Handwriting Velcro: Endowing AR Glasses with Personalized and Posture-adaptive Text Input using Flexible Touch Sensor

FENGYI FANG, Xiamen University, China and Tsinghua University, China
HONGWEI ZHANG, Xiamen University, China
LISHUANG ZHAN, Xiamen University, China
SHIHUI GUO*, Xiamen University, China
MINYING ZHANG, Alibaba Group, China
JUNCONG LIN, Xiamen University, China
YIPENG QIN, Cardiff University, United Kingdom
HONGBO FU, City University of Hong Kong, China

Text input is a desired feature for AR glasses. While there already exist various input modalities (e.g., voice, mid-air gesture), the diverse demands required by different input scenarios can hardly be met by the small number of fixed input postures offered by existing solutions. In this paper, we present *Handwriting Velcro*, a novel text input solution for AR glasses based on flexible touch sensors. The distinct advantage of our system is that it can easily stick to different body parts, thus endowing AR glasses with posture-adaptive handwriting input. We explored the design space of on-body device positions and identified the best interaction positions for various user postures. To flatten users' learning curves, we adapt our device to the established writing habits of different users by training a 36-character (i.e., A-Z, 0-9) recognition neural network in a *human-in-the-loop* manner. Such a personalization attempt ultimately achieves a low error rate of 0.005 on average for users with different writing styles. Subjective feedback shows that our solution has a good performance in system practicability and social acceptance. Empirically, we conducted a heuristic study to explore and identify the best interaction Position-Posture Correlation. Experimental results show that our *Handwriting Velcro* excels similar work [6] and commercial product in both practicality (12.3 WPM) and user-friendliness in different contexts.

CCS Concepts: • **Human-centered computing → Text input**.

Additional Key Words and Phrases: handwriting input, AR glasses, flexible touch sensor, personalized and posture-adaptive text input

ACM Reference Format:
Fengyi Fang, Hongwei Zhang, Lishuang Zhan, Shihui Guo, Minying Zhang, Juncong Lin, Yipeng Qin, and Hongbo Fu. 2022. Handwriting Velcro: Endowing AR Glasses with Personalized and Posture-adaptive Text Input using Flexible Touch Sensor. 1,

*Corresponding author

Authors' addresses: Fengyi Fang, Xiamen University, Xiamen, China and Tsinghua University, Shenzhen, China, 34520182201469@stu.xmu.edu.cn; Hongwei Zhang, Xiamen University, Xiamen, China, zhhhw@stu.xmu.edu.cn; Lishuang Zhan, Xiamen University, Xiamen, China, zzzlis@stu.xmu.edu.cn; Shihui Guo, Xiamen University, Xiamen, China, guoshihui@xmu.edu.cn; Minying Zhang, Alibaba Group, Hangzhou, China, minying.zmy@alibaba-inc.com; Juncong Lin, Xiamen University, Xiamen, China, jclin@xmu.edu.cn; Yipeng Qin, Cardiff University, Cardiff, United Kingdom, qiny16@cardiff.ac.uk; Hongbo Fu, City University of Hong Kong, Hong Kong, China, fuplus@gmail.com.

1 INTRODUCTION

Text input is one of the essential functionalities in promoting the popularity of AR glasses amongst the mass population. This task is important for applications such as instant messages and information editing on AR glasses. In addition to input efficiency, the portability of AR glasses requires an input method to adapt to a wide range of environments and user states, including indoor/outdoor scenes and moving/stationary postures in daily life [62] (as illustrated in Fig. 1). However, such a demanding requirement to cope with diverse usage scenarios has not been fully met so far.

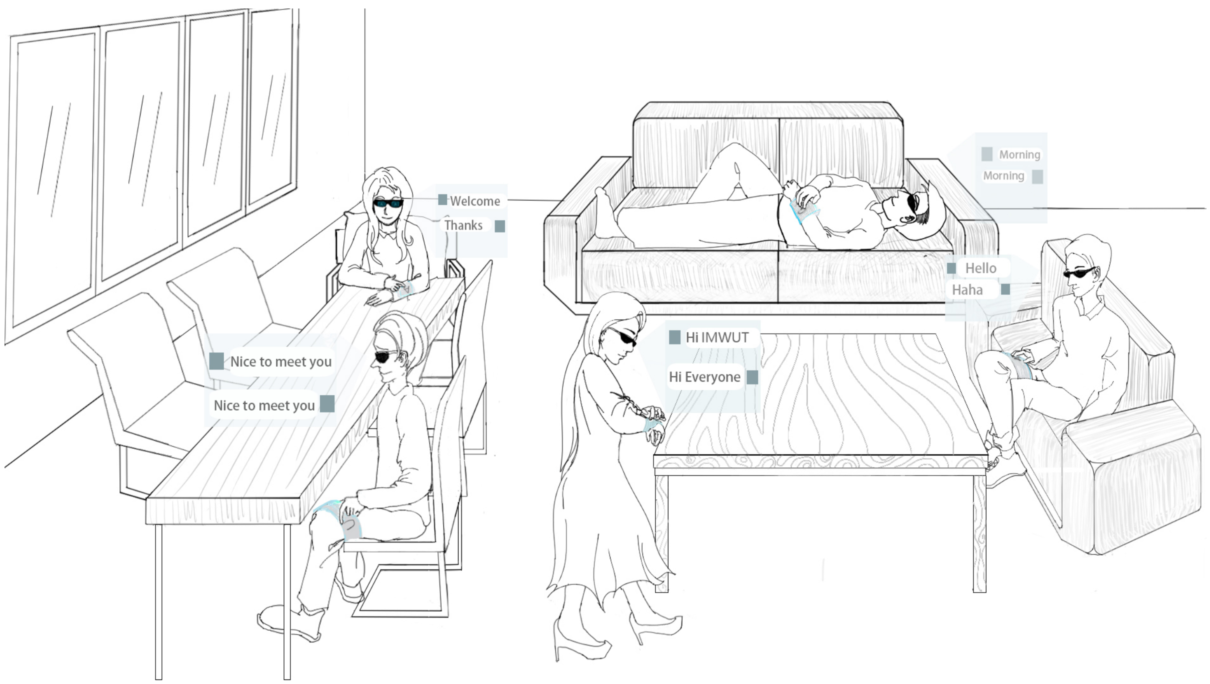


Fig. 1. Our work provides handwriting input for augmented reality (AR) glasses, enabled by on-body flexible touch sensors. Our method adapts to both personal writing styles and diverse user postures, offering accurate recognition performance and a user-friendly experience.

Existing text input solutions on AR glasses include voice input [1], gaze tracking [3, 24, 37], mid-air gesture [10], on-device touch bars [83, 86], and wearable accessories [6, 82]. Nevertheless, the diverse usage scenarios of AR glasses pose challenges to each of these methods. Specifically, voice input is natural but highly selective to external environments, i.e., not suitable for use in a quietly shared environment [14] or a noisy environment [81]. Touch bars on AR glasses are ideal for simple menu operations, but their small size and number only allow limited input expression and speed in the task of text input [14, 83]. Mid-air gestures can easily cause arm fatigue [25] and might lead to misunderstood social cues [14]. Additionally, most of existing wearable accessory devices restrict the interaction positions to the vicinity of the hand, such as finger [39, 79], arm [6], wrist [19], etc. However, these novel wearable devices often implement interactive control and text input by designing different touch gestures [51, 60] or finger combinations [6], thus often leading to low user affordance and additional learning costs. Therefore, each of the above interaction paradigms have their own pros and cons. We consider

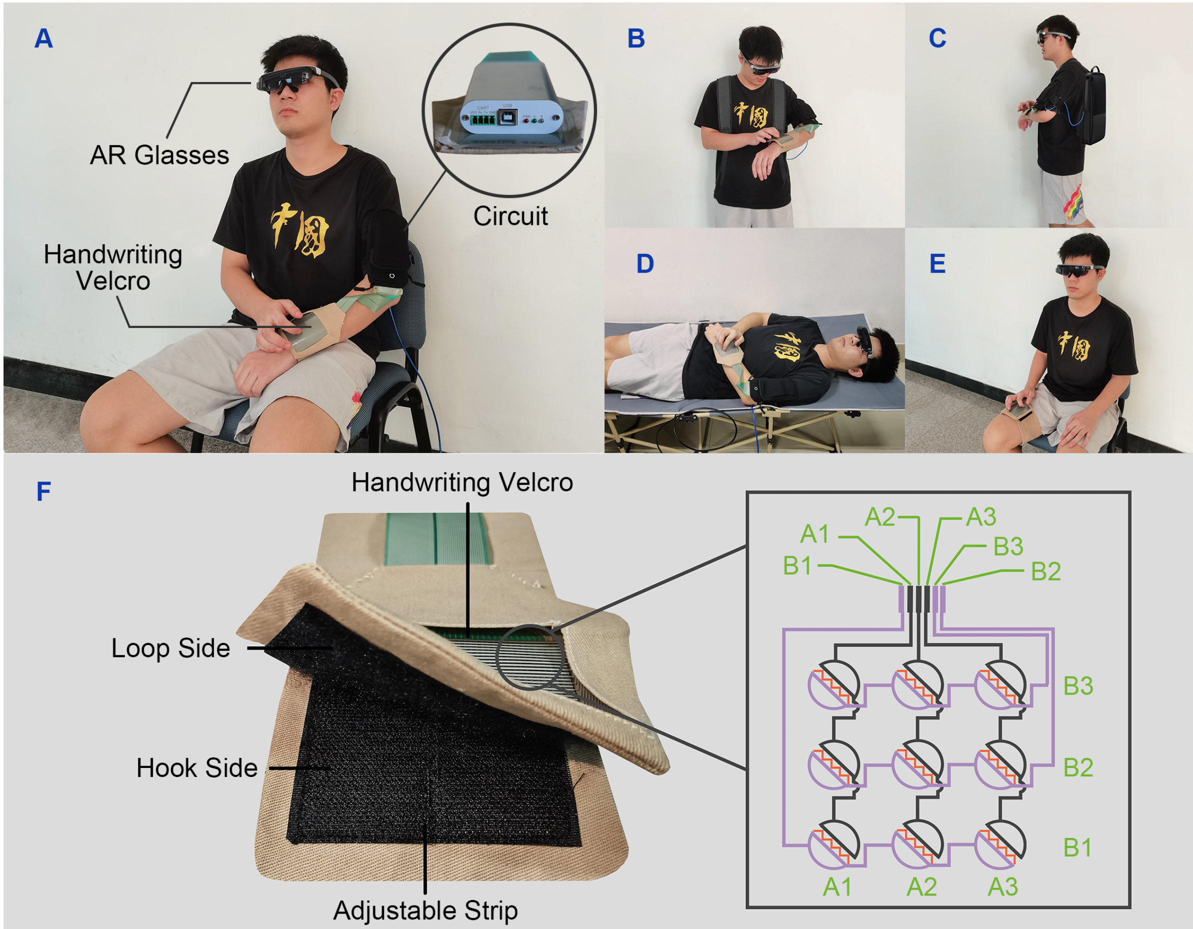


Fig. 2. *Handwriting Velcro* in use (A) under four different postures (B-E). A close photo with a diagram (F) to show the layout of sensing points and connection to the circuit.

our work as an additional contribution to the collection, with a specific focus on natural posture adaptability and user-friendliness, which provides a robust input experience for common challenging input scenarios.

We propose a novel text input solution for AR glasses, *Handwriting Velcro*, using flexible touch sensors for personalized and posture-adaptive handwriting input (Fig. 1). Our design rationale is inspired by the fact that multi-posture adaptation is an inevitable step towards anywhere and anytime text input for AR glasses. Specifically, the large number of AR text input scenarios (e.g., time, place, movement, etc., and their combinations) lead to diverse demands that can hardly be met by a small number of fixed input postures (i.e., sitting or walking) offered by existing solutions [22, 28, 40, 41, 70]. The flexible touch sensors employed in our work could fill this gap since they can be easily attached to any positions of body surfaces (i.e., interaction positions) comfortably and naturally, thereby giving users complete freedom to explore the best interaction positions and postures for different AR text input scenarios. To make the most use of the flexible film, we choose the handwriting input paradigm. As the original form of creating human-readable text [45], handwriting is highly user-friendly. However, being flexible implies shape deformation, which produces jerky signals and other nonlinear artifacts when monitoring continuous position input [64]. To directly address the challenges in signal processing, we propose dedicated data preprocessing methods and use a deep convolutional neural network (CNN) to implement handwritten character recognition (Sec. 5). In order to cope with a personalized handwriting style, we introduce the "human-in-the-loop" idea [26] and apply active learning [61] to implement personalized learning on a small labeled sample set for a single user with minimal cost.

To summarize, our work makes the following contributions:

- We present *Handwriting Velcro* for multi-position and multi-posture AR text input, which can easily adapt to diverse input scenarios (Fig. 2). In addition, our *Handwriting Velcro* uses flexible touch sensors for handwriting input, allowing for a user-friendly input method on lightweight and flexible film that can be less intrusively integrated into daily clothes.
- We introduce a novel method to learn a personalized handwritten character classifier for each single user to cope with a personalized handwriting style. The experimental results show that, compared with the non-personalized classifier, the total error rate of personalized classifiers for individual users (6 in total) decreases by an average of 0.026 after six sessions, reaching 0.005.
- We designed and conducted extensive user experiments to obtain enlightening guidance and to evaluate the performance of our system under various conditions of different users, postures, and interaction positions. Subjective feedback shows that our solution has a good performance in system practicality and social acceptance. We also compared the performance between our solution and the existing work (TEXTile [6]) and commercial product (physical mini QWERTY keyboard) and linked our research to other similar wearable text input schemes in the field (Table. 1). Results show that our method excels the existing technique in both practicality (12.3 WPM) and user-friendliness in different daily contexts.

2 RELATED WORK

2.1 Text Input On AR Glasses

The small size of AR glasses does not allow the installation of a large-area touch screen. This severely limits the tactile interaction of AR glasses and thus makes text input on AR glasses a tough problem [41, 72]. In this regard, researchers have conducted numerous research on the text input mechanisms of smart glasses.

One group of methods aim to enter text through the limited interactive area of AR glasses. For example, Yu et al. [83] proposed a one-dimensional writing gesture system, and SwipeZone [20] divides the touch interface of Google Glass into two parts for letter selection. However, due to the small interaction area of AR glasses, the interaction of such a method is not very convenient, highly limiting its typing speed.

In order to obtain a larger operating space, mid-air gesture interaction is still a commonly adopted solution. For instance, HIBEY [42] achieves keyboard-like text input through vision-based freehand interactions. Meanwhile, voice recognition [1] and eye tracking [3, 37] are also popular interaction means. For example, Hololens integrates these three interaction methods at the same time. Hummer [24] introduces text entry by gaze and hum as a novel hands-free text entry tool. However, these methods often have strict requirements on usage scenarios and equipment as discussed in Sec. 1.

It is also common to use external wearable devices for text input, which have the advantages of being wearable, mobile, and flexible. Most of existing works have focused on the interaction positions near the hand, such as finger [49, 79, 82], arm [6], and wrist [19]. For example, TipText [79], RotoSwype [22] and FingerText [39] provide finger-based gesture or typing input. In addition, the interaction on the forearm is also highly popular. For example, GestureSleeve [60] is a fabric gesture recognition sleeve for smartwatches. Some studies like WrisText [19] also chose the wrist as an interactive position. Inspired by these studies, we believe that fingers and forearms are highly potential interactive spaces. Additionally, our work does not limit the interaction positions to the vicinity of the hand and has explored a variety of interactive positions, including the forearm, upper arm, thigh, abdomen, etc.

There are also various research studies exploring the input methods in the VR field [31, 52, 80, 84], most of which are based on a physical keyboard [9, 21] or virtual keyboard [2, 16] for interaction in virtual environments. Many of these interaction methods can also be used in augmented reality [10, 17, 43, 76]. Meanwhile, researchers have developed new keyboards [56] based on hybrid methods [31] and touch-sensitive physical keyboards [47, 52, 53]. Besides varieties of keyboards, external input devices [30, 84] have been explored as well, enriching the diversity of interaction. More gesture-based [15], multi-modal [1] and hands-free [44, 77] input methods have been proposed in order to adapt to various mobile usage scenarios.

In this article, we focus on a novel text input method for text entry on AR glasses by using handwriting input based on flexible touch sensors. It is non-invasive, flexible, and easy to integrate with clothing. Another work, TEXTile [6], shares a similar goal for text input on AR glasses, but fixes the interaction position only on the forearm and lacks exploration of different interaction postures. At the same time, their complicated finger combination also increases a user's learning cost to a certain extent. In contrast, our method conforms with writing habits and supports head-up text input while moving, and is applicable to both single- and two-handed input. Most importantly, our work features personalized recognition and posture adaptability. In Sec. 7, we will further compare our solution with TEXTile for system performance and subjective feedback.

2.2 Interaction On Flexible Touch Sensor

Flexible sensors have the advantages of being flexible and wearable [48]. Thus, they demonstrate potential advantages over rigid sensors in application fields [27] such as human-computer interaction [59, 69], medical health [29], and robotic haptics [66, 71]. In various input interaction methods, wearable touch input based on flexible sensors is considered to have exceedingly high input expressiveness [14].

With the development of tactile sensor [67], numerous flexible touch sensors [73, 74] have appeared. Some researchers use flexible sensors as wearable interactive interfaces to achieve interactive control of mobile smart devices. Jacquard [57] and GestureSleeve [60] integrate touch textiles and sleeves, realizing the control of smart earphones and watches through simple touch gestures. I/O Braid [50, 51] advances cord-based interfaces, implementing both the continuous and discrete gesture control for headphones. Some other studies explore the interaction modality on the skin [4, 85] and pocket [14, 68, 75] and combine touch with other modal interactions. The zPatch [65] is an eTEXTile patch using both resistive and capacitive sensing, supporting hovering, touch, and pressure input. The Project Tasca [75] is a pocket-based textile sensor integrating four distinct types of sensing methods. These works all demonstrate the advantages and potential usages of flexible sensors.

In addition to simple interactive control, there are also various text input tasks based on flexible touch sensors. Fingers [5, 34, 72] are the most common input position. In the case of the forearm, TEXTile [6] uses touch (capacitive) textiles for text input on smart glasses.

Among various flexible touch sensors, pressure and strain sensors are the most popular categories [48, 54]. Pressure-based position input used in wearable systems is highly common [64], including on-body gesture input [60], body posture classification [63], detection of advanced deformation gestures [55], etc. However, because the pressure sensor matrix often encounters problems such as jitter, jumping, or other nonlinear artifacts in practice, the current application of pressure matrix for precise and continuous position control is very rare [64]. This might limit the wearable touch interaction to basic gestures [14] using pressure sensors.

Our work detects continuous position input on a flexible pressure touch sensor, using a resistive pressure sensor matrix. We carry out a series of data preprocessing techniques to eliminate the signal quality problems caused by the sensor acquisition process.

2.3 Handwriting Input Recognition

Nowadays, research in mobile text entry is flourishing, and there are two main text entry technologies: typing and handwriting [45]. In some human-computer interaction interfaces, some research has proven that handwriting recognition is generally superior to an on-screen keyboard [11, 33]. Because handwritten text entry must be coupled with a recognition technology, there have been a large number of open source handwritten text databases [12, 13] and recognition algorithms [38]. The most widely known dataset is MNIST, which was first introduced in 1998 by LeCun et al. [38]. Deep neural networks show consistently superior performance on the MNIST dataset [13]. More specifically, a convolutional neural network (CNN) designed to handle 2D shape variability, outperforming all other solutions [38]. Since MNIST contains only numerals, Cohen et al. propose EMNIST (Extended MNIST) [12], constituting a more challenging classification task involving letters and numerals.

Although handwriting input is rather mature, few research studies have explored its use as a text input method for wearable smart devices such as AR glasses. One of the difficulties is that AR glasses lack convenient and suitable handwriting input media [41, 72]. For most text input methods, it is hard to meet the characteristics of being wearable and flexible required by AR glasses simultaneously [18]. Thus, flexible touch sensors may provide a suitable potential handwriting interface.

3 HARDWARE

Handwriting Velcro. As a handwriting input interface of AR glasses, our Handwriting Velcro (Fig. 2) consists of a flexible touch sensor and a circuit for signal collection. We use an off-shelf flexible touch sensor provided by *Legact*. The sensor is composed of flexible polyester films, highly-conductive materials, and nano-scale piezoresistive materials. It is divided into the bottom and top layers of flexible pressure-sensitive films, which are bonded by double-sided adhesive to isolate the pressure-sensitive areas of the upper and lower layers. When the sensing area is under pressure, the two pressure-sensitive layers are in contact with each other, and the resistance output value of the channel changes with different positions. The resistance value of each channel is represented with 12 bits and digitized into a value within [0, 4095]. The size of the sensing region is 8.36×8.36 cm^2, and the resolution is 44×44. This implies that the size of each sensing point is 1.6×1.6 mm^2, and the distance between two sensing points is 0.3 mm. To avoid data disruption, we discard the four sensing points along the edges and focus on the central region. This reduces the resolution to 40×40. The sensing threshold is 20 grams and the response time is <10 μs, which are both sufficiently small for convenient and interactive handwriting.

The sensor is attached to the loop side of a *velcro* patch (Fig. 2F). The sensor weighs 30 grams and its thickness is 0.2 mm, significantly smaller than a standard *velcro* strip (around 1 cm). The hook side of the velcro is placed on three wearable strips with different lengths to fit the girths at different body locations. To facilitate flexible and stable handwriting, we further use three adjustable strips (see Fig. 2 (F)) to secure our Velcro to the body.

The sensor is connected to an 80-pin self-designed circuit for signal collection. 40 pins are connected to the horizontal channels (denoted as A1, A2, A3$\cdots$), and another 40 pins are connected to the vertical ones (denoted as B1, B2, B3$\cdots$). For protection and portability, the circuit is covered by an aluminum shell (shown in Fig. 2A). This circuit for signal collection can be purchased from the sensor vendor upon request. The sampling channel for all sensing points is selected by a multiplexing micro-processing unit. Serial sensor data is collected and transmitted to a computer via USB and processed by a dedicated program in real-time. The frequency of data collection for all sensing points is 60 fps.

AR Glasses. The AR glasses in our work is a Rokid Glass 2, which has a field of view (FOV) of 40°, a 1280×720 resolution screen, a touch bar (2cm wide), and an Android operating system supporting 4G, WIFI, and Bluetooth connections. Only the right lens of the glasses is equipped with a visual screen, which will not block the user's vision. It weighs approximately 96 grams.

Processing Unit. In our experiments, sensor data are collected by the circuit and transmitted to a computer (Intel (R) Core (TM) i7-8700K CPU @ 3.70GHz, GTX 1080Ti, 16.0GB RAM) via USB. The program for both data collection and recognition algorithm currently runs on the computer. The processed information is then transferred to the AR glasses via a local wireless network.

4 HEURISTIC STUDY

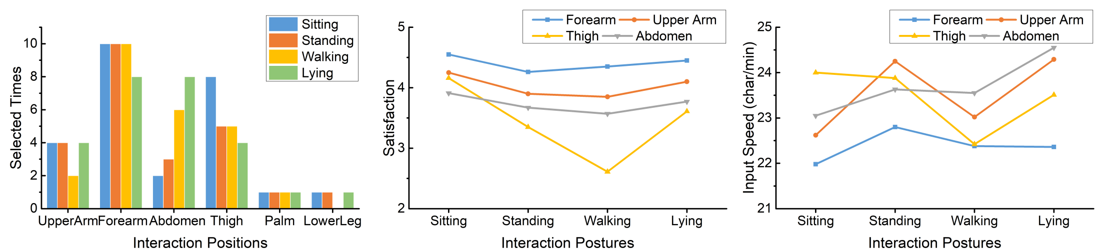
4.1 Free Exploration of Interaction Positions

Previous works in touch interaction explored various interaction positions on the surfaces of different body parts, including forearm [60], fingers [39], wrist [19], pocket [14], etc. Following this paradigm, we have conducted the following free exploration experiment to gain insights into users' preferred interaction positions in different postures when using our Handwriting Velcro without guidance.

4.1.1 Participants. We recruited 10 participants (5 male, 5 female). One was left-handed while others were right-handed. We have applied for IRB for all experiments of this work, and each participant in this work received a gratuity of $10 for their contribution.

4.1.2 Task and Procedure. Prior to the experiments, we asked the participants to sign written consent forms and introduced them to our Handwriting Velcro and AR glasses. They were informed of the experiment's purpose, as well as provided with a short tutorial on handwriting input. The participants were then instructed to freely explore and try different interactive positions to place our *Handwriting Velcro* when they were performing four postures, *i.e., sitting, standing, walking and lying down*. These four postures were selected based on heuristic analysis and literature findings so as to cover most daily use scenarios [7].

During the experiments, we recorded the interaction positions tried by each participant and asked them to demonstrate how they wore the Handwriting Velcro in the above-mentioned 4 postures, to explain why they chose them, and to comment on their attempts.



(a) (b) (c)

Fig. 3. Results from the heuristic study. (a) The number of selections of different interaction positions tried by users in the *free exploration* phase. (b) User satisfaction of the 4 candidate interaction positions in 4 postures. (c) Input speed of the 4 candidate interaction positions in 4 postures.

4.1.3 Results. As Fig. 3(a) shows, although the participants were guided to freely explore interaction positions, these attempts were eventually gathered in the four most representative positions (*i.e.,* upper arm, forearm, abdomen, and thigh) and two outliers (*i.e.,* palm and lower leg):

Upper Arm. The position was attempted by 2 to 4 participants in different postures, respectively. All of them verbally praised this choice, indicating that it could be a hidden gem. One participant who chose the upper arm gave a high evaluation, "Possibly due to its large area, the experience of writing on the upper arm was better than expected." Another participant said, "I didn't choose the upper arm because it was too high and unnatural to write on it."

Forearm. We think the forearm may be the most appropriate interaction position, since almost all the participants tried it in each of the postures. One of them said that it was as acceptable as wearing a watch. Another participant said, "I was used to operating on my hand, so the forearm was the most appropriate choice."

Abdomen. The statistics of attempts for the abdomen in different postures show a large variance, 2 when sitting and 8 when lying down, indicating that the abdomen may be especially suitable for the lying posture. As one participant said, "it was really convenient and interesting to write on the abdomen while lying down, while in other postures, it would be a little awkward."

Thigh. Similarly, the high variance in different postures indicates that the thigh is dedicated to the sitting posture. Some participants said their hands rested naturally on their thighs when they sat. This natural posture endows the thigh as a fine choice for handwriting in a sitting posture.

Palm and Lower Leg. We consider them outliers because in each pose, at most one participant tried them, and expressed it was inconvenient to interact in these positions. They complained that the lower leg was too far away, and it was weird to wear a Velcro on their palm, which would affect the activity of their hands.

4.1.4 Findings. As aforementioned, we observed that i) the forearm is the most popular interaction position in all postures; ii) the thigh and abdomen are special-purpose positions for the sitting and lying down postures, respectively; iii) the upper arm could be a hidden gem ignored by many participants; iv) the palm and lower leg are unsatisfactory positions for flexible handwriting input. Accordingly, we identify four interaction positions: upper arm, forearm, abdomen, and thigh, to be further studied in the following experiments.

4.2 Evaluation of Position-Posture Correlation

To find the best interactive positions for flexible handwriting input, we conducted another experiment to evaluate *user satisfaction* and *input speed* when using the four common interaction positions (*i.e.,* upper arm, forearm, abdomen, and thigh) in four postures (*i.e., sitting, standing, walking, and lying down*).

4.2.1 Participants. We recruited 10 participants (6 male, 4 female). One was left-handed and others were right-handed. All the participants were different from the ones involved in the previous experiment (described in Sec. 4.1).

4.2.2 Task and Procedure. Prior to the experiments, we did the same preparation as described in Sec. 4.1.2. The participants were asked to repeatedly input a sequence of 36 characters (A to Z and 0 to 9) one by one using each of the four interaction positions in the four selected postures. Specifically, when the experiment began, the participants were presented with the character sequence and started writing the first character by touching the Handwriting Velcro. A no-input period of 0.4 seconds was treated as a "completion" event and the system emitted a "beep" tone. The participants could start writing the next character after hearing the sound.

Overall, each participant repeated the same character input task 16 times (4 postures × 4 interaction positions). For each task, we recorded their associated metadata (*e.g.,* participant ID, timestamps).

At the end of the experiment, we invited the participants to fill in a questionnaire and rate their satisfaction on a scale between 0 and 5. The score serves as a subjective evaluation based on their perception of comfort and ease-of-use.

4.2.3 Results. In this section, we define the "input speed" as the average number of characters entered per minute (CPM), and "satisfaction score" as the average of user ratings for each task, and get the results as follows.

Input Speed. Overall, there is no much difference in the input speed across all (interaction position, posture) pairs: all input speed values fall in a small range between 22 and 25 (Fig. 3(c)). Among them, the input speed of the forearm is slightly slower than those of the other interaction positions. We ascribe such a slow speed to the relatively small area of the forearm, which leads to a curved writing surface. In addition, the input speed of the thigh is the fastest in the sitting posture.

Satisfaction. As Fig. 3(b) shows, in all four postures, the satisfaction scores of the forearm, upper arm, and abdomen all ranged from 3.5 to 4.5 (forearm >upper arm >abdomen). Among them, the forearm ranks the highest with an average score of 4.4. Interestingly, we observed that although the thigh has the lowest satisfaction scores in standing, walking, and lying down postures, its score in the sitting posture is high as 4.16.

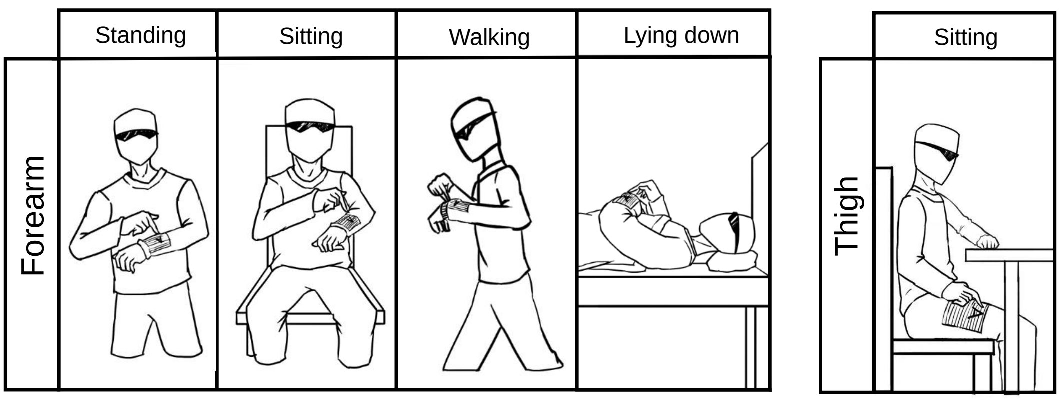


Fig. 4. The design space of Handwriting Velcro inspired by Heuristic Study.

4.2.4 Findings.

Forearm. Although being slightly slower in the input speed, the forearm shows a good advantage over the other interaction positions in user satisfaction scores in all postures.

Thigh. As expected, the thigh has the lowest satisfaction scores in the standing, walking, and lying down postures due to the longer interaction distance between the thigh and the hand in these postures. However, when it comes to the sitting posture, the thigh becomes a favorable interaction position, which enables fast handwriting input due to its large and flat area within easy reach.

Upper Arm. Surprisingly, the upper arm achieves the second-highest satisfaction scores. This may originate from its stability and relatively large area. However, it was a minority choice and we received feedback from some participants complaining that its high position caused fatigue in the writing process.

Abdomen. As for the abdomen, although its input speed is slightly higher than those of the other interaction positions in the walking and lying down postures, this advantage does not hold in the sitting and standing postures. In addition, its satisfaction scores are mediocre in all postures.

To sum up, the forearm is the best interactive position for *standing, walking, and lying-down*; the thigh is the best interactive position for *sitting*. Thus, these two positions are selected for our design space (as shown in Fig. 4) and used for the following experiments.

5 METHODOLOGY

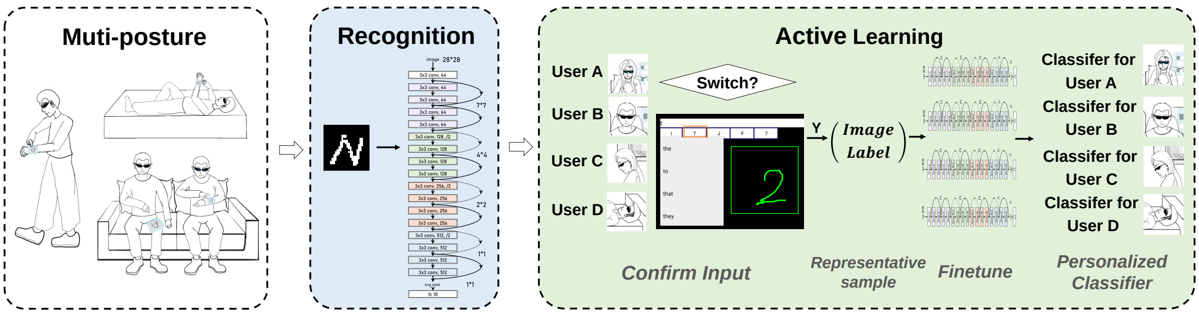


Fig. 5. Handwriting input processing pipeline, endowing personalized and posture-adaptive input for AR glasses. (Left: Multi-posture handwriting input event monitoring (sitting, standing, lying down, and walking); Middle: Handwritten character recognition using a variant of the ResNet18 [23, 36] deep residual CNN; Right: User-centered active learning for personalization. As Sec. 5.3 described, we obtain "representative" samples during users' *SWITCH* process, so as to iteratively fine-tune our model to match the user's writing habits.)

5.1 Handwriting Input Event Monitoring

We monitor the continuous data stream collected frame by frame from the sensor in our Handwriting Velcro, and represent each frame of data as a 40×40 matrix of pressure values. For each data matrix, we eliminate outliers and noise by thresholding valid pressure values to a range of [250, 4095] and denote the elements with pressure values in-between as "effective pressing points". The position coordinates and pressure values of all "effective pressing points" are recorded in real-time as an "effective trace" and wirelessly transmitted to the AR glasses. The "effective trace" data is then rendered as real-time handwriting animation on the AR glasses display interface, providing users with direct visual feedback. Note that the real-time visual feedback of our system enables "*head-up*" handwriting input.

5.2 Handwritten Character Recognition

Data Preprocessing. We first convert a complete sequence of "effective pressing points" as an image to represent the character. A set of "effective pressing points" is constructed between their first appearance and the completion sign (*i.e.,* 0.4 seconds of data without a single "effective pressing point"). Specifically, we first identify the coordinates of the point with the highest pressure value for each matrix in the data sequence. Then, a "drawing trajectory" can be created by connecting the points with these coordinates one by one in a 2D plane. Note that we cut the trajectory into two separate ones if the coordinate distance of two successive points is too large. For example, the lower right point and the upper right point of character 'X' are not connected. Then, we employ the Bresenham connection method to convert the trajectories into a rasterized grayscale image. Finally, we remove small noises in the resulting image using an opening operation in morphology, and connect broken parts in the image using a closing operation in morphology. In this way, we can convert each data sequence into an image

x_i. Together with its corresponding character label y_i, we can create our dataset $D = \{x_i, y_i\}_{i=1}^{n}$, where n is the number of samples in the dataset.

To reduce data variance and improve training, we eliminate the effect of different pressures among users by binarising all images in our dataset D. This ensures that only valid character structure information is retained for input to the recognition neural network in Sec. 5.2.

Data Augmentation. To reduce overfitting and improve generalization, we perform data augmentation on our training dataset. This helps to increase the diversity of writing styles covered during training and thus reduces the potential domain gaps between the training and testing data. Specifically, we augment our training dataset with a series of data augmentation techniques [32], including:

- Zooming, which randomly scales an image with a scaling factor between 0.6 and 1.1 on each independent axis;
- Translation, which randomly translates an image by a value between -20% and +20% on the x- and y-axis independently;
- Rotation, which randomly rotates an image in $(-20°, 20°)$;
- Shearing, which randomly shears an image in $(-16°, 16°)$.

These techniques are superimposed and applied to each image in random order. Note that we did not apply augmentation techniques that disrupt the orientation or structure of character images and cause confusion (e.g., image flipping).

Model Architecture and Training. We formulate our handwritten character recognition task as a multiclass (i.e., 36 classes, including A-Z and 0-9) classification problem. Note that we did not differentiate between upper and lower cases of English letters, but instead provided flexible case selection in an Android text input application. Users can switch between upper and lower cases freely on the AR glasses. Additionally, given the high similarity between the two pairs (0/O, 2/Z), we decided to ignore the incorrect predictions in such conditions in the subsequent experiments of accuracy evaluation. Users are offered the choice to select the correct label when needed.

Accordingly, we build a variant of the ResNet18 [23, 36] deep residual CNN by adjusting its input layer to match our input image size and its output layer to match our output dimension of 36. Then, we can train our ResNet18 model in a supervised manner using our dataset D. We implement this model in Python using the *keras* framework. [1]

5.3 User-centered Active Learning for Personalization

To further improve the recognition accuracy and smooth the user's learning curve, we propose a novel personalization method that adapts the character recognition classifier to each single user. Specifically, we borrow the idea of "human-in-the-loop" and apply active learning [26] to iteratively fine-tune a pre-trained ResNet18 model based on a specific user's real-time feedback when using our Handwriting Velcro.

Interactive Input Modes. To obtain the user's real-time feedback and realize the interactive active learning process, we design 3 *input modes* on the text input app for AR glasses:

- *CHAR.* Default mode. In this mode, the user is free to do handwritten text input on Handwriting Velcro.
- *SWITCH.* Alternative mode. Mimicking the recommendation feature of English grammar checkers, we provide the user with the top-5 candidate characters based on their classification confidence scores after each input. If the top-1 candidate is incorrect, the user can correct it by selecting the correct character

[1] All the source code, dataset, and the trained model will be released to the public upon the acceptance of this paper.

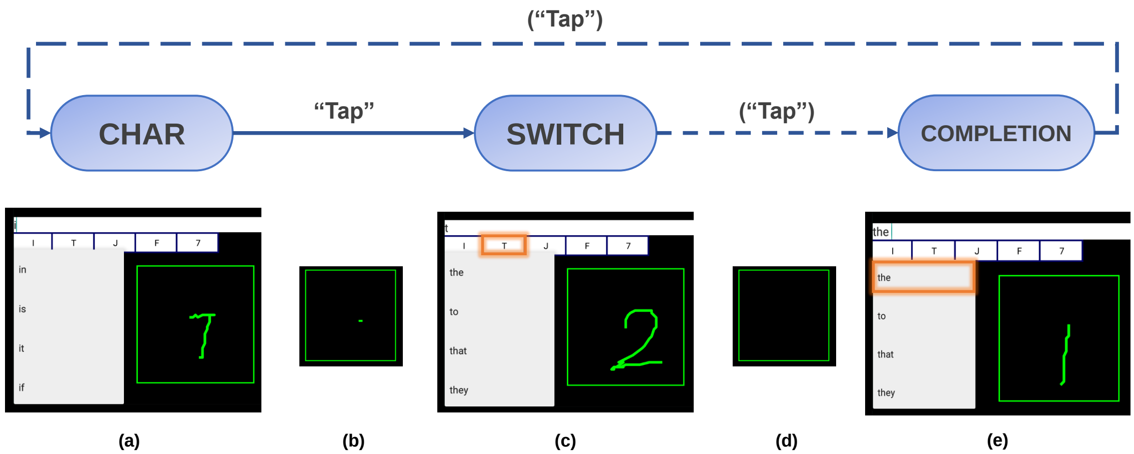


Fig. 6. User enters the word "The" using three input modes.

from the rest candidates. The selection is implemented by handwriting the serial number (1-5) of the corresponding correct candidate character on our Handwriting Velcro.

- *COMPLETION.* Completion mode. To improve the performance of our Handwriting Velcro in actual use, we also design an auto-completion function similar to an existing IME (input method editor). Users can select a complete word from the candidate words list, by handwriting the corresponding serial number (1-4) of the word on our Handwriting Velcro.

The switching between different modes is achieved by a short *tap*, and the mode switching sequence is: CHAR ->SWITCH->COMPLETION->CHAR. In addition, after the user performs the corresponding selection operation in the SWITCH or COMPLETION mode, it will automatically switch to the next mode.

Fig. 6 shows the switching flow of the three input modes and an example of a user entering the word "The" using the three input modes. (a) CHAR mode. A recognition error occurs when the user handwrites "T": the top-1 candidate is "I". (b) Tap. Mode switching: CHAR -> SWITCH. (c) SWITCH mode. The user handwrites "2" to correct the "I" in the text box to "T". (d) Automatic Mode switching: SWITCH -> COMPLETION. The dotted line indicates that if the corresponding switching/completion operation is performed in the previous mode, this mode switching does not need to be performed manually, and the program will automatically switch to the next mode. (e) COMPLETION mode. The user handwrites "1" to autocomplete the word "The".

In addition, we design a "backspace" key to allow free modification, which is achieved by two consecutive Taps.

Active Learning Pipeline. Recognizing the fact that not all feedback is equally useful, we propose a querying strategy that only extracts a small number of "representative" samples from the continuous data stream collected by our Handwriting Velcro to fine-tune the model. We implement our querying strategy by defining such "representative" samples as those with incorrect predictions:

- First, as mentioned above, the user can correct the input character by selecting the correct character from the rest candidates in the SWITCH mode.
- Then, we record the error image and its correct label (*i.e.,* the character selected by the user) when the user corrects an input character. These $(image, label)$ pairs construct the "representative" samples.
- Finally, we create a small dataset with these "representative" samples to iteratively fine-tune our model to match the user's writing habits.

Overall, a longer usage produces more "representative" samples and leads to higher recognition accuracy with a more personalized model. Please see Fig. 5 for an illustration of our user-centered active learning pipeline.

Remarks. As an alternative solution to our user-centered active learning, one can also collect a personalized dataset and train a recognition model for each user respectively. However, this requires a long and tedious data collection process, which is undesirable. In contrast, our human-in-the-loop strategy amortizes the data collection cost to the fragmented uses of our device, and our solution is thus more user-friendly.

6 EXPERIMENT

6.1 Experimental Setup

Datasets. We used two datasets in our experiment:

- EMNIST dataset [12], which contains 533,933 images (in the category of "By_Class") of the same 36 types of characters as those used in our task. Note that training our recognition neural network on EMNIST alone yields poor testing performance (68.81% accuracy) on our data, due to the significant style differences between EMNIST and our user data.
- Our dataset. Based on the conclusions of our heuristic study, we recruited two participants (one male and one female, both right-handed) and collected their handwriting data for two interaction positions (*i.e.*, the forearm and the thigh) in the sitting posture. We asked them to repeatedly write each of the 36 characters for 100 times and obtained 2 people $\times$ 2 positions $\times$ 36 characters $\times$ 100 = 14,400 data points. We further doubled the number of data points in our dataset to 28,800 using the data augmentation techniques described in Sec. 5.3.

Training Details. We trained our character classifier in two stages: first, we trained the model on the EMNIST dataset with a batch size of 128 for 20 epochs; second, we continued training the model on our dataset (after data augmentation) with a batch size of 32 for another 20 epochs. The model trained on EMNIST serves as a warm-up model for the subsequent training task on our dataset. We used the Adam optimizer [35] to train our neural network, and set its initial learning rate to 0.001, $\epsilon = 10^{-8}$, $\beta = (0.9, 0.009)$.

6.2 Performance Evaluation of Handwriting Velcros

In this section, we conducted three experiments to evaluate the performance of our Handwriting Velcro among i) different users, ii) different interactive positions, and iii) different postures. Each experiment was a within-subject design. Noting that in this experiment: i) We did not conduct the active learning (*SWITCH* mode), which will be further explored in Sec. 6.3. ii) *COMPLETION* mode was also not invoked since this experiment was conducted at the character level.

6.2.1 Participants. Overall, we recruited 32 participants (16 male, 16 female; 2 left-handed) in the three experiments. All the participants were different from the ones involved in the previous experiment (described in Sec. 4). Gender-balance is guaranteed in all the experiments: i) 10 users (5 male, 5 female) participated in our "Performance vs. Users" experiment; ii) 10 users (5 male, 5 female; 1 left-handed) participated in our "Performance vs. Interaction Positions" experiment; iii) 12 users (6 male, 6 female; 1 left-handed) participated in our "Performance vs. Postures" experiment. The participants in all three experiments are mutually exclusive.

6.2.2 Task and Procedure. Based on our heuristic study (Sec. 4), we designed the three experiments as follows:

- *Performance vs. Users.* **10 users** $\times$ 1 "Forearm" position $\times$ 1 "Sitting" posture = 10 task units.
- *Performance vs. Interaction Positions.* 10 users $\times$ **2 interactive positions (Forearm and Thigh)** $\times$ 1 "Sitting" posture = 20 task units.

- *Performance vs. Postures.* 12 users × 1 "Forearm" position × **4 postures (Sitting, Standing, Walking and Lying down)** = 48 task units.

All three experiments are within-subjects designs. To ensure counterbalance, in our "Performance vs. Interaction Positions" experiment we divided the participants into two groups: A (Forearm first, then thigh) and B (Thigh first, then forearm), according to the order of experiments. In our "Performance vs. Postures" experiment, we constructed a Latin square to divide the participants into 4 groups.

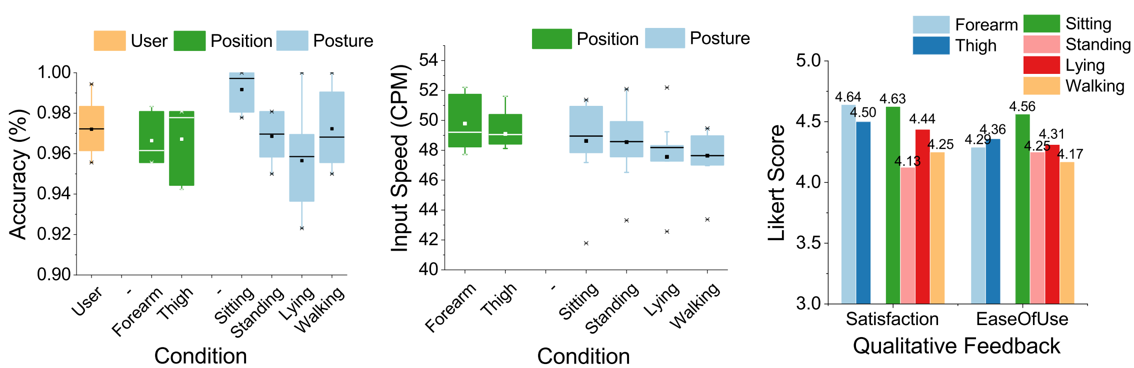
In each task unit (*i.e.,* 1 user × 1 interactive position × 1 posture), we instructed each participant to repeatedly enter each of the 36 characters (A-Z, 0-9) 5 times. This experiment contains a total of 78 task units (Performance vs. Users: 10 task units, Performance vs. Interaction Positions: 20 task units, and Performance vs. Postures: 48 task units).

Prior to the experiments, we did the same preparation as described in Sec. 4.1.2. During the experiments, we collected sensor data of all 78 task units along with their metadata (*e.g.,* participant ID and timestamps), collected 78 task units × 36 characters × 5 repetitions = 14,040 pieces of data. In this way, we constructed the total test set D_{test}, including 3 sub-test sets, namely: $D_{test} = D_{user} \bigcup D_{position} \bigcup D_{posture}$.

At the end of the experiments, we asked the participants for feedback about their overall satisfaction and ease of use. Questionnaire responses were on a five-point Likert scale: the higher, the better. See Fig. 7(c) for the statistics of the results.

6.2.3 Results and Discussions.

Performance vs. Users. As shown in Fig. 7(a), the average recognition accuracy of D_{user} is 97.21% (sd = 1.36×10^{-2}). The highest accuracy among all the participants was as high as 99.44%, and even the lowest one reached 95.56%. We believe that the relatively low accuracy in some cases is rooted in the different writing styles of specific users. This implies the demand for personalized classifiers in order to adapt to different writing styles. This will be further addressed in Sec. 6.3.



(a) (b) (c)

Fig. 7. Results of performance evaluation experiment. (a) Accuracy (%). (b) Input speed (CPM). (c) User feedback.

Performance vs. Interaction Positions. Both forearm and thigh average accuracy of $D_{position}$ are above 96% (Fig. 7(a)), indicating good performance for both interaction positions. In terms of input speed, both forearm ($mean = 49.45CPM$, $sd = 1.71$) and thigh ($mean = 48.74CPM$, $sd = 1.67$) reached close to 50 CPM (Fig. 7(b)).

Regarding satisfaction and ease-of-use, all the scores are above 4 (Fig. 7(c)), indicating that our Handwriting Velcro performed well and stably for both interaction positions. In addition, we performed the Levin test for homogeneity of variances on each performance of the forearm and thigh positions to confirm that the variances between the data of each performance at the two positions were homogeneous. Then we used one-way ANOVA to test the significance of each performance at the positions of the forearm and thigh. The results show that the interaction position has no significant effect on the recognition accuracy ($F = 0.011$, $p = 0.920$), input speed ($F = 0.620$, $p = 0.446$), user satisfaction ($F = 0.300$, $p = 0.594$), and ease-of-use ($F = 0.120$, $p = 0.735$). To conclude, the two interaction positions of the forearm and thigh both performed well in the above four performances.

Performance vs. Postures. As Fig. 7(a) shows, the recognition accuracy of the sitting pose is about 99%, and those for the other three postures are above 95%. We visually inspected the handwriting data for all postures and observed that the characters written in the standing, walking, and lying down postures were slightly more scribbled and irregular than those written in the sitting posture. This may account for the small performance drop. These are also reflected in the satisfaction scores, ease-of-use scores, and input speed. In addition, we performed the Levin test and used one-way ANOVA to test the significance of recognition accuracy ($F = 5.042$, $p = 0.006$), input speed ($F = 0.308$, $p = 0.819$), user satisfaction ($F = 1.657$, $p = 0.199$), and ease-of-use ($F = 1.072$, $p = 0.377$) in the different postures of sitting, standing, walking and lying down. For different postures, the variance of each performance is not significant ($p > 0.05$) except for accuracy, indicating that our Handwritten Velcro adapted well to different postures.

Ablation Study on Data Augmentation. As discussed in Sec. 5.2, we applied data augmentation to reduce overfitting and help the trained recognition model to generalize to testing data that are "unseen" during training. To verify its effectiveness, we used D_{test} (consisting of all 14,040 pieces of data) to evaluate the performance of our recognition model with and without data augmentation:

- When data augmentation is not applied, although the training and validation accuracy of the trained model achieve 99.95% and 98.70%, respectively, the testing accuracy is only 84.60%, indicating that the model overfits to the training dataset and generalizes poorly to testing data.
- When data augmentation is applied, the testing accuracy increases by more than 10% and achieves 97.21%, verifying the effectiveness of data augmentation.

Fig. 8 shows the confusion matrix of the prediction results when applying our recognition model (with data augmentation) to the D_{test} dataset.

6.3 Evaluation of Active Learning-based Personalization

As discussed above, although our character recognition model generally performs well, its performance may be worse if a user's writing style deviates too much from those in the training dataset. To address this issue, we propose to personalize our model by adapting it to a single user in a "human-in-the-loop" manner (Sec. 5.3). Empirically, we verify the effectiveness of this approach as follows. Note that during this experiment, the users were free to use our different input modes (as described in Sec. 5.3) for text input, character correction, and word completion.

6.3.1 Participants. We recruited 10 participants (5 male, 5 female; all right-handed) to participate in our personalization experiment. All of them were novices and did not participate in the previous experiments (described in Sec. 4 and Sec. 6.2).

6.3.2 Task and Procedure. To simulate the actual use process, we asked each user to repeat a phrase repetition task: complete 7 sessions, each involving repeated input of a 20-word phrase. The design and vocabulary were taken from previous work [8, 39]. The phrase was *"the and you that is in of know not they get have were are bit*

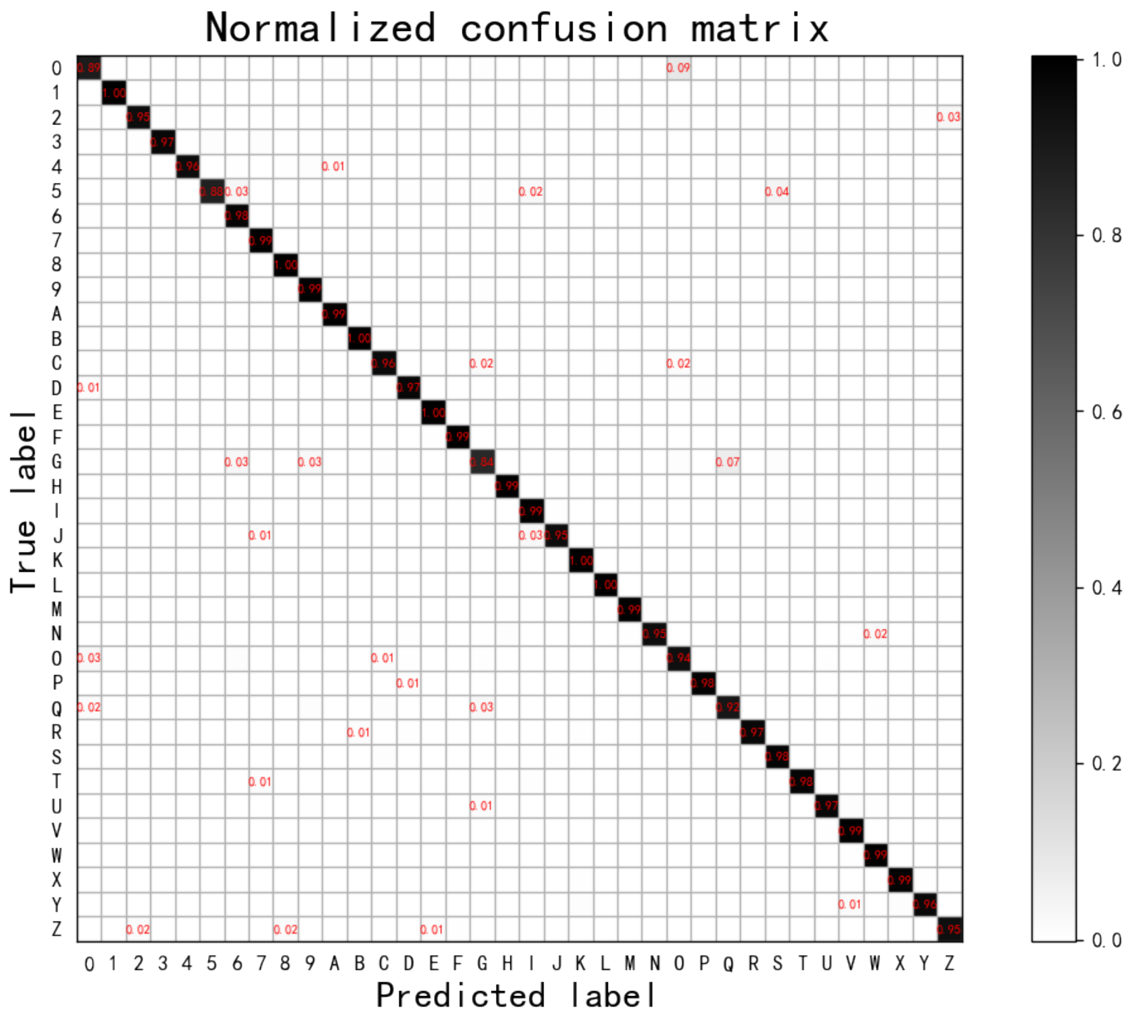


Fig. 8. Confusion matrix of our trained model on the D_{test} dataset.

quick fox jumps lazy on" (66 char). It contains all English letters and approximates monograms and bigrams. The experiment was conducted in a representative task unit: "Forearm" position in "Sitting" posture. The first repetition was discarded as an exercise. Prior to the experiment, we followed the same preparation as described in Sec. 4.1.2.

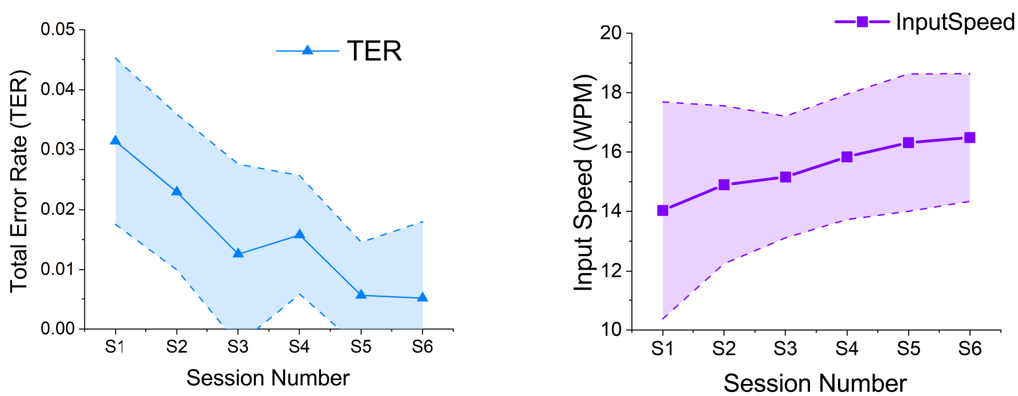
As described in Sec. 5.3, "representative" samples naturally generated in the SWITCH mode weresequentially added to the representative dataset $D_{representative}$. Utilizing $D_{representative}$, our model was iteratively fine-tuned to generate a personalized classifier adapted to the current user's writing habits.

At the end of the experiment, we asked the participants to fill in a 20-point Likert scale for feedback about "Performance", "Comfort", "Effort (ease-of-learning)", "Frustration", and "Ease-of-use". Higher scores are better.

In the end, we invited the participants to fill in a questionnaire, which addressed social acceptance in different locations and in front of different audiences as proposed by Rico [58] and Hsieh [28].

In order to measure the subjective feeling of social acceptance[28] when using the system in a public place, and further strengthen the validity of the collected responses, the experimental site was chosen in a relatively open public space: a coffee restaurant, surrounded by corridors, office areas, rest areas, toilet, etc.

6.3.3 Experimental Setup. We set the batch size=32. Whenever the size of $D_{representative} > batchsize$, we iteratively fine-tuned our model with $D_{representative}$ for 5 epochs. Specifically, we froze the first 60 layers of the model and only fine-tuned the neural network weights of the last 7 layers. Note that we optimized the model using an Adam optimizer with a small learning rate of 0.0001 (10 times smaller than the one used in model training), $\epsilon = 10^{-8}$, $\beta = (0.9, 0.009)$.



(a) (b)

Fig. 9. Results of active learning experiment in different sessions. (a) Total Error Rate (TER). (b) Input speed (WPM).

6.3.4 Quantitative Analysis. After the experiment, we recorded the user's input error rate and input speed in each session for statistical analysis and discussion. Four participants show a quite low error rate (≤ 0.010) in S1, which indicates that the generalization classifier already has a good recognition performance for them. In order to verify the improvement of the recognition performance of the personalized classifier for the users with high initial error rates, we analyzed the data of the remaining six users (the error rate of S1 >0.010).

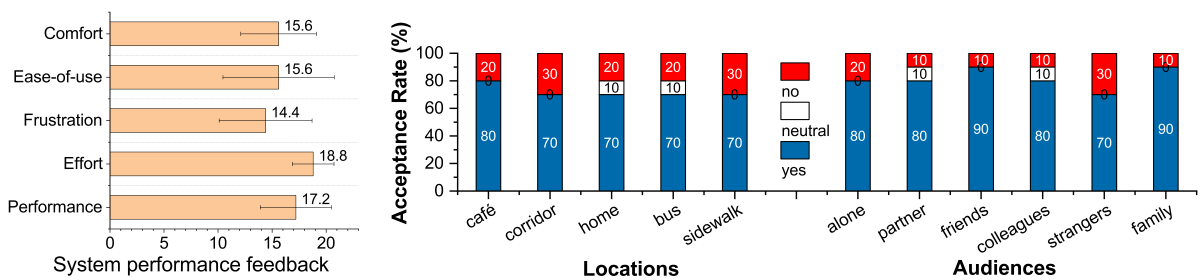
Error Rate. We distinguish three different degrees of error:

- *Corrected Errors (CE):* A recognition error resulted in the entry of a wrong character, but was corrected by our SWITCH mode.
- *Uncorrected Errors (UE):* A recognition error resulted in the entry of a wrong character, which was not corrected.
- *Total Errors (TE):* The sum of Corrected Errors (CE) and Uncorrected Errors (UE).

We observed that all participants consciously corrected all errors without being reminded, making UE always 0 (TE = CE). Fig. 9(a) shows the variation trend of the total error rate for multiple sessions in the experiment. From S1 to S6, the TER of each user decreases by 0.026 on average (from 0.010 to 0.048). After 6 sessions, the TER of $D_{personalized}$ finally reaches 0.005. ($D_{personalized} = \sum_{i=1}^{6} D_{personalized,i}$). Paired samples T-test shows that TER of S6 (mean=0.005, sd=0.013) is significantly lower (p=0.009) than that of S1 (mean=0.031, sd=0.014). The above results show that with the help of active learning, the recognition performance of the personalized classifiers for each user improve significantly (p=0.009) over usage time. Our personalized classifier can adapt to the writing style of different users, thus smoothing the users' learning curve and improving the use experience.

Input Speed. As Fig. 9(b) shows, the user's text input speed improves with sessions. Particularly, from S1 to S6, the input speed of the 10 participants increased by an average of 2.45 WPM. This shows that our Handwriting Velcro has a very friendly learning curve for "novices", and after a few simple repetitions of sessions, the participants could achieve more proficient input. The average text input speed of each participant finally reaches 16.49 WPM after 6 sessions. One participant said the SWITCH and COMPLETION modes improved input efficiency.

Remark. Note that in practice, instead of collecting data in advance, we amortize the data collection and training into users' daily use with little additional cost. This design makes our method user-friendly.



(a)

(b)

Fig. 10. Subjective feedback of evaluation experiment. (a) System performance feedback (20-point Likert scale, higher scores are better). (b) Social acceptance in different locations and in front of different audiences.

6.3.5 Subjective Feedback.

System Performance Feedback. Fig. 10(a) shows various subjective scoring results of system performance. Higher scores are better. For "Performance" ($mean$ = 17.2, sd = 3.29), "Comfort" ($mean$ = 15.6, sd = 3.50), "Effort" ($mean$ = 18.8, sd = 1.93) and "Ease-of-use" ($mean$ = 15.6, sd = 5.15), all received high ratings above 15 points. This shows that our system has usability and comfort while being easy to learn. "Frustration" ($mean$ = 14.4, sd = 4.30) is slightly lower, which may be caused by limitations such as the *linearity* of the current hardware, which we will further improve in future research.

Social Acceptance. The measuring of social acceptance follows the method proposed by Rico [58] and Hsieh [28]. Each question provides three answer options: "Yes, I would use the system in that situation", "No, I wouldn't use the system in that situation," and "I don't know". The percentage of positive, negative, and uncertain responses with respect to each situation can be seen in Fig. 10(b).

Overall, the participants expressed positive attitudes toward the possibility of using Handwriting Velcro in both private and public places. 80% of the respondents will use Velcro in cafes, and 70% will use it at home, corridor, bus, and sidewalks respectively. Some participants said that the reason they were less inclined to use Velcro on sidewalks was due to the bumps in the process of vehicle driving and the safety of the sidewalk, rather than social acceptance. The above results show that the respondents have a positive attitude towards using the system at work and in private situations (on average, yes: 72%, no: 24%). The "Audiences" survey shows that only 15% of the participants explicitly oppose the use of the system in front of others, and most of them show a positive attitude towards using the system in front of people with different levels of intimacy (on average, yes: 81.67%, neutral: 3.33%). This shows that the system has a quite high acceptance for different groups. Among them, the participants' use attitude when using alone or facing acquaintances (yes: 85%, no: 12.5%) is more positive than that of colleagues or strangers (yes: 75%, no: 20%).

6.3.6 Discussions. In general, Handwriting Velcro exhibits practical, reliable, and friendly performance in both quantitative and qualitative analysis of the experimental results.

After 6 sessions, the TER of each user decreases by 0.026 on average and finally reaches 0.005. The results of Paired samples T-test show that with the help of active learning, the recognition performance of the personalized classifier for a single user improves significantly (p=0.009). The average text input speed of each participant finally reaches 16.49 WPM in S6, which indicates our system has practical input efficiency.

In the meanwhile, Handwriting Velcro has received highly positive reviews from the users' subjective feedback. The first is the system performance: the results of the 20-point Likert scale show that the system is practical and comfortable. Second, findings on social acceptance reveal that the participants express positive attitudes towards the possibility of using Handwriting Velcro in both private and public places. Moreover, only 15% of the participants explicitly object to using the system in front of others with different levels of intimacy, and the system has a broad social acceptance.

7 COMPARISON WITH EXISTING BASELINES

We compared our Handwriting Velcro with two baselines selected from the literature and commercial products respectively:

- *TEXTile:* The wearable touch text input work proposed by Belkacem et al [6] provides a text input technology for smart glasses based on eight touched or released finger combinations. We chose TEXTile as the baseline due to its highly similar method to our work: similar wearable flexible sensing media, similar touch text input approaches, and similar mobile input scenarios. However, its interaction is restricted to the forearm, whereas our Handwriting Velcro allows for handwriting interaction in multi-position/posture.
- *Physical Mini QWERTY Keyboard:* As a naive solution using existing commercial products, we replace our Handwriting Velcro with a Physical Mini QWERTY Keyboard ($114mm \times 60mm \times 9mm$, Bluetooth) attached to the user's body.

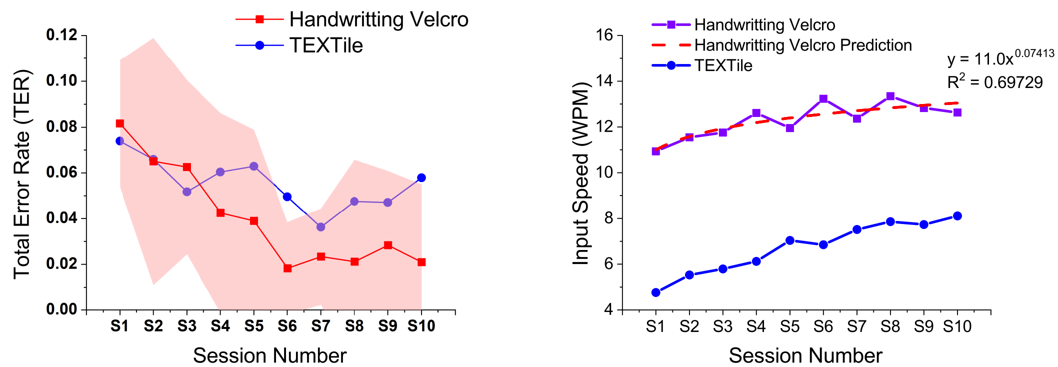
For TEXTile, we obtained its performance data metrics from [6]. To make a fair comparison, we follow [6] and restrict the use of our Handwriting Velcro to the *"forearm"* position and *"standing"* posture. For the Physical Mini QWERTY Keyboard, we perform a stress test in which we compare it with our Handwriting Velcro on four common *challenging contexts*: obstacle walking, dim, one-hand, and hidden, which has been shown to be a key factor in the ultimate viability of any wearable text input technology [40]. To make a fair comparison, the active learning module (*SWITCH* mode) of our Handwriting Velcro is disabled. However, users are allowed to use the "backspace" key and *COMPLETION* mode to assist with handwriting.

7.1 Comparison with TEXTile

7.1.1 Context Setup. To ensure the consistency of experimental conditions, we restrict the interaction position to "forearm" and the interaction posture to "standing" in this experiment.

7.1.2 Participants. We recruited 10 participants (8 male, 2 female, all right-handed) to participate in this comparison. All the participants were novices and had not participated in our previous studies.

7.1.3 Task and Procedure. We asked each user to repeat a phrase repetition task: complete 10 sessions, each involving repeated input of 4 phrases (total 22 words, 97 characters) randomly selected from MacKenzie & Soukoreff phrase set [46]. Prior to the experiment, we followed the same preparation as described in Sec. 4.1.2. During the experiment, we encouraged the participants to write as "fast" and "accurately" as possible, and recorded the input speed and error rate of the current session when each session was completed.



(a) (b)

Fig. 11. Results of comparison with TEXTile. (a) Error rate comparison between Handwriting Velcro and TEXTile. (b) Input speed (WPM) comparison between Handwriting Velcro and TEXTile.

7.1.4 Results.

Error Rate. We define three types of errors in the same way as in Sec. 6.3.4: Corrected Error Rate (CER), Uncorrected Error Rate (UER), and Total Error Rate (TER). Note that in the comparison experiment, error correction was done through the "backspace" key (two "Taps") similar to TEXTile. The error rates of the two methods are shown in Fig. 11(a). We observed that i) all the participants consciously corrected all errors without being reminded, making UER always 0 (TER = CER); ii) TER shows a decreasing trend with sessions, and achieves higher accuracy than TEXTile from S4 onwards; iii) over the 10 sessions, our Handwriting Velcro achieved an average TER of 4.02% ($sd = 2.2 \times 10^{-2}$), outperforming TEXTile (mean = 5.53%) by 1.51%.

Input Speed. As Fig. 11(b) shows, Handwriting Velcro's input speed increases with sessions, reaching a maximum of 13.34 WPM in S8. The rate increases according to the Power Law of Learning, fitting to the curves at $R^2 = 0.697$ (Fig. 11(b)). Over the 10 sessions, Handwriting Velcro's average input speed was 12.32 WPM (sd=0.77), higher than TEXTile's average input speed of 6.73 WPM (sd=1.13). To verify whether there is a significant difference between the input speeds of the two solutions, we performed the Levin test and one-way ANOVA. The results indicate that our Handwriting Velcro is significantly faster (p=0.00) than TEXTile in terms of input speed. This shows that our system outperforms in input efficiency.

7.1.5 Discussions. In terms of recognition accuracy, Handwriting Velcro achieves an average error rate of 4.02% ($sd = 2.2 \times 10^{-2}$) in 10 sessions, which is 1.51% lower than TEXTile (mean = 5.53%). The average input speed of Handwriting Velcro in 10 sessions (mean=12.32 WPM, sd=0.77) is significantly higher (p=0.00) than TEXTile (mean=6.73 WPM, sd=1.13). Such performance is sufficient for short text input tasks. In the meanwhile, compared with TEXTile, this system has a smoother and more friendly learning curve: users can directly use traditional handwriting input instead of additionally learning complex finger combinations, and the use fatigue is also reduced.

7.2 Comparison with Physical Mini QWERTY Keyboard

7.2.1 Context Setup. To demonstrate the superiority and robustness of our Handwriting Velcro in various application scenarios, we conduct a stress test in which we compare the performance of the Physical Mini QWERTY Keyboard and our Handwriting Velcro in four common but challenging contexts:

- *Obstacle Walking.* Users were asked to walk at a normal walking speed (about $1m/s$) around a path of about $50m \times 20m$, with 1 to 2 obstacles set in the way, and complete the input task while walking.
- *Dim.* To simulate input scenarios of no light or traveling at night, we asked users to complete the input task in low visibility. Note that in this case, the characters on the keys of the mini keyboard cannot be clearly seen in the *"sitting"* posture.
- *One-hand.* Users were asked to complete text input with a tote bag in their left hand to simulate a one-handed input scenario.
- *Hidden.* Since hidden input protects the privacy and is more socially acceptable [28], we asked users to complete input tasks in the "thigh" position and "sitting" posture (at a desk) to simulate a hidden input scenario. The users were asked to use only peripheral vision for the input task.

The *forearm* was chosen as the default interaction location for all contexts, except for the hidden context where the device was attached to the *thigh*. As described in Sec. 3, the AR glasses used in this study do not block the user's vision, which will allow users to see the keyboard through the glasses clearly. In addition, there were no other visual obstructions other than those due to the experimental scenarios.

7.2.2 Participants. We recruited 12 participants for this experiment, all of whom were novices and had not participated in any of our previous studies.

7.2.3 Task and Procedure. Participants were randomly divided into four groups to participate in the studies in four contexts respectively. The study in each context used a within-participant design with Device (our Handwriting Velcro, physical mini QWERTY keyboard) and Session (1-10) as independent variables.

Same as Sec. 7.1, we asked each user to repeat a phrase input task using both devices in the corresponding context. The phrase to be entered was displayed on the interface of the AR glasses. We collected 1 context $\times$ 2 devices $\times$ 10 sessions $\times$ 22 words = 440 words/person for a total of 5280 acquisitions. To ensure that we are testing the performance of novices, we introduced the users how to use the device before the start of the experiment, with no additional practice. Users could apply for breaks between experiments. To ensure counterbalance, we grouped users equally according to the order in which the different devices were used. When switching between devices, we ensure users are well rested to remain in a similar state to that at the start of the experiment.

We recorded two quantitative metrics: *total error rate (TER)* and *input speed (WPM)*, and collected *subjective feedback* after the experiment to analyze users' preferences for using the two devices in each context.

7.2.4 Results and Discussions.

Total Error Rate. As shown in Fig. 12(a), our Handwriting Velcro achieves a high accuracy (TER $\leq$ 5%) in all four contexts. The results of the Levin test and one-way ANOVA showed that "Context" has no significant effect on the TER (F = 1.003, p = 0.440) of our Velcro, indicating the superiority and robustness of our Handwriting Velcro against challenging contexts. It is worth noting that the "eyeless" nature of our Handwriting Velcro allows for accurate and stable text input when the line of sight is obstructed and hidden (*Dim* and *Hidden*). Specifically, our Velcro achieves low average TERs of 4.31% and 2.22% in *Dim* and *Hidden* respectively, outperforming those

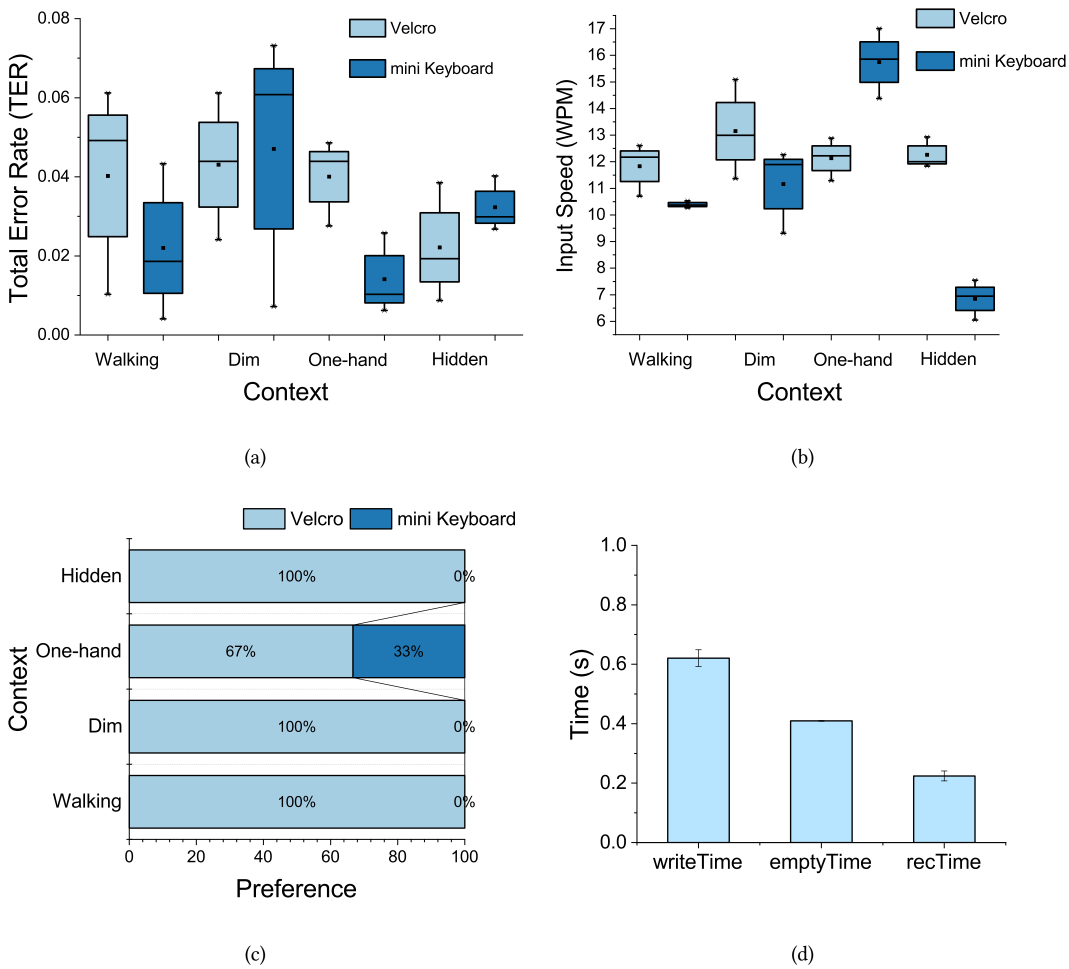


Fig. 12. (a-c) Comparison of Total Error Rate, Input speed (WPM), and User Preferences with physical mini QWERTY keyboard in different contexts. (d) Per-character response time for all participants.

(4.71% and 3.23%) of the physical mini QWERTY keyboard. This has been verified by the participants' comments, *e.g.,* "The mini keyboard is difficult to use when you can't see it clearly, especially when the keys are still small. While Velcro only needs to determine the approximate location of the device and can write without looking. The glasses screen will provide visual feedback." For *obstacle walking* and *one-hand* contexts, the physical mini keyboard is slightly more accurate due to users' established proficiency in keyboard interaction and visual aids. However, with more practice (20mins) on our Handwriting Velcro, users can achieve comparable or even lower TER.

Input Speed. Our Handwriting Velcro achieves an average input speed (12.34 WPM) higher than the physical mini keyboard (11.04 WPM) for all the participants in all the contexts over the 10 sessions. The results of the Levin test and one-way ANOVA showed that "Context" had no significant effect on the input speed (F = 0.706,

$p = 0.575$) of our Velcro as well, indicating the efficiency and robustness of our Handwriting Velcro against challenging contexts. Specifically, the input speeds of our Handwriting Velcro in the Walking (11.83WPM), Dim (13.15WPM) and Hidden (12.26WPM) contexts are faster than those of the mini keyboard in the corresponding contexts (10.39WPM, 11.16WPM and 6.85WPM, respectively). While in one-hand context, the mini keyboard achieves a higher input speed than our Handwriting Velcro. We attribute this to the good input conditions unique to the one-hand context: standing posture, sufficient light, no need to move, etc., which are difficult to meet in many complex real-world contexts. For example, suppose that a businessman is sitting in a busy and noisy bus on his way home late at night and needs to text his family privately, then the mini keyboard might not be as effective as we have demonstrated its drawbacks in *Dim* and *Hidden*. Nevertheless, our Handwriting Velcro shows good adaptability across challenging input conditions.

Response Time. The response time for interacting with our Handwriting Velcro consists of three components:

- *writeTime.* The amount of time the user actually touches the Velcro for handwriting.
- *emptyTime.* The no-input waiting time after a character is handwritten.
- *recTime.* The program processing time for handwritten character recognition and result output.

Fig. 12(d) shows the means and sds of per-character response time for all the participants in this experiment. It can be observed that: i) the writeTime (mean=0.621s) is the longest, ii) emptyTime (mean=0.409s) takes 0.4s fixedly, iii) recTime (mean=0.224s) is the shortest, about 0.2s. Thus, it can be concluded that the total response time for the input of a single character is approximately 1.2s (about 50CPM), indicating the high input efficiency of our Handwriting Velcro. Note that the emptyTime can be further reduced by choosing a more appropriate shorter fixed value, and recTime can be further reduced by using more computationally powerful devices.

User Preference. At the end of the experiment, we recorded the participants' preferences for the use of the two devices in each context. As shown in Fig. 12(c), the participants showed a very positive preference for our Handwriting Velcro. Among the 12 participants, only one expressed a preference for the mini keyboard in the one-hand context. We interviewed the participants to find out their reasons for their choices: i) Four participants mentioned that "eyeless, mobile input" is a major advantage for our Handwriting Velcro. They said that our Velcro only requires a rough location of the device for handwriting, while the keyboard requires constant visual attention. ii) Some participants complained that "Hanging the keyboard on the arm is too strange", showing the superiority of our Velcro as a flexible wearable device in comfort and social acceptance. iii) Some participants were bothered with the too-small keys on the keyboard, "sometimes two keys were pressed at the same time," they complained. iv) The only participant who chose the mini keyboard said that using the keyboard while standing was in line with their daily use habits.

8 DISCUSSIONS

8.1 Comparison with Previous Literature

As Table 1 shows, in addition to the comparison with the baselines in Sec. 7, we further demonstrate the superiority of our Handwriting Velcro to other relevant works in terms of *input speed* and *total error rate (TER).* To make the comparison more concise and informative, we focus on one-hand wearable text input for smart glasses.

Input Speed. As Sec. 7.1 shows, our Handwriting Velcro achieves a high average WPM of 12.32, which outperforms i) TEXTile (6.73 WPM), which uses similar flexible fabric for touch input; ii) Haptic Glove [28] (5.39 WPM), TipText [79] (11.9 WPM), Thumb-to-Finger [41] (5.12 WPM), and ThumbText [34] (7.5 - 9 WPM), which use different types of wearable devices (e.g., gloves, ring) for one-hand mobile text input. We attribute the faster input speed of our Handwriting Velcro to its larger interaction area and the user-friendliness of handwriting. On the other hand, FingerText [40] uses fingernails for text input, and achieves a higher average WPM of 22.38 - 31.3

Table 1. Comparison with relevant works in previous literature. "∼": values inferred from the figures and data in the original papers. "(lX)": the mean of the last X session(s) as reported in the original papers.

Method	Average WPM	Last WPM	Average TER%	Practice Time
Velcro	12.32	12.63	4.02%	20 mins
TEXTile[6]	6.73	8.11	5.53%	120 mins
Haptic Glove[28]	5.39	(l4) 5.42	5.45%	20 mins
TipText[79]	11.90	13.30	4.89%	40 mins
Thumb-to-Finger[41]	5.12	6.47	10.35%	∼ 30 mins
FingerText[40]	(l3) 22.38 - 31.3	∼ 23 - 31	(l3) ∼ 4% - 12%	90 mins
ThumbText[34]	∼ 7.5 - 9	8.46 - 11.41	∼ 10%	∼ 60 mins

than our Handwriting Velcro. However, its high input speed comes at the cost of accuracy (4% - 12% TER) and the high learning costs (90 mins of practice time) associated with its optimized nail keyboard layout. In contrast, our Handwriting Velcro is accurate (4.02% TER) and easy-to-learn (handwriting input). Note that for fairness we did not compare with two-hand input solutions with faster text input rates, such as *DigiTouch* [72] (13 WPM) and *BiTipText* [78] (23.4 WPM).

Error Rate. As Sec. 7.1 shows, our Handwriting Velcro achieves a low TER of 4.02%, which outperforms all relevant works listed in Table 1. Specifically, FingerText [40], TipText [79], Thumb-to-Finger [41] (5.12 WPM), and ThumbText [34] suffer from the limited interaction area of fingers; Haptic Glove [28] and TEXTile [6] suffer from their complicated gesture recognition. In contrast, our Handwriting Velcro offers a large interaction area to enable intuitive handwriting, which is part of human nature and easy to recognize. As a result, our Velcro allows for accurate text input (TER of 4.02%) with a low learning cost, i.e., our Velcro requires comparably shorter practice time (20 mins) than other methods in Table 1.

8.2 Multi-Posture Adaptation

We believe that multi-posture adaptation is an inevitable step towards anywhere and anytime text input for AR glasses. Specifically, a large number of AR text input scenarios (e.g., time, place, movement, etc., and their combinations) lead to diverse demands that can hardly be met by a small number of fixed input postures (i.e., sitting or walking) offered by existing solutions [22, 28, 40, 41, 70]. To meet such demands, we propose Handwriting Velcro, which is based on a flexible sensing film that can be easily attached to any positions of body surfaces (i.e., interaction positions) in a comfortable and natural way, thereby giving users complete freedom to explore the best interaction positions and postures for different AR text input scenarios. To make the most use of the flexible film, we choose the handwriting input paradigm (with a low learning cost but high speed/accuracy) and implement it with the latest deep learning technology (Sec. 5).

For evaluation purposes, we first select 6 representative interaction positions and investigate their adaptability to 4 representative interaction postures (Sec. 4). After obtaining the best interaction positions for the 4 postures (Fig. 4), we benchmark the performance of the proposed Handwriting Velcro and demonstrate its superiority with a series of experiments (Sec. 6). Finally, we show the effectiveness of our Handwriting Velcro against a similar baseline (TEXTile [6]) that also uses a flexible film, and demonstrate that it is superior to the physical mini keyboard (commercial product) in meeting the demands of various challenging contexts/scenarios for AR text input (Sec. 7). Specifically, the comparison with mini keyboard shows that "context" has no significant effect on the input speed ($F = 0.706$, $p = 0.575$) and TER ($F = 1.003$, $p = 0.440$) of the Handwriting Velcro. For all users in all contexts, our Velcro achieved an average input speed of 12.34 WPM over 10 sessions, which is faster than physical mini keyboards (11.04 WPM). In terms of error rate, our Velcro achieved an average TER of 3.64% over 10 sessions, and the results of one-way ANOVA showed that there was no significant difference in TER between the two devices (F = 0.826, p = 0.373). Subjective feedback (Fig. 12(c)) showed that 91.67% (11 out of 12) users preferred Velcro in the four contexts. Since the successful design of common situational barriers, such as obstacle walking, maybe a key factor in the ultimate viability of any wearable text input technology [40], we believe the multi-posture design space explored in this paper will benefit the community in creating more effective wearable text entry systems for diverse scenarios in the future.

In addition, thanks to the multi-posture feature of our Handwriting Velcro, it can be adapted to various challenging contexts and is thus more socially acceptable. Specifically, our Velcro allows hidden operations that are more acceptable in social settings [28]; our assessment of social acceptance (Sec. 6.3.5) indicates that users are widely positive about using Velcro in different locations and in front of different audiences; our comparative study (Sec. 7.2) about hidden context shows that our Velcro allows significantly faster (12.26 WPM, F = 95.91, p = 0.001) and more accurate (TER of 2.22%) hidden input operations than the mini keyboard (6.85 WPM, TER of 3.23%).

8.3 User-centered Personalization

Based on the previous discussion (Sec. 8.2), we choose the handwriting input paradigm (low learning cost, high speed/accuracy, and suitable for the flexible film). However, due to the inherent differences in handwriting styles among different users, it is difficult for generalized classifiers to achieve high accuracy for different users (Sec. 6.2). Therefore, we believe that user-centered personalized handwritten character classification is indispensable. We borrow the idea of "human-in-the-loop" and implement the personalized classifiers through active learning (Sec. 5.3). After experimental evaluation, our personalized classifier eventually reached a low TER of 0.005 among different users after 6 sessions (Sec. 6.3), which outperforms the performance of the generalized classifier not calling the active learning module (*SWITCH* mode) in the same situation (“forearm" position, “sitting" posture) in the comparative experiment (as shown in Sec. 7.1, 0.021 of TER after 10 sessions). The lower error rate also reduces the time required for error correction, resulting in faster input speed (15.46WPM vs 12.32WPM, on average). The benefits of user-centered personalization are also reflected in the user's subjective feedback on system performance and social acceptance (Sec. 6.3.5).

8.4 Limitations and Future Work

Limited Sample Sizes. We acknowledge that the sample size in our user studies is relatively small. For example, in the personalization experiment, although 10 participants were recruited in total, only 6 of them had eligible data for subsequent analysis, i.e., their initial error rate is large (TER>0.01) in the S1 experiment (Sec. 6.3.4) to allow sufficient room for improvement in personalization. Although all six users show improvement in terms of both reduced error and increased input speed for all sessions, their statistics show minor fluctuations (0.015+/-0.005 for average TER and 15.46+/-4.09WPM for average speed). Therefore, despite their effectiveness in justifying our major claims, we expect that the reported statistics may demonstrate some variation if using a different sample size.

Hardware Improvement. Currently, our device is connected to a desktop computer via USB, limiting its mobility. In the future, we will switch from a wired cable to wireless transmission methods such as Bluetooth. Additional hardware upgrade, such as reducing the size of the circuit, is also critical to achieving a more affordable and wearable experience.

Exploring More Interaction Positions. The current study carefully evaluated the forearm and the thigh as the two interaction positions. However, as we mentioned in the heuristic study, the upper arm and the abdomen are also promising interactive positions. They may show unique advantages or user preferences in certain interaction scenarios. In the future, we will explore more interaction positions, including the upper arm and the abdomen.

Advanced Text Input and Editing Functions. In addition to basic characters (*e.g.*, A-Z, 0-9), modern text input methods are required to support the input of special characters (*e.g.*, punctuation) as well as basic editing functions such as selection and modification. These can be achieved by increasing the types of recognized characters or combining with other input modalities (such as voice, touch strips, etc.). In addition, we can also control the text input process by designing start/end options to prevent accidental touches of sensors in daily life.

Real-world Usage. In this work, we used a *strip*-based prototype of our Handwriting Velcro, which allows for free exploration of interaction positions and postures at low cost. Although sufficient for lab use, the strips used in our prototype might be inconvenient for use in real-world scenarios. In the future, we will replace our strip-based prototype with customized ones integrated into clothing (Fig. 13), making them more suitable for practical daily use.

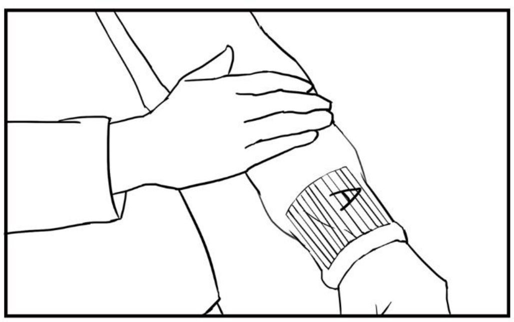


Fig. 13. The possible integrated wearable prototype design for daily use in the future. (left: integrated sleeve design, right: integrated trousers design.)

More Interaction Options. Although effective, the proposed Handwriting Velcro focuses on a limited number of interaction options associated with handwriting text input. However, flexible film-based touch interfaces can sense continuous input, a feature that enables them to be used for various tasks beyond text input [72]. Thanks to its expressive 40x40 resolution, larger touch area, and accurate continuous input monitoring, our Handwriting Velcro allows for many more interaction options (e.g., sketch drawing, touch control, 3D interaction, etc.), and we will address them in future work.

9 CONCLUSION

We have presented a novel method, *Handwriting Velcro*, for handwriting input on AR glasses, using flexible touch sensors. The distinct advantage of our method is that it adapts to different postures and users at little cost, providing a user-friendly, accurate, and robust input experience for diverse, challenging scenarios. The heuristic study reveals the optimal layout positions of the input device when the user is under different postures. Our method achieved accurate recognition performance by building a personalized classifier for each single user based on the concept of “human-in-the-loop” and active learning. With this approach, the error rate of personalized classifiers for individual users (6 in total) reaches an average of 0.005. The results show that our handwriting Velcro achieves good performance and high user feedback in system practicability and social acceptance. We also compared our solution with the existing similar work TEXTile [6] and commercial grade physical mini QWERTY keyboard for performance evaluation. Results show that our method excels the existing technique in both practicality (12.3 WPM) and user-friendliness in different daily contexts. In the future, we can achieve more complete and practical text input and editing solutions by upgrading hardware and adding more types of input characters and text editing functions.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (62072383, 61702433, 62077039), the Fundamental Research Funds for the Central Universities (20720210044, 20720190006), and the Royal Society (IEC\NSFC\211022), grants from the City University of Hong Kong (Project No. 7005590 and 9667234) and the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, CityU.

REFERENCES

[1] Jiban Adhikary and Keith Vertanen. 2021. Text Entry in Virtual Environments using Speech and a Midair Keyboard. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2648–2658.

[2] Jiban Adhikary and Keith Vertanen. 2021. Typing on Midair Virtual Keyboards: Exploring Visual Designs and Interaction Styles. In *IFIP Conference on Human-Computer Interaction*. Springer, 132–151.

[3] Sunggeun Ahn and Geehyuk Lee. 2019. Gaze-assisted typing for smart glasses. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 857–869.

[4] Takumi Azai, Shuhei Ogawa, Mai Otsuki, Fumihisa Shibata, and Asako Kimura. 2017. Selection and Manipulation Methods for a Menu Widget on the Human Forearm. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 357–360. https://doi.org/10.1145/3027063.3052959

[5] Bartosz Bajer, I. Scott MacKenzie, and Melanie Baljko. 2012. Huffman Base-4 Text Entry Glove (H4 TEG). In *2012 16th International Symposium on Wearable Computers*. IEEE. https://doi.org/10.1109/iswc.2012.28

[6] Ilyasse Belkacem, Isabelle Pecci, Benoît Martin, and Anthony Faiola. 2019. TEXTile: Eyes-Free Text Input on Smart Glasses Using Touch Enabled Textile on the Forearm. In *Human-Computer Interaction – INTERACT 2019*. Springer International Publishing, 351–371. https://doi.org/10.1007/978-3-030-29384-0_22

[7] Brenda AJ Berendsen, Marike RC Hendriks, Kenneth Meijer, Guy Plasqui, Nicolaas C Schaper, and Hans HCM Savelberg. 2014. Which activity monitor to use? Validity, reproducibility and user friendliness of three activity monitors. *BMC Public Health* 14, 1 (2014), 1–11.

[8] Xiaojun Bi and Shumin Zhai. 2016. *IJQwerty: What Difference Does One Key Change Make? Gesture Typing Keyboard Optimization Bounded by One Key Position Change from Qwerty*. Association for Computing Machinery, New York, NY, USA, 49–58. https://doi.org/10.1145/2858036.2858421

[9] Sabah Boustila, Thomas Guégan, Kazuki Takashima, and Yoshifumi Kitamura. 2019. Text typing in VR using smartphones touchscreen and HMD. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 860–861.

[10] Eugenie Brasier, Olivier Chapuis, Nicolas Ferey, Jeanne Vezien, and Caroline Appert. 2020. ARPads: Mid-air Indirect Input for Augmented Reality. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 332–343.

[11] GE Burnett, SM Lomas, B Mason, JM Porter, and SJ Summerskill. 2005. Writing and driving: An assessment of handwriting recognition as a means of alphanumeric data entry in a driving context. *Advances in Transportation Studies* (2005).

[12] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre van Schaik. 2017. EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE. https://doi.org/10.1109/ijcnn.2017.7966217

[13] Li Deng. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine* 29, 6 (nov 2012), 141–142. https://doi.org/10.1109/msp.2012.2211477

[14] David Dobbelstein, Christian Winkler, Gabriel Haas, and Enrico Rukzio. 2017. PocketThumb. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (jun 2017), 1–17. https://doi.org/10.1145/3090055

[15] Jacqui Fashimpaur, Kenrick Kin, and Matt Longest. 2020. PinchType: Text Entry for Virtual and Augmented Reality Using Comfortable Thumb to Fingertip Pinches. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–7.

[16] Conor R Foy, John J Dudley, Aakar Gupta, Hrvoje Benko, and Per Ola Kristensson. 2021. Understanding, Detecting and Mitigating the Effects of Coactivations in Ten-Finger Mid-Air Typing in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–11.

[17] Maite Frutos-Pascual, Clara Gale, Jake M Harrison, Chris Creed, and Ian Williams. 2021. Character Input in Augmented Reality: An Evaluation of Keyboard Position and Interaction Visualisation for Head-Mounted Displays. In *IFIP Conference on Human-Computer Interaction*. Springer, 480–501.

[18] Debjyoti Ghosh, Pin Sym Foong, Shengdong Zhao, Can Liu, Nuwan Janaka, and Vinitha Erusu. 2020. EYEditor. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/3313831.3376173

[19] Jun Gong, Zheer Xu, Qifan Guo, Teddy Seyed, Xiang 'Anthony' Chen, Xiaojun Bi, and Xing-Dong Yang. 2018. *WrisText: One-Handed Text Entry on Smartwatch Using Wrist Gestures*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3173574.3173755

[20] Tovi Grossman, Xiang Anthony Chen, and George Fitzmaurice. 2015. 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* (Copenhagen, Denmark) *(MobileHCI '15)*. Association for Computing Machinery, New York, NY, USA, 144–152. https://doi.org/10.1145/2785830.2785867

[21] Jens Grubert, Lukas Witzani, Eyal Ofek, Michel Pahud, Matthias Kranz, and Per Ola Kristensson. 2018. Text entry in immersive head-mounted display-based virtual reality using standard keyboards. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 159–166.

[22] Aakar Gupta, Cheng Ji, Hui-Shyong Yeo, Aaron Quigley, and Daniel Vogel. 2019. *RotoSwype: Word-Gesture Typing Using a Ring*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300244

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[24] Ramin Hedeshy, Chandan Kumar, Raphael Menges, and Steffen Staab. 2021. Hummer: Text Entry by Gaze and Hum. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/3411764.3445501

[25] Juan David Hincapié-Ramos, Xiang Guo, Paymahn Moghadasian, and Pourang Irani. 2014. Consumed endurance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/2556288.2557130

[26] Andreas Holzinger. 2016. Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Informatics* 3, 2 (mar 2016), 119–131. https://doi.org/10.1007/s40708-016-0042-6

[27] Xing-Yu Hou and Chuan-Fei Guo. 2020. Sensing mechanisms and applications of flexible pressure sensors. *Acta Physica Sinica* 69, 17 (2020), 178102. https://doi.org/10.7498/aps.69.20200987

[28] Y. T. Hsieh, Antti Jylh, V. Orso, L. Gamberini, and G. Jacucci. 2016. Designing a Willing-to-Use-in-Public Hand Gestural Interaction Technique for Smart Glasses. In *Chi Conference on Human Factors in Computing Systems*.

[29] Tan-Phat Huynh and Hossam Haick. 2018. Autonomous Flexible Sensors for Health Monitoring. *Advanced Materials* 30, 50 (2018), 1802337. https://doi.org/10.1002/adma.201802337 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/adma.201802337

[30] Haiyan Jiang and Dongdong Weng. 2020. HiPad: Text entry for head-mounted displays using circular touchpad. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 692–703.

[31] Haiyan Jiang, Dongdong Weng, Zhenliang Zhang, Yihua Bao, Yufei Jia, and Mengman Nie. 2018. Hikeyb: High-efficiency mixed reality system for text entry. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 132–137.

[32] Jung. 2020. imgaug. https://github.com/aleju/imgaug/.

[33] Dagmar Kern, Albrecht Schmidt, Jonas Arnsmann, Thorsten Appelmann, Nillakshi Pararasasegaran, and Benjamin Piepiera. 2009. Writing to your car. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/1520340.1520724

[34] Junhyeok Kim, William Delamare, and Pourang Irani. 2018. ThumbText: Text Entry for Wearable Devices Using a Miniature Ring. In *Graphics Interface*.

[35] Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs.LG]

[36] Kotikalapudi. 2016. keras-resnet. https://github.com/raghakot/keras-resnet.

[37] Chandan Kumar, Ramin Hedeshy, I. Scott MacKenzie, and Steffen Staab. 2020. *TAGSwipe: Touch Assisted Gaze Swipe for Text Entry*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376317

[38] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324. https://doi.org/10.1109/5.726791

[39] DoYoung Lee, Jiwan Kim, and Ian Oakley. 2021. FingerText: Exploring and Optimizing Performance for Wearable, Mobile and One-Handed Typing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/3411764.3445106

[40] DoYoung Lee, Jiwan Kim, and Ian Oakley. 2021. FingerText: Exploring and Optimizing Performance for Wearable, Mobile and One-Handed Typing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.

[41] 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. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (sep 2019), 1–27. https://doi.org/10.1145/3351252

[42] Lik Hang Lee, Kit Yung Lam, Yui Pan Yau, Tristan Braud, and Pan Hui. 2019. HIBEY: Hide the Keyboard in Augmented Reality. In *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE. https://doi.org/10.1109/percom.2019.8767420

[43] Xueshi Lu, Difeng Yu, Hai-Ning Liang, and Jorge Goncalves. 2021. iText: Hands-free text entry on an imaginary keyboard for augmented reality systems. *Proc. UIST. Association for Computing Machinery, New York, NY, USA* (2021).

[44] Xueshi Lu, Difeng Yu, Hai-Ning Liang, Wenge Xu, Yuzheng Chen, Xiang Li, and Khalad Hasan. 2020. Exploration of Hands-free Text Entry Techniques For Virtual Reality. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 344–349.

[45] I Scott MacKenzie and R William Soukoreff. 2002. Text entry for mobile computing: Models and methods, theory and practice. *Human–Computer Interaction* (2002).

[46] I. Scott MacKenzie and R. William Soukoreff. 2003. *Phrase Sets for Evaluating Text Entry Techniques*. Association for Computing Machinery, New York, NY, USA, 754–755. https://doi.org/10.1145/765891.765971

[47] Tim Menzner, Alexander Otte, Travis Gesslein, Jens Grubert, Philipp Gagel, and Daniel Schneider. 2019. A capacitive-sensing physical keyboard for VR text entry. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 1080–1081.

[48] Anindya Nag, Subhas Chandra Mukhopadhyay, and Jurgen Kosel. 2017. Wearable Flexible Sensors: A Review. *IEEE Sensors Journal* 17, 13 (jul 2017), 3949–3960. https://doi.org/10.1109/jsen.2017.2705700

[49] Shahriar Nirjon, Jeremy Gummeson, Dan Gelb, and Kyu-Han Kim. 2015. TypingRing: A Wearable Ring Platform for Text Input. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (Florence, Italy) *(MobiSys '15)*. Association for Computing Machinery, New York, NY, USA, 227–239. https://doi.org/10.1145/2742647.2742665

[50] Alex Olwal, Jon Moeller, Greg Priest-Dorman, Thad Starner, and Ben Carroll. 2018. I/O Braid: Scalable Touch-Sensitive Lighted Cords Using Spiraling, Repeating Sensing Textiles and Fiber Optics. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 485–497. https://doi.org/10.1145/3242587.3242638

[51] Alex Olwal, Thad Starner, and Gowa Mainini. 2020. *E-Textile Microinteractions: Augmenting Twist with Flick, Slide and Grasp Gestures for Soft Electronics*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376236

[52] Alexander Otte, Tim Menzner, Travis Gesslein, Philipp Gagel, Daniel Schneider, and Jens Grubert. 2019. Towards utilizing touch-sensitive physical keyboards for text entry in virtual reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 1729–1732.

[53] Alexander Otte, Daniel Schneider, Tim Menzner, Travis Gesslein, Philipp Gagel, and Jens Grubert. 2019. Evaluating text entry in virtual reality using a touch-sensitive physical keyboard. In *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 387–392.

[54] Patrick Parzer, Florian Perteneder, Kathrin Probst, Christian Rendl, Joanne Leong, Sarah Schuetz, Anita Vogl, Reinhard Schwoediauer, Martin Kaltenbrunner, Siegfried Bauer, and Michael Haller. 2018. RESi: A Highly Flexible, Pressure-Sensitive, Imperceptible Textile Interface Based on Resistive Yarns. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 745–756. https://doi.org/10.1145/3242587.3242664

[55] Patrick Parzer, Adwait Sharma, Anita Vogl, Jürgen Steimle, Alex Olwal, and Michael Haller. 2017. SmartSleeve. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM. https://doi.org/10.1145/3126594.3126652

[56] Duc-Minh Pham and Wolfgang Stuerzlinger. 2019. Hawkey: Efficient and versatile text entry for virtual reality. In *25th ACM Symposium on Virtual Reality Software and Technology*. 1–11.

[57] Ivan Poupyrev, Nan-Wei Gong, Shiho Fukuhara, Mustafa Emre Karagozler, Carsten Schwesig, and Karen E. Robinson. 2016. Project Jacquard. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/2858036.2858176

[58] Julie Rico and Stephen Brewster. 2010. Usable Gestures for Mobile Interfaces: Evaluating Social Acceptability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 887–896. https://doi.org/10.1145/1753326.1753458

[59] Deepak Ranjan Sahoo, Kasper Hornbæk, and Sriram Subramanian. 2016. *TableHop: An Actuated Fabric Display Using Transparent Electrodes*. Association for Computing Machinery, New York, NY, USA, 3767–3780. https://doi.org/10.1145/2858036.2858544

[60] Stefan Schneegass and Alexandra Voit. 2016. GestureSleeve. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*. ACM. https://doi.org/10.1145/2971763.2971797

[61] Burr Settles. 2009. *Active learning literature survey*. University of Wisconsin-Madison Department of Computer Sciences.

[62] Gaganpreet Singh, William Delamare, and Pourang Irani. 2018. D-SWIME: A design space for smartwatch interaction techniques supporting mobility and encumbrance. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[63] Sophie Skach, Rebecca Stewart, and Patrick G. T. Healey. 2018. Smart Arse. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*. ACM. https://doi.org/10.1145/3242969.3242977

[64] Paul Strohmeier, Victor Håkansson, Cedric Honnet, Daniel Ashbrook, and Kasper Hornbæk. 2019. Optimizing Pressure Matrices. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM. https://doi.org/10.1145/3294109.3295638

[65] Paul Strohmeier, Jarrod Knibbe, Sebastian Boring, and Kasper Hornbæk. 2018. zPatch. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM. https://doi.org/10.1145/3173225.3173242

[66] Felix Sygulla, Felix Ellensohn, Arne-Christoph Hildebrandt, Daniel Wahrmann, and Daniel Rixen. 2017. A flexible and low-cost tactile sensor for robotic applications. In *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. 58–63. https://doi.org/10.1109/AIM.2017.8013995

[67] Mohsin I. Tiwana, Stephen J. Redmond, and Nigel H. Lovell. 2012. A review of tactile sensing technologies with applications in biomedical engineering. *Sensors and Actuators A: Physical* 179 (jun 2012), 17–31. https://doi.org/10.1016/j.sna.2012.02.051

[68] Radu-Daniel Vatavu. 2017. Smart-Pockets. *Int. J. Hum.-Comput. Stud.* 103, C (July 2017), 1–21. https://doi.org/10.1016/j.ijhcs.2017.01.005

[69] Nicolas Villar, Daniel Cletheroe, Greg Saul, Christian Holz, Tim Regan, Oscar Salandin, Misha Sra, Hui-Shyong Yeo, William Field, and Haiyan Zhang. 2018. *Project Zanzibar: A Portable and Flexible Tangible Interaction Platform.* Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3174089

[70] Cheng-Yao Wang, Wei-Chen Chu, Po-Tsung Chiu, Min-Chieh Hsiu, Yih-Harn Chiang, and Mike Y Chen. 2015. PalmType: Using palms as keyboards for smart glasses. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 153–160.

[71] Yancheng Wang, Xin Wu, Deqing Mei, Lingfeng Zhu, and Jianing Chen. 2019. Flexible tactile sensor array for distributed tactile sensing and slip detection in robotic hand grasping. *Sensors and Actuators A: Physical* 297 (2019), 111512. https://doi.org/10.1016/j.sna.2019.07.036

[72] Eric Whitmire, Mohit Jain, Divye Jain, Greg Nelson, Ravi Karkar, Shwetak Patel, and Mayank Goel. 2017. DigiTouch. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (sep 2017), 1–21. https://doi.org/10.1145/3130978

[73] Tony Wu, Shiho Fukuhara, Nicholas Gillian, Kishore Sundara-Rajan, and Ivan Poupyrev. 2020. ZebraSense: A Double-Sided Textile Touch Sensor for Smart Clothing. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 662–674. https://doi.org/10.1145/3379337.3415886

[74] Te-Yen Wu, Lu Tan, Yuji Zhang, Teddy Seyed, and Xing-Dong Yang. 2020. Capacitivo: Contact-Based Object Recognition on Interactive Fabrics Using Capacitive Sensing. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 649–661. https://doi.org/10.1145/3379337.3415829

[75] Te-Yen Wu, Zheer Xu, Xing-Dong Yang, Steve Hodges, and Teddy Seyed. 2021. Project Tasca: Enabling Touch and Contextual Interactions with a Pocket-Based Textile Sensor. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 4, 13 pages. https://doi.org/10.1145/3411764.3445712

[76] Wenge Xu, Hai-Ning Liang, Anqi He, and Zifan Wang. 2019. Pointing and selection methods for text entry in augmented reality head mounted displays. In *2019 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 279–288.

[77] Wenge Xu, Hai-Ning Liang, Yuxuan Zhao, Tianyu Zhang, Difeng Yu, and Diego Monteiro. 2019. Ringtext: Dwell-free and hands-free text entry for mobile head-mounted displays using head motions. *IEEE transactions on visualization and computer graphics* 25, 5 (2019), 1991–2001.

[78] Zheer Xu, Weihao Chen, Dongyang Zhao, Jiehui Luo, Te-Yen Wu, Jun Gong, Sicheng Yin, Jialun Zhai, and Xing-Dong Yang. 2020. Bitiptext: Bimanual eyes-free text entry on a fingertip keyboard. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[79] Zheer Xu, Pui Chung Wong, Jun Gong, Te-Yen Wu, Aditya Shekhar Nittala, Xiaojun Bi, Jürgen Steimle, Hongbo Fu, Kening Zhu, and Xing-Dong Yang. 2019. TipText: Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM. https://doi.org/10.1145/3332165.3347865

[80] Powen Yao, Vangelis Lympouridis, Tian Zhu, Michael Zyda, and Ruoxi Jia. 2020. Punch Typing: Alternative Method for Text Entry in Virtual Reality. In *Symposium on Spatial User Interaction*. 1–2.

[81] Shanhe Yi, Zhengrui Qin, Ed Novak, Yafeng Yin, and Qun Li. 2016. GlassGesture: Exploring head gesture interface of smart glasses. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*. IEEE. https://doi.org/10.1109/infocom.2016.7524542

[82] Sang Ho Yoon, Ke Huo, Vinh P. Nguyen, and Karthik Ramani. 2015. TIMMi. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM. https://doi.org/10.1145/2677199.2680560

[83] Chun Yu, Ke Sun, Mingyuan Zhong, Xincheng Li, Peijun Zhao, and Yuanchun Shi. 2016. One-Dimensional Handwriting. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/2858036.2858542

[84] Difeng Yu, Kaixuan Fan, Heng Zhang, Diego Monteiro, Wenge Xu, and Hai-Ning Liang. 2018. PizzaText: Text entry for virtual reality systems using dual thumbsticks. *IEEE transactions on visualization and computer graphics* 24, 11 (2018), 2927–2935.

[85] Yang Zhang, Junhan Zhou, Gierad Laput, and Chris Harrison. 2016. SkinTrack. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/2858036.2858082

[86] Fengyuan Zhu and Tovi Grossman. 2020. BISHARE: Exploring Bidirectional Interactions Between Smartphones and Head-Mounted Augmented Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/3313831.3376233