 90 words. For the correction task, 15 error characters were introduced to each text (Figure 3), which were later highlighted in green in the gedit editor using the *yals* plugin. This way, we made sure that task time was not confounded by participants searching for errors in the text.

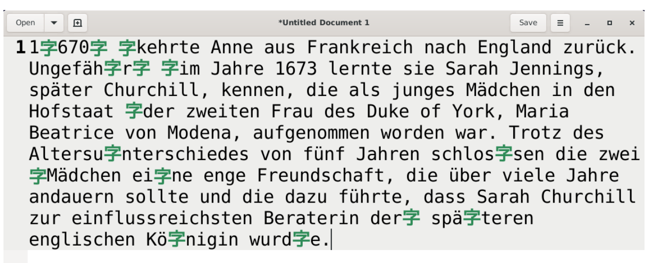

Figure 3: A template text displayed in gedit. For the correction task, each template text contains 15 error characters, represented as 字, which need to be deleted rather than replaced. To make them easier to spot, the error characters are highlighted in green.

3.3 Study Design

We designed a within-groups lab study to evaluate the effect on user experience when typing with added end-to-end latency. Participants were given two tasks: copying texts (copying task) and correcting error characters from template texts (correcting task). For the copy task, participants were presented a template text which they had to type into the text editor. For the correction task, the text editor contained a template text with introduced error characters, which they had to remove from the text. Participants were not allowed to use the mouse or the *replace all* function but could use Arrow Keys, Ctrl + Arrow Keys, End and Pos1. Both tasks were repeated five times for both latency conditions. We randomly assigned one of our twenty template texts to each repetition to counteract potential effects of text complexity on the overall results. Accordingly, each participant worked with each template text exactly once.

For our two latency conditions, we chose 20 ms (which is the end-to-end latency of the unmodified system) as the condition with low latency, and 200 ms as the condition with high latency. Even though 200 ms is higher than the end-to-end latency of most modern computers, we selected this value to ensure that an effect is found in case it exists. Furthermore, when using web-based text editors in combination with a bad internet connection, high text input latency can occur in practice. Text editor latencies (excluding the latency of the remainder of the system) of over 100 ms have also been reported by Fatin [13] for worst-case scenarios. For the condition with high latency, latency was added to each input event with a program similar to Liu and Claypool's *EvLag* [23].

After each task, participants were asked to answer the short version of the NASA TLX questionnaire [17]. Additionally, we saved all texts typed and corrected by participants and logged key presses and timestamps during the study.

After welcoming our participants to the laboratory, we explained the purpose and the procedure of the study, as well as which data we collected. After participants had given informed consent, we started the study with the first task. The order of tasks and latency conditions was counterbalanced using a latin square. Between tasks, participants were allowed to take a break and were offered sweets. After completing both tasks in both latency conditions, we asked participants for demographic data and conducted a semi-structured post-study interview. In this interview, we asked participants whether they noticed the added latency while typing and whether they changed their behavior if they did notice latency.

The whole study procedure took 40 to 60 minutes per participant.

3.4 Participants

We recruited 31 participants (20 identified as male, 11 identified as female) with convenience sampling, using the online forum of our university. They were aged 19 – 33 (mean: 25.7) and a majority of them were computer science students.

4 RESULTS

In this section, we present the results of our experiment. Every inferential analysis was preceded by a Shapiro Wilk test to test for normal distribution of residuals. If data was normally distributed, we used a dependent samples two-tailed t-test. If normal distribution was violated, we used the non-parametric Wilcoxon signed rank test. We use Cohen's d [9] as a measure for effect size.

4.1 Copy Task

We first compare typing speed during the copy task between the two latency conditions. Typing speed is operationalized as the average time needed to type one correct character and can therefore be calculated by dividing the the task completion time by the number of characters in the copied text. Of the three repetitions of the copy task in each latency condition, we discarded the first repetition as a warm-up round for participants to get used to the current latency. We aggregated the remaining two repetitions by calculating the mean typing speed for each participant.

Mean time per character in the low latency condition was 290.9 ms (SD: 80.4 ms). Mean time per character in the high latency condition was 302.7 ms (SD: 83.0 ms). A dependent samples t-test could not find a significant difference between typing speed for the two latency conditions ($p = 0.086, d = 0.324$, Fig. 4).

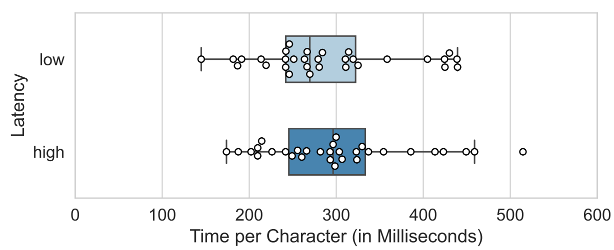

Figure 4: Typing speed under different latency conditions for the copying task.

To calculate the fraction of incorrect key presses during the copying task, we first compared the template text to the texts typed by participants. We then quantified the remaining errors in the text using the Damerau–Levenshtein distance [11]. As correcting each error would require a minimum of two key presses (one to delete the wrong character, one to type the correct character), we multiplied the result by two. This number was added to the total number of key presses made by the participant when copying the text. By dividing the number of characters in the template text by this number and subtracting the result from one, we derived the fraction of incorrect key presses.

Analogous to our evaluation of time per character, we discarded the warm-up round for each latency condition and aggregated data for the remaining rounds. We could not find a significant difference in the fraction of incorrect key presses when copying texts with low (mean: 0.11, SD: 0.04) or high latency (mean: 0.11, SD: 0.04) using a dependent sample t-test ($p = 0.388, d = 0.160$, Fig. 5).

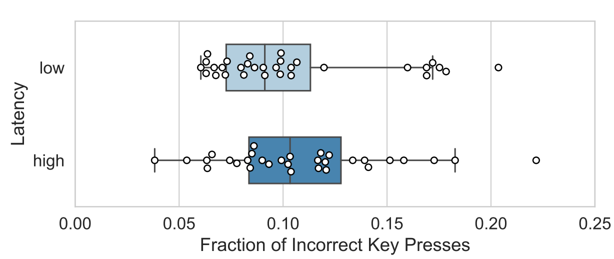

Figure 5: Error rate under different latency conditions for the copying task.

We measured participants' perceived task load during the copy task after each latency condition using the NASA TLX. Table 1 lists means and standard deviations, as well as test results for each subscale. A repeated measures t-test shows that participants perceived higher Effort when copying texts with high latency ($p = 0.022, d = 0.442$). A Wilcoxon signed rank test shows that participants perceived higher Frustration when copying texts with high latency ($p = 0.004, d = 0.484$). Both effect sizes were *medium*. We could not find significant effects for the remaining subscales or the total TLX score.

Table 1: NASA-TLX scores under different latency conditions for the copying task. Values in latency columns are mean (SD). The last column shows Cohen's d as a measure of effect size. For subscales marked with a dagger (†), we employed a Wilcoxon signed rank test. In all other cases, we employed a dependent-sample two-tailed t-test.

TLX subscale	Low Latency	High Latency	p	d
Mental	33.81 (20.46)	38.26 (22.28)	0.096	0.313
Physical†	22.23 (19.55)	20.77 (21.43)	0.387	0.096
Temporal†	30.26 (20.66)	32.13 (22.69)	0.724	0.09
Performance	66.29 (22.16)	64.55 (20.26)	0.473	0.133
Effort	31.45 (19.65)	38.84 (19.18)	0.022	0.442
Frustration†	20.94 (20.45)	31.32 (23.58)	0.004	0.484
Total†	34.16 (11.36)	37.65 (12.99)	0.081	0.341

4.2 Correction Task

For the correction task, we analyzed participants' task completion time, as well as results from the NASA TLX. Again, we discarded the first repetition for each latency condition so participants could get used to the new latency without affecting our data. We aggregated task completion time – the time it took to remove all 15 error characters – of the remaining four repetitions for each participant.

Using a dependent sample t-test, we could find a highly significant difference with a *very large* effect size ($p < 0.001, d = 1.673$) in task completion time between the two latency conditions. Participants were significantly faster in correcting texts with low latency (mean: 38.9 s, SD: 8.3 s) than with high latency (mean: 46.5 s, SD: 9.0 s). Results are depicted in Fig. 6.

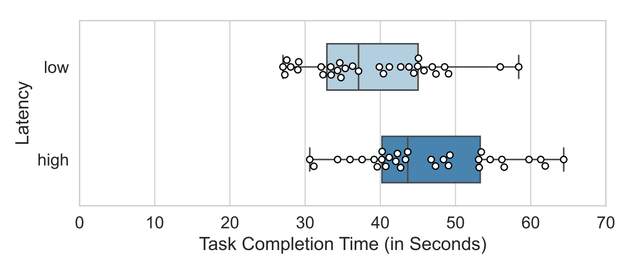

Figure 6: Task completion time under different latency conditions for the correction task.

Again, participants were asked to answer the NASA TLX questionnaire after both latency conditions. We could find significant differences between latency conditions for each subscale, as well as the total TLX score (all $p < 0.05$). Effect size was *Small* for Physical Demand, *Medium* for Temporal Demand and Effort, *Large* for Performance, Frustration and Total, and *Very Large* for Mental Demand. Detailed results and descriptive statistics can be seen in Table 2.

Table 2: NASA-TLX scores under different latency conditions for the correction task. Values in latency columns are mean (SD). The last column shows Cohen's d as a measure of effect size. For subscales marked with a dagger (†), we employed a Wilcoxon signed rank test. In all other cases, we employed a dependent-sample two-tailed t-test.

TLX subscale	Low Latency	High Latency	*p*	*d*
Mental†	21.26 (24.43)	36.45 (25.61)	< 0.001	0.925
Physical†	14.71 (20.49)	19.39 (19.13)	0.048	0.345
Temporal	24.97 (23.19)	30.90 (23.26)	0.014	0.476
Performance	83.71 (16.06)	75.29 (17.50)	0.001	0.669
Effort†	25.26 (24.33)	31.42 (22.63)	0.042	0.408
Frustration†	13.55 (17.08)	26.61 (23.98)	0.001	0.739
Total	30.58 (15.18)	36.68 (14.54)	< 0.001	0.755

4.3 Control Variables

To control for learning effects and exhaustion, participants absolved control rounds before and after the copy task and the correction task. No additional latency was introduced to those control rounds. Furthermore, we used the NASA TLX to measure participants' perceived task load for each control round. By comparing pre-task and post-task control rounds, we can observe to which degree participants' performance changed over the course of the experiment.

For the copy task, we found a significant difference ($p = 0.03, d = 0.423$) in time per character between pre-task control round (mean: 312.9 ms, SD: 102.0 ms) and post-task control round (mean: 292.0 ms, SD: 83.0 ms). However, we could not find a significant difference regarding the fraction of incorrect key presses ($p = 0.474, d = 0.132$). As revealed by a Wilcoxon signed rank test, participants perceived significantly higher Frustration in the post-task control round of the copy task ($p = 0.015, d = 0.511$). We could not find significant effects for the other TLX subscales or the total TLX score.

For the correction task, a repeated measures t-test ($p < 0.001, d = 0.975$) has shown that participants were significantly faster in the post-task control round (mean: 37.7 s, SD: 9.2 s) than in the pre-task control round (mean: 51.3 s, SD: 16.2 s). However, we could not find a significant effect on total task load ($p = 0.471, d = 0.133$) or any of the TLX subscales (all $p > 0.4$, all $d < 0.19$).

4.4 Post-Experiment Interview

Once both tasks were completed, we conducted a short interview with the participants as described in subsection 3.3. Overall, participants stated that latency affected them more in the correction task than in the copy task. According to our participants, they tended to focus on the template texts and the keyboard during the copying task, whereas they looked at the editor window during the correction task. According to some participants, the change in latency between conditions was more noticeable than the latency itself. They had the feeling that they would type slower and more tentative, and paid more attention while being affected by latency. However, they also pointed out that they were able to adapt to changes in latency relatively quickly. When typing with higher latency, they also experienced that holding or quickly tapping arrow keys could lead to overshooting.

4.5 Exploratory Analysis: Latency Noticed Versus Not Noticed

During the short post-experiment interview, we asked participants whether they noticed a higher than normal latency at any point during the experiment. We split the full sample into two groups corresponding to the answers given. Twenty people responded that they did notice a higher latency, and eleven people responded that they did not. Because of the uneven distribution of those two groups, we only performed an exploratory analysis of this data. We therefore only report descriptive statistics. Even though we can not draw reliable conclusions from this exploratory analysis due to the small and imbalanced sample, our data indicates a clear difference between the two groups. Therefore, this analysis can be a solid starting point for future research.

During the copy task, participants who did notice latency were clearly faster than those who did not (Fig. 7). In the low latency condition, mean time per character for participants who did not notice latency was 347.9 ms (SD: 80.1 ms) opposed to 259.6 ms (SD: 65.3 ms) for participants who did notice latency. Similarly, in the high latency condition, mean time per character for participants who did not notice latency was 349.4 ms (SD: 96.9 ms) and thus clearly higher than the 277.0 ms (SD: 65.9 ms) for participants who did notice latency.

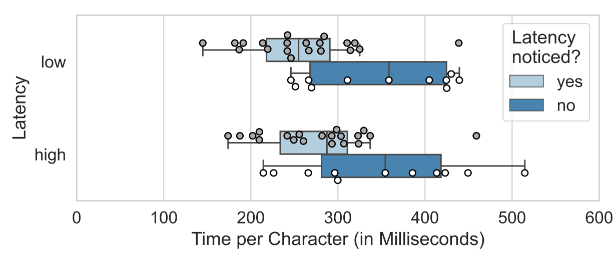

Figure 7: Copying task: comparison of typing speed between users who noticed latency and users who did not.

Similarly, participants who did notice latency were clearly faster in the correction task (Fig. 8). In the low latency condition, participants who did not notice latency achieved a mean task completion time of 44.8 s (SD: 8.8 s), while participants who did notice latency were faster with a mean task completion time of 35.7 s (SD: 6.4 s). In the high latency condition, mean task completion time was 53.3 s (SD: 8.2 s) for participants who did not notice latency and 42.7 s (SD: 7.4 s) for participants who did notice latency.

5 DISCUSSION

We investigated the influence of text input latency on text input in a within-groups user study. We used two tasks: copying text from a template, and correcting a given text by deleting characters. In this section we summarize our findings and describe how latency influenced users' performance and perceived task load during those two

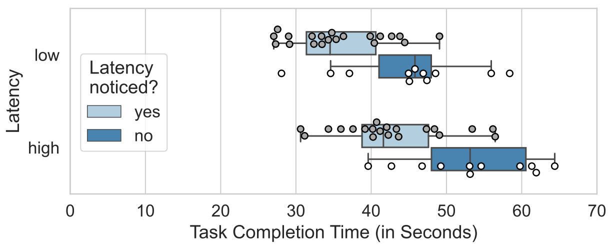

Figure 8: Correction task: comparison of task completion time between users who noticed latency and users who did not.

tasks. Afterwards, we discuss our study's limitations and conclude with implications of our findings for practical applications, as well as future research.

We found that during the copy task, time per character was slightly higher when typing with high latency, but this effect was not statistically significant. The error rate, operationalized by the fraction of incorrect key presses, was not influenced by latency. We could find significant effects with medium effect sizes for the NASA TLX subscales effort and frustration. In contrast, during the correction task, participants were significantly slower when latency was added to the system. Furthermore, we could find medium to large effects of latency on perceived task load, with significant differences on every NASA TLX subscale.

5.1 Interpretation of Results

Even though the results from both tasks seem to contradict each other, our findings can be explained by users' behavior during the study, as well as the intrinsic nature of the tasks. Firstly, many participants focused on the template text or the keyboard during the copy task. This behavior goes in line with Rieger and Bart [29], how found that experienced typists using all ten fingers tend to focus on template texts when copying text. As latency was only apparent when looking at the editor window, users did not perceive latency while looking at the template text in our study.

Furthermore, latency influences users by disrupting the feedback loop of a user's action, the system's processing of the input and response to the user, as well as the user's perception of the system's response [6]. When copying a text, users are only interested in the system's output when they either want to check if they made a mistake, or they finished typing and intend to finalize the task. Thus, even though copying text is comprised of individual keystrokes, users perceive the task as subdivided into more abstract concepts, such as words, sentences, or paragraphs.

On the other hand, the correction task required users to navigate to distinct positions within the text and then perform a specific action by deleting a character. This task is subdivided into more distinct, fine grained steps: navigating vertically, navigating horizontally, and deleting the character. During navigation, users have to constantly wait for the system's response, as the current cursor position determines which action to take next. If the current cursor position is not taken into consideration, users could under or overshoot the target, leading to additional correction steps. As every query of the system's state is affected by latency, its effect on the correction task was larger than with the copy task.

This difference between the copy task and the correction task is related to the concept of *deadline* introduced by Claypool et al. [7, 8]. In video games, an action's *deadline* is the latest point in time at which an action could be performed. For example in a first-person shooter, a player can only perform game actions until they get fragged by an opponent. For the correction task, this concept can be applied in reverse: there is an earliest point in time for when a user can decide about their next action. Only when a user has perceived the current position of the cursor, they can decide in which direction to move or whether they want to perform a delete operation. This theory is also supported by the findings of Tolia et al. [35]. In their analysis of replayed event logs under different simulated network latencies, they found that highly interactive systems are affected strongly by latency as the next action can only be performed when the previous action has been completed. In particular, they found that tracking changes in a word processor (which is similar to our correction task) is influenced more severely by high latency than pure typing.

Furthermore, our exploratory analysis suggests that users who did not notice latency during the study, were on average slower in both tasks than users who did notice latency. Even though our sample was too small for a statistical analysis, this potential effect can be again explained with the feedback loop of interaction. Latency only affects users if it causes temporal bottlenecks and therefore disrupts the feedback loop. If a user would be ready to plan their next action, but they have to wait for the system's response, interaction gets delayed. However, if the time between user inputs is higher than the system's latency, latency does not disrupt the feedback loop as users have all required information when they perform their next action. Accordingly, the perception threshold for latency is inversely proportional to interaction frequency. Applied to the concrete example of a typing task, this means that users who type faster are affected by lower amounts of latency than users who type slowly. However, even though our data clearly indicates that users who noticed latency were faster during our tasks than users who did not, we can not draw conclusions on the reason for this effect. On the one hand, typing faster might make latency more obvious, causing faster typers to recognice lower amounts of latency. On the other hand, noticing latency might cause users to adapt to the system, compensate latency by adjusting their inputs, and thus be faster in absolving typing tasks. Follow-up studies are required to analyze and explain this effect.

5.2 Limitations and Future Work

Even though we could show that text input latency can affect users in different typing tasks, our study still has some limitations that have to be addressed in future research. First and foremost, to maximize internal validity, we selected tasks that could be easily operationalized in terms of success and required time. However, those tasks are only related to free writing to some degree. While atomic interactions, such as keystrokes and navigation, are the same, free writing involves a complex combination of mentally drafting a text, typing the text, reading already written text, as well as correcting mistakes. Therefore, we do not know to which degree our findings regarding the effect of text input latency can be applied to free writing. A middle way between our strictly controlled copying and correction tasks and the complex task of free writing could be to ask users to write a dictated text or write a protocol of a spoken conversation. This approach would address the problem of users focusing on a template text, which occurred during our copy task.

Additionally, we only compared two latency conditions with a large amount of added latency in the high latency condition. As our study is the first to investigate the effect of latency on text input, we deliberately focused on finding out whether there is an effect or not. However, future studies should include more latency conditions to analyze the relationship between the amount of added latency and users' performance, as well as the just noticeable difference for text input latency in different scenarios.

Furthermore, we observed that participants became better in both tasks over the course of the study. We compensated this learning effect to some degree by discarding the first repetition of each task from our analysis. However, to fully control for this potential confounding effect, future studies should consider either a between-groups design with very short tasks, or start their study with a long warm-up phase.

During the post-study interview, participants reported that latency was most noticeable when switching between conditions. This goes in line with Halbhuber et al.'s findings [16] that switching between blocks of low and high latency can affect users more than operating at a constant high latency. On the other hand, Schmid et al. [30] found that small latency variations do not affect users when playing a first person shooter. Therefore, the effect of changes in latency seem to depend on the amount and frequency of changes, as well as the particular task. Future studies should further investigate how varying latency affects users when working with texts, or at least control for potential effects with an appropriate study design, for example by using a between-subjects design or by including long pauses between latency conditions.

Finally, typing skill is very heterogeneous across the population [29]. Even though there are experienced typers who use the ten finger system, and unexperienced typers who have to have to search for the next key, the majority of typers falls into a middle ground. This group uses idiosyncratic typing styles and is therefore hard to compare among itself. As findings for individual groups with different typing styles might be interesting, future studies on text input latency should recruit participants with respect to their typing skill. Another option would be to take typing skill into consideration during the evaluation, for example by measuring participants' strokes per minute before the study.

6 CONCLUSION

In our study, we investigated how input latency affects performance and task load in text input tasks. Using two latency conditions – 20 and 200 milliseconds – we let users copy and correct short text snippets with a physical keyboard. While we could not find significant effects of latency on users' performance during the copy task, they perceived significantly higher effort and frustration when typing with high latency. During the correction task, participants were significantly slower and perceived higher task load when latency was applied to the system. We can explain this discrepancy with users' behavior: during the copy task, they focused on the template text or their keyboard and did therefore not perceive the system's latency.

Our work contributes to the existing body of research concerning the effects of latency on users. Even though there have been numerous studies investigating the effect of latency on users in atomic tasks [12, 20, 26] or video games [8, 15, 16, 18, 24], to our knowledge, our study is the first to investigate the effects of text input latency on users. As we could show that latency can affect performance and user experience during typing in a controlled lab study, future research should investigate to which degree this effect influences typing in more realistic settings. Furthermore, an investigation regarding the perception threshold of text input latency could help extending design guidelines for text input systems.

REFERENCES

[1] Rui Alexandre Alves, São Luís Castro, Liliana de Sousa, and Sven Strömqvist. 2007. Influence of typing skill on pause–execution cycles in written composition. In *Writing and Cognition*. Brill, 55–65.

[2] Christiane Attig, Nadine Rauh, Thomas Franke, and Josef F Krems. 2017. System latency guidelines then and now–is zero latency really considered necessary?. In *Engineering Psychology and Cognitive Ergonomics: Cognition and Design: 14th International Conference, EPCE 2017, Held as Part of HCI International 2017, Vancouver, BC, Canada, July 9-14, 2017, Proceedings, Part II 14*. Springer, 3–14. https://doi.org/10.1007/978-3-319-58475-1_1

[3] Tom Beigbeder, Rory Coughlan, Corey Lusher, John Plunkett, Emmanuel Agu, and Mark Claypool. 2004. The Effects of Loss and Latency on User Performance in Unreal Tournament 2003®. In *Proceedings of 3rd ACM SIGCOMM Workshop on Network and System Support for Games* (Portland, Oregon, USA) *(NetGames '04)*. Association for Computing Machinery, New York, NY, USA, 144–151. https://doi.org/10.1145/1016540.1016556

[4] François Bérard and Renaud Blanch. 2013. Two Touch System Latency Estimators: High Accuracy and Low Overhead. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces* (St. Andrews, Scotland, United Kingdom) *(ITS '13)*. Association for Computing Machinery, New York, NY, USA, 241–250. https://doi.org/10.1145/2512349.2512796

[5] C.H. Björnsson. 1968. *Läsbarhet:*. Liber; [Solna, Seelig]. https://books.google.de/books?id=MlhDnQEACAAJ

[6] Stuart K. Card, Thomas P. Moran, and Allen Newell. 1986. The model human processor- An engineering model of human performance. *Handbook of perception and human performance.* 2, 45–1 (1986), 6.

[7] Mark Claypool and Kajal Claypool. 2006. Latency and player actions in online games. *Commun. ACM* 49, 11 (Nov. 2006), 40–45. https://doi.org/10.1145/1167838.1167860

[8] Mark Claypool and Kajal Claypool. 2010. Latency Can Kill: Precision and Deadline in Online Games. In *Proceedings of the First Annual ACM SIGMM Conference on Multimedia Systems* (Phoenix, Arizona, USA) *(MMSys '10)*. Association for Computing Machinery, New York, NY, USA, 215–222. https://doi.org/10.1145/1730836.1730863

[9] Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences*. Routledge. 567 pages. https://doi.org/10.4324/9780203771587

[10] Stephan F. Dahm and Martina Rieger. 2022. A little doubt saves many mistakes: Early and late error detection in copy-typing. *Open Psychology* 4, 1 (2022), 115–131. https://doi.org/doi:10.1515/psych-2022-0006

[11] Fred J. Damerau. 1964. A Technique for Computer Detection and Correction of Spelling Errors. *Commun. ACM* 7, 3 (mar 1964), 171–176. https://doi.org/10.1145/363958.363994

[12] Jonathan Deber, Ricardo Jota, Clifton Forlines, and Daniel Wigdor. 2015. How Much Faster is Fast Enough? User Perception of Latency & Latency Improvements in Direct and Indirect Touch *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1827–1836. https://doi.org/10.1145/2702123.2702300

[13] Pavel Fatin. 2015. Typing with pleasure. https://pavelfatin.com/typing-with-pleasure/

[14] Sebastian Friston and Anthony Steed. 2014. Measuring Latency in Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics* 20, 4 (April 2014), 616–625. https://doi.org/10.1109/TVCG.2014.30 00063.

[15] David Halbhuber. 2022. To Lag or Not to Lag: Understanding and Compensating Latency in Video Games. In *Extended Abstracts of the 2022 Annual Symposium on*

Computer-Human Interaction in Play (CHI PLAY '22). Association for Computing Machinery, New York, NY, USA, 370–373. https://doi.org/10.1145/3505270.3558364

[16] David Halbhuber, Valentin Schwind, and Niels Henze. 2022. Don't Break my Flow: Effects of Switching Latency in Shooting Video Games. *Proceedings of the ACM on Human-Computer Interaction* 6, CHI PLAY (Oct. 2022), 229:1–229:20. https://doi.org/10.1145/3549492

[17] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Advances in Psychology*. Vol. 52. Elsevier, 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9 12600.

[18] Zenja Ivkovic, Ian Stavness, Carl Gutwin, and Steven Sutcliffe. 2015. Quantifying and Mitigating the Negative Effects of Local Latencies on Aiming in 3D Shooter Games. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 135–144. https://doi.org/10.1145/2702123.2702432

[19] Roger Johansson, Åsa Wengelin, Victoria Johansson, and Kenneth Holmqvist. 2010. Looking at the keyboard or the monitor: relationship with text production processes. *Reading and writing* 23 (2010), 835–851. https://doi.org/10.1007/s11145-009-9189-3

[20] Ricardo Jota, Albert Ng, Paul Dietz, and Daniel Wigdor. 2013. How Fast is Fast Enough? A Study of the Effects of Latency in Direct-Touch Pointing Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 2291–2300. https://doi.org/10.1145/2470654.2481317

[21] Ronald T. Kellogg. 1996. A model of working memory in writing. In *The Science of Writing*. Lawrence Erlbaum Associates, Inc.

[22] Joohwan Kim, Josef Spjut, Morgan McGuire, Alexander Majercik, Ben Boudaoud, Rachel Albert, and David Luebke. 2019. Esports Arms Race: Latency and Refresh Rate for Competitive Gaming Tasks. *Journal of Vision* 19, 10 (2019), 218c–218c. https://doi.org/10.1167/19.10.218c

[23] Shengmei Liu and Mark Claypool. 2021. EvLag – A Tool for Monitoring and Lagging Linux Input Devices. (2021), 6. 00000.

[24] Shengmei Liu, Mark Claypool, Atsuo Kuwahara, James Scovell, and Jamie Sherman. 2021. The Effects of Network Latency on Competitive First-Person Shooter Game Players. In *2021 13th International Conference on Quality of Multimedia Experience (QoMEX)*. 151–156. https://doi.org/10.1109/QoMEX51781.2021.9465419

[25] Gordon D. Logan and Matthew J.C. Crump. 2009. The left hand doesn't know what the right hand is doing: The disruptive effects of attention to the hands in skilled typewriting. *Psychological Science* 20, 10 (2009), 1296–1300. https://doi.org/10.1111/j.1467-9280.2009.024

[26] I. Scott MacKenzie and Colin Ware. 1993. Lag as a Determinant of Human Performance in Interactive Systems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems* (Amsterdam, The Netherlands) *(CHI '93)*. Association for Computing Machinery, New York, NY, USA, 488–493. https://doi.org/10.1145/169059.169431

[27] Anne Mangen, Liss G. Anda, Gunn H. Oxborough, and Kolbjørn Brřnnick. 2015. Handwriting versus keyboard writing: effect on word recall. *Journal of writing research* 7, 2 (2015), 227–247. https://doi.org/10.17239/jowr-2015.07.02.1

[28] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. 2012. Designing for Low-Latency Direct-Touch Input. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 453–464. https://doi.org/10.1145/2380116.2380174

[29] Martina Rieger and Victoria K.E. Bart. 2016. Typing style and the use of different sources of information during typing: an investigation using self-reports. *Frontiers in Psychology* 7 (2016), 1908. https://doi.org/10.3389/fpsyg.2016.01908

[30] Andreas Schmid, David Halbhuber, Thomas Fischer, Raphael Wimmer, and Niels Henze. 2023. Small Latency Variations Do Not Affect Player Performance in First-Person Shooters. *Proceedings of the ACM on Human-Computer Interaction* 7, CHI PLAY (Oct. 2023), 381:197–381:216. https://doi.org/10.1145/3611027

[31] Andreas Schmid and Raphael Wimmer. 2021. Yet Another Latency Measuring Device. (2021). https://doi.org/10.5283/epub.45570

[32] Andreas Schmid and Raphael Wimmer. 2023. Measuring the Latency of Graphics Frameworks on X11-Based Systems. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3544549.3585779

[33] Seth Schneider. 2020. Introducing NVIDIA Reflex: Optimize and Measure Latency in Competitive Games. https://www.nvidia.com/en-us/geforce/news/reflex-low-latency-platform/

[34] Ben Shneiderman, Catherine Plaisant, Maxine S. Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. 2016. *Designing the user interface: strategies for effective human-computer interaction*. Pearson.

[35] N. Tolia, D.G. Andersen, and M. Satyanarayanan. 2006. Quantifying interactive user experience on thin clients. *Computer* 39, 3 (March 2006), 46–52. https://doi.org/10.1109/MC.2006.101 Conference Name: Computer.

[36] Raphael Wimmer, Andreas Schmid, and Florian Bockes. 2019. On the Latency of USB-Connected Input Devices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300650