Check for updates

Tap to Sign: Towards using American Sign Language for Text Entry on Smartphones

SAAD HASSAN*, Tulane University, USA
ABRAHAM GLASSER*, Gallaudet University, USA
MAX SHENGELIA*, Rochester Institute of Technology, USA
THAD STARNER*, Google, USA
SEAN FORBES† and NATHAN QUALLS†, Deaf Professional Arts Network, USA
SAM S. SEPAH, Google, USA

227

Soon, smartphones may be capable of allowing American Sign Language (ASL) signing and/or fingerspelling for text entry. To explore the usefulness of this approach, we compared emulated fingerspelling recognition with a virtual keyboard for 12 Deaf participants. With practice, fingerspelling is faster (42.5 wpm), potentially has fewer errors (4.02% corrected error rate) and higher throughput (14.2 bits/second), and is as desired as virtual keyboard texting (31.9 wpm; 6.46% corrected error rate; 10.9 bits/second throughput). Our second study recruits another 12 Deaf users at the 2022 National Association for the Deaf conference to compare the walk-up usability of fingerspelling alone, signing, and virtual keyboard text entry for interacting with an emulated mobile assistant. Both signing and virtual keyboard text entry were preferred over fingerspelling.

CCS Concepts: • **Human-centered computing** → *Accessibility design and evaluation methods*; **Empirical studies in accessibility**.

Additional Key Words and Phrases: Sign Languages, ASL, Fingerspelling, text entry, Signing Interface, Mobile Assistant

ACM Reference Format:
Saad Hassan, Abraham Glasser, Max Shengelia, Thad Starner, Sean Forbes, Nathan Qualls, and Sam S. Sepah. 2023. Tap to Sign: Towards using American Sign Language for Text Entry on Smartphones. *Proc. ACM Hum.-Comput. Interact.* 7, MHCI, Article 227 (September 2023), 23 pages. https://doi.org/10.1145/3604274

1 INTRODUCTION

There are over 70 million Deaf and hard of hearing (DHH) people in the world [1, 2]. Increasing numbers of people who are DHH—as well as some hearing individuals—use one or more of around 150 sign languages to communicate [17]. In the U.S. alone, American Sign Language (ASL) is used by about 500,000 people as a primary form of communication [38]. Conversational ASL is a complex language with spatial constructions that are difficult to address with current machine learning

*These authors contributed equally to this research.
†Both authors contributed equally to this research.

Authors' addresses: Saad Hassan, Tulane University, New Orleans, Louisiana, USA, 70118, shassan4@tulane.edu; Abraham Glasser, Gallaudet University, Washington, District of Columbia, USA, 20002, abraham.glasser@gallaudet.edu; Max Shengelia, Rochester Institute of Technology, Rochester, New York, USA, 14623, mas5627@rit.edu; Thad Starner, Google, Atlanta, Georgia, USA, 30309, thadstarner@google.com; Sean Forbes, NathanQualls; Nathan Qualls, nathan@dpan.tv, Deaf Professional Arts Network, Detroit, Michigan, USA; Sam S. Sepah, Google, Mountain View, Georgia, USA, 94043, sepah@google.com.

2573-0142/2023/9-ART227
https://doi.org/10.1145/3604274

techniques and datasets. While sign-language recognition has been a topic of interest by the machine learning community for decades [11, 56], most systems have little usefulness or usability for the Deaf community [11, 25]. Fingerspelling should be easier to recognize in theory, but most attempts have been technical demonstrations instead of working toward a goal that might be helpful to the Deaf community. In addition, much more data must be collected to make an accurate fingerspelling recognition system. *Our primary objective is to evaluate whether a significant data collection effort for fingerspelling is justified by identifying a practical application for a fingerspelling recognition system, namely text entry on mobile phones, and testing an emulated version of that application with the Deaf community. Furthermore, we examine the desirability of creating a recognition system for an additional, closely-related subset of ASL: signs that could be used to command a mobile assistant.*

Like all active languages, ASL is evolving [28, 39]. With the advent of smartphones and being able to communicate "on-the-go," some Deaf users have begun altering their signing style while using smartphone cameras [39], a phenomenon we refer to as "smartphone-signing." When ASL signers are engaged in a video call while mobile, it is typical for them to hold their smartphone in one hand and sign using their free hand. In a recent study, 53% Deaf users reported holding their smartphone in one hand while signing with the other [34]. Some others use gadgets or cases that allow them to sign with both hands while recording. Smartphone-signing naturally addresses some of the difficulties of ASL recognition. For example, in order to be understood, the signer is already accustomed to adjusting the smartphone's camera and their signing style such that the user's hand and face can be seen and adequately lit [12]. Signing gestures when using a smartphone tend to be smaller, as the field of view tends to be too narrow to capture the full body. Smartphone-signing often adapts two-handed signs to one-handed versions, and fingerspelling might be used if the one-handed version of the sign would be too ambiguous. Fingerspelling, which is the equivalent to letter-by-letter English text entry, is often used to specify proper nouns or to communicate precise numbers in signed conversations. In conversational ASL discourse, fingerspelling can contribute between 12% - 35% of the signed conversation [26, 40], and we expect to see even more in smartphone-signing.

One of the most requested use cases for sign recognition by the Deaf community is for communicating with speech-based assistants [21], such as Apple's Siri or Google's Assistant [3], where fingerspelling will be a major component. While there has been research on using sign language to command an assistant in a home scenario [21], few studies have focused on smartphone-signing. Mobile assistant commands often involve proper names (for example, "VIDEOCALL M-E-L-O-D-Y" or "DIRECTIONS 1-2-3 P-E-A-C-H-T-R-E-E A-V-E"), and a sign recognition system for a mobile assistant will need to perform well on such fingerspelling text entry. DHH users prefer not to use speech recognition due to having low or no voice, being unable to tell if they are speaking too loudly (or too softly) in public to their phone, or having a "deaf accent" which makes speech recognition difficult. Thus, looking ahead, there is potential for fingerspelling to become a more common method of text entry for smartphones and/or interacting with a mobile assistant, especially if it proves faster, less error-prone, or less effortful than virtual keyboards. We summarize our contributions below:

- We establish a learning curve for fingerspelling text entry by measuring the performance of 12 Deaf smartphone users as they fingerspell 1000 phrases.
- After the 12 Deaf users are trained to expertise, we compare their text entry speed and subjective preference with a Wizard-of-Oz fingerspelling interface versus a standard Android virtual smartphone keyboard. Fingerspelling is preferred by 50% of the participants; fingerspelling speeds are significantly faster (42.5 wpm vs. 31.9 wpm) and error rates are lower (4.89% vs. 7.23%) in the ideal case.

- We recruit a second group of 12 Deaf participants to explore the difference between their walk-up performance and subjective preferences when communicating with an emulated mobile assistant using a standard Android virtual smartphone keyboard versus fingerspelling versus signing. Signing and the virtual keyboard are preferred over the fingerspelling interface. An analysis of open-ended feedback revealed potential use cases and design improvements for a mobile assistant capable of taking sign language input.

2 BACKGROUND AND RELATED WORK

ASL fingerspelling can be very rapid, with the literature reporting between 60-96 wpm [45]. Given that the average words per minute (wpm) in a study of 37,000 smartphone typists was 36, this reported fingerspelling speed is encouraging [41]. However, at such speeds, individual fingerspelled letters are hard to distinguish or may be completely skipped. In addition, understanding fast fingerspelling can be difficult in conversations, providing co-articulation challenges similar to those found in distinguishing individual phonemes in speech recognition. In fact, fingerspelling recognition accuracy of the "in-the-wild" ChicagoFSWild+ dataset is 62.3%. Even human recognition accuracy of the fingerspelling in that dataset is around 80% [52]. Improved recognition of fast fingerspelling could open up new interfaces for the Deaf community that are both useful and user-friendly. Faster and more convenient fingerspelling recognition could motivate the development of fingerspelling (and more generally, signing) interfaces for smartphones, especially for Deaf individuals with limited hand dexterity or low vision, where interactions with small keyboards can be difficult[1].

2.1 Sign recognition and potential use cases

While sign language recognition is complex [30], deep learning advancements in hand and face landmark detection and pose estimation suggest sign recognition may be possible in the future. However, no robust sign language translation system exists yet [11]. Continuous sign recognition is challenging due to limited available research datasets [29, 52] and the need for skilled annotators to create new ones from on-line data. Fingerspelling recognition faces similar issues but has seen success with models like Kim et al.'s semi-Markov conditional random field [29]. The recent release of fingerspelling datasets such as the ChicagoFSWild+ dataset [52, 53] has motivated the use of transformer models such as Gajurel et al.'s fine-grained visual attention model [18], Papadimitriou and Potamianos' convolutional multi-step attention-based method [43], and snapshot learning-based approaches [13, 31, 50]. Wearable glove recognition systems have also been developed but are generally impractical and not favored by the Deaf community [16].

While current fingerspelling recognition accuracy is improving, it is still not practical. Given that we are interested in smartphone text entry, can we use the interface to make the recognition task easier? To investigate this idea, we include an experimental condition in our study in which users push and hold a button on the phone's touchscreen to fingerspell individual words. This walkie-talkie style interface constrains the task to isolated word recognition, which reduces the complexity of the task significantly. A similar mechanism was used in early speech recognition interfaces.

Practical applications of fingerspelling recognition have been limited. Some prior work proposed a fingerspelling practice tool that used a realistic 3D animated character [8] to facilitate self-study for ASL learners, with a focus on receptive skills [58, 65]. There are also some games and AR Lenses

[1] Although one might assume that fingerspelling requires more hand dexterity than typing, our pilot testing with Deaf participants showed that some found typing on a virtual keyboard to be challenging, while fingerspelling was a more comfortable and natural alternative. Fingerspelling interfaces could also be useful for text entry on smaller devices, such as smartwatches or smart glasses, which have limited touchscreen surface area.

that are built using fingerspelling recognition models. Instead of being an interface for the Deaf, these systems focus on beginners and expect slow letter-by-letter fingerspelling [2]. Text entry on mobile phones, however, is a vastly different use case than educational games. It will require the recognition system to adapt to higher speeds and greater signer and background diversity.

2.2 Linguistic studies on fingerspelling

There has been substantial research on fingerspelling within the sign language linguistics community [10, 19, 24, 26, 27, 45, 47, 63]. Notably, Reich and Bick found that medial letters in manually coded English words were fingerspelled quicker than initial and final letters [47]. Within the medial letters, letters in later positions and those with movement and orientation changes tend to be held for shorter periods of time. Some research has also looked into holds (the time duration in which the hand configuration is stable) and transitions (the time between holds), finding that holds last 40 milliseconds and transitions 100 milliseconds for four participants. Understanding these temporal properties and variations is valuable for models of fingerspelling production and recognition.

Geer compared fingerspelling between native and non-native signers [19]. Visible English (VE), also known as the Rochester Method, has been investigated for its understandability, with research finding that it is not effective for in-class communication [47]. Hanson's research showed that Deaf participants do not read fingerspelled words as individual letters [24], and that misspellings in fingerspelling comprehension tasks are similar to those in written English, suggesting that auto-spell and auto-correct features would benefit the Deaf community.

Co-articulation, where sequential handshapes become more similar to one another, is a common phenomenon in fingerspelling, but has been underexplored for ASL. A study on fingerspelling speed found significant differences across signers and for word length, with signers fingerspelling at 5-8 letters per second and faster in formal settings [45]. These results further support the exploration of fingerspelling as a potential text entry method for mobile devices.

A key limitation of many of these studies is the number of fingerspelling examples used, which are often selected from a small number of manually coded English words. The number of participants, specifically Deaf participants, involved in these studies tends to be less than 10, which is insufficient to confirm that the results can be extended to a variety of signers. Here, we focus our effort on a specific practical task, fingerspelling as a text entry method on a smartphone, and recruit 24 Deaf participants for our user studies.

2.3 Text entry on different devices

Improved mobile text entry methods have been a focus of HCI innovation since the dominance of SMS with mobile feature phones [48, 49, 57]. Mini-QWERTY keyboards, whether physical or virtual, achieve faster speeds (up to 60 wpm) than multitap or T9 methods, enabling longer-form communication [51]. Deaf communities were early adopters of physical mini-QWERTY keyboards for remote communication, often using unique grammar and meanings [4–6, 44]. As capacitive touchscreens became more accurate, virtual mini-QWERTY keyboards became the standard, achieving speeds and accuracies previously only seen on physical keyboards. While few users currently write full documents using a mobile phone keyboard instead of a desktop, users no longer balk at typing emails and social media posts using these devices. Text entry using fingerspelling and smartphone signing may enable new interfaces or alternatives for the Deaf community.

Advancements in hand landmark detection models have enabled researchers to investigate novel interaction approaches based on hand gestures. These methods are more expressive, possibly faster,

[2]https://fingerspelling.xyz/

setting independent, and versatile in their usage. In a closely related prior work, researchers have investigated typing in mid-air for text entry and achieved 22 words per minute [54]. They also looked at other aspects like finger dexterity, typing accuracy, and individual differences between fingers [54]. Other studies have also looked at finding the optimum size of screen for QWERTY keyboard by assessing typing performance, finger postures, and user preference [66]. Turner et al. compared different input modalities as well for typing on smartwatches (tap, trace, and handwriting) while standing and walking [60]. Prior studies have also explored error rates and learnability with novel text entry methods on mobile phones of novice learners over multiple sessions [33, 55, 59, 64, 68]. These studies provided methodological guidance, including what phrase set to use for evaluation of our prototype systems, and the experimental protocol.

3 STUDY OVERVIEW AND RESEARCH QUESTIONS

We compare the performance of fingerspelling on two "Wizard-of-Oz" interfaces with typing on the standard Android virtual keyboard. The first interface, Tap to Sign (Figure 1), involves the user tapping the virtual button at the beginning and end of their signing. The second interface, Push to Sign (Figure 2), requires the user to press and hold a virtual button while fingerspelling each word in a sentence, releasing between words. Both interfaces always display correct outputs after the participants submit words or phrases since our goal is to compare the best possible fingerspelling text entry system to virtual keyboard typing. Although the Android virtual keyboard (Figure 3) allows gesture typing, all of our users use variants of two-thumb typing. Section 4.1 describes the three prototypes in greater detail.

Deaf participants were recruited by a Deaf professional organization. Smartphone texting is prevalent in the Deaf community as a way to communicate, and these participants were already expert in the use of texting interfaces. For an accurate comparison between texting and fingerspelling, we need to train our participants to expert levels with our fingerspelling interface by collecting several hours of fingerspelled phrases, leading to our first research question:

RQ1 How much training is required to reach expertise in a fingerspelling text entry interface?

After our participants were trained to expertise (which could be done with 200-300 phrases in approximately 2-3 hours, though we ran all participants to 1000 phrases), we could then better compare our prototypes to current texting interfaces, leading to the second research question:

RQ2 Are the Tap to Sign and Push to Sign fingerspelling prototypes faster and more preferred for text entry versus a virtual keyboard for Deaf users trained to expertise with the fingerspelling interface?

Participant's speeds were significantly higher when using fingerspelling as compared to the virtual keyboard, and subjective experience and satisfaction were roughly the same. Given these results and the desire by the Deaf community for signed access to virtual assistants [20, 22, 37], we hoped that a fingerspelling interface to such assistants might be a sufficient first interface while a full signing interface is being developed. This hope motivated our third question:

RQ3 In a walk-up usability evaluation, do Deaf users prefer fingerspelling and/or smartphone-signing for text entry over typing on a virtual smartphone keyboard to command an emulated virtual assistant?

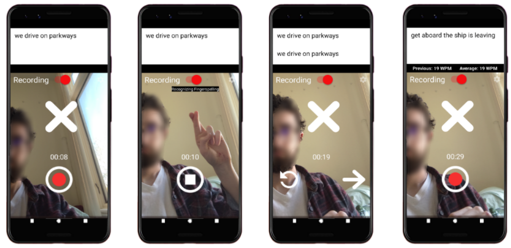


Fig. 1. Tap to sign emulates fingerspelling recognition of entire sentences at a time. The X means that the emulated recognizer is not running whereas “recognizing fingerspelling” displayed on the screen means it is. The user must fingerspell the entire prompted sentence (top of screen) before given feedback on what they signed below the prompt. Since recognition is emulated, the correct phrase is shown no matter what the user signs. No partial recognition is shown. Pressing the undo icon (third image) erases the entire sentence. Participants can see their average speed and texting rate for the last phrase (rightmost figure).

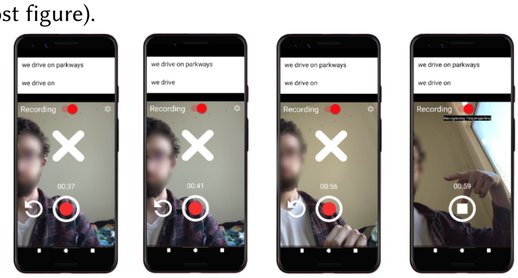


Fig. 2. Push to Sign emulates fingerspelling recognition one word at a time, requiring the user to hold the "recognition" button while fingerspelling each word and releasing between words. Recognition is emulated by always showing the correct next word in the sentence. Participants could still undo, however, if they felt they signed incorrectly. In this interaction sequence, a user fingerspells the first three words of the prompt, then presses undo to remove the word “on,” and then repeats the word, and finishes the phrase.

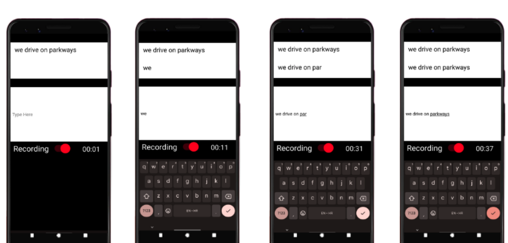


Fig. 3. Virtual Keyboard emulates normal text entry on smartphones. Participants could see individual characters being typed, and auto-correct was available. Predictive text was turned off.

To address this last question, study 2 creates a simple emulation of a mobile assistant commanded through text entry and recruits 12 new Deaf signers at NAD2022 who are not familiar with the prior work to evaluate the interface.

4 STUDY 1: COMPARING SPEED AND PREFERENCE BETWEEN FINGERSPELLING AND VIRTUAL KEYBOARD FOR TEXT ENTRY

Our aim was to compare the usefulness and desirability of using fingerspelling as a text entry method on phone as compared to standard virtual keyboard typing. To investigate this question, we first conducted an experimental study with 12 DHH users who interacted with three wizard-of-oz prototypes for text entry on mobile phone: Push to Sign, Tap to Sign, and the Android virtual keyboard. We designed three Android mobile applications to capture the speeds of users while interacting with different text entry methods and record their performance. All three prototypes prompted the users to enter phrases either by fingerspelling (in case of Tap to Sign and Push to Sign) or two-thumb typing (virtual keyboard). In all three conditions, participants are prompted to enter randomly chosen phrases from the standard MacKenzie text entry phrase set [36][3] As our aim was to ascertain whether text entry on a mobile phone justifies investing significant resources in data collection for the underlying sign recognition system, we employed a Wizard-of-Oz approach to simulate sign recognition performance that always produced accurate phrases.

4.1 Prototypes

4.1.1 Tap to Sign (Figure 1): This interface prompted the participants to fingerspell entire phrases shown on the screen. After entering their username, users were asked to check the correct camera view, resolution, and aspect ratio, which were adjustable in settings. Clicking the record button displayed the first phrase and showed a cross indicating non-recognition. Tapping the circular button initiated text entry, replacing the cross with *"recognizing fingerspelling."* The phrase appeared in a text field upon completion, and users could repeat or move to the next phrase. Clicking the recording slider button saved the video and typing speed data locally and remotely using Firebase.

4.1.2 Push to Sign (Figure 2): This interface required the participants to fingerspell each word in a phrase separately. The design of the Push to Sign interface was similar to Tap to Sign except for the following:

(1) Participants recorded one word at a time. In order to record a word, a user would press and hold the record button. Once the button is released the word appears in the text field underneath the text prompt as a confirmation of an entry.
(2) Participants could click on undo to repeat the word if they felt that they misspelled it. Unlike the Tap to Sign interface, only the last word was deleted from the confirmation text.

4.1.3 Virtual Keyboard (Figure 3): The virtual keyboard interface used the standard Android keyboard, which can be used to type a single letter at a time or with gestures as the user preferred. Notably, we added a recording button and a timer to make sure the interface has all the UI elements in the other two conditions. Predictive text options were disabled for this study but auto-correct was enabled so as to better match the signing conditions where we showed perfect output after each entry. That is, if the participant was supposed to fingerspell "back " and instead fingerspelled "bak", the system would display "back ". For the virtual keyboard, "bak " would auto-correct resulting in a key sequence of "bak «ck " and a resulting text string of "back ".

[3]The Mackenzie phrase set is a standard that has been used in the evaluation of text entry methods. In prior studies, participants also enter phrases of text using a technique of interest while performance and usability data are collected. The data set consists of a collection of 500 phrases for such evaluations. A phrase approximately has 30 characters and consists of roughly 5-6 words. Some of the words with British English spellings were converted to American English spellings since the study was being conducted across the U.S. The participants in study 1 completed the entire dataset twice before the study.

4.2 Participants and Recruitment

12 Deaf participants (9 females and 3 males) were recruited by a Deaf professional organization based in Detroit. Permission was sought from participants to include their images in this work. The participants were recruited for the main purpose of providing example signing for a large fingerspelling database. All considered ASL their dominant language[4]. The average age of participants was 33.6 years (σ = 13.5). Three participants said that they use video messaging apps to send signed video messages every day, one said more than half of the days, four said about 50% of the days, three said less than half of the days, and one reported never doing it. Nine of our participants reported holding their phone in one hand while recording signed videos; two reported putting it in a stationary location. To train to expertise and for data collection purposes, participants were asked to fingerspell 1000 phrases from our dataset [36] using the Tap to Sign interface. This procedure was suggested by the Deaf professional organization with whom we were working. Participants were asked to fingerspell the phrases correctly, which may have limited their speed compared to normal conversational fingerspelling, which tends to elide characters. Participants averaged around 100 phrases per hour.

4.3 Training to Proficiency

Figure 4 shows how participants' speed increased over time as they captured 1000 fingerspelled phrases with the Tap to Sign app. This learning curve was a bit surprising in that we had assumed that fingerspelling phrases would be straightforward for people who used ASL as their primary language. However, fingerspelling is normally used in bursts in conversation. Thus, some training proved beneficial both to learn the interface and to reach proficiency for fingerspelling at speed.

4.4 Experiment Design for Comparing Text Entry Methods

The study was conducted in-person and started with sharing an informed consent form that was acknowledged, signed, and returned. The experiment consisted of three conditions, each using a different text entry method described in the previous section, using a within-subject design. The order in which participants engaged with each condition was predetermined using a Latin-square protocol for counter-balancing. Each condition involved three sessions of five minutes each, with a one-minute break between each session. After completing each experimental condition, participants were asked to complete a form with the NASA-TLX scales and a subjective preference questionnaire, along with an open-ended field for feedback. Participants were also given the opportunity to sign their feedback, which was transcribed by the researcher. At the end of the experiment, participants could also provide additional feedback. Our metrics for measuring task performance and user satisfaction were adapted from prior HCI research by Ruan et al. on comparing different text entry methods for short messages in two different languages [48].

4.5 Results

4.5.1 Speed. Our primary hypotheses was that Push to Sign and Tap to Sign would be faster than virtual keyboard. We ran two paired, one-tailed t-tests with Benjamini-Hochberg correction comparing the methods.

[4]Eleven participants reported using ASL every day and 1 participant reported using it about half of the days.

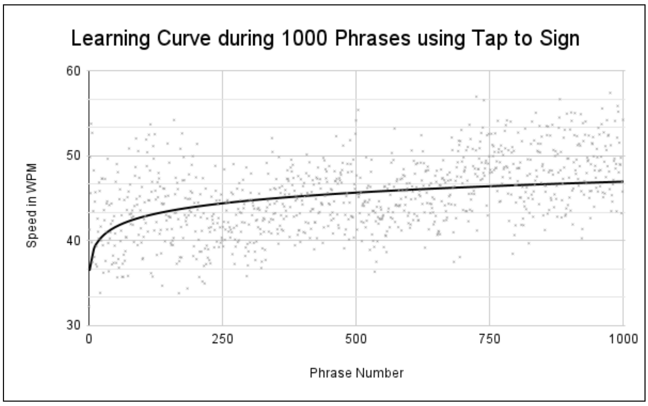


Fig. 4. Plot showing speed in words per minute over 1000 phrases that participants in study 1 completed prior to participating in the study. A smooth power-series best-fit curve is also plotted to visualize the trend.

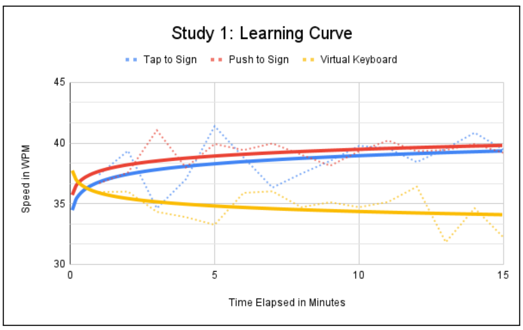


Fig. 5. Change in speed in wpm during 15 minutes of interacting with each interface across the three conditions. Individual smooth power-series best-fit curves are also plotted to visualize the trends for all three conditions.

Condition	WPM	CER	UCER	Bandwidth [48]	Throughput [68]
Tap to Sign	39.3 (10.1)	4.97% (6.19%)	0.761% (1.23%)	**94.0% (3.52%)***	13.2 (3.28)
Push to Sign	**42.5 (11.3)***	4.02% (4.27%)	0.866% (1.13%)	**95.4% (3.40%)***	**14.2 (3.78)***
Virtual Keyboard	31.9 (10.1)	6.46% (2.26%)	0.770% (0.372%)	88.3% (3.60%)	10.9 (3.45)

Table 1. Comparison of the three conditions in study 1 across five different metrics of speed, accuracy, and efficiency. * indicates a significant difference with virtual keyboard condition. CER is the character error rate. UCER is uncorrected character error rate.

Table 4 (Appendix A) shows the average speeds in words per minute for each of the 12 participants while interacting with the three prototypes. We remove P20 from the data as an outlier based on the difference in typing speeds between their fingerspelling and virtual keyboard methods being greater than three standard deviations (3.04) of the average of the rest of the participants. The average text entry speed on the virtual keyboard was 31.9 wpm. Tap to Sign was 39.3, and Push to Sign was 42.3[5]. Significance testing supported the hypothesis (t(10) = 2.35, p = 0.0145, medium

[5] P20's speeds are Tap to Sign 34.7; Push to Sign 29.8; Virtual Keyboard 62.6

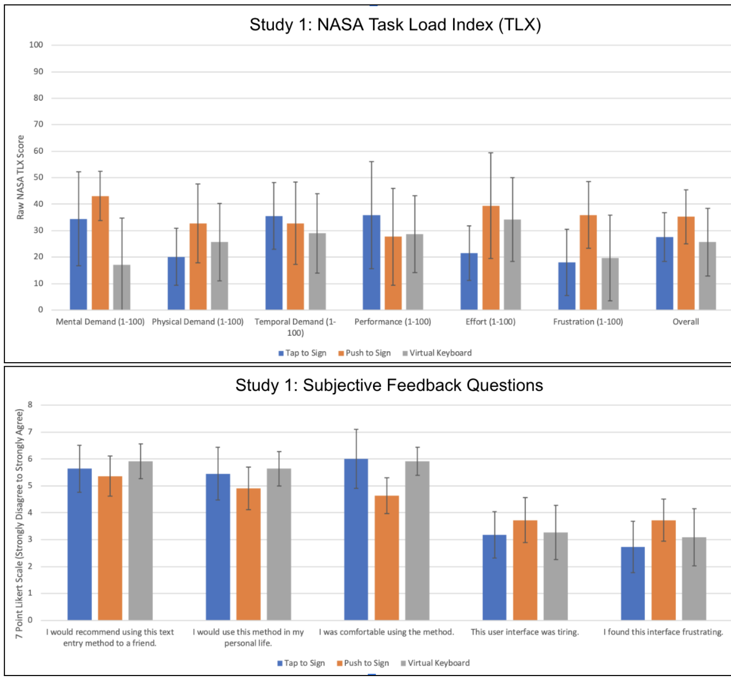


Fig. 6. Average raw NASA TLX and subjective feedback scores for study 1. The error bars indicate one standard deviation.

effect) that Push to Sign (μ=42.5, σ=11.3) was faster than virtual keyboard (μ=31.9, σ=10.1). The average text entry speed for Tap to Sign was also significantly (t(10) = 1.74, p = 0.0482, small effect) higher (μ = 39.3, σ = 10.1) as compared to the virtual keyboard. A post-hoc, two-tailed, paired t-test comparing the average speeds between Push to Sign to Tap to Sign did not show significance.

As mentioned earlier in the paper, our users typed as they would do in real-life scenarios. We mostly observed different variants of two-thumb typing and combinations with gesture. We compared the proportions of characters entered during virtual keyboard conditions in study 1 that can be characterized as auto-corrected, gesture typed, or manually entered using an Intelligent Text Entry (ITE) recognition system which determines which type of text entry is being used by the speed at which the characters appear in the transcript [42][6]. For example, gesture-based typing typically has all the letters in a word appear at once whereas tapping each letter tends to have the letters appear in the transcript with pauses between each. For study 1, 8.3% of the characters were autocorrected, 14.6% entered by gesture, and 77.1% were entered by normal-text entry.

4.5.2 Error Analysis. For all three conditions in Study 1, we were also interested in what happens *during* the text entry. To this end, for the last recorded session in the two signing conditions, a Deaf member of our team carefully transcribed the input stream. The input stream was automatically logged for the virtual keyboard condition in our backend. By comparing the input stream with the actual phrases users were intending to sign, we investigated the types of errors that users were making and the fixes they made using the undo button in the signing conditions or the backspace

[6]https://userinterfaces.aalto.fi/typing37k.

button in the virtual keyboard condition. We computed metrics that measure error rates, efficiency, and text entry method-agnostic speed-accuracy trade-off [48, 68].

While we understand that the errors made with a functional fingerspelling recognition system may significantly differ from this approach, **our goal is to determine whether attempting a fingerspelling recognition system may be useful** before spending significant resources on the data collection necessary to make such a recognition system. The approach here reflects the best situation we might expect from a fingerspelling interface.

Based on the approach used in Wobbrock's work that compared speech and keyboard text entry for short text messages in two languages on mobile phones [48], we classified each character entered in the input stream as *Correct* (C), *Incorrect-not-fixed* (INF), or *Incorrect-fixed* (IF). Correct was computed as the difference of the larger length among transcribed and presented phrases and the edit distance between them (*MAX(P,T) - ED(P,T)*). Incorrect-not-fixed is the number of incorrect characters in the final transcript or simply the edit distance between presented and transcribed phrase *ED(P,T)*. Incorrect-fixed is the number of characters deleted during the text entry or the sum of lengths of all the deletions during *undos* or backspace in the virtual keyboard. The lengths of individual deletions were typically the longest for Tap to Sign since the users deleted the entire phrase or a significant portion of it in each deletion as compared to Push to Sign in which users only deleted a single word in each undo attempt. In the virtual keyboard condition users used backspace to erase one character in a single deletion. *Fixes* (F) used in some of the calculations below was a sum of all delete actions (*undos* in case of Tap to Sign or Push to Sign) and backspace in case of miniQWERTY. Below we summarize how each of the metrics were calculated. Table 1 shows a summary of our findings.

- **Utilized Bandwidth** represents the proportion of keystrokes or gestures that contributed to the correct entry in the final transcribed string. This measure characterizes the "efficiency" of the input. The formula is:
  $utilized\ bandiwdth = \frac{C}{C+INF+F+IF}$. Significance testing revealed that utilized bandwidth was significantly higher for Push to Sign as compared to Virtual Keyboard ($t(10) = 4.30$, $p < 0.0003$, large effect). Significance testing also revealed that utilized bandwidth was significantly higher for Tap to Sign as compared to Virtual Keyboard ($t(10) = 3.73$, $p < 0.0015$, large effect). No significant difference between Tap to Sign and Push to Sign was observed.
- **Throughput** represents the amount of information transmitted in a text entry method per unit of time. It is computed using text entry speed and uncorrected error rate. We use the code shared by Zhang, Zhai, and Wobbrock [68] to compute throughput for our three conditions. Like utilized bandwidth, significance testing revealed that throughput was also significantly higher for Push to Sign as compared to Virtual Keyboard ($t(10) = 2.15$, $p < 0.0456$, small effect). No other pairwise significant differences were observed.
- **Corrected Error Rate** (CER) represents fixed errors during text entry. Virtual keyboard users could see their mistakes and correct them, while signers had to use an "undo" feature to re-spell words. The formula for CER is: $CER = \frac{IF}{C+INF+F}$. CER was highest for virtual keyboard but there were no significant differences between conditions. **Uncorrected Error Rate** (UCER) is the rate of errors that remain in the transcribed string. For fingerspelling, we transcribed what participants spelled, not the perfect words displayed to them. The formula for UCER is: $UCER = \frac{INF}{C+INF+F}$. UCER overcounts errors corrected by auto-correct, not available to fingerspelling participants. There were no significant differences between conditions.

One of our Deaf team members conducted a qualitative analysis of the errors observed in participant videos. It was noted that participants did not move their wrists with enough clarity to distinguish between similar letters like 'P' and 'K' or 'U' and 'H'. Additionally, some participants did not execute the twisting motion of 'J' and instead flicked their fingers slightly. Groups of letters with similar handshapes like 'U' and 'R', or 'O', 'E', and 'C' were often challenging to distinguish, especially when co-articulated with adjacent letters. In some cases, it was unclear that a participant was fingerspelling two of the same letters in a row, which can be done by sliding the letter, bouncing it, or loosening and reforming the handshape. Another significant finding was that participants frequently made false starts with their phrases in Tap to Sign (more common) or with words in Push to Sign (less common). They would start a word, make a mistake or second-guess themselves, and then restart the entire word (sometimes going back multiple words, or redoing the entire phrase in Tap to Sign). This behavior unfavorably biases the results against our fingerspelling interface as the virtual keyboard smartphone texting interface has a well-established, refined, and familiar editing system.

4.5.3 Subjective judgements. Figure 6 describes the participants' assessment of subjective workload and their feedback on other subjective questions. The average overall NASA TLX scores were Tap to Sign 27.6, Push to Sign 35.3, and virtual keyboard 25.7. Paired t-tests with Benjamini-Hochberg correction between all three pairs of conditions revealed no statistically significant differences. In a final thoughts survey, six participants ranked Tap to Sign highest on overall preference, and six ranked virtual keyboard highest. Figure 5 (right) shows the averaged speeds over phrases for Study 1. The R^2 values for fitting the curves were 0.175, 0.299, and 0.134 for Tap to Sign, Push to Sign, and virtual keyboard, respectively.

4.6 Study 1 Discussion

4.6.1 Speed and Learning effect. Relative typing speed affects the types of tasks that users will attempt (for example, short messages versus email versus document preparation). It also correlates to method preference [35]. Our participants fingerspelled fastest with the Push to Sign interface, even though they had just finished fingerspelling 1000 phrases using Tap to Sign. One would expect the experience with Tap to Sign would give that method a significant advantage. However, the dominant language of participants in the study was ASL. As observed by Hanson [24], Deaf individuals may have difficulty with English spellings as the gestures in ASL have little relation to English. (To provide some perspective, learning ASL is considered as hard for English speakers as learning Japanese by the Language Institute at the Naval Postgraduate School. Imagine expecting a native English speaker to be able to spell in Japanese!) Perhaps being able to fingerspell one word at a time with Push to Sign was easier than trying to transcribe the entire phrase at once (though the TLX mental workload, effort, and frustration scores do not support this hypothesis). Similarly, for the virtual keyboard condition, the participants' visual attention was on the keys, not allowing continual observation of the spelling of the individual words themselves (as with Push to Sign).

4.6.2 Subjective judgments. The difference between the workloads of participants across the three conditions was narrow. Note that the relative ratings of Tap to Sign and virtual keyboard invert for the mental, effort, and frustration scores between the two studies. Interestingly, our participants rated Push to Sign worse than Tap to Sign on almost every metric (NASA TLX and Subjective Likert Questions) except performance (lower performance numbers are better). This last subjective rating is consistent with the superior objective speed and error metrics of the Push to Sign system. In a final questionnaire, 50% of the participants preferred fingerspelling with Tap to Sign for text entry.

4.6.3 Errors. The total error rates (corrected plus uncorrected) for the virtual keyboard are similar to other mobile keyboard studies which show 5-8% character error on the same phrase set [14, 15, 32, 33]. However, both the total error rates tended lower with fingerspelling, which, combined with the increased texting speed, led to an advantage in throughput for the fingerspelling methods over the virtual keyboard [68]. Granted, the error rates are not really comparable as the fingerspelling recognition was emulated, but the actual total error rate from fingerspelling should be lower with a live recognition system providing real-time auto-correction feedback to the user. It is possible that an actual recognition system could have better speeds, accuracies, and subjective ratings than our emulated system.

For example, one could imagine that a fingerspelling interface that used auto-correct and predictive text might be more effective than the equivalent virtual keyboard. With virtual smartphone keyboards, the user is focusing on the keys in order to hit them correctly. Little visual attention can be spared for lines above the Android keyboard which shows potential completions and the current word's spelling. Fingerspelling, on the other hand, is more like touch typing, where the typist can monitor the output simultaneously while hitting the keys. As we know from Wickens's multiple resource model for attention [62], simultaneous tasks are performed better when spread between multiple pools of attention (in this case, visual and haptic) versus single modalities (visual). Thus, perhaps fingerspelling users will be able to better take advantage of intelligent text entry features versus smartphone virtual keyboard users.

Manually analyzing the participant videos revealed the lack of articulation of individual characters. Compared to the fingerspelling one might see in an academic/professional setting (such as by interpreter students), there was a drastic difference in wrist mobility, such as twisting the letter 'J', as well as clarity of handshapes like 'R' and 'U'. When fingerspelling in the community, this kind of muddling is generally accepted, and it is expected that the recipient will make the distinctions on their own, especially given context clues. For example, it is rare to see someone spell every letter of "Arnold Schwarzenegger." The situation is somewhat analogous to gesture keyboard typing in which users speed through a series of sometimes poorly approximated letters and the keyboard pieces together the most likely correct word(s). As with the gesture keyboard, leveraging dictionaries and stochastic grammars should buoy recognition rates significantly, perhaps allowing faster and less precise fingerspelling of words.

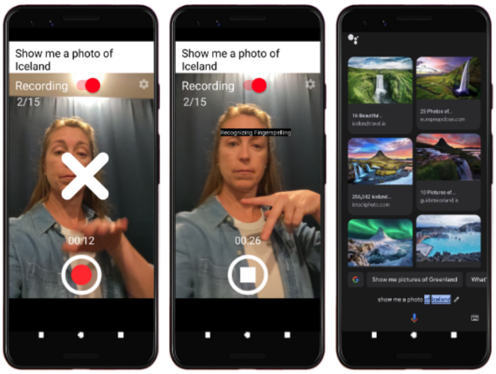
Users commonly made mistakes while fingerspelling and often needed to redo the entire word or phrase. This observation suggests potential interface improvements, such as having the "backspace" button restart the entire word (similar to gesture keyboards) rather than just erasing the last character. This behavior is already commonly accepted in the Deaf community, where individuals will fingerspell a word, make a mistake, and then restart the word from the beginning by shaking their hand as if wiping a whiteboard clean.

4.6.4 Overall observations. The results support the hypothesis that fingerspelling might be valued by the Deaf community for text entry. 50% of participants preferred a fingerspelling interface, and the studies suggest a fingerspelling interface might be faster and more accurate.

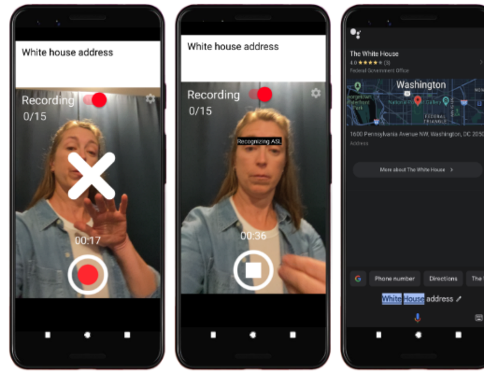
A closely related application domain for fingerspelling recognition is commanding a mobile assistant (personal assistant on a smartphone). Many mobile assistant commands will naturally involve fingerspelling for proper nouns such as contact names and locations ("CALL M-E-L-O-D-Y"). The commands themselves are relatively limited in number and scope by the assistant's capabilities. In the next study, we investigate whether fingerspelling these commands would make an adequate first interface or whether the system needs to recognize signed commands as well as fingerspelling in order to be accepted by the Deaf.

5 STUDY 2: WALK-UP USABILITY OF A MOBILE ASSISTANT THAT CAN BE COMMANDED WITH SIGN LANGUAGE

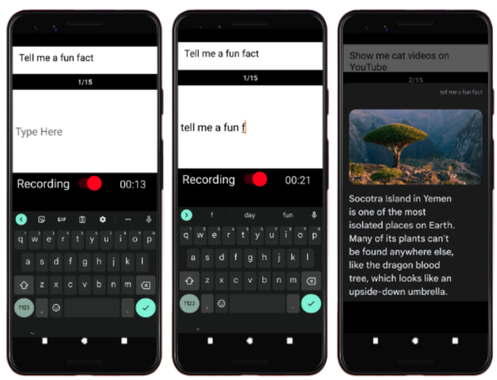
Findings from the first study motivated a walk-up usability study into another potential use of signing for text entry on smartphones: using a mobile assistant. Findings from study 1 revealed



(a) Screenshots for the *Fingerspelling Only* condition.



(b) Screenshots for *Smartphone-signing* condition.



(c) Screenshots for *Virtual Keyboard* condition.

Fig. 7. Wizard-of-Oz prototype screenshots before (left), during (middle), and after (right) text entry. Fingerspelling recognition, sign recognition, and mobile assistant responses are emulated.

that fingerspelling could be faster than virtual keyboard typing after users acclimatize to this method. In study 1 the subjective preference of users was comparable with the virtual keyboard, and the reported workload was higher for both fingerspelling conditions. This result motivates a comparison of fingerspelling with smartphone signing (i.e., using one hand for signing to a smartphone while the other hand is holding the smartphone).

Similar to the first study, we designed three wizard-of-oz prototypes and recruited 12 participants. However, unlike the first study, which was a lab-based controlled experimental study, this study was conducted at the biennial National Association for the Deaf 2022 conference. Therefore, we had less control over the demographics of our participants and no time to acclimatize them to these new proposed methods of smartphone text entry.

5.1 Prototypes

Figure 7 shows screenshots of the three different prototypes designed and used for this study:

(1) Smartphone-Signing: The prototype used a Tap to Sign design. Participants were instructed to use signs wherever possible to complete the prompted task. To emulate the function of a mobile assistant, a screenshot of a typical response from Google's mobile Assistant appeared

on the screen after the participant finished their signing (instead of simply a copy of the prompt text as in the previous study). As before, no sign language recognition was used, and the response was the same no matter what the user signed.

(2) Fingerspelling: Identical to the Signing prototype except that participants were asked to only fingerspell.
(3) Virtual keyboard: Participants were asked to type the given prompt into the phone, and as with the other two input styles, they were shown a screenshot which emulated Google's mobile assistant's response. Unlike study 1, predictive text options remained on in this study since we were not measuring speeds and errors.

5.2 Study Design and Choice of Prompts

A total of 45 phrases were chosen, and 15 were used for each of the three conditions: Smartphone-signing, Fingerspelling Only, and virtual keyboard. The mobile assistant commands were selected and counterbalanced to include commands from five categories of popular commands used with mobile Assistant: "Entertainment", "Information", "Home Control", "Personal", and "Navigation". A list of these prompts are provided in Table 5 (Appendix B). Unlike the first study, we did not measure the text entry speeds of participants since the goal was to investigate a potential use case (the mobile assistant) based on the findings from study 1.

Twelve participants were recruited while visiting our vendor booth based on their willingness to schedule a study session while at the conference. Two participants had movement disabilities that affected their signing noticeably. Participants signed a consent form and then completed a pre-experiment questionnaire. This questionnaire asked about their familiarity with personal assistant devices and asked about where they have seen or used them. Then, participants used each of the three Wizard-of-Oz prototypes (Latin square counterbalanced to combat order effects), answering questions for each condition.

For each condition, participants answered System Usability Scale questions [23], as well as user friendliness, and net promoter score questions [23] (all questions were presented in ASL, using Berke et al.'s resources [9]). These questions are standardized system usability scales, which have traditionally been used in Human-Computer Interaction (HCI) studies, and they have been translated into ASL to ensure accessibility for the Deaf and Hard-of-hearing participants.

In the end, after participants used and answered questions for all three conditions, they were asked closing questions. These consisted of questions asking about preference and naturalness of the different conditions. Then, a brief semi-structured interview was conducted in ASL by the Deaf members of our team. Questions included (1) *Can you suggest some ideas on how you would use a mobile assistant that takes sign-language input?* (2) and *Any other comments or feedback?* We analyzed the recommendations for the mobile assistant applications using an initial deductive framework based on prior research regarding the interest of deaf and hard-of-hearing (DHH) users in sign-language interactions with personal assistant devices [20, 22]. However, we also remained open to the emergence of new use-case categories or the possibility of previously identified categories becoming irrelevant in the context of smartphone assistants. Responses to second question were analyzed using an iterative thematic analysis, in which the Deaf team member who conducted the interviews examined the whole list of transcribed responses, added labels, and then combined them into major categories. The quotes presented are edited for clarity and brevity.

5.3 Quantitative Results

Condition	Average SUS scores [23]	Average "User-friendliness" score	NPS score [23]
Smartphone-signing	75.8 (13.7)	6.17 (1.03)	83.3%
Fingerspelling Only	63.5 (16.9)	4.75 (2.30)	0%
Virtual Keyboard	79.8 (11.9)	6.10 (0.994)	100%

Table 2. Comparison of SUS and user-friendliness scores across the three conditions.

Table 2 summarizes the average scores on system usability scale [7] and user-friendliness across the three conditions. Figure 8 shows the subjective preferences of users when using the three prototypes. To compare conditions, a t-test with Benjamini-Hochberg correction for multiple hypotheses was employed:

- **"How likely is it that you would recommend [this] to a friend or colleague?" [0=Not at all likely to 10=Extremely likely]:** A statistically significant difference was found between fingerspelling and signing (p=0.02, n(22)=1.96, small effect) and fingerspelling and virtual keyboard (p=0.019, n(20)=2.22, medium effect).
- **System Usability Scale (SUS) questions**: The difference in mean SUS score between fingerspelling and signing reached significance (p=0.031, n(22)=1.96, small effect) as well as fingerspelling and virtual keyboard (p=0.0096, n(20)=-2.54, medium effect)[7].
- **"Overall, I would rate the user-friendliness of this as [1=Worst Imaginable to 7=Best Imaginable]":** The difference between fingerspelling (average = 4.75) and signing (average = 6.17) was statistically significant (p=0.032, n(22)=1.95, medium effect).

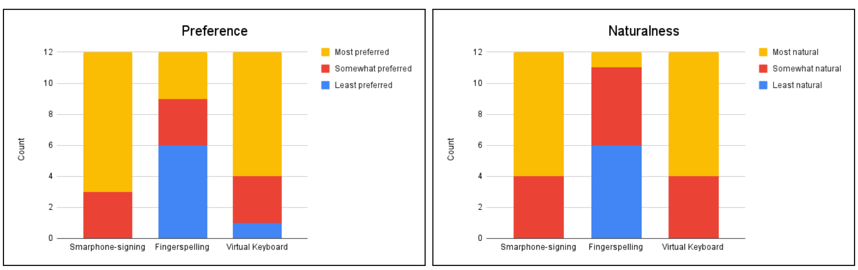


Fig. 8. Participants' subjective feedback on preference and naturalness across the three conditions in study 2.

[7]The number of responses for virtual keyboard condition were 10 instead of 12. Therefore, there is a difference in degrees of freedom reported for any calculation involving virtual keyboard data. To perform an appropriate effect size calculation, we use Cohen's D when comparing ASL and fingerspelling, and Hedges' G for all calculations involving virtual keyboard since it provides a measure of effect size weighted according to the relative size of each sample

5.4 Qualitative Results

Category	Use-case
Navigation	Navigating to an address (P2, P3, P6), Viewing traffic (P11), Restaurants (P9, P11), Ride-sharing services (P9), Food-ordering services
Communication	Communication with others (P1), Messages (P7), Email (P8)
Information	Color of a flag (P4), Weather (P6, P7, P8, P10), News (P7), Technology release (P7), Recipes (P9), Gas Prices (P12), Deaf Events (P12)
Device Control	Smartphone commands (P5), Temperature control (P5), Home information, e.g. lights on, doors locked (P5), Open Apps (P12), Record (P10)
Personal Use	Appointments (P6, P7), Calendar (P6), Reminders (P6), Calendar (P6, P7), Productivity stats (P6), Reservations (P7), Tasks (P7), Paying Bills (P7), Movie releases (P7), Flights (P7), Timezones (P7), Health (P7), Pandemic (P7), Scriptures (P7), Documents (P8), Grocery List (P9)

Table 3. Tasks suggested by DHH users for a mobile assistant that understands sign-language.

5.4.1 Excitement and uses of mobile assistant. Participants were enthusiastic about the potential of commanding the mobile assistant with signing. For example, P5 said that if machine learning is truly able to recognize one-handed sign, he will use it (mobile assistant) for more things. Similarly, P7 expressed their enthusiasm *"I would use this everyday. I hope this could be ready today"*. P10 commented that a mobile assistant that can take sign-language input could be especially useful when mobile: *"It is a fantastic resource, even when I am moving or traveling."* Participants suggested several uses of a mobile assistant based on signing which broadly fall under the five categories of commands we incorporated in our study: Navigation (n=7), Communication (n=3), Information (n=8), Device control (n=3), Personal uses (n=3). Table 3 lists these different suggestions from the participants. Notably, our analysis revealed that many participants would like to use the mobile assistant for navigation and information-seeking related tasks.

5.4.2 Concerns and suggested improvements. Six participants expressed concerns. Some related to the limitations of our wizard-of-oz prototype. For example, P8 mentioned that they had trouble adjusting to the dark mode. They also suggested the need for better video stabilization: *"Could we use a stabilizer for people moving around a lot? Running, walking or otherwise on the go?"* P7 mentioned that at times it was difficult for them to adjust to the limited field-of-view of a mobile screen and suggested he would like to use a *"fishbowl camera to capture more space."* P5 asked *"will it pan to follow them, if they set it down?"* Two participants were also curious about how the mobile assistant would work *"while its dark, if you're in bed"* or *"Can it read from a distance? or from up really close?"*

Three participants noted that the auto-complete feature in the virtual keyboard condition was useful, and such a feature for signing based text entry would be useful as well. Two participants mentioned more feedback would make the signing condition better: *"While using text I can see input clearly, with sign it's not so clear. If I can see more feedback as I sign, that is helpful. It combines both worlds of natural ASL and quicker texting, and increases the visual accuracy."*

Unlike study 1, the participants were not given a chance to acclimatize to this new style of text entry. Three participants (one of whom had a movement disability) reported that smartphone signing challenges their natural habits of signing. Participants noted that there is no one-to-one correspondence between ASL and English. While using the fingerspelling only mobile assistant, P3 commented *"ASL drops words [Signs a, the, and is]. It is hard to not sign when told to fingerspell. The prompts used are highly English-y (words like brighten, navigate). English grammar causes contrived* *signing and English word order (PSE)."* Another participant mentioned how one-handed versions of two handed signs can look the same (e.g., OPEN vs. DOOR).

5.5 Study 2 Discussion

In study 2, run at NAD 2022 which is a large public Deaf conference, participants preferred the virtual keyboard and Smartphone-signing over fingerspelling. There was no significant difference between virtual keyboard and Smartphone-signing. There was a significant difference, however, between those two and Fingerspelling Only; most, if not all participants did not enjoy and were not used to fingerspelling every word used in mobile assistant commands. It is suspected that participants' prior experience with using a smartphone (particularly video chatting with other Deaf people in ASL while holding their phone in one hand) strongly influences their experiences [34, 39].

Since the participants did not have practice with signing as a text entry method, it was remarkable that the scores were similar to that of the virtual keyboard (a method commonly used by most of the participants). This result suggests that there is great potential in signing for text entry on smartphones.

Our qualitative findings, particularly the list of suggested use cases presented in Table 3, can motivate the creation of datasets of signs that are most often used during smartphone signing (e.g., APPOINTMENT TIME 7 or CALL M-E-L-O-D-Y). Whereas prior research has investigated the use of personal assistant devices with a focus on collecting data to support smart-home hubs [20, 22, 37], mobile assistants are likelier to be used for navigation and communication tasks. This difference may be due to the portability and convenience of mobile devices, as users can easily access their personal assistant while on-the-go. Even so, some use cases, such as setting alarms or appointments, are consistent across different types of personal assistants. In the future, smartphone video corpora could be collected based on our identified categories to train and test underlying sign recognition models for mobile assistants.

6 INFORMING FUTURE RESEARCH AND DESIGN

Sign recognition is still relatively crude but is improving [46]. Identifying practical use cases with the DHH community for sign recognition while it is still in its infancy is important for progress in the field [11, 46]. With smartphone signing, the recognition task is substantially simplified due to the phone's affordances and limited field-of-view, which naturally encourages users to sign more compactly and closer to the camera. Advancements in mobile phone cameras [61] and hand landmark detection [67] are already at a stage where practical applications like fingerspelling keyboards or mobile assistants can be implemented in the near future.

Our findings inform several audiences. For HCI and accessibility researchers, our findings motivate investigations on using fingerspelling and smartphone-signing as input to several types of devices, especially handheld devices like mobile phones. Our study has provided the needed empirical support to motivate the design of future systems that can allow Deaf users to use signing keyboards instead of the standard virtual keyboard on their phones. Future researchers can closely work with AI researchers to develop the underlying sign recognition algorithms for this task to best support the user and achieve good recognition results. For sign language linguists, our findings motivate future research on how signing is changing with advancements in mobile computing. To our knowledge, this is the first study that recruited Deaf participants and rigorously investigated their interactions with a sign language-based text entry system on mobile phone. For computer vision researchers, our work serves as a "problem setting" paper and presents a new direction for developing underlying technology to support sign recognition models for text entry on mobile phones.

7 LIMITATIONS AND FUTURE WORK

Our study has several limitations that future research can address. Firstly, our prototypes did not have an underlying AI system, and the performance of AI was merely mimicked. Future studies can either mimic the imperfect nature of an underlying AI or test user performance with prototypes that have actual underlying AI systems. Participants in both studies could use backspace to remove individual characters on the virtual keyboard, but in all signing conditions, participants could not remove a single character. They could only undo individual words or phrases (depending on the condition). Improving the editing systems will allow for more sophisticated comparisons between the conditions. We did not provide participants in all conditions of study 1 with an auto-complete option. Once fingerspelling recognition improves to a stage where it can be used for text entry, researchers can evaluate the benefits of word and/or sentence completion options. Study 2 only used subjective preference instruments and received some open-ended feedback. Future studies can use a more ecologically valid approach and assess other measures of task performance. We used English prompts in our studies, including for smartphone signing. English prompts might have inadvertently encouraged participants to sign with a more English-y grammar. In the future, creating scenarios and tasks for users to complete may elicit more natural sign. Study 2 was also designed as a walk-up usability study. More research is needed to investigate how DHH users would interact with an actual mobile assistant once they have more experience. Lastly, although we present a thematic organization of potential use cases of a mobile assistant that takes sign language input, research is needed to create an organized corpus of the most requested mobile assistant commands from DHH individuals.

8 CONCLUSION

Our goal was to investigate the usability of smartphone-signing, especially fingerspelling, as a text entry method. We conducted two studies. Our first study found that if participants are given time to acclimatize to a text entry interface based on fingerspelling, their speeds and subjective preferences regarding using fingerspelling as a text entry method improve over time. With experience, half of the participants preferred fingerspelling for text entry, and fingerspelling proved faster. Study 2 showed that walk-up users prefer signing and the virtual keyboard over fingerspelling for interacting with a mobile assistant. Even so, fingerspelling recognition does seem to be a worthwhile endeavor for sign recognition efforts for interfaces where the user can be enticed to gain sufficient experience with the interaction or as part of a more general signing system for interacting with a mobile assistant.

References

[1] 2021. International Day of Sign Languages. https://www.un.org/en/observances/sign-languages-day
[2] 2022. Deafness and hearing loss. https://www.who.int/health-topics/hearing-loss#tab=tab_1
[3] Mehdi Assefi, Guangchi Liu, Mike P Wittie, and Clemente Izurieta. 2015. An experimental evaluation of apple siri and google speech recognition. *Proccedings of the 2015 ISCA SEDE* 118 (2015).
[4] Judith E Harkins St Matthew Bakke. 2003. Technologies for Communication. *Oxford Handbook of Deaf Studies, Language, and Education* (2003), 406.
[5] F Bakken. 2004. The Inside Text: Social, Cultural and Design Perspectives on SMS, chapter SMS Use Among Deaf Teens and Young Adults in Norway.
[6] Frøydis Bakken. 2005. SMS use among deaf teens and young adults in Norway. In *The inside text*. Springer, 161–174.
[7] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies* 4, 3 (2009), 114–123.
[8] Souad Baowidan. 2021. Improving realism in automated fingerspelling of American sign language. *Machine Translation* 35, 3 (2021), 387–404.
[9] Larwan Berke, Matt Huenerfauth, and Kasmira Patel. 2019. Design and Psychometric Evaluation of American Sign Language Translations of Usability Questionnaires. *ACM Trans. Access. Comput.* 12, 2, Article 6 (jun 2019), 43 pages.

https://doi.org/10.1145/3314205

[10] Harry Bornstein. 1965. READING THE MANUAL ALPHABET–A RESEARCH PROGRAM FOR DEVELOPING A FILMED PROGRAM FOR TEACHING THE MANUAL ALPHABET. (1965).

[11] Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris. 2019. Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility* (Pittsburgh, PA, USA) *(ASSETS '19)*. Association for Computing Machinery, New York, NY, USA, 16–31. https://doi.org/10.1145/3308561.3353774

[12] Anna Cavender, Richard E Ladner, and Eve A Riskin. 2006. MobileASL: Intelligibility of sign language video as constrained by mobile phone technology. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*. 71–78.

[13] Ananya Choudhury, Anjan Kumar Talukdar, Kandarpa Kumar Sarma, and Manas Kamal Bhuyan. 2021. An adaptive thresholding-based movement epenthesis detection technique using hybrid feature set for continuous fingerspelling recognition. *SN Computer Science* 2, 2 (2021), 1–21.

[14] JAMES Clawson, Kent Lyons, Edward Clarkson, and Thad Starner. 2006. Mobile text entry: An empirical study and analysis of mini–qwerty keyboards. *Submitted to the Transaction on Computer Human Interaction Journal* (2006).

[15] James Clawson, Kent Lyons, Thad Starner, and Edward Clarkson. 2005. The impacts of limited visual feedback on mobile text entry for the twiddler and mini-qwerty keyboards. In *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)*. IEEE, 170–177.

[16] Michael Erard. 2017. Why Sign-Language Gloves Don't Help Deaf People. *The Atlantic* (Nov 2017).

[17] Jordan Fenlon and Erin Wilkinson. 2015. Sign languages in the world. *Sociolinguistics and deaf communities* 1 (2015), 5.

[18] Kamala Gajurel, Cuncong Zhong, and Guanghui Wang. 2021. A Fine-Grained Visual Attention Approach for Fingerspelling Recognition in the Wild. *CoRR* abs/2105.07625 (2021). arXiv:2105.07625 https://arxiv.org/abs/2105.07625

[19] Leah Geer. 2010. ASL fingerspelling: Comparing native and non-native signers. *Unpublished manuscript Gallaudet University* (2010).

[20] Abraham Glasser, Vaishnavi Mande, and Matt Huenerfauth. 2021. Understanding Deaf and Hard-of-Hearing Users' Interest in Sign-Language Interaction with Personal-Assistant Devices. In *Proceedings of the 18th International Web for All Conference* (Ljubljana, Slovenia) *(W4A '21)*. Association for Computing Machinery, New York, NY, USA, Article 24, 11 pages. https://doi.org/10.1145/3430263.3452428

[21] Abraham Glasser, Matthew Watkins, Kira Hart, Sooyeon Lee, and Matt Huenerfauth. 2022. Analyzing Deaf and Hard-of-Hearing Users' Behavior, Usage, and Interaction with a Personal Assistant Device That Understands Sign-Language Input. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 306, 12 pages. https://doi.org/10.1145/3491102.3501987

[22] Abraham Glasser, Matthew Watkins, Kira Hart, Sooyeon Lee, and Matt Huenerfauth. 2022. Analyzing Deaf and Hard-of-Hearing Users' Behavior, Usage, and Interaction with a Personal Assistant Device That Understands Sign-Language Input. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 306, 12 pages. https://doi.org/10.1145/3491102.3501987

[23] DF Hamilton, Judith V Lane, P Gaston, JT Patton, DJ Macdonald, AHRW Simpson, and CR Howie. 2014. Assessing treatment outcomes using a single question: the net promoter score. *The bone & joint journal* 96, 5 (2014), 622–628.

[24] Vicki L Hanson. 1981. When a word is not the sum of its letters: Fingerspelling and spelling. *Haskins Laboratories Status Report on Speech Research* 65 (1981), 145–155.

[25] Joseph Hill. 2020. Do deaf communities actually want sign language gloves? *Nature Electronics* 3, 9 (2020), 512–513.

[26] Jonathan Keane, Diane Brentari, and Jason Riggle. 2012. Coarticulation in ASL fingerspelling. In *Proceedings of the North East Linguistic Society*, Vol. 42.

[27] Jonathan Keane, Jason Riggle, and Diane Brentari. 2013. The timing of ASL fingerspelling. In *Unpublished conference presentation, Texas Linguistics Society*, Vol. 14.

[28] Elizabeth Keating and Gene Mirus. 2003. American Sign Language in virtual space: Interactions between deaf users of computer-mediated video communication and the impact of technology on language practices. *Language in Society* 32, 5 (2003), 693–714.

[29] Taehwan Kim, Jonathan Keane, Weiran Wang, Hao Tang, Jason Riggle, Gregory Shakhnarovich, Diane Brentari, and Karen Livescu. 2016. Lexicon-Free Fingerspelling Recognition from Video: Data, Models, and Signer Adaptation. *Computer Speech and Language* 46 (09 2016). https://doi.org/10.1016/j.csl.2017.05.009

[30] Oscar Koller. 2020. Quantitative survey of the state of the art in sign language recognition. *arXiv preprint arXiv:2008.09918* (2020).

[31] Hamzah Luqman, El-Sayed M El-Alfy, and Galal M BinMakhashen. 2021. Joint space representation and recognition of sign language fingerspelling using Gabor filter and convolutional neural network. *Multimedia Tools and Applications* 80, 7 (2021), 10213–10234.

[32] Kent Lyons, Daniel Plaisted, and Thad Starner. 2004. Expert chording text entry on the twiddler one-handed keyboard. In *Eighth International Symposium on Wearable Computers*, Vol. 1. IEEE, 94–101.

[33] Kent Lyons, Thad Starner, Daniel Plaisted, James Fusia, Amanda Lyons, Aaron Drew, and E. W. Looney. 2004. Twiddler Typing: One-Handed Chording Text Entry for Mobile Phones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vienna, Austria) *(CHI '04)*. Association for Computing Machinery, New York, NY, USA, 671–678. https://doi.org/10.1145/985692.985777

[34] Kelly Mack, Danielle Bragg, Meredith Ringel Morris, Maarten W Bos, Isabelle Albi, and Andrés Monroy-Hernández. 2020. Social app accessibility for deaf signers. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–31.

[35] I Scott MacKenzie. 2012. Human-computer interaction: An empirical research perspective. (2012).

[36] I Scott MacKenzie and R William Soukoreff. 2003. Phrase sets for evaluating text entry techniques. In *CHI'03 extended abstracts on Human factors in computing systems*. 754–755.

[37] Vaishnavi Mande, Abraham Glasser, Becca Dingman, and Matt Huenerfauth. 2021. Deaf Users' Preferences Among Wake-Up Approaches during Sign-Language Interaction with Personal Assistant Devices. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI EA '21)*. Association for Computing Machinery, New York, NY, USA, Article 370, 6 pages. https://doi.org/10.1145/3411763.3451592

[38] Ross Mitchell, Travas Young, Bellamie Bachleda, and Michael Karchmer. 2006. How Many People Use ASL in the United States? Why Estimates Need Updating. *Sign Language Studies* 6 (03 2006). https://doi.org/10.1353/sls.2006.0019

[39] Amanda Morris. 2022. How Sign Language Evolves as Our World Does. *The New York Times* (Jul 2022).

[40] Carol A Padden and Darline Clark Gunsauls. 2003. How the alphabet came to be used in a sign language. *Sign Language Studies* (2003), 10–33.

[41] Kseniia Palin, Anna Maria Feit, Sunjun Kim, Per Ola Kristensson, and Antti Oulasvirta. 2019. How do people type on mobile devices? Observations from a study with 37,000 volunteers. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*. 1–12.

[42] Kseniia Palin, Anna Maria Feit, Sunjun Kim, Per Ola Kristensson, and Antti Oulasvirta. 2019. How Do People Type on Mobile Devices? Observations from a Study with 37,000 Volunteers. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (Taipei, Taiwan) *(MobileHCI '19)*. Association for Computing Machinery, New York, NY, USA, Article 9, 12 pages. https://doi.org/10.1145/3338286.3340120

[43] Katerina Papadimitriou and Gerasimos Potamianos. 2019. End-to-End Convolutional Sequence Learning for ASL Fingerspelling Recognition. In *INTERSPEECH*.

[44] Mary R Power and Des Power. 2004. Everyone here speaks TXT: Deaf people using SMS in Australia and the rest of the world. *Journal of deaf studies and deaf education* 9, 3 (2004), 333–343.

[45] David Quinto-Pozos. 2010. Rates of fingerspelling in american sign language. In *Poster presented at 10th Theoretical Issues in Sign Language Research conference, West Lafayette, Indiana*, Vol. 30.

[46] Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2021. Sign Language Recognition: A Deep Survey. *Expert Systems with Applications* 164 (2021), 113794. https://doi.org/10.1016/j.eswa.2020.113794

[47] Peter A Reich and Margaret Bick. 1977. How visible is visible English? *Sign Language Studies* 14, 1 (1977), 59–72.

[48] Sherry Ruan, Jacob O. Wobbrock, Kenny Liou, Andrew Ng, and James A. Landay. 2018. Comparing Speech and Keyboard Text Entry for Short Messages in Two Languages on Touchscreen Phones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 159 (jan 2018), 23 pages. https://doi.org/10.1145/3161187

[49] Rudy Schusteritsch, Shailendra Rao, and Kerry Rodden. 2005. Mobile Search with Text Messages: Designing the User Experience for Google SMS. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems* (Portland, OR, USA) *(CHI EA '05)*. Association for Computing Machinery, New York, NY, USA, 1777–1780. https://doi.org/10.1145/1056808.1057020

[50] Shikhar Sharma and Krishan Kumar. 2021. ASL-3DCNN: American sign language recognition technique using 3-D convolutional neural networks. *Multimedia Tools and Applications* 80, 17 (2021), 26319–26331.

[51] Jitesh Shetty and Jafar Adibi. 2004. The Enron email dataset database schema and brief statistical report. *Information sciences institute technical report, University of Southern California* 4, 1 (2004), 120–128.

[52] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2018. American sign language fingerspelling recognition in the wild. In *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 145–152.

[53] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019. Fingerspelling recognition in the wild with iterative visual attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5400–5409.

[54] Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. 2015. Investigating the Dexterity of Multi-Finger Input for Mid-Air Text Entry. 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, 3643–3652. https://doi.org/10.1145/2702123.2702136

[55] T. Starner. 2004. Keyboards redux: fast mobile text entry. *IEEE Pervasive Computing* 3, 3 (2004), 97–101. https://doi.org/10.1109/MPRV.2004.1321035

[56] Thad E Starner. 1995. *Visual Recognition of American Sign Language Using Hidden Markov Models.* Technical Report. Massachusetts inst of tech Cambridge dept of brain and cognitive sciences.

[57] Caroline Tagg. 2012. *Discourse of text messaging: Analysis of SMS communication.* Bloomsbury Publishing.

[58] Jorge Andres Toro, John C McDonald, and Rosalee Wolfe. 2014. Fostering better deaf/hearing communication through a novel mobile app for fingerspelling. In *International Conference on Computers for Handicapped Persons*. Springer, 559–564.

[59] Jason Tu, Angeline Vidhula Jeyachandra, Deepthi Nagesh, Naresh Prabhu, and Thad Starner. 2021. *Typing on Tap: Estimating a Finger-Worn One-Handed Chording Keyboard's Text Entry Rate.* Association for Computing Machinery, New York, NY, USA, 156–158. https://doi.org/10.1145/3460421.3480428

[60] Colton J Turner, Barbara S Chaparro, and Jibo He. 2021. Typing on a smartwatch while mobile: a comparison of input methods. *Human factors* 63, 6 (2021), 974–986.

[61] Neal Wadhwa, Rahul Garg, David E Jacobs, Bryan E Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T Barron, Yael Pritch, and Marc Levoy. 2018. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics (ToG)* 37, 4 (2018), 1–13.

[62] Christopher D Wickens. 2008. Multiple resources and mental workload. *Human factors* 50, 3 (2008), 449–455.

[63] Sherman Wilcox. 1992. *The phonetics of fingerspelling*. Vol. 4. John Benjamins Publishing.

[64] Jacob O Wobbrock. 2007. Measures of text entry performance. *Text entry systems: Mobility, accessibility, universality* (2007), 47–74.

[65] Rosalee Wolfe, John McDonald, Jorge Toro, Souad Baowidan, Robyn Moncrief, and Jerry Schnepp. 2015. Promoting Better Deaf/Hearing Communication Through an Improved Interaction Design for Fingerspelling Practice. In *Universal Access in Human-Computer Interaction. Access to Today's Technologies*, Margherita Antona and Constantine Stephanidis (Eds.). Springer International Publishing, Cham, 495–505.

[66] Xin Yi, Chun Yu, Weinan Shi, and Yuanchun Shi. 2017. Is it too small?: Investigating the performances and preferences of users when typing on tiny QWERTY keyboards. *International Journal of Human-Computer Studies* 106 (2017), 44–62.

[67] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214* (2020).

[68] Mingrui Ray Zhang, Shumin Zhai, and Jacob O. Wobbrock. 2019. Text Entry Throughput: Towards Unifying Speed and Accuracy in a Single Performance Metric. 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, 1–13. https://doi.org/10.1145/3290605.3300866

A SPEEDS IN WPM IN STUDY 1

Participant	Tap to Sign	Push to Sign	Virtual Keyboard
P13	29.889	29.062	27.523
P14	29.471	38.179	26.381
P15	38.506	34.042	19.757
P16	37.193	48.902	35.564
P17	40.616	41.439	32.616
P18	52.656	65.145	48.117
P19	20.718	24.612	13.674
P20	34.695	29.843	62.633
P21	47.227	41.567	34.732
P22	41.914	44.897	30.291
P23	54.869	52.314	45.382
P24	39.414	47.643	36.779
Average	38.931	41.470	34.454
Average w/o P20	39.316	42.527	31.892
Standard Deviation	10.138	11.292	10.111

Table 4. Average speeds of participants in study 1 across three conditions during the last 5-minute session.

B SELECTED MOBILE ASSISTANT COMMANDS FOR STUDY 2

Below is a table with the list of the 45 mobile assistant commands that were used for Study 2. The top row has the 5 category labels, and there are 9 commands for each category.

Entertainment	Information	Home Control	Personal	Navigation
Hey <assistant>, play my Cool playlist	OK <assistant>, when is Hanukkah	Hey <assistant>, set an alarm for 8 AM	OK <assistant>, what's on my calendar	Hey <assistant>, how's the traffic home
OK <assistant>, play some music	Hey <assistant>, 100 euros in dollars	OK <assistant>, set a timer for 2 hours from now	Hey <assistant>, when is my next appointment	OK <assistant>, navigate home
OK <assistant>, show me my pictures from Japan	OK <assistant>, what is 7 * 18	Hey <assistant>, turn off the living room light	OK <assistant>, remember that I parked in garage A	Hey <assistant>, where can I get coffee
Hey <assistant>, open Netflix	Hey <assistant>, how do you say where's the bathroom in Spanish	OK <assistant>, set the thermostat to 68	Hey <assistant>, check my email	OK <assistant>, show me gas stations on my route
Hey <assistant>, tell me a joke	OK <assistant>, what time is sunset	Hey <assistant>, lock my door	OK <assistant>, tomorrow morning remind me to buy eggs	Hey <assistant>, how far is Walmart
OK <assistant>, tell me a fun fact	Hey <assistant>, what time is it in Paris	OK <assistant>, turn on battery saver	Hey <assistant>, make a dinner reservation	OK <assistant>, where am I
Hey <assistant>, show me cat videos on YouTube	OK <assistant>, how many cats are there in America	Hey <assistant>, show me my backyard	OK <assistant>, show me my flight	Hey <assistant>, when does Costco close
OK <assistant>, give me a motivational quote	Hey <assistant>, how far away is Mars	OK <assistant>, brighten the living room light	Hey <assistant>, read messages	OK <assistant>, navigate to work
OK <assistant>, show me a photo of Iceland	OK <assistant>, what's the weather this week	Hey <assistant>, turn off the bedroom light	OK <assistant>, add flour to my list	Hey <assistant>, white house address

Table 5. List of 45 selected mobile assistant commands for Study 2

Received January 2023; revised May 2023; accepted June 2023