



Quadmetric Optimized Thumb-to-Finger Interaction for Force Assisted One-Handed Text Entry on Mobile Headsets

LIK HANG LEE, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong and Center for Ubiquitous Computing, University of Oulu, Finland

KIT YUNG LAM, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong

TONG LI, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong

TRISTAN BRAUD*, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong

XIANG SU, Department of Computer Science, The University of Helsinki, Finland

PAN HUI, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong and Department of Computer Science, The University of Helsinki, Finland

Augmented reality head-worn computers often feature small-sized touch interfaces that complicate interaction with content, provide insufficient space for comfortable text input, and can be awkward to use in social situations. This paper presents a novel one-handed thumb-to-finger text entry solution for augmented reality head-worn computers. We design a glove composed of 12 force-sensitive nodes featuring an ambiguous keyboard layout. We first explore the viability of force disambiguation to evaluate the force division within the force spectrum. We select a 3-level force division as it allows to considerably reduce the number of keys while featuring a high (83.9%) accuracy. Following this pilot study, we map the 26 English characters onto the 9 nodes located on the index, middle and ring fingers in a 3-3-3 configuration, and attribute the space, enter and backspace keys to the remaining three nodes. We consider text entry performance as a quadmetric optimization problem considering the following criteria: goodness of character pairs, layout similarity to the QWERTY keyboard, easiness of force interaction, and comfort level of thumb reach. The resulting layout strikes a balance between performance and usability. We finally evaluate the quadmetric optimized layout over 6 sessions with 12 participants. The participants achieve an average text entry rate of 6.47 WPM with 6.85% error rate in the final session, which is significantly faster than existing thumb-to-finger solutions. In addition, our one-handed text entry system enhances the user mobility compared to other state-of-the-art solutions by freeing one hand, while allowing the user to direct his visual attention to other activities.

*This is the corresponding author: braudt@ust.hk

Authors' addresses: Lik Hang LEE Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong Center for Ubiquitous Computing, University of Oulu, Finland; Kit Yung LAM Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong; Tong LI Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong; Tristan BRAUD Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong; Xiang SU Department of Computer Science, The University of Helsinki, Finland; Pan HUI Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong; Department of Computer Science, The University of Helsinki, Finland.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2019 Association for Computing Machinery.

2474-9567/2019/9-ART94 \$15.00

<https://doi.org/10.1145/3351252>

CCS Concepts: • **HUMAN-CENTERED COMPUTING**; • **Human-computer interaction**;

Additional Key Words and Phrases: Human-computer interaction, Thumb-to-finger interaction, Text input, Smart wearable

ACM Reference Format:

Lik Hang LEE, Kit Yung LAM, Tong LI, Tristan BRAUD, Xiang SU, and Pan HUI. 2019. Quadmetric Optimized Thumb-to-Finger Interaction for Force Assisted One-Handed Text Entry on Mobile Headsets. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 3, Article 94 (September 2019), 27 pages. <https://doi.org/10.1145/3351252>

1 INTRODUCTION

Augmented reality (AR) head-worn computers are emerging as the next major smart wearables trend after smartphones and smartwatches. Various applications such as emails, social media and instant messaging are heavily dependent on text entry. However, the current text input approaches are either socially awkward [1] or lack mobility [2], which prevents the wider application of head-worn computers in mobile scenarios. For instance, both the touch interface on the frame of Google Glass and the mid-air hand gestures of Microsoft HoloLens require lifting up the hand to the eye level, which is not only tedious but also draws unwanted attention from the surroundings. Speech input, although efficient for simple navigation, may cause disturbance and is often not socially acceptable [1]. VR goggles mainly rely on handheld controllers. Handheld controllers prevent the user to use his hands for other purposes, and are thus not appropriate in mobile scenarios. In order to facilitate the adoption of head-worn computers, there is a need for new text entry approaches considering the portability, social acceptance, and easiness of input.

Text entry through thumb-to-finger interaction has appealing properties such as discreet [1] and subtle [3] movements, and provides comfortable yet accurate input [4] with no visual demand on the keyboard layout [5]. Thumb-to-finger interaction is distinctive from other input approaches that require users to be in specific in-use postures [6][7]. Furthermore, users can subtly choose the characters within the finger space without noticeable movements [8]. Thumb-to-finger interaction enables off-hand text entry. As such, the user can maintain his forearm in a relaxed position and input text through a series of thumb-to-finger touches. Thumb-to-finger driven text entry is challenging as the limited-sized finger space can barely accommodate the full QWERTY keyboard on two hands [8]. Furthermore, two-handed thumb-to-finger interaction limits mobility as both hands are required to input text. In this paper, we propose a single-handed text entry approach in which users distinguish characters on an ambiguous layout by varying the force applied to the keys. As each ambiguous key consists of 3 characters, our work shrinks the full two-handed QWERTY keyboard into a one-handed ambiguous layout, while force dimension serves as an alternative modality to distinguish the target keys in the ambiguous keypads.

Unlike the sizable amount of keyboard optimization studies on soft keyboards [9] for smartphones [10] and smartwatches [11], we investigate in this paper the optimal key placement in the finger space to facilitate thumb-to-finger interaction for one-handed text entry. Although prior works have explored the feasibility of thumb reaches in the finger space [4], demonstrated the possible use of thumb-to-index finger reach for augmenting touchscreen interaction [12], and developed thumb-to-finger two-handed full QWERTY gloves [8], none of the existing works consider the optimal keyboard layout within the one-handed finger space.

The key goal of this paper is to design a thumb-to-finger interaction system for one-handed text entry on augmented reality head-worn computers. To this end, we have formulated a quadmetric optimization problem for designing optimal 3-character ambiguous key layouts by considering the following factors: the goodness of character pairs (reducing the thumb movement in the finger space, and minimizing the ambiguity to the statistical decoder of word auto-correction), the user familiarity to the new layout, the ease of force interaction, and the level of comfort for thumb-to-finger interaction. In this work, we discuss the effects of the optimization criteria to the design of keyboard layouts. Finally, we validate the quadmetric optimized layout through a user study.

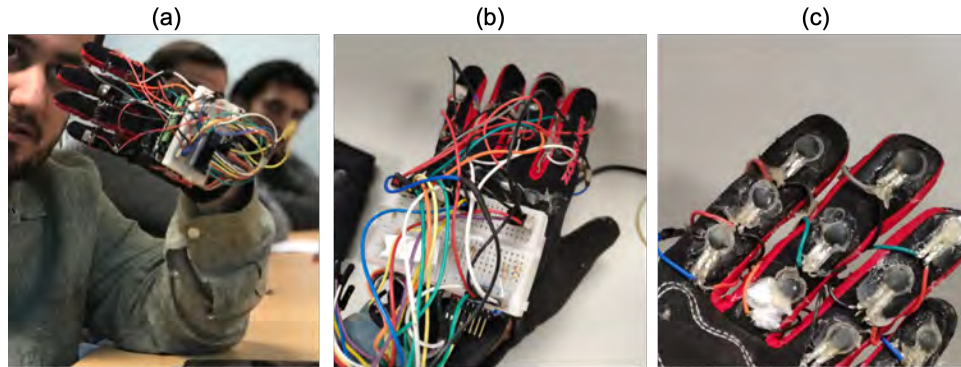


Fig. 1. The prototype of one-handed text entry glove: (a) the glove worn on a user's hand; (b) back-side of the glove with Arduino wired to the keypads located on the rear-side; (c) rear-side of the glove with a close zoom to the force-sensitive keypads weld on index, middle and ring fingers.

Figure 1 gives an overview of the prototypical glove. The glove consists of 9 force-sensitive nodes on the phalanges of the index, middle, and ring fingers in a 3-3-3 configuration and 3 touch nodes on the small finger. Each force-sensitive node contains three characters discriminated through force exertion. The 3 touch nodes on the small finger contain the whitespace, enter and backspace keys. The user can perform off-hand and off-sight text entry through small and unnoticeable finger movements. The novelty of the glove is that the force-assisted interaction is applied to achieve one-handed text entry within the small area of finger space, while the keyboard layout on the glove is optimized to enhance the easiness of force-assisted interaction. The system is designed particular for the Augmented Reality, where the one-handed text entry enables the users to reserve a free hand for holding objects such as briefcase and handrail in various mobile situations. Our contribution is fivefold:

- We design a force-assisted, hand-mounted ambiguous keyboard for the constrained interface on smart wearables such as Augmented Reality (AR) head-worn computers.
- The proposed solution is a hands-off, single-handed, and eyes-free text entry approach, which considers the mobile situation for interacting with the AR environment.
- We solve the optimization problem for character placement within our ambiguous keyboard and propose an intuitive yet efficient keyboard layout for thumb-to-finger interaction.
- We evaluate our solution through a 6-session user experiment with 12 participants. Participants with quadmetric optimized layout achieve an average text entry rate of 6.47 WPM with error rate 5.64% in the final session, which is significantly faster than other one-handed solutions.
- On the premise that the initial evaluation of the 3-level configuration (Section 4) results in an error rate of 16.1%, the quadmetric optimized layout significantly reduces the average error rate to 6.47%.

The rest of this paper is organized as follows. After a quick review of the key related work in Section 2, we describe our system implementation (Section 3). We then validate our initial intuition through a pilot study and calibrate the force levels in Section 4. According to user feedback, we formulate our optimization problem in Section 5 and present our keyboard layout optimization in Section 5.5. We then evaluate its performances in Section 6. We finally discuss our findings in Section 7.

2 RELATED WORK

Our research on Force-assisted Glove is highly related to the existing studies on designing systems and devices for text entry on augmented reality (AR) head-worn computers, finger-space interaction, force-assisted text input, and optimization of the character position in the keyboard layout.

2.1 Text Entry for AR Head-worn Computers

Various systems and devices facilitate text entry on AR head-worn computers. Several swipe-based gestural systems use the touch interface on the spectacle frame of Google Glass [6][7]. In One-Dimensional Handwriting [7], each character is encoded using multiple bidirectional unistrokes that mimic the handwriting of the character. For instance, the character ‘c’ is composed of three swipes – ‘up-down-up’. SwipeZone [6] divides the touch interface on the spectacle frame into three areas. First, the user selects one of the zones, then, in a second step, targets the character within the zone. One-Dimensional Handwriting and SwipeZone achieve respectively 4.67 WPM and 8.73 WPM. However, reaching for the spectacle’s frame can attract attention and be socially awkward [1]. Some other works describe external controllers to enable off-hand text entry. These systems allow the user to operate a cursor and select keys on a virtual on-screen keyboard such as the Dasher input system [13]. A ring wearable [14] enables two-stage character selection on a virtual QWERTY keyboard in which the characters composing the QWERTY layout are grouped into a sequence of 3 consecutive keys. This system achieves 6 – 10 WPM. Finally, keyboards on Smartwatches enable text entry on smartglasses (7.66 - 10.85 WPM) [15]. Although external controllers and ring wearables provide off-hand touch interfaces, text entry on external devices consumes the user’s visual attention on the keyboard layout and occupies the screen real estate in the AR environment [16].

2.2 Thumb-to-finger Text Entry

On-body interaction can leverage human proprioception as an additional feedback mechanism [17]. Recent works have studied thumb-to-finger interaction [4], which enables users to locate the character key without visual attention. PalmType [3] uses a computer-vision approach to detect the movement of the index finger on one hand to type on the palm and fingers of the other hand, and achieves 4.6 WPM. DigiTouch [8] is a split-QWERTY keyboard across two gloves, in which the characters are 1-to-1 mapped to the key positions, and achieves an average of 13.0 WPM. Even though two-handed thumb-to-finger interaction is a good candidate for direct 1-to-1 key mapping, no spare hand is reserved for augmented reality in outdoor scenario (i.e. one hand is reserved for carrying a shopping bag or holding a handrail).

One-handed and thumb-to-finger text entry free up one of the user’s hand, however, 1-to-1 key mapping is more difficult due to the smaller available area. Merging characters into one key is thus unavoidable. Argot [5] is a glove prototype with 15 conductive patches (keys) located at the radical side of fingers, finger pads and fingernails. Each key accommodates four characters or symbols. Multi-tap T9 are Predictive T9 [18] serve to disambiguate the multiple characters on each key. FingerT9 [19] applies the multi-tap T9 on an 11-key glove for text entry on smartwatches (3.43 WPM). However, key disambiguation techniques such as multi-tap T9 and Predictive T9 can suffer from the high number of Key-Stroke-Per-Character (KSPC) and word collisions [20]. These studies raise the need for further exploration of alternative key disambiguation methods. To the best of our knowledge, we are the first work on force-assisted key disambiguation for one-handed thumb-to-finger text entry on AR head-worn computers.

2.3 Force-assisted Text Input on Smart Wearables

Users manage to distinguish between subtle amounts of force up to 6 levels [21]. The error rate rises from 4.9% for one layer to 35% for six layers [22]. If adequate feedback is given, users can reach up to 10 levels [23]. Existing studies on force-assisted text input for mobile devices are very limited and only two prior works apply

force-sensitive interaction on tiny touchscreens. Hsiu et al. [24] design a 2-level ambiguous QWERTY keyboard on smartwatches (12.4 WPM), in which two discrete levels of force applied to the screen discriminate between two characters located within the same key. A force-assisted scanning ambiguous keyboard (SAK) [25] has a thumb-sized button for character selection in a 1-line character layout. It achieves an average of 4.2 WPM for character-level input solely by user judgment and 11 WPM for word-level input supported by probabilistic word auto-completion. In contrast, we investigate the feasibility of designing an ambiguous keyboard with force augmentation within the finger space.

2.4 Keyboard Layout Optimization

The original QWERTY layouts are inherited from mechanical typewriters and designed to accommodate the constraints of physical keyboards. Text entry in mobile environments results in conflicting objectives for smartphones and smartwatches [26]. Designing a new keyboard layout can thus be regarded as a multi-metric optimization problem. Bi et al. [10] investigate a quasi-optimized layout considering the long-term efficiency and ease of visual search for smartphone keyboards, achieving a typing speed between the original QWERTY layout and an unfamiliar, speed-optimized layout. Dunlop et al. [27] study the optimization of semi-ambiguous keyboards on smartphones. They design a QWERTY-like keyboard layout to balance the keypad size with the prediction problems. In another work [9], the proposed smartphone keyboard optimizes the speed, familiarity, and spell-checking ambiguity. Smith et al. [28] evaluate a Pareto frontier between the following three metrics: gesture clarity, gesture speed and the similarity to the QWERTY smartphone keyboard. Similarly, Gong et al. [29] formulate the optimization problem with four metrics including comfort, accuracy, word disambiguation, and learnability. Optimal T9 on a tiny touchscreen [11] considers the speed, avoidance of word collision, and learnability.

Our solution considers four metrics – 1) Character pair arrangement to reduce both the finger movements and the spell corrector ambiguity, 2) User familiarity with the QWERTY layout, 3) Easiness of force touch interaction, and 4) Comfort level in the finger space. Our solution shares the same values of speed, clarity, and familiarity present in most of the prior works. As the text entry is performed on the glove, the fourth metric particularly considers the thumb-to-finger interaction and the third metric integrates the force augmentation into the finger space. To the best of our knowledge, this paper is the first study to leverage the force dimension to disambiguate the characters in an ambiguous layout within the finger space. We therefore introduce the first optimized force-assisted thumb-to-finger keyboard layout for AR Head-worn computers in this paper.

3 SYSTEM IMPLEMENTATION

In this section, we first describe the implementation of our test system and its different components. We then discuss the calibration and behavior of the system. Finally, we discuss the potential modifications to adapt this prototype in real-life situations.

System Description

The glove prototype system is composed of three key components: the tactile surface of the glove, a 16 channels multiplexer, and an arduino microcontroller, as shown in Figure 2. Each finger of the glove contains three force sensors, one for each phalanx. The initial iteration of the prototype shown in Figure 1 consisted of pressure-sensitive resistors glued to a glove. However, this setup presents several issues for experimental purposes. First of all, the glove suffers from a lack of adaptability to various hand sizes. For users with larger or smaller hands, the sensors would end up on the joints instead of the phalanxes, which would trigger erroneous inputs. Furthermore, users with larger hands would experience reduced thumb mobility, while users with smaller hands had trouble pressing the last sensor. Finally, as a prototype, this system may encounter unexpected breakdowns during the usage-intensive experiments, and this static setup does not allow for rapid replacement of broken sensor.

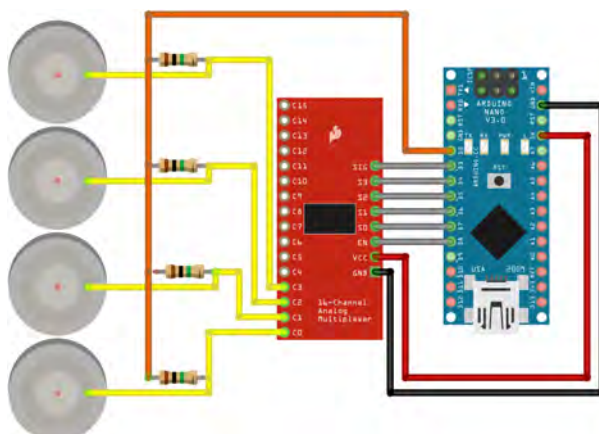


Fig. 2. System schematics. The force sensors are connected to the 16-channels multiplexer that forwards the measurements to the Arduino.

Table 1. Minimum voltage, maximum voltage, and voltage range for each sensor.

Sensor	1	2	3	4	5	6	7	8	9	10	11	12
Min (V)	0.48	0.58	0.58	0.43	0.58	0.48	0.58	0.58	0.24	0.39	0.39	0.58
Max (V)	3.17	4.15	3.41	2.92	3.41	3.66	4.39	2.92	1.70	1.95	3.41	3.66
Range (V)	2.69	3.57	2.83	2.49	2.83	3.18	3.81	2.34	1.46	1.36	3.02	3.08
Normalized Min	0.15	0.13	0.20	0.17	0.20	0.15	0.15	0.20	0.14	0.20	0.09	0.19

We thus adopt the following improved design. These sensors are made using small 5x5 mm squares of pressure-sensitive conductive sheets. This type of plastic was originally designed to protect items from electric discharges. Thanks to its property of changing resistance with pressure, it can easily be used as a cheap alternative to pressure sensors for prototype systems. In order to keep constant pressure over the plastic, the wires are connected using gaffer tape. Other alternatives such as conductive glue lead to unpredictable changes in the resistance of the material, rendering the measurements unreliable. The sensors are fixed to the hand using Velcro fasteners. We use one Velcro fastener per phalanx, for a total of 12 fasteners by hand. Although this solution makes the initial setup more cumbersome, it allows customizing the system to each user during the experiment phase.

The sensors are connected to the 16-channel analog multiplexer that aggregates the measured pressures and sends them to the Arduino. The multiplexer converts the analog signals received to digital signals and serializes them to the Arduino. As such, only four pins on the Arduino are required instead of 16. The Arduino microcontroller contains the whole logic to convert numeric measurements of force to key presses on the computer. To this purpose, we use an Arduino Uno R3 that provides the standard USB Human Interface Device (HID) profiles for mouse and keyboard.

Calibration and Behaviour

The conductive plastic we use for the pressure sensor is extremely sensitive to external parameters. The wire connection and the attachment to the glove strongly impact the values measured. As such, we have to calibrate each sensor individually. To this purpose, we empirically measure the minimum (no pressure, accounting for the

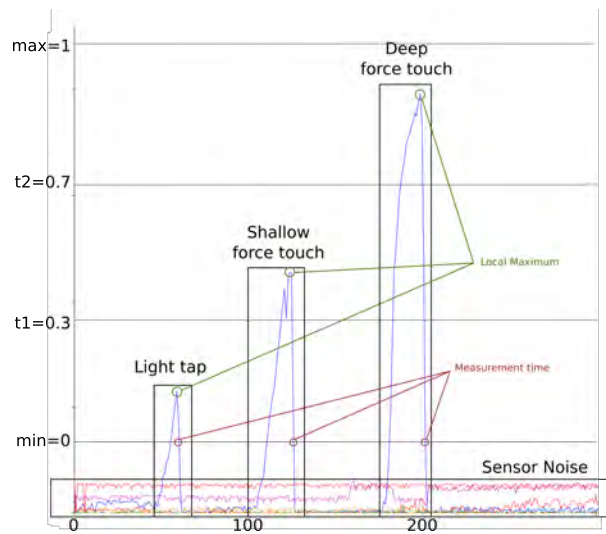


Fig. 3. Normalized signal analysis for sensor 1. Succession of light tap, shallow force touch and deep force touch

variance of the sensor) and maximum (maximum force measurable) voltage output for each sensor. The minimum and maximum values are represented in Table 1. We observe a wide disparity between the sensors. Although most of the minimum voltages are around 0.5 V, the maximum voltage ranges from 1.70 V to 4.39 V. As such, the effective voltage range is between 1.36 V and 3.81 V. The 16-channel multiplexer converts this voltage to a digital signal with values ranging between 0 and 1024. As such, even the lowest range of 1.36 v corresponds to 300 discrete values, which is acceptable precision for our usage.

We normalize the aforementioned values between 0 and 1, 0 corresponding to the minimum value for sensor activation and 1 being the maximum measured value. We then divide this range into three sections (0.000 – 0.299, 0.300 – 0.699, 0.699 – 1.000), corresponding respectively to ‘light tap’, ‘shallow force touch’ and ‘deep force touch’. When a sensor’s value exceeds the minimum threshold, we wait until that value drops back down below the minimum threshold and record the maximum value obtained in that interval. In case several sensors are activated during the movement, we only consider the largest normalized value. We summarize this process in Figure 3. This process allows users to perform force-assisted clicks on the ambiguous keys. Figure 4 shows the force disambiguation, where three force levels can be detected for character selection in the ambiguous key for the characters T, U, and F.

Limitations and Potential Improvement Areas

As a prototype system (Figure 4), this implementation is bulky and cumbersome to use in real-world applications. However, this design is dictated by the conditions of the test environment. The initial prototype presented in Figure 1 is closer to a real-life application, although still at a very early stage. Through the following modifications, we can integrate the full system within an easy and plug-and-play solution:

- (1) Use piezoelectric sensors or force sensitive resistors instead of conductive plastic. These sensors present a much more reliable response while allowing to solder wires without significantly impacting the measurements.
- (2) Integrate the sensors within a glove adapted to the user’s hand size.
- (3) Use conductive fabric, ink or glue to reduce the number of cables (points of failure) on the glove.

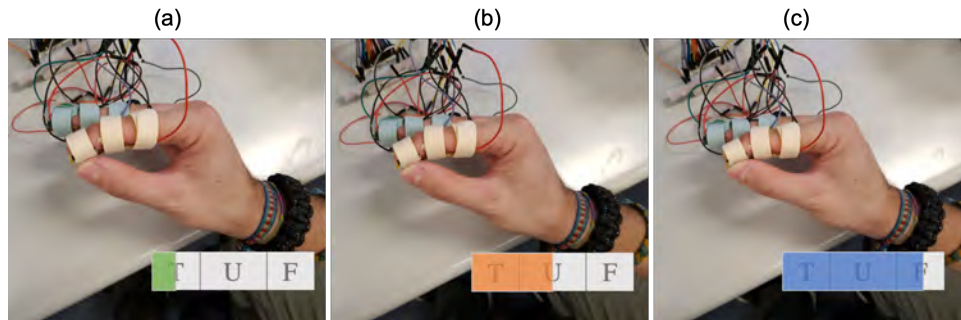


Fig. 4. Three force levels inside the ambiguous located in the first phalanx of index finger: (a) light tap; (b) Shallow force touch; and (c) Deep force touch.

- (4) Integrate the multiplexer and the microcontroller within the same circuit board. This would allow to keep the active elements board to a size lower than 2x2 cm, and embed it on the back of the user's hand.
- (5) Add a battery and a Bluetooth module for the system to be used in complete autonomy.

These actions would result in a compact wireless glove, weighing less than a few hundred grams.

4 FORCE-ASSISTED GLOVE DESIGN

After describing the implementation of the glove infrastructure, we now focus on the keypad design. The 2-level configuration is neglected in the experiment due to the following limitations. The 2-level configuration leads to a 5-5-4 keypad arrangement on the index, middle and ring fingers. On each finger, 2 (ring finger) or 3 keypads (index and middle fingers) are located on the joints. As our glove design is sensitive to the force exertion, the joint movements between the phalanxes will trigger erroneous character input. For instance, a user intending to reach the keypad on the first phalanx of the middle finger will activate the joints of his middle finger to reach the keypad, triggering unintended keystrokes. Furthermore, previous studies [21] show that configurations with more than 4 levels lead to unacceptable error rates. The evaluation is necessary because the prior studies consider the tangible and external interfaces, while our studies perform on the finger space with the prominent features of proprioception. Therefore, we only consider the 3-level (9 ambiguous keypads) and 4-level (7 ambiguous keypads) configurations on the index, middle and ring fingers as the feasible candidates. The remainder of this section describes the pilot test to understand the user performance in 3-level and 4-level force-assisted thumb-to-finger interaction.

4.1 Participants and Apparatus

We recruited 20 participants (Age 18 – 30) on our local campus. None of them had prior experience in force-assisted interaction on the thumb-to-finger space. However, 6 out of 20 participants had tried 3D touch on iOS or emulated Force-touch on the Android Q. We implemented two layouts (3-level and 4-level configurations as shown in Figure 5) on the prototypical glove described Section 3. The two layouts include 26 characters in alphabetical order. The characters are first assigned to the most comfortable key positions (Figure 13b) on the index, middle and ring fingers, and the 3 discrete keys on the small finger are disabled. The ambiguous keys are distinguished by the exerted force level, represented by a normalized value between 0 and 1. Either 3 or 4 characters are included in each ambiguous key. Figure 15 shows the two layouts on the prototypical glove. As there are either 27 or 28 character slots in the 3-level and 4-level configurations, the symbol ‘-’ indicates the empty slot. Our trial tests suggest that the middle range(s) between two ends should be large enough to enable

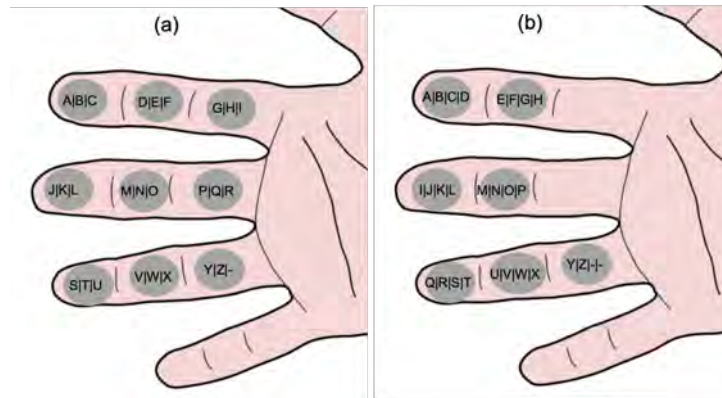


Fig. 5. Testing layouts for pilot tests, where the symbol ‘-’ represents no character (empty key) in the ambiguous keys: (a) 3-level configurations; (b) 4-level configurations.



Fig. 6. The testing interfaces of 3-level configuration with illustration.

users to distinguish the force easily. We design the normalized range as follows: 3-level configuration – (1st level: 0.000 – 0.299; 2nd level: 0.300 – 0.699; and 3rd level: 0.700 – 1.00); and 4-level configuration – (1st level: 0.000 – 0.199; 2nd level: 0.200 – 0.499; 3rd level: 0.500 – 0.799; and 4th level: 0.800 – 1.00).

A computer monitor is used to display a character (A – Z) in the testing interface. The users are required to type the requested character. The usage of the monitor was requested by the participants who reflected difficulties to see the content on smartglasses. Figure 6 shows the testing interfaces for the 3-level configuration. The characters are randomly generated and a visual aid showing the force level assists the user. By using Quick Release [30], users validate the character by releasing their thumb from the keypad on the finger space. The system will then compute the force level based on the 15th frame capture, equivalent to 60 ms [31]. After release, a new random character is displayed for the users to type. No feedback for the typed character is shown in the interface to avoid bias. It also ensures that the participants only focus on the next character entry.

4.2 Design and Procedure

A prior work [8] uses audio instructions to request the participants to touch on large surfaces on either the left-hand or the right-hand glove with the target pressure level (level 1/2/3). In comparison, our pilot test requires

the participants to exert force touch on a specific position of the finger space (proximal, middle and distal phalanx of the index, middle and ring finger), guided by the requested character at a certain force level. In addition, our study investigates the feasibility of both 3-level and 4-level configurations.

Prior to the experiment, the users had 10 minutes to familiarize with the interface and the glove typing environment. The users were sitting in a sedentary environment during the whole experiment and typed on the prototypical glove worn on their hand. We instructed them to put their hand under the table to avoid looking on the glove and type as fast and accurately as possible. We collected 50 force-assisted clicks with each layout per participant to avoid long-term training leading to a bias in terms of force selection accuracy. Overall, this experiment consists of 2,000 character inputs or force-assisted clicks (50 force-assisted clicks x 20 participants x 2 interfaces).

4.3 Results and Implications

With a total of 1000 force inputs for each configuration, users identify the correct characters through force inputs for the 3-level and 4-level configuration with a respective accuracy of 83.9% and 76.6%. Figures 7 and 8 show the distribution of the force levels. The dotted lines specify the division of the force spectrum. If the force is applied in the incorrect division, an error is logged. For example, green bar charts with values of 0.4 (0.300 – 0.399) and 0.5 (0.400 – 0.499) are counted as force-assisted clicks (Figure 7).

In both the 3-level and 4-level configurations, we find three common phenomenons among the participants. First, the participants intended to exert the maximum force to the third level for the 3-level configuration and the fourth level for the 4-level configuration, where the highest number of force-assisted clicks (146 counts for 3-level and 139 counts for 4-level) were recorded. Second, we observe that the occurrences of the middle level(s) between the two ends of the force spans are evenly distributed. This can be explained by that the participants constantly relying on the visual feedbacks on the screen and performing the quick release once the force level is appropriately displayed. Third, participants had trouble in distinguishing two levels between the two ends of the force span. In the 3-level configuration, 57.14% of the errors are triggered by trying to distinguish the second level (92 out of 161 errors). In the 4-level configuration, the situation deteriorates, and we record an error sum of 90.17% for the second level and the third level (211 out of 234 errors). In addition, the total number of errors increases by 45.34% between 3-level to 4-level configurations.

We conducted a short questionnaire after the experiment to collect the user qualitative feedback. We asked the following questions: "What do you think about the design of force spectrum (middle levels / levels of the two ends)?" and "What is your preference between the 3-level and 4-level configurations?". The majority of participants (14 out of 20) feel confident with applying force-assisted click at the two ends of the force span. They show a higher preference than for the middle levels of 3-level and 4-level configurations. The participants preferred the force level at the lower end (light tap) due to its similarity to the tap gesture on the smartphones. In addition, most of them (18 out of 20) reflected that the 3-level configuration is more user-friendly and usable than the 4-level configuration as it induces a lower temporal load (faster decision) for the force control.

Text entry is regarded as a repetitive task and hence the error rate is one of the key considerations. Considering that more levels will introduce a significantly higher error rate, the 3-level configuration is preferable to the 4-level configuration. Although the 3-level configuration is subject to a 16.1% error rate, the participants in the preliminary survey also indicate a higher preference for the 3-level configuration than for the 4-level configuration. In the next section, we address this error rate through a better layout design formulated as an optimization problem.

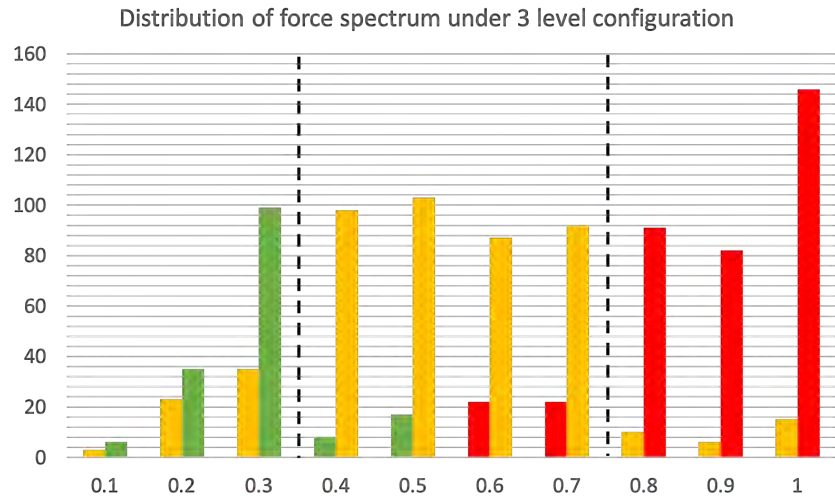


Fig. 7. The distribution of force input in 3-level configuration (Green Colour - first level; Yellow Colour - second level; Red Colour - third level).

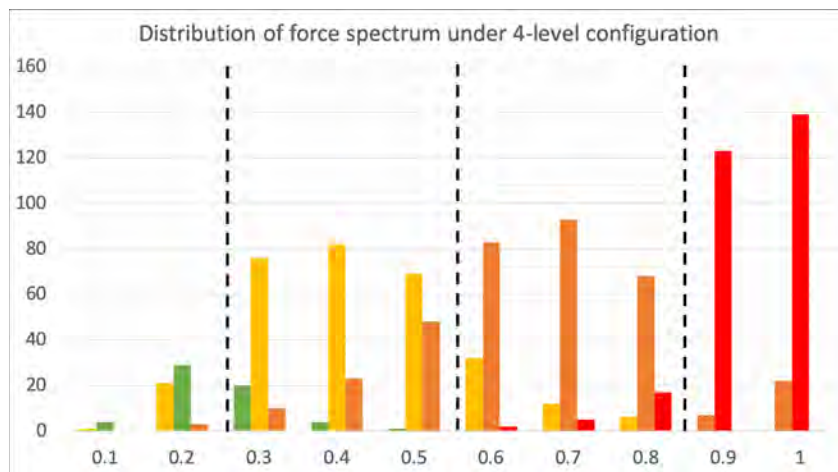


Fig. 8. The distribution of force input in 4-level configuration (Green Colour - first level; Yellow Colour - second level; Orange Colour - third level; Red Colour - forth level).

5 OPTIMIZATION OF THE KEYBOARD LAYOUT

Our pilot test studies the user performance on 3-level and 4-level force-assisted thumb-to-finger interaction. We selected the 3-level configuration as an optimal solution due to its significantly higher accuracy. A prior work [4] indicates that users can accurately distinguish multiple buttons on their fingers that can include the 26 characters, space, enter, and delete key in the finger space of one hand, under an off-sight situation. We thus define a keyboard of 9 force-touch buttons and 3 touch buttons, where the 26 alphabet characters are assigned to

the 3 rows of 3 force-touch keys (3-3-3) located on the index, middle and ring finger. The space, enter, and delete keys are assigned to the 3 touch buttons on the small finger. Our optimization problem assigns the 26 characters into the ambiguous force-touch keys in the 3-3-3 configuration.

We consider the following optimization metrics: (1) the goodness of character pair and (2) the familiarity with the QWERTY layout facilitates the text entry performance, while (3) the easiness of force keypad interaction and (4) the comfort level of the finger space consider the human ability to input text. A genetic algorithm [32] is applied to solve the optimization problem. Finally, we present the optimized 3-3-3 thumb-to-finger layout, striking a balance between the text entry performance and the human ability to input text.

5.1 Maximizing the Goodness of Character Pair

The goodness of a character pair in a candidate solution keyboard $k \in K$ is subject to two key factors: (i) the finger movement distance and (ii) the ambiguity for auto-correction.

Regarding the finger movement distance, Fitts' law describes two temporal factors for the time to input a character on keyboard – the time for finger movement from one key to another and the time for targeting the character key. The nearer and bigger a key is, the quicker the character input. In our glove configuration, we design all keys to be of identical size and hence Fitts' law can be simplified by considering only the finger movement between the keys. In addition, the thumb has a lower degree of movement [33] than the index finger and hence the optimized layout can reduce the amplitude of thumb movements.

We compute the frequencies $a_{i,j}$ of key pairs in English by building the Bigram of character pairs in the selected English corpus [34]. Let $\alpha = \{a, b, c, \dots, x, y, z, _\}$ be the alphabet we consider, $Bigram_{i,j}$ refers to the probability of the finger moving from character i to character j . We build a character pairing table of size 676 (26×26), $Bigram_{i,j}$, by normalizing the frequencies as $Bigram_{i,j} = a_{i,j} / \sum_{i,j \in \alpha} a_{i,j}$. Zhai et al. [35] indicate that the source of the corpus does not significantly influence the optimization of the keyboard layout.

Figure 9 shows the 26×26 character pairs as a Bigram. Our analysis shows that the most common character pair is ER , with 90,510,491,354 occurrences in our corpus. The probability of finger movement between E to R is 0.0391. The next most frequent character pairs is ER with $P_{IN} = 0.0275$, ES with $P_{ES} = 0.0250$, TI with $P_{TI} = 0.0238$, TH with $P_{TH} = 0.0225$, and OR with $P_{OR} = 0.0224$, while the least probable character pairing was QJ with a 0.0000005 probability. Next, we normalize the probability to a Bigram score, $Bi_{i,j}$ defined as follows: $Bi_{i,j} = \frac{100 * Bigram_{i,j}}{\max(Bigram_{i,j})}$. For instance, the most common character pairs ER and IN will respectively get 100.00 and 70.49 points.

Performing eye-free text entry is error-prone [36]. Automatic correction algorithms alleviate the issue but they have limitations on checking valid words. For example, the words 'hit' and 'hat' are both valid and the ambiguity issue rises if the characters 'A' and 'I' are arranged together [27]. To address this issue, we build the table of *bad-bigrams* or *badgrams* [27] showing the 676 (26×26) possible character pairs. As the white space character is used to separate words, we exclude it from the Badgram. We scan all words of identical length in the corpus and count the Badgram frequency of the character pair, $b_{i,j}$, by checking whether a character substitution in a word leads to another valid word. We then normalize the frequencies as probabilities $Badgram_{i,j} = b_{i,j} / \sum_{i,j \in \alpha} b_{i,j}$. Figure 10 shows the overview of the Badgrams in English Corpus. Our analysis gives the following top Badgrams: AE with $P_{AE} = 0.0175$, AO with $P_{AO} = 0.0146$, AI with $P_{AI} = 0.0125$, EO with $P_{EO} = 0.0121$, EI with $P_{EI} = 0.0118$. Having A and E as a neighboring pair leads to many ambiguous words such as *and* instead of *end*, *ha* instead of *he*, *bat* instead of *bet*, etc. Normalizing the probability of Badgram results in the Badgram score, $Bad_{i,j} = 100 * Badgram_{i,j} / \max(Bigram_{i,j})$.

Finally, we obtain the satisfaction scores of the character pairs, $CP_{i,j} = Bi_{i,j} - Bad_{i,j}$ [9]. For combinations including the white space, $CP_{i,j} = Bi_{i,j}$. We aim to find out the most common character pairs which are not likely to trigger ambiguity with the auto-correction software. Figure 11 shows the satisfaction scores of the character

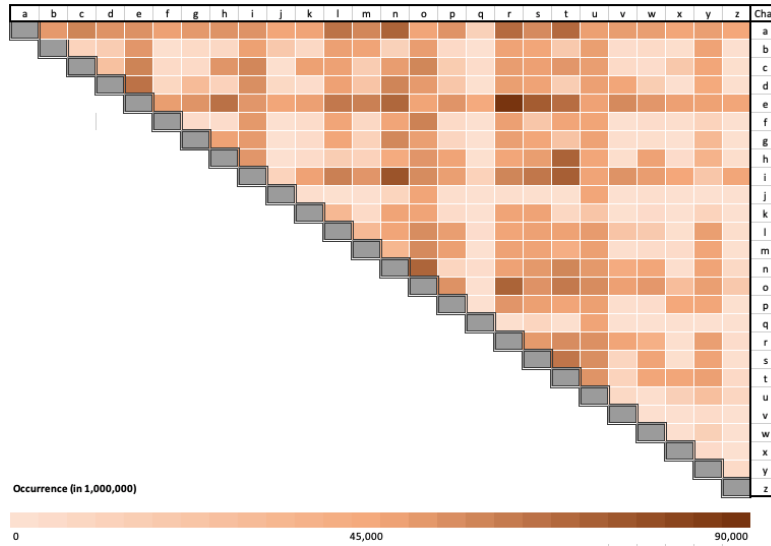


Fig. 9. Bigrams Matrix of 26 x 26 character pairs.

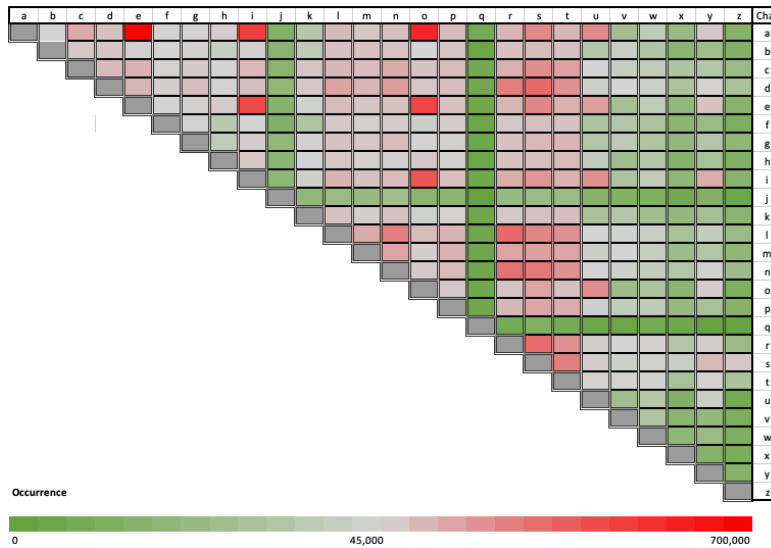


Fig. 10. Badgrams Matrix of 26 x 26 character pairs.

pairs, according to the computed Bigrams and Badgrams. The 10 most preferred character pairs ER, IN, NO, OR, AN, TH, TI, EN, and EH are ranked in the top 10 (descending preference), while the 10 least preferred character pair are EA, OA, EO, AI, EI, LR, DS, NR, LN, and RS (ascending preference).

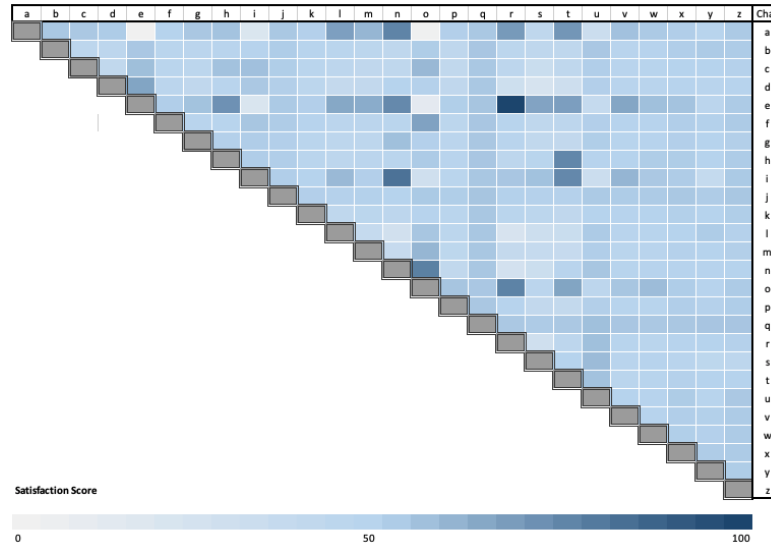


Fig. 11. Satisfaction Score Matrix of 26 x 26 character pairs.

The goodness of the character pair configuration, M_{pair} , can be computed by summing the satisfaction score if the characters are neighbors. Below is the first objective function that we derive.

$$O_1 : M_{pair} = \sum_{\forall i, j \in \alpha} \begin{cases} CP_{i,j} & \text{if character } i \text{ \& } j \text{ are adjacent keys} \\ 0 & \text{otherwise} \end{cases}$$

Previous optimized layouts for virtual and physical keyboards define adjacent keys by their 2D geometric proximity. The glove prototype system introduces a third dimension driven by the force level division. We regard two characters within a key as adjacent to each other if the associated force levels are continuous. In the alphabetical layout, the character 'a' is adjacent to 'b' but not 'c' and character 'b' is adjacent to both 'a' and 'c'. Regarding the geometric proximity, we consider three separate planes corresponding to the three force levels. For instance, 'a' is geometrically adjacent to 'd', 'j', and 'm', while 'h' is geometrically adjacent to 'e', 'n', and 'q'. We normalize the M_{pair} into a normalized score, M_{pair_n} , ranging from 0 to 1 by $M_{pair_n} = M_{pair} / 1.1 * M_{pair_best}$, M_{pair_best} being the highest scoring keyboard and 1.1 an offset index to facilitate the optimal search.

5.2 Maximizing the Familiarity with the QWERTY Layout

Most computer and smartphone users are accustomed to the QWERTY keyboard layout. Previous studies show that users learn and adopt QWERTY-like layouts faster than alphabetical layouts, as QWERTY-like layouts reduce the user's efforts on the visual search for characters [37]. Among physical and virtual keyboards, the trapezoid-shaped QWERTY configuration consists of 10, 9 and 8 characters on the first, second and third row. As we create a QWERTY-like keyboard on our 3-3-3 configuration, we make minor modifications to fit all the characters as shown in Figure 13a.

We compute the familiarity between the candidate keyboard and the 'starter layout' (Figure 13a) by rating the summed Euclidean distance of all the characters $i \in \alpha$ between the geometrical center of a given key in the candidate solution k_i to the center of the corresponding character position in the 'starter layout' s_i , as the

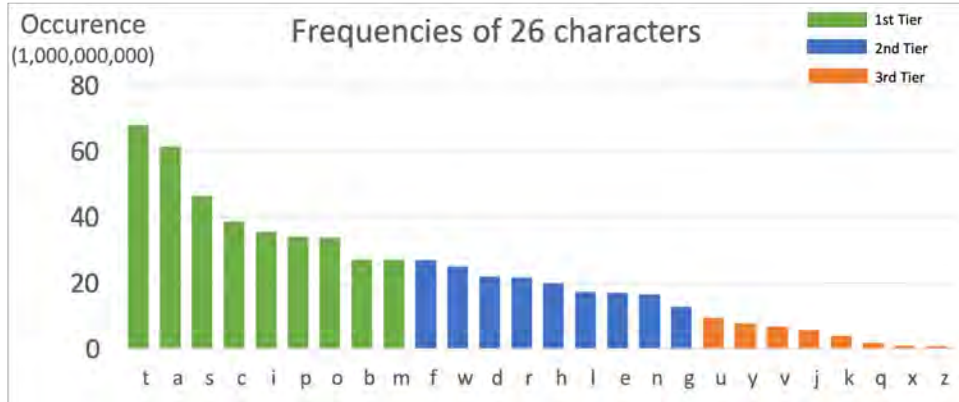


Fig. 12. Frequencies of characters in the English corpus (Green Colour - first tier; Blue Colour - second tier; Orange Colour - third tier;).

reference of home positions. The character positioned far away from their home position in the candidate solution will be punished. We evaluate the summed distance of a candidate solution, k_{dist} in the candidate solution set K as:

$$k_{dist} = \sum_{\forall i \in \alpha} Distance(k_i, s_i)^2$$

We obtain the normalized familiarity score as follows.

$$O_2 : M_{familiarity} = 1 - \frac{k_{dist}}{\max_{\forall k \in K}(k_{dist})}$$

The lesser the distance generated from the character alternation, the higher the familiarity in this metric.

5.3 Maximizing the Easiness of Force Keypad Interaction

In our pilot study, the participants displayed individual preferences in the choice of the three characters within a key. Taking into account the user feedback, the light tap is easier than the deep force touch in the 3-level ambiguous keyboard. We assign the most frequent character tier to the light tap force level. The second most frequent character tier occupies the deep force touch force level. We rank the characters in descending order and pair the two tiers with two force levels, based on the 26 character frequency analysis: $Tier_1$ [t, a, s, c, i, p, o, b, m] – light tap, $Tier_2$ [f, w, d, r, h, l, e, n, g] – deep force touch, $Tier_3$ [u, y, v, j, k, q, x, z] – shallow force touch – force touch. Considering our corpus, the full ranking of 26 characters, CF_i for all $i, j \in \alpha$ is listed in Figure 12. The characters of the $Tier_1$ account for 79.56% of the occurrence and light-tap can fulfill most of the character inputs. Even though force-assisted interaction is error-prone [21][23], the optimization criterion can assign the less frequently-used character in the second tier to minimize triggering erroneous inputs.

The intuition behind force interaction is analogous to the prior work of LetterEase [38] that modifies the multi-press T9 keyboard by arranging the most frequently used characters in the first position among the three characters. We estimate that the lower force needed for frequent characters will increase the input speed accordingly.

We compute the ease of use of our force keyboard as the summed weighted score of character-tier matches. We define a reward scheme to give a score to those configurations if the characters in the candidate solution are assigned to the matched tiers. The weighted score is summed by the CF_i and no normalization is needed. As we

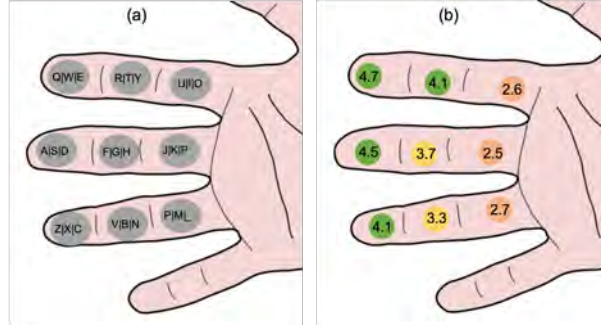


Fig. 13. Reference layouts: (a) A ‘starter’ QWERTY layout on the finger space, where the symbol ‘_’ represents a dummy character in the combination of optimization as only 26 English characters exist among 27 possible choices; (b) Rating of comfort for thumb-to-finger interaction.

consider the most frequently used character will be triggered in the user preferred position that is the two ends of the force spectrum, the user performance in terms of error rate will improve. To maintain a lower error rate from force-assisted interaction, we treat this optimization metric as a constraint so the normalized value of the metric is predefined as 1.00.

$$O_3 : M_{force} = \sum_{\forall i \in \alpha} \begin{cases} CF_i & \text{if character } i \text{ in the matched tier} \\ 0 & \text{otherwise} \end{cases}$$

5.4 Maximizing the Comfort Level of the Finger Space

Even though the thumb has the unique characteristics of reaching all the finger phalanxes in the finger space, users show different perception to the phalanxes of the fingers from the thumb reach. We use the results of the prior work [4] showing the comfort level of thumb-to-finger interaction. We apply their score directly in our evaluation. Figure 13b shows the computed value of thumb-to-finger interaction.

We compute the comfort level of the finger space by rating the summed product of all the characters $i \in \alpha$ and the button positions $z \in Z$ in the 3-3-3 configuration. And the comfort level is denoted by $Comf_z$. The frequently used characters positioned in a comfortable position will be rewarded in the candidate solution k_i . The evaluation of the candidate solution k_i in the candidate solution set K is as follows.

$$O_4 : M_{Comfort} = \sum_{\forall z \in Z} \sum_{\forall i \in \alpha} \begin{cases} CF_i * Comf_z & \text{if character } i \text{ in key } z \\ 0 & \text{otherwise} \end{cases}$$

5.5 Optimized Keyboard Layouts

Table 2 shows the optimization results under the quadmetric consideration. In this layout, the symbol ‘_’ represents a dummy character in the layout as only 26 English characters exist among 27 possible choices. Under the 3-3-3 configuration, we intend to choose a layout that balances all the metrics and maximizes the overall score as shown in the last column. The keyboards on the 1st, 3rd and 4th row prioritize a single criterion such as the goodness of character pair (Figure 14a), the familiarity with the QWERTY layout (Figure 14b), and the level of comfort in the finger space (Figure 14c) as top-priority in the respective layouts.

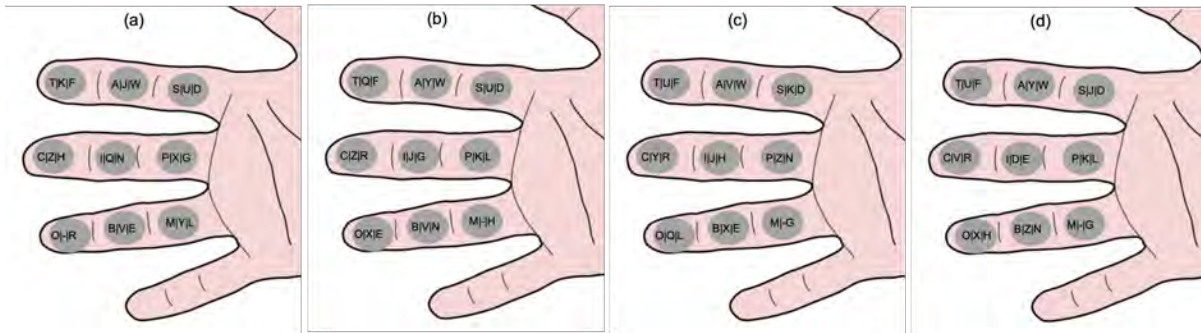


Fig. 14. Resultant layouts from the optimization problem: (a) The layout considering the goodness of character pair as top priority; (b) The layout considering the similarity to QWERTY keyboard as top priority; (c) The layout considering thumb-to-finger interaction as top priority; (d) The layout considering the four metrics in holistic view.

We observe that the over-emphasis on a specific criterion significantly deteriorate the performance of other criteria. For example, the keyboards on the 3rd and 4th rows show contradicting results when they go into one end of the spectrum. Satisfying the familiarity with the QWERTY layout will significantly impact the comfort level of the finger space and vice versa. However, these sub-optimal keyboard layouts shed light on the viability of choosing a balanced point between multiple criteria. We incrementally make trade-offs between four metrics. The quadmetric optimized keyboard (Figure 14d) in the fifth row strikes a balance between all the metrics and scores the highest among all the candidate keyboards. This keyboard layout can maintain both the goodness of character pair and easiness of force keypad interaction. The optimized quadmetric layout reduces the goodness of character pair by 17.00% compared to the 1st-row keyboard. And in return, the normalized score for the familiarity with the QWERTY layout improves by 121.05% from 0.38 to 0.84, while the normalized score for the comfort level of the finger space enhances by 73.58% from 0.53 to 0.92. As a result, the overall score of the quadmetric layout outperforms the 1st-row keyboard by 23.37% through enabling a satisfactory trade-off between the three criteria (O_1 , O_2 , O_4). Similarly, it scores 10.46% and 8.46% higher than the optimized layouts on the 3rd and 4th row respectively. Interestingly, the alphabetically constrained layout (Figure 15b) and the QWERTY starter layout (Figure 13a) obtain lower scores than the above layouts due to their lack of easiness for force keypad interaction.

All the optimized layouts score 1.00 regarding the easiness of force interaction that is highly complementary to the other criteria. As each key consists of three characters and force is applied to disambiguate the keys, the 9 frequently used characters on the first level can be easily selected by light tap and the remaining characters on the next level are triggered by force touch. The full score facilitates the force disambiguation in the finger space.

In summary, although the existing literature [4] proposes that the performance of thumb-to-finger interaction is subject to the level of comfort in the finger space, broader considerations on the goodness of character pair (O_1) as well as the familiarity with the QWERTY layout (O_2) should be applied to the text entry problem. The keyboard on the 3rd row shows a counterexample of over-emphasis on the level of comfort in the finger space (O_4), resulting in the lowest score between all the optimized layouts. Through balancing all the criteria, the quadmetric optimized layout maintains a reasonable satisfaction level to the character positions and user familiarity when text entry in the finger space relies on the force disambiguation.

6 USER EVALUATION

After computing the quadmetric optimized layout, we investigate how the users perform with the new keyboard design in this section.

Table 2. Optimization results under Quadmetric consideration. The four metrics are defined as O_1 , O_2 , O_3 , O_4 where the higher the value of the normalized score, the better the objective function is achieved.

Character layout	O_1	O_2	O_3	O_4	$O_1+O_2+O_3+O_4$
Character Pair OPT Layout (Figure 14a)	1.00	0.38	1.00	0.53	2.91
QWERTY Starter Layout (Figure 13a)	0.61	1.00	0.15	0.64	2.40
QWERTY OPT Layout (Figure 14b)	0.73	1.00	1.00	0.52	3.25
Finger Space OPT Layout (Figure 14c)	0.83	0.48	1.00	1.00	3.31
Quadmetric OPT (Figure 14d)	0.83	0.84	1.00	0.92	3.59
Alphabetical constrained Layout (Figure 15b)	0.91	0.16	0.31	0.98	2.36

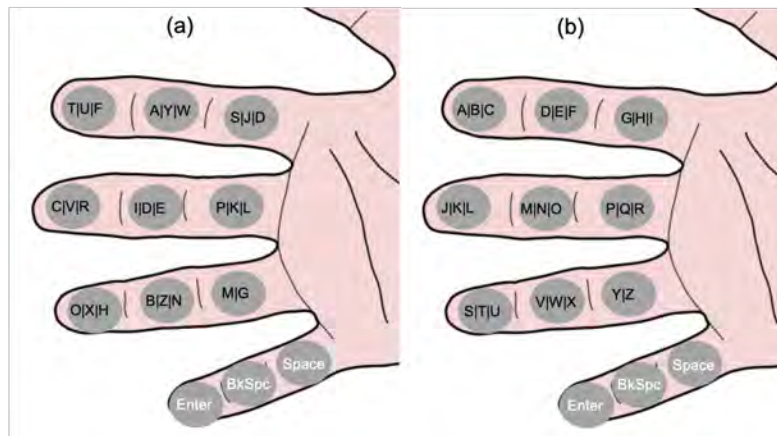


Fig. 15. The two testing layouts, where the ambiguous keys (English alphabets) are colored in black and the discrete keys (Enter, Backspace (BkSpC) and White Space (Space)) are colored in white: (a) Quadmetric optimized layout; (b) Alphabetical constrained layout.

6.1 Participants and Apparatus

We recruited 12 participants from the pilot study (Age 18 – 30). Indeed, through this within-subject experiment, we aim at investigating the pickup and usage rate of the quadmetric optimized layout inside the finger space compared to the alphabetical layout. We implemented the layouts on the prototypical glove described Section 3. The ambiguous keys are distinguished by the force detection in a normalized value from 0 to 1. Three characters are included in each key and the force spectrum is divided based on the 3-level configuration shown in Figure 7. The remaining three keys contain the space, enter and backspace characters. Figure 15 shows the two layouts on the prototypical glove. The normalized ranges for the 3-level configuration are as follows: (1st level: 0.000 – 0.299; 2nd level: 0.300 – 0.699; and 3rd level: 0.700 – 1.00). The alphabetical layout has been chosen as a baseline for the quadmetric optimized layout, with the below reasons. First, the participants know the sequence of English alphabets instinctively, leading to performance improvement [39] and better usability [40] for novices. Second, this baseline layout is similar to the T9 layout [41], with some reference performance from prior studies, even though those studies are conducted on larger oval-shaped physical keys with multi-tap disambiguation. Novice users with multi-tap T9 can achieve between 5.33 WPM [42] and 6.64 WPM [43].

A computer monitor is used to display word phrases on the testing interface. The users are required to type the requested phrases character by character. As in the pilot study, a computer monitor was requested by the

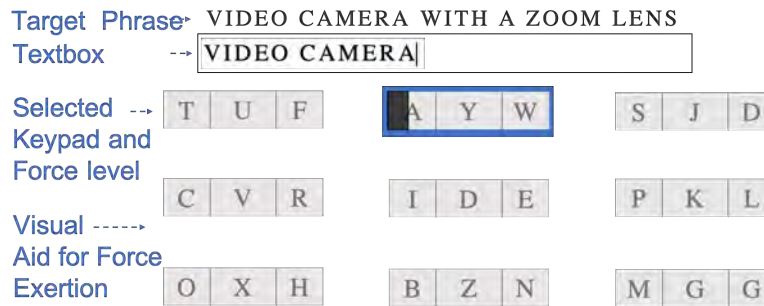


Fig. 16. The text entry interfaces (the quadmetric optimized layout as example) in which target word phrases, text boxes, and visual aids are displayed.

participants as they had difficulties in distinguishing the instructions on smartglasses. As shown in Figure 16, the testing interface also shows the target word phrases within a text box and displays the current force level in the 3-level ambiguous keys. A visual aid showing the force level exerted by the user helps the participants in determining the force level in the continuous spectrum. Users perform Quick Release [30] to end the character selection.

6.2 Design and Procedure

Before the experiments, we briefed the users about the two keyboard layouts (Alphabetical layout and quadmetric optimized layout). After the briefing, the participants had ten minutes to familiarize with each keyboard layout. The participants are instructed at the beginning of each session to type as fast and accurately as possible and can correct typing mistakes only for the current word phrase. For the two layouts, we ran a total of six sessions, three sessions per day for two days. During each session, the participants typed 15 word phrases [20]. The material is written in a sophisticated yet recognizable style so the experiments are reasonable mock-ups of everyday typing tasks. We measure and compare the participants' typing speeds (WPM) and error rates (KSPC).

The total experimental setting of the two separate layouts corresponds to 5,400 characters (5 average characters in a word x 15 word phrases x 6 sessions x 12 participants). Each session lasts no longer than 5 minute. After the first session on the first day and final sessions on the second day, we asked the participants to complete the NASA TLX questionnaire [44] about the two layouts. After the experiment, snacks and soft drinks were served as remuneration.

6.3 Results and Implications

Text entry rate. Figure 17 shows the character-level text entry rate with the quadmetric optimized layout, where the error bars represent the standard deviation. Two-way RM-ANOVA demonstrate a significant effect of the layout ($F_{1,132} = 215.59$, $p < .001$) and the Session ($F_{5,132} = 238.89$, $p < .001$), which indicates a learning effect on the new layout. Throughout the sixth sessions, participants achieved mean values of 5.12 WPM (std. = 1.09) with the quadmetric optimized layout and 4.30 WPM (std. = 0.95) with alphabetical layout. The text entry performance of the quadmetric optimized layout increases from 3.53 WPM (std. = 0.17) in the first session to 6.47 WPM (std. = 0.53) in the sixth session, showing an 83.36% speed improvement. In contrast, the participants achieved 3.01 WPM (std. = 0.16) with the alphabetical layout in the first session and 5.55 WPM during the sixth session (std. = 0.33). Therefore, our results show a performance gap between the quadmetric optimized layout and alphabetical layout throughout the 6-session text entry tasks, where quadmetric optimized layout is 17.07% and 16.64% faster than



Fig. 17. Mean text entry rate over 6 sessions (S1 – S6).

the alphabetical layout in the first and sixth sessions respectively. The quadmetric optimized layout outperforms the alphabetical layout that neglects the easiness of force keypad interaction. In the alphabetical layout, two 1st-tier (T and B) and three 2nd-tier (E, N, and W) characters are assigned to the least preferred middle level, named ‘shallow force touch’, and the participants require more time and cognitive efforts to estimate the force exertion applied to the keypads of the glove. Among the three types of touch interaction (‘light tap’, ‘shallow force touch’ and ‘deep force touch’), the average holding times are recorded for the correctly typed characters in the sixth session, showing the duration between the time the users hold their thumbs down to control force level and the release to accomplish the character input. One-way ANOVA demonstrate a significant effect of the force click types ($F_{2,33} = 731.21, p < .001$). The holding time shows the participants’ confidence to the touch types, in the following descending order: The participants are most confident to the ‘light tap’ with a mean response time of 332 ms (std. = 59 ms). The ‘deep force touch’ has slightly longer mean response time of 552 ms (std. = 79 ms), while the participants are least confident to the ‘shallow force touch’ with a mean response time of 857 ms (std. = 117 ms). Figure 12 indicates that ‘light tap’, ‘shallow force touch’ and ‘deep force touch’ respectively correspond to 63.09%, 6.54%, and 30.37% of the character occurrences. Based on the occurrence of characters, the expected mean time is 433 ms among three types of clicks. Considering the quadmetric optimized layout achieved 5.12 WPM, click interaction occupies only 18.48% of the testing time, and the rest of the testing time are used for the users’ cognitive loading and thumb movements.

Error rate. Figure 19 shows the error rate of the quadmetric optimized layout, where the error bars represent the standard deviation. Two-way RM-ANOVA demonstrate a significant effect of the layout ($F_{1,132} = 215.58, p < .001$) and the session ($F_{5,132} = 238.89, p < .001$), which indicates a learning effect on the new layout. The error rate of the quadmetric optimized layout improves significantly from 14.88% (std. = 0.0058) in the first session and to 5.64% (std. = 0.0120) in the sixth session. The quadmetric optimized layout achieves a mean error rate of 10.35% (std. = 0.0304). In comparison, the alphabetical layout results in a mean error rate of 17.21% (std. = 0.0353). The error rate of 23.35% (std. = 0.0164) in the first session decreases to 14.04% (std. = 0.0181) in the sixth session. The error rate of 14.04% in the sixth session is close to the findings of the pilot test (16.10%). It shows the effects of optimization, the ease of force interaction, in particular, can alleviate error-prone force interaction for text entry tasks.

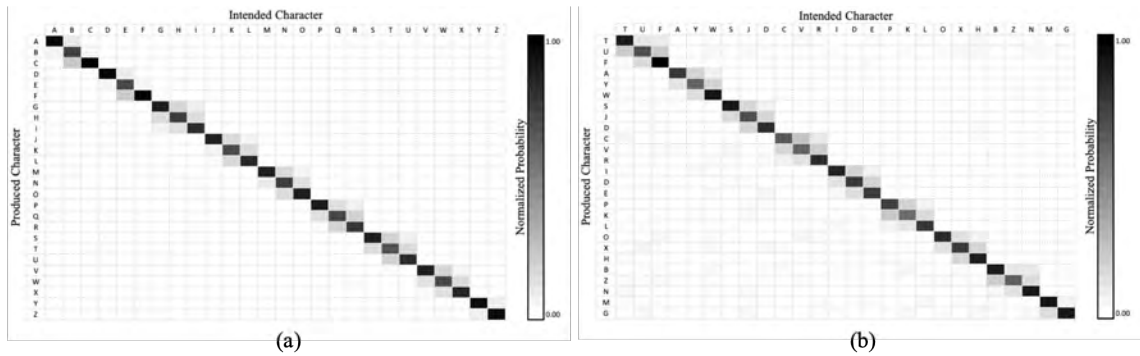


Fig. 18. Confusion matrix for two layouts: (a) Alphabetical constrained layout; (b) Quadmetric optimized layout.

The initial high error rate on the first day was mainly caused by the unfamiliarity with the new layout. The error rates improves in the later sessions as the users become more skillful. In particular, as the quadmetric optimized layout addresses the easiness of force keypad interaction, the quadmetric optimized layout improves more prominent than the alphabetical layout. Figure 18 shows the confusion matrix between the intended character and produced character. We classify the errors into two categories:

- (1) The user mistypes on another key, due to the unfamiliarity with the new layout.
- (2) The user mistakenly chooses an adjacent character inside the target ambiguous key, as an unintended force level is applied.

In the first session, more than half of the erroneous characters on the quadmetric optimized layout are caused by the first type of error (53.70%), while 46.30% of the erroneous characters are caused by the second type of error. In the final session, the overall error rate drops and the percentage of the first type of error diminishes to 19.10%. This phenomenon shows the users have learned about the new layout and avoid mistyping on the neighboring keys. Even though the second type of error on the fifth day surges to 80.90%, this does not imply that the easiness of force-assisted character selection was getting worse. In fact, the absolute number of average errors belonging to the second category drops by 33.87% from 62 in the first session to 41 in the sixth session. In addition, the participants show an improvement in controlling the force exertion for the new input modality under the quadmetric optimized layout. Our pilot test in Section 4 indicates that the 3-level force configuration leads to an error rate of 16.10% on average. The quadmetric optimized layout in the final session shows a significantly lower mean error rate of 5.64%. This difference can be explained by the character arrangement on the Quadmetric optimized layout, in which the most frequent characters (Figure 12) are located at the two ends of the 3-level force span ('light tap' and 'deep force touch'), minimizing the occurrence of characters located at the middle position ('shallow force touch') inside the keys.

NASA Task Load Index. We distributed NASA TLX forms [44] to the participants after the text entry tasks on the first and sixth sessions. Figure 20 shows the results of this survey. Student's T-tests with Bonferroni and Holm multiple comparisons between the alphabetical layout in the first session and the quadmetric optimized layout in the first session show a significantly higher mental load ($p < 0.05$), temporal factor ($p < 0.05$) and frustration ($p < 0.05$) but no significant difference in the physical load ($p = 1.000$), performance ($p = 0.310$) and effort ($p = 0.269$). T-tests with Bonferroni and Holm multiple comparisons yield significant differences in frustration ($p < 0.05$) between the alphabetical layout in the sixth session and quadmetric optimized layout in the sixth session but no significant differences in other five indexes: mental ($p = 0.196$), physical ($p = 0.069$), temporal

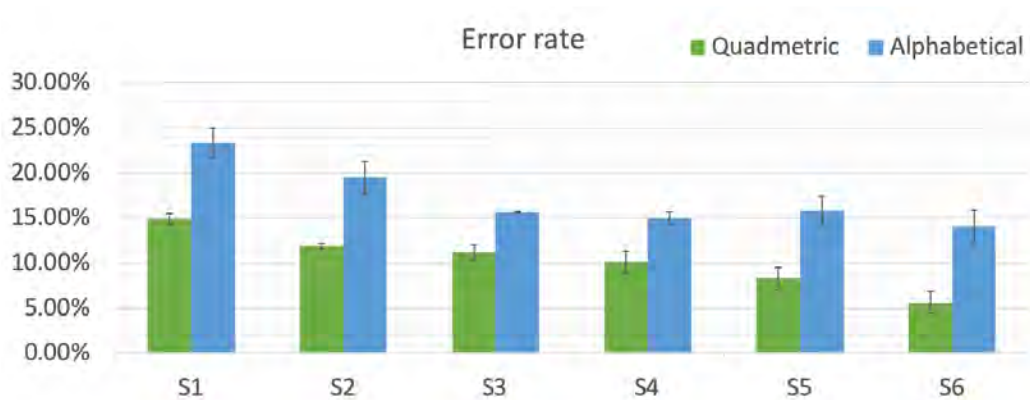


Fig. 19. Mean error rate over over 6 sessions (S1 – S6)

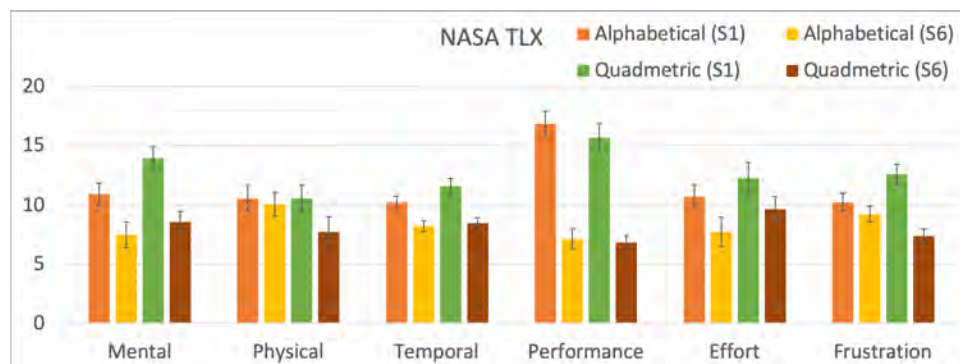


Fig. 20. NASA Task Load Index Results (Session 1 and 6). After 6 session, users perceive similar load for using quadmetric-optimized layout compared to the (baseline) alphabetical layout.

($p = 0.578$), performance ($p = 0.681$) and effort ($p = 0.173$). We can conclude that the participant's perceived load to the quadmetric layout significantly decreased over the 6-session text entry tasks, where T-tests with Bonferroni and Holm multiple comparisons (1st vs 6th session) yielded significant differences in all indexes ($p < 0.05$). In contrast, the participant's perceived load to the alphabetical layout over 6-session text entry tasks shows a significant decrease to the indexes of mental, temporal, performance and effort ($p < 0.05$) but no significant effects on physical ($p = 1.796$) and frustration ($p = 0.112$).

At the end of the study (6th session), we asked all the 12 participants the following question: *Would you use the quadmetric optimized keyboard on augmented reality headsets for typing tasks?* Participants replied on a 10-point scale, 1 meaning "definitely no" and 10 "definitely yes". 7 out of 10 participants replied positively on a 10-point scale (more than 5). The mean score is 6.5 out of 10.0 with a standard deviation of 2.4, and the ratings from participants are ranged from 3 to 9. Five participants felt that the quadmetric optimized layout was more counter-intuitive than the alphabetical layout in the first session but got used to the new layout in the

sixth session. In addition, 7 out of 12 participants commented that the quadmetric optimized layout made the character entry easier with the optimized character arrangement. In conclusion, the improved user perception of the quadmetric keyboard during the 6-session period can be seen as follows. In the first session, the participants were not familiar with the quadmetric optimized layout while the alphabetical order is deeply rooted in the mind of the participants. After 6-session practice with the quadmetric optimized layout, they felt more confident with the character positions and significantly improved their performance in the text input tasks.

7 DISCUSSION

The proposed thumb-to-finger interaction method on the quadmetric optimized layout achieves a mean text entry rate of 5.12 WPM throughout the 6 sessions and a mean text entry rate of 6.47 WPM during the final session. Other text entry systems using touch interfaces located on the frame of smartglasses, ring devices, or smartwatches, for instance, One-Dimensional Handwriting [7] (4.67 WPM), SwipeZone [6] (8.73 WPM), and Typing ring [14] (6 – 10 WPM) are faster than our system, as they involve more dexterous fingers such as the index and the middle finger. Azenkot et al. [33] give evidence that index-finger typing is faster than single-thumb typing on smartphones as the thumb suffers from a lower degree of movement. Therefore, it is expected that our proposed system has a reduced text entry rate compared to systems interacting with more dexterous fingers. However, our system allows for a concealed operation which is more acceptable in social situations [1]. Regarding text entry within the finger space, our system outperforms PalmType [3] (4.6 WPM), which detects the movement of the index finger of one hand to tap on the imaginary palm keyboard on the other hand. Digitouch [8] (13.0 WPM) applies two-handed gloves in which a full QWERTY keyboard is directly mapped to the touch-sensitive non-ambiguous keys. In the two-handed environment, the user can leverage both thumbs to type simultaneously and hence gain significant speed. Therefore, one-thumb interaction as proposed in this paper is expected to achieve only half of their text entry rate. However, one-handed interaction frees the other hand for performing other tasks, which is a non-negligible advantage in most mobile scenarios.

To the best of our knowledge, FingerT9 [19] (4.7 WPM) is the only work using a one-handed glove in which a multi-tap T9 layout is applied. Their text entry rate reaches 5.42 WPM in the final session, which is 16.23% slower than our approach (6.47 WPM). The force-assisted quadmetric optimized layout performs better than the multi-tap T9 layout for the following reasons. The multi-tap T9 layout requires the user to perform multiple taps for the 2nd- and 3rd-level characters. In contrast, the force-assisted quadmetric optimized layout enables the user to use a single force-assisted touch to select the 2nd- and 3rd-level characters directly. For example, the word phrase 'COOL' requires 12 taps with multi-tap T9 layout but only 4 taps with force-assisted layout (3 light tap and 1 deep force touch). The situation worsens with the number of characters, for instance, 'FORCEFULLY' requires 27 taps with multi-tap T9 layout and 10 taps with force-assisted layout (2 light tap; 2 shallow force touch; and 6 deep force touch). In addition, the 2nd- and 3rd-level characters in multi-tap T9 will trigger more taps than expected. For instance, the user may sometime over-tap on the key and need to tap further to correct his mistake. If character 'B' is over-tapped as 'C', the user needs 2 additional taps to revert to 'B', which deteriorates the user performance. In our scenario, the Fitts' law describes the relationship between thumb movement distance and the text entry performance. The Multi-tap T9 layout involves bigger thumb movement distance than the quadmetric optimized layout and thus leads to a slower text entry rate. On the other hand, a force-assisted quadmetric optimized layout enables users to change the exerted force with more subtlety, which minimizes the number of taps required.

Regarding the error rate, our system achieves a mean error rate of 10.35% across 6 sessions and a mean error rate of 6.85% in the final session. Even though the quadmetric optimized layout aims to alleviate the errors in force interaction [21, 23], the mean error rate of the proposed system is undoubtedly high compared to the multi-stroke character selection system such as One-Dimensional Handwriting [7] (0.78%), SwipeZone [6] (5.62%),

and Typing ring [14] (1.33%). With multiple discrete gestures, these state-of-the-art works provide a small number of options at the bottom of the decision hierarchy. Among the studies of two-handed text entry approach within the finger space, PalmType [3] (1.74%) relies on a high-grade motion tracking system in a controlled laboratory environment, which is not appropriate for both mobility and scalability. Digitouch [8] (15.8%) results in a higher error rate than our system. The users select the characters on a continuous rectangle-shaped strip of conductive fabric on a finger, in which the character selection is computed by its relative position on the strip. In contrast, with the advantages of human proprioception as an additional feedback mechanism [17], the separate keypads on the glove enable to accurately target the ambiguous keys once familiar with the quadmetric optimized layout. Our findings in Section 6 show that the absolute number of first type error (mistype on neighboring keys) decreases from 72 in the first session to 10 in the final session. In FingerT9 [19] (6.3%), taps on the separate keypads ensures the lower error rate and users can accurately find the keypad locations [4]. In our system, the users are less confident with the shallow force touch. Additional training sessions are required to become proficient with force-assisted clicks, even though the findings in Section 6 show that the second type error (misuse of force level on the same key) drops by 33.87% after only 6 sessions. We strike a balance between the text entry rate and the accuracy. Although achieving a slightly (0.45%) higher error rate than FingerT9, our work achieves a significantly higher text entry rate (37% higher). In the future work, user-specific optimization will be performed to further enhance the text entry rate and error rate.

With our proposed one-handed glove, users can perform text entry small and unnoticeable finger movements in an off-hand posture. Some in-use postures such as lifting up the arm (e.g. One-Dimensional Handwriting [7]), waving the arm [14], and constantly tapping the touchscreen on smartwatches [6], are often regarded as socially awkward [1]. Additionally, the one-handed off-sight feature of our system can enhance the use in mobile situations. For instance, the other hand can hold the handrail in commuting while the visual attention is reserved for the physical environment and AR objects.

8 CONCLUSION

In this paper, we introduced a novel one-handed thumb-to-finger text entry solution for head-worn computers. This solution targets more particularly head-worn computers such as AR headsets, that are used in mobile scenarios. During the pilot study, we investigated the user perception of the force control in the 3-level and 4-level configurations. We chose the 3-level configuration, for the design of ambiguous nodes on the glove, due to its lower error rate (16.1%). We developed a force-sensitive prototypical glove consisting of 9 force-sensitive nodes on the phalanges of the index, middle, and ring fingers in a 3-3-3 configuration, and 3 touch nodes on the small finger. We mapped the 26 English characters on the 9 force-sensitive nodes, that are distinguished through the force-sensitive interaction. The 3 touch nodes on the small finger were mapped to the whitespace, enter and backspace keys. We then formulated our optimization problem for the keyboard layout with the following four metrics: 1) Goodness of character pair 2) User familiarity with the QWERTY layout, 3) Easiness of force touch interaction, and 4) Comfort level of finger space, and introduced the resulting quadmetric optimized layout. The optimized layout was assessed over 6 sessions with 12 participants. The participants achieved an average text entry rate of 5.12 WPM with 10.35% error rate throughout the 6 sessions. In the final session, the participants reached an average text entry rate of 6.47 WPM with 6.85% error rate, which is noticeably faster than other thumb-to-finger text entry solutions proposed in the literature. In addition, the quadmetric layout alleviated error-prone force interaction. This study serves as a groundwork for key disambiguation by force interaction for text entry on limited-sized interfaces. As force disambiguation combined with the quadmetric optimized layout can reduce the necessary numbers of nodes on the glove significantly, our solution allows shrinking the layout from two-handed to one-handed typing. The reserved hand can enhance user mobility, for instance, carrying a briefcase, shaking hands, or handling other daily tasks. In the future, we will deeply investigate the

thumb performance and install an electrical impedance tomograph on the glove to track the thumb movement. Combined with our findings on the mean time of force-assisted touches, we can understand better about the user reaction time on various (sub-)optimized layouts. In addition, we observed that the thumb movement is less dexterous than other fingers. We will conduct in-depth studies by equipping the glove with alternative feedback sources such as haptic feedback, guiding the user's thumb to the most probable characters of the frequently used word phrases and hence achieve improvement in text entry rate. In the current evaluation, the participants were seated in the laboratory. We will further investigate the performance in other mobile scenarios such as walking, standing, and analyze scenarios in which users focus on another task while typing.

ACKNOWLEDGMENTS

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong and the 5GEAR project from the Academy of Finland.

REFERENCES

- [1] Yi-Ta Hsieh, Antti Jylhä, Valeria Orso, Luciano Gamberini, and Giulio Jacucci. Designing a willing-to-use-in-public hand gestural interaction technique for smart glasses. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 4203–4215, New York, NY, USA, 2016. ACM.
- [2] Chun Yu, Yizheng Gu, Zhican Yang, Xin Yi, Hengliang Luo, and Yuanchun Shi. Tap, dwell or gesture?: Exploring head-based text entry techniques for hmds. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 4479–4488, New York, NY, USA, 2017. ACM.
- [3] Cheng-Yao Wang, Wei-Chen Chu, Po-Tsung Chiu, Min-Chieh Hsiu, Yih-Harn Chiang, and Mike Y. Chen. Palmtype: Using palms as keyboards for smart glasses. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '15, pages 153–160, New York, NY, USA, 2015. ACM.
- [4] Da-Yuan Huang, Liwei Chan, Shuo Yang, Fan Wang, Rong-Hao Liang, De-Nian Yang, Yi-Ping Hung, and Bing-Yu Chen. Digitspace: Designing thumb-to-fingers touch interfaces for one-handed and eyes-free interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1526–1537, New York, NY, USA, 2016. ACM.
- [5] Anna Peshock, Julia Duvall, and Lucy E. Dunne. Argot: A wearable one-handed keyboard glove. In *Proceedings of the 2014 ACM International Symposium on Wearable Computers: Adjunct Program*, ISWC '14 Adjunct, pages 87–92, New York, NY, USA, 2014. ACM.
- [6] Tovi Grossman, Xiang Anthony Chen, and George Fitzmaurice. Typing on glasses: Adapting text entry to smart eyewear. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '15, pages 144–152, 2015.
- [7] Chun Yu, Ke Sun, Mingyuan Zhong, Xincheng Li, Peijun Zhao, and Yuanchun Shi. One-dimensional handwriting: Inputting letters and words on smart glasses. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 71–82, 2016.
- [8] Eric Whitmire, Mohit Jain, Divye Jain, Greg Nelson, Ravi Karkar, Shwetak Patel, and Mayank Goel. Digitouch: Reconfigurable thumb-to-finger input and text entry on head-mounted displays. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):113:1–113:21, September 2017.
- [9] Mark Dunlop and John Levine. Multidimensional pareto optimization of touchscreen keyboards for speed, familiarity and improved spell checking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2669–2678, New York, NY, USA, 2012. ACM.
- [10] Xiaojun Bi, Barton A. Smith, and Shumin Zhai. Quasi-qwerty soft keyboard optimization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 283–286, New York, NY, USA, 2010. ACM.
- [11] Ryan Qin, Suwen Zhu, Yu-Hao Lin, Yu-Jung Ko, and Xiaojun Bi. Optimal-t9: An optimized t9-like keyboard for small touchscreen devices. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*, ISS '18, pages 137–146, New York, NY, USA, 2018. ACM.
- [12] Hsin-Ruey Tsai, Te-Yen Wu, Da-Yuan Huang, Min-Chieh Hsiu, Jui-Chun Hsiao, Yi-Ping Hung, Mike Y. Chen, and Bing-Yu Chen. Segtouch: Enhancing touch input while providing touch gestures on screens using thumb-to-index-finger gestures. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, pages 2164–2171, New York, NY, USA, 2017. ACM.
- [13] R. McCall, B. Martin, A. Popleteev, N. Louveton, and T. Engel. Text entry on smart glasses. In *2015 8th International Conference on Human System Interaction (HSI)*, pages 195–200, June 2015.
- [14] Shahriar Nirjon, Jeremy Gummeson, Dan Gelb, and Kyu-Han Kim. Typingring: A wearable ring platform for text input. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '15, pages 227–239, 2015.

- [15] Sunggeun Ahn, Seongkook Heo, and Geehyuk Lee. Typing on a smartwatch for smart glasses. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pages 201–209, New York, NY, USA, 2017. ACM.
- [16] Yui Pan Yau Tristan Braud Lik-Hang Lee, Kit Yung Lam and Pan Hui. Hibey: Hide the keyboard in augmented reality. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications*, PERCOM '19, Kyoto, JAoan, 2019. IEEE.
- [17] L. Lee and P. Hui. Interaction methods for smart glasses: A survey. *IEEE Access*, 6:28712–28732, 2018.
- [18] Dale L. GroverMartin T. KingClifford A. Kushler. Reduced keyboard disambiguating computer, May 7 1995. US Patent 5818437A.
- [19] Pui Chung Wong, Kening Zhu, and Hongbo Fu. Fingert9: Leveraging thumb-to-finger interaction for same-side-hand text entry on smartwatches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 178:1–178:10, New York, NY, USA, 2018. ACM.
- [20] I. Scott MacKenzie, Hedy Kober, Derek Smith, Terry Jones, and Eugene Skepner. Letterwise: Prefix-based disambiguation for mobile text input. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, UIST '01, pages 111–120, New York, NY, USA, 2001. ACM.
- [21] Sachi Mizobuchi, Shinya Terasaki, Turo Keski-Jaskari, Jari Nousiainen, Matti Ryyanen, and Miika Silfverberg. Making an impression: Force-controlled pen input for handheld devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1661–1664, New York, NY, USA, 2005. ACM.
- [22] Jibin Yin, Xiangshi Ren, and Shumin Zhai. Pen pressure control in trajectory-based interaction. 29:137–148, 03 2010.
- [23] Graham Wilson, Craig Stewart, and Stephen A. Brewster. Pressure-based menu selection for mobile devices. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '10, pages 181–190, New York, NY, USA, 2010. ACM.
- [24] Min-Chieh Hsiu, Da-Yuan Huang, Chi An Chen, Yu-Chih Lin, Yi-ping Hung, De-Nian Yang, and Mike Chen. Forceboard: Using force as input technique on size-limited soft keyboard. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, MobileHCI '16, pages 599–604, New York, NY, USA, 2016. ACM.
- [25] Mingyuan Zhong, Chun Yu, Qian Wang, Xuhai Xu, and Yuanchun Shi. Forceboard: Subtle text entry leveraging pressure. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 528:1–528:10, New York, NY, USA, 2018. ACM.
- [26] James Clawson, Ahmed Sabbir Arif, Stephen Brewster, Mark Dunlop, Per Ola Kristensson, and Antti Oulasvirta. Text entry on the edge. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pages 2381–2384, New York, NY, USA, 2015. ACM.
- [27] Mark D. Dunlop, Naveen Durga, Sunil Motaparti, Prima Dona, and Varun Medapuram. Qwerth: An optimized semi-ambiguous keyboard design. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services Companion*, MobileHCI '12, pages 23–28, New York, NY, USA, 2012. ACM.
- [28] Brian A. Smith, Xiaojun Bi, and Shumin Zhai. Optimizing touchscreen keyboards for gesture typing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 3365–3374, New York, NY, USA, 2015. ACM.
- [29] Jun Gong, Zheer Xu, Qifan Guo, Teddy Seyed, Xiang 'Anthony' Chen, Xiaojun Bi, and Xing-Dong Yang. Wristext: One-handed text entry on smartwatch using wrist gestures. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 181:1–181:14, New York, NY, USA, 2018. ACM.
- [30] Sachi Mizobuchi, Shinya Terasaki, Turo Keski-Jaskari, Jari Nousiainen, Matti Ryyanen, and Miika Silfverberg. Making an impression: Force-controlled pen input for handheld devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1661–1664, New York, NY, USA, 2005. ACM.
- [31] Christian Corsten, Simon Voelker, and Jan Borchers. Release, don't wait!: Reliable force input confirmation with quick release. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pages 246–251, New York, NY, USA, 2017. ACM.
- [32] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA, 1996.
- [33] Shiri Azenkot and Shumin Zhai. Touch behavior with different postures on soft smartphone keyboards. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '12, pages 251–260, New York, NY, USA, 2012. ACM.
- [34] T. Segaran and J. Hammerbacher. *Beautiful Data: The Stories Behind Elegant Data Solutions*. Theory in practice. O'Reilly Media, 2009.
- [35] Shumin Zhai, Michael Hunter, and Barton A. Smith. The metropolis keyboard - an exploration of quantitative techniques for virtual keyboard design. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pages 119–128, New York, NY, USA, 2000. ACM.
- [36] Yiqin Lu, Chun Yu, Xin Yi, Yuanchun Shi, and Shengdong Zhao. Blindtype: Eyes-free text entry on handheld touchpad by leveraging thumb's muscle memory. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(2):18:1–18:24, June 2017.
- [37] Sunyu Hwang and Geehyuk Lee. Qwerty-like 3x4 keypad layouts for mobile phone. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1479–1482, New York, NY, USA, 2005. ACM.
- [38] Hokyoung Ryu and Katrina Cruz. Letterease: Improving text entry on a handheld device via letter reassignment. In *Proceedings of the 17th Australia Conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future*, OZCHI '05, pages

- 1–10, Narrabundah, Australia, Australia, 2005. Computer-Human Interaction Special Interest Group (CHISIG) of Australia.
- [39] Shumin Zhai and Barton A Smith. Alphabetically biased virtual keyboards are easier to use: Layout does matter. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '01, pages 321–322, New York, NY, USA, 2001. ACM.
- [40] Jun Gong and Peter Tarasewich. Alphabetically constrained keypad designs for text entry on mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pages 211–220, New York, NY, USA, 2005. ACM.
- [41] Dale L. Grover; Martin T. King; Clifford A. Kushler. Reduced keyboard disambiguating computer, May 7 1995. US Patent 5818437A.
- [42] Christina L. James and Kelly M. Reischel. Text input for mobile devices: Comparing model prediction to actual performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, pages 365–371, New York, NY, USA, 2001. ACM.
- [43] Lee Butts and Andy Cockburn. An evaluation of mobile phone text input methods. In *Proceedings of the Third Australasian Conference on User Interfaces - Volume 7*, AUIC '02, pages 55–59, Darlinghurst, Australia, Australia, 2002. Australian Computer Society, Inc.
- [44] NASA AMES Research Center Human Performance Research Group. Nasa task load index (tlx), 1999.