HandPad: Enabling On-the-Go Writing on Your Hand via Human Capacitance

Yu Lu*
yulu01@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Hao Pan
panh09@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Dian Ding*†
dingdian94@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Yijie Li*
yijieli@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Juntao Zhou*
juntaozhou@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Yongjian Fu
fuyongjian@csu.edu.cn
Central South University
Changsha, China

Yongzhao Zhang
zhangyongzhao@uestc.edu.cn
University of Electronic Science and Technology of China
Chengdu, China

Yi-Chao Chen*
yichao@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

Guangtao Xue*
gt_xue@sjtu.edu.cn
Shanghai Jiao Tong University
Shanghai, China

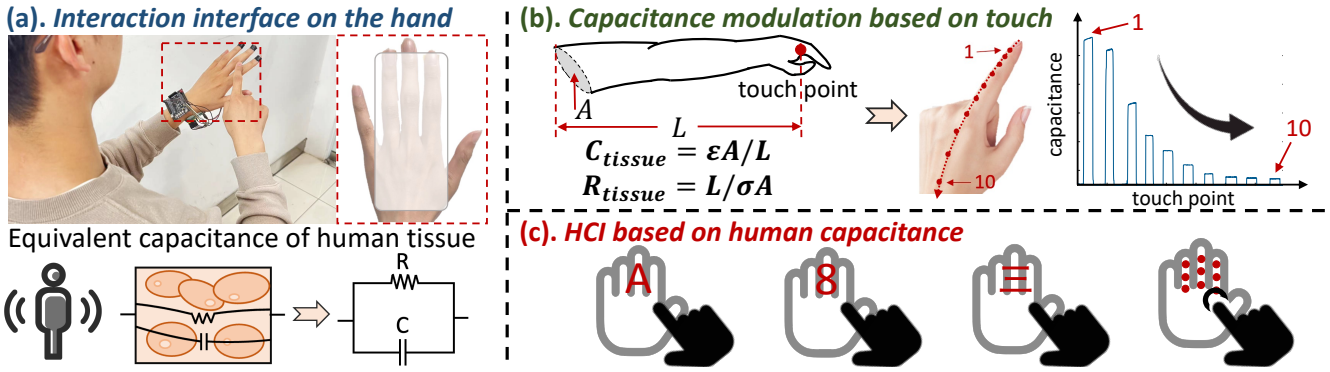


Figure 1: A taste of *HandPad*: (a) Our bodies can be constructed as capacitive circuits. (b) When we touch different locations on one hand with a finger from the other, the intrinsic capacitance of our arm is altered. (c) Utilizing changes in human capacitance during touch interactions, we engineer a skin-based touchscreen system that supports multiple input modalities.

ABSTRACT

The convenient text input system is a pain point for devices such as AR glasses, and it is difficult for existing solutions to balance portability and efficiency. This paper introduces *HandPad*, the system that turns the hand into an on-the-go touchscreen, which realizes interaction on the hand via human capacitance. *HandPad* achieves keystroke and handwriting inputs for letters, numbers, and Chinese characters, reducing the dependency on capacitive or pressure sensor arrays. Specifically, the system verifies the feasibility of touch point localization on the hand using the human capacitance model and proposes a handwriting recognition system based on Bi-LSTM and ResNet. The transfer learning-based system only needs a small amount of training data to build a handwriting recognition model for the target user. Experiments in real environments verify the feasibility of *HandPad* for keystroke (accuracy of 100%) and handwriting recognition for letters (accuracy of 99.1%), numbers (accuracy of 97.6%) and Chinese characters (accuracy of 97.9%).

CCS CONCEPTS

• **Human-centered computing** → **Interaction techniques**.

KEYWORDS

Human Capacitance; Key Stroke; Handwriting Input

*Also with Shanghai Key Laboratory of Trusted Data Circulation and Governance in Web3.

†Dian Ding is the corresponding author.

UIST '24, October 13–16, 2024, Pittsburgh, PA, USA

ACM ISBN 979-8-4007-0628-8/24/10
https://doi.org/10.1145/3654777.3676328

ACM Reference Format:
Yu Lu, Hao Pan, Dian Ding, Yijie Li, Juntao Zhou, Yongjian Fu, Yongzhao Zhang, Yi-Chao Chen, and Guangtao Xue. 2024. *HandPad*: Enabling On-the-Go Writing on Your Hand via Human Capacitance. In *The 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24), October 13–16, 2024, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3654777.3676328

1 INTRODUCTION

Augmented reality devices such as AR glasses provide users with an intuitive visual experience, but a convenient text input system for mobile scenarios is an unsolved pain point [19]. Due to the problem of thick fingers, traditional methods based on keyboards or touchscreens have limitations on device size, making it difficult to meet the portability requirement of mobile scenarios. To enhance the portability and efficiency of the system, researchers proposed different ways of text input methods for AR systems, including speech recognition [1, 17, 68], mid-air gestures [25, 45, 51, 65] and touch-based input interfaces [18, 62]. However, speech recognition is affected by ambient noise and can cause privacy concerns for users in public places [17, 68]. Wireless signal-based mid-air gestures are less robust to environmental disturbances [25, 43, 51]. In addition, researchers designed different types of wearable interaction interfaces for AR glasses, including fingers [45, 62], and hands [18]. However, the limited contact area of the finger [62] only supports limited keystrokes. The contact area of the hand can be sufficient for handwriting input, but the existing work relies on the sensor matrix and the commonly used pressure sensor matrix suffers from jitter, jumps, and nonlinear artifacts [18].

We propose an innovative capacitance-based interaction system, which seeks to expand the user's palm into a powerful "touchscreen" input interface. Starting from the unique perspective of intrinsic human body capacitance, as shown in Fig. 1, we explore modulation techniques based on this inherent property. When a user touches different locations on their left hand with a finger from their right hand, changes in the touch points alter the length of two sections of the left arm tissue, subsequently affecting their equivalent capacitance and the human capacitance measured by the sensor. Based on this principle, we develop a novel human-computer interaction (HCI) system named *HandPad*. Although related work has utilized capacitance for HCI systems, such as recognizing different fingers [26, 40], parts of a finger [32, 50], varying touch pressures [4, 31], and finger angles [47, 66], our system differs from these works in two main aspects:

- While related work requires traditional multi-channel capacitive sensors [10, 18], *HandPad* uses a single three-channel capacitive sensor to transform the hand into a portable interaction interface, significantly simplifying system complexity and reducing reliance on sensor matrices.
- Related work is typically limited to specific applications. *HandPad* not only identifies different touch modes but also actively modulates the intrinsic human capacitance to sense touch locations and sliding trajectories. Thus, *HandPad* can realize touch-based keystroke input, allowing users to perform different keystrokes by touching distinct positions on the left hand. Furthermore, it enables hand-based handwriting input, supporting various languages including letters, numbers, and Chinese characters capabilities not previously achieved in earlier studies.

Designing *HandPad*, a human capacitance interaction system, is challenging. The core issue is achieving stable modulation of human body capacitance, requiring precise measurement and interpretation of capacitive signals while preventing interference from environmental factors. Electromagnetic (EM) interference can affect sensor readings, leading to inaccurate touch point localization. Additionally, different users' writing styles and habits make handwriting recognition difficult. Collecting substantial writing for an effective recognition model is time-consuming and burdensome. The complexity of Chinese character strokes further complicates model training and handwriting recognition.

To address these challenges, we adopt a series of innovative strategies. The *HandPad* system initially uses the user's hand as an interactive interface, modulating the body's intrinsic capacitance through finger touches to enable diverse input operations. During this process, the system has designed a precise finger model that matches the capacitance values of 14 fixed touch points on the hand and can be adjusted with the capacitance of 5 specific calibration points to accommodate the hand characteristics of different users. To ease the burden of dataset acquisition for the target user, *HandPad* employs transfer learning techniques. By leveraging a pre-trained model, we can fine-tune it with a limited amount of target user data, thereby speeding up the model training process and improving the accuracy of handwriting recognition. Finally, to simplify the difficulty of Chinese character handwriting recognition, the *HandPad* system does not recognize the entire character directly but decomposes it into fundamental strokes. Chinese characters typically consist of 32 strokes, and *HandPad* categorizes these strokes into 19 types for recognition. The system not only identifies individual strokes but also matches the sequences of strokes, thus accomplishing the recognition of the Chinese character. This stroke-level recognition greatly reduces the recognition complexity while also enhancing recognition speed and accuracy.

We develop a prototype of *HandPad* by strategically placing ultra-miniature electrodes on three fingertip locations of the user's hand. This innovative approach transforms the dorsum of the hand into an extended touch-sensitive interface for interaction, enabling both keystroke and handwriting input modalities. To validate the effectiveness of the proposed *HandPad*, we conduct a comprehensive series of experiments. The experimental results confirm the high efficacy of *HandPad* in facilitating user input. Specifically, the system achieves a keystroke input accuracy of 100% when recognizing 3×4 keystrokes. In terms of handwriting input, *HandPad* shows remarkable precision across various categories: letters were recognized with an accuracy of 99.1%, numerical digits with 97.6%, and Chinese strokes with 94.6%. Furthermore, the system demonstrated a 97.9% accuracy rate in recognizing complete Chinese characters. The contributions of this research are detailed below:

- We propose a novel interaction system, called *HandPad*, demonstrating the practicality of constructing an augmented touchscreen interface on the human hand by employing a model of human capacitance. Our system's robustness permits accurate touch inputs on the hand skin's surface, paving the way for innovative wearable technologies.

- In keystroke input mode, we utilize a sophisticated finger model to accurately localize touchpoints to ensure intuitive and efficient keystroke interaction across users.
- In handwriting mode, we employ Bi-LSTM and ResNet architectures to analyze the sliding trajectories. Additionally, we leverage a compact personal dataset to facilitate efficient transfer learning for accommodating new users.
- We implement the prototype of *HandPad* and conduct extensive experiments in real-world environments to assess its keystroke and handwriting interaction capabilities. The results indicate that *HandPad* is an innovative interaction system, and can be deployed on wearable devices to extend the hand into a touchscreen interface.

2 RELATED WORK

2.1 Handwriting and Key Stroke Methods

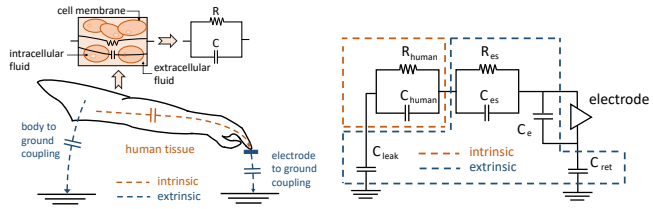
Recent research has proposed handwriting recognition systems based on wireless signals such as acoustic [51, 65, 69], magnetic [25, 44], vibration signal [14, 15], and external interface [18]. Acoustic signals can also track the pen's writing trajectory [51, 65] or capture the friction sound of the pen tip [6, 69] to recognize handwriting inputs. MagHacker [44] and WiReader [25] recognize the writing trajectory via the magnetic field and Wi-Fi signal. VibWriter [14, 15] detects handwriting by analyzing the vibration signals on a surface.

Mechanical keys-based systems are challenging to configure on small mobile devices, and researches provides effective solutions based on acoustic signal [28, 36], and vibration signal [8, 9]. TapSense [28] distinguished between different interactions by the acoustic signals of the finger. SonarID [36] sent ZC sequences [71] using a speaker and identified the keystroke. ViType [9] and Taprint [8] identified the different inputs via the vibration signals generated by tapping different hand positions. PrinType [45] used the fingerprint sensor to determine the different positions of the thumb clicks on the other fingers.

2.2 HCI based on Human Capacitance

There is a high degree of differentiation between the different hand gestures [3, 10, 11, 22, 35, 38, 39, 42, 63], such as fist clenching, open palm, etc. The finger recognition-based interaction approaches [26, 27, 40] use the finger as a complementary dimension to the human-computer interaction system. Different parts of the finger (e.g., finger belly, nail) have different electrical properties [7, 32, 50], which can be captured and recognized by capacitive sensors. Touch pressure recognition in conventional touch screen devices relies on an additional sensing layer; capacitive sensor-based finger pressure recognition offers a low-cost solution [4, 12, 23, 31, 64]. Finger angle estimation based on the touch screen provides information about the interaction in 3D space [47, 55, 66].

HandPad allows for a wider range of interactions, including keystrokes, handwriting letters, numbers, etc. Furthermore, the interaction extensions based on the finger touch method require the user to learn and adapt. In contrast, the handwriting input method is more commonly used and therefore more familiar and easier for users to master.



(a) The intrinsic and extrinsic capacitance of the human body. (b) The circuit model of human capacitance.

Figure 2: Human capacitance model: (a). Human capacitance measured by the capacitive sensor during finger touch; (b). The extrinsic capacitance includes the coupling capacitance of the human body (C_{leak}) and the electrode (C_{ret}) to the ground, the capacitance between the skin and the electrode (C_{es}), and the intrinsic capacitance of the electrode (C_e).

3 PRELIMINARY STUDY

3.1 Human Capacitance Model

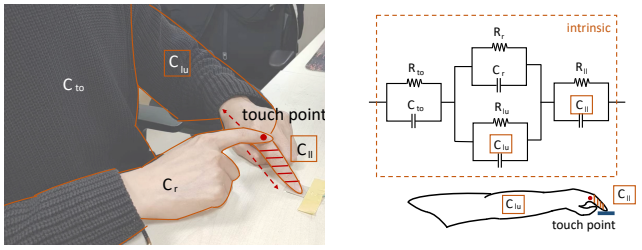
HCI systems based on human capacitance recognize interaction contents utilizing different signal patterns of human capacitance. We need to build the human capacitance model to verify the feasibility of *HandPad*.

The system measures the human capacitance at the fingertip, as shown in Fig. 2(a). Human capacitance can be divided into the intrinsic part and the extrinsic part. The intrinsic part is the body tissue; the extrinsic part consists of the body or electrodes and the external ground plane [34, 53]. The intrinsic part is usually considered to be static and related to the electrical properties of the body, and the equivalent capacitance of the body part tissue is only related to the size of the part being measured. The external part depends mainly on the external environment, such as objects in the return path (i.e., air), the contact between the electrodes and the skin, etc.

HandPad builds interaction systems based on finger touch. The human capacitance C_{human} measured from the fingertip contains two components, C_{in} and C_{ex}. As shown in Fig. 2(b), C_{in} represents the intrinsic part of the human capacitance, and C_{ex} represents the extrinsic part. The internal part is related to the electrical properties of the human body and the capacitance of part of the tissue C_{tissue} can be written as $C_{tissue} = \varepsilon A/L$ and $R_{tissue} = L/\sigma A$, where A represents the cross-sectional area of the tissue, L represents the length, ε and σ represents the relative permittivity and conductivity. The extrinsic part contains the following components: C_{leak} represents the human body capacitance coupling to the ground, and C_{ret} is the capacitance representing the electrode coupling to the ground. Furthermore, C_{es} is the capacitance between the skin and the electrode, and C_e represents the intrinsic capacitance of the electrode.

3.2 HCI based on Human Capacitance

Recent researches on HCI using human capacitance is mainly based on two-dimensional capacitive images. Considering the limitations of device size and user comfort, these systems only enable a limited expansion of interaction methods. We attempted to build the



(a) Modulation the intrinsic capacitance with finger touch. (b) The circuit model of intrinsic capacitance.

Figure 3: The intrinsic part of human capacitance.(a): *HandPad* enables different inputs based on different touch points; (b): The touch point divides the left arm into two parts, and the position of the touch point changes the size of these two body tissues, thus affecting their equivalent capacitances.

interaction system with a capacitive sensor. The system transfers the touch interface from the device side to the user's hand, thus alleviating the dependence of current interaction systems on the size of touch screens.

The feasibility of *HandPad* is further demonstrated with the human capacitance model. Specifically, the user touches the left hand with the finger of the right hand at different positions (including the knuckles, back of the hand, etc.) to achieve different inputs, as shown in Fig. 3(a).

During this process, the user's right-hand moves a limited amount and the torso part remains almost stationary, so the extrinsic part of the human capacitance can be seen as constant, which verifies the modulation of human capacitance by touch behavior through the control variable method. We then analyze the intrinsic part of the human capacitance and the influence of different touch positions on it. We can divide the human body into four parts and create corresponding equivalent circuits for each, as shown in Fig. 3(b). The equivalent capacitance corresponding to the torso part and the right arm is independent of the touch position. The left arm is divided into two parts by the touch point of the right-hand finger, and the position of the touch point affects the length of the body tissue in both parts, thus changing the equivalent capacitance of the two parts.

With a constant dielectric constant and cross-sectional area of the human tissue, its corresponding equivalent capacitance is inversely proportional to the length of the body. The intrinsic body capacitance corresponding to the four parts of human tissue can be written as:

$$C_{in} = \frac{1}{\frac{1}{C_{to}} + \frac{1}{C_r + C_{lu}} + \frac{1}{C_{ll}}} \tag{1}$$

where C_{to} and C_r represent the equivalent capacitance of the torso and right arm, while C_{lu} and C_{ll} represent the upper and lower parts of the left arm divided by the touch point and the length of two parts concerning the position of the touch point. The two parts can be written as $C_{lu} = \varepsilon A_{lu}/L_{lu}$ and $C_{ll} = \varepsilon A_{ll}/L_{ll}$; A_{lu} and A_{ll} is the cross-sectional area of two parts; L_{lu} and L_{ll} are the length of two parts, and the sum of the two is the length of the left arm.

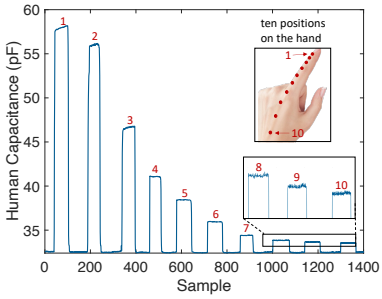


Figure 4: Feasibility of keystroking based on the human capacitance. Touch positions can be modulated with the human capacitance values.

3.3 Feasibility of *HandPad*

We illustrate the feasibility of implementing touch-based interaction on the human body theoretically with the human capacitance model, and we further verify it with experiments. As shown in Fig. 3(a), a volunteer naturally touched the electrode connected to the capacitance sensor (TI FDC2214 [20]) with the fingertip of the left hand, and the right-hand finger touches the left hand. Ten locations on the fingers and back of the left hand were selected to verify the effect of the touch position on human capacitance. The sampling rate of the capacitive sensor is only $30Hz$, which requires a low performance of the device.

The experimental results are shown in Fig. 4, where the volunteer touches ten locations in sequence from the fingertip to the wrist. The human capacitance measured by the sensor decreases as the distance between the touch point and the fingertip (i.e. L_{ll}) increases, and the human capacitance is inversely proportional to the distance. Because the distance to the upper part of the left arm (L_{lu}) is much greater than the lower part (L_{ll}), and the equivalent capacitance of human tissue is inversely proportional to its length, the equivalent capacitance corresponding to the upper part (C_{lu}) varies more weakly with distance. Therefore, the human capacitance measured by the sensor mainly reflects the change in the equivalent capacitance of the lower part (C_{ll}).

4 SYSTEM DESIGN

4.1 Signal Acquisition

As shown in Fig. 5, the capacitive sensor unit [33] consists of two parallel electrodes (C_t) [56] and the physical contact of the user corresponds to a human capacitance C_{human} in series with the branched circuit. The sensor monitors the touch behavior based on the capacitance of the branch.

The charging loop: First, the capacitance C_t receives an excitation signal (usually an alternating current (AC) signal of a square or sine wave) with the amplitude V_{ex}. As the voltage across the capacitance changes, the charge Q_t stored in the capacitance C_t can be written as [70]:

$$Q_t = 2C_t V_{ex} \tag{2}$$

The transfer loop: The charge stored by capacitance C_t is then transferred to the feedback circuit, and the amount of charge gained

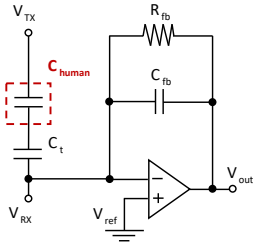


Figure 5: The capacitive sensor.

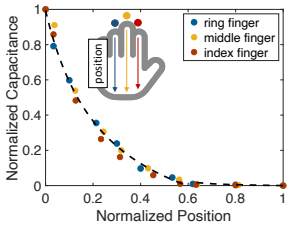


Figure 6: The model of three fingers.

by capacitance C_{fb} is equal to the charge of C_t during the charging process [49]. The output voltage V_{out} of the amplifier at the end of the excitation cycle is determined by the feedback capacitance C_{fb} [41]:

$$2C_t V_{ex} = C_{fb}(V_{ref} - V_{out}) \tag{3}$$

$$V_{out} = V_{ref} - 2C_t V_{ex}/C_{fb} \tag{4}$$

where V_{ref} is the reference voltage [49] of operational amplifier. As the user touches the sensor, the branch capacitance changes, and the human capacitance on the branch can be calculated based on the output of the amplifier V_{out}.

The discharge loop: The loop current flows through the feedback capacitor C_{fb} and the feedback resistor R_{fb} and is then released.

4.2 Signal Pre-processing

The user touching the left hand with a finger can also significantly increase the measured value. The internal capacitance can be written as:

$$C'_{in} = \frac{1}{\frac{1}{C_t} + \frac{1}{C_l}} \tag{5}$$

where C_l is the equivalent capacitance of the left arm. Comparing the intrinsic capacitance of two states (C_{in} and C'_{in}), C_{lu} is greater than C_l and C_l is much smaller than C_{ll} (i.e., $2C_l < C_{ll}$). The system can extract segments of the human capacitance signal for performing different interactions. The system uses a Gaussian filter to smooth the signal segments of the human capacitance, which mitigates the fluctuations caused by sensor noise.

4.3 Finger Modeling

Based on the human capacitance model, we verified the relationship between the touch position and the measurement. While only the amplitude can differentiate the touch point, the two-dimensional information of the touch points enables a wealth of application extensions. We collected human capacitance with the capacitive sensor of three channels at the fingertips of the index, middle, and ring fingers and built a simple writing pad on the three fingers.

For the three fingers that make up the interaction interface, the volunteer was asked to touch each finger at ten locations and record the measurements of human capacitance and the distance from the touch point to the fingertip.

As shown in Fig. 6, we normalized the human capacitance signals and it can be seen that the measurements are inversely proportional to the distance. We fitted the finger model using an inverse proportional function, and as the back of the hand and the finger possess different cross-sectional areas, we fitted them separately. Then, considering the inconsistency in finger lengths, we scaled and translated three finger models. The finger models can be written as :

$$d = \frac{\alpha_i}{b * C_{human} + 1} + \beta_i \tag{6}$$

where C_{human} represents the human capacitance measurement, b is the fitting parameter common to the three finger models, α and β are the scaling and translation factors for the three fingers, and d represents the location of the touch point. With a single channel, the system can only acquire one-dimensional information about the contact point.

5 APPLICATION

HandPad implements an extended touchscreen on the hand based on human capacitance, enabling keystrokes and handwriting input for letters, numbers, and Chinese characters. The architecture of the system is shown in Fig. 7.

5.1 Keystroke Input

Keystroke-based interaction systems are widely present in various devices in people's everyday lives. We attempt to implement the keystroke input using the capacitive sensor based on human capacitance on the hand. The keystroke input system consists of four parts:

Registration: The length ratio of the finger joints varies from user to user and the position of the keystroke also changes. Therefore, before performing the keystroke, the system needs to record the position of the touch point on the fingers, as shown in the keystroke interface in Fig. 8.

Calibration: Before pressing a keystroke, the user needs to touch the calibration points, the capacitance at the fingertip (P_1, P_2, and P_3) and the back of the hand (P_5) can be used to adjust the capacitance model of the corresponding fingers.

Detection: According to the effect of finger touch on the intrinsic capacitance, a user touching the left hand causes an abrupt change in the body capacitance. The system can propose touch segmentation based on the amplitude, and we set the segmentation threshold as $T_{touch} = 0.7 \times C_{no\_touch} + 0.3 \times C_{P_5}$.

Recognition: The four keys on three fingers form a classic 3×4 keystroke interface, with two buttons on the back of the hand for "*Delete*" and "*Enter*". The user touches different locations on the finger or the back of the hand for key press operation. The system uses the human capacitance and finger model to calculate the position of the touch point P_{input} and selects the touch point closest to the recorded point P_{key} as the key to *HandPad*.

5.2 Handwriting Recognition

The feasibility of using human capacitance to image finger movements was confirmed, and we tried to implement the handwriting interface as shown in Fig. 8. The handwriting input system also consists of four parts:

Registration: For different handwriting input tasks, the system collects the handwriting training set $Dataset_{user}$ of the target user.

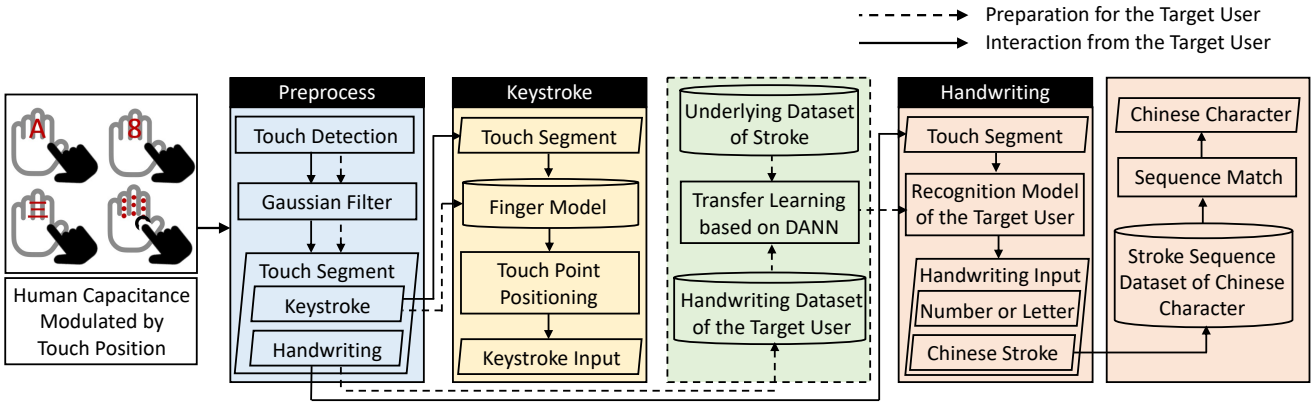


Figure 7: The interaction system architecture of *HandPad*: the dashed lines indicate the preparation for the target user, and the solid lines indicate the recognition process for keystrokes and handwriting inputs.

(a) calibration point (b) keystroke interface (c) handwriting interface

Figure 8: The calibtration points and keystroke & handwriting interfaces of *HandPad*.

Calibration: The user touches two calibration points, P_4 and P_5, to provide the capacitance value information of "*Delete*" and "*Enter*".

Detection: The handwriting input system uses the capacitance value of P_4 as the amplitude threshold for handwriting to extract the input segment.

Recognition: For the handwritten fragments, the system carries out the normalization process for recognition. The handwriting recognition model for the target user is implemented based on transfer learning, which reduces the burden of dataset collection. The judgment of "*Delete*" and "*Enter*" is achieved by comparing the magnitude of calibration points.

Considering the large number of Chinese characters, we divide Chinese character recognition into two parts: stroke recognition and stroke sequence matching. We create a Chinese character stroke dataset containing `20795 Chinese characters` and their corresponding stroke sequences. The system recognizes Chinese characters based on stroke sequence matching.

Unlike the keystroke input, *HandPad* recognizes the handwriting input without relying on the positioning based on the finger model. We can convert the human capacitance measured by three capacitance sensors into 2D images of finger movements. Although the number of channels limits the imaging accuracy, we can mine the trace information of the handwriting from the time series, and the slight deformation does not affect the recognition of the handwriting input. As shown in Fig. 18, to display the finger movement trajectory clearly, the finger model was used to adjust. It can be seen that the capacitance sensor can reflect finger movements and infer handwriting input.

5.2.1 Letter and Number Recognition. Considering that the collected capacitance signals can exhibit variability depending on the different biometric characteristics and writing habits of users, the effectiveness of the neural network model for a specific user is closely determined by the quantity and quality of the training data. However, collecting sufficient training data on the target user is time-consuming and economically costly. Consequently, transfer learning [52] is hugely appropriate. Based on a substantial amount of underlying data and a trace amount of target user data, the training mode of transfer learning can improve the test performance of the network model on target user data.

5.2.2 Transfer Learning. We consider recognition tasks and define X as the input space and $Y = \{0, 1, ..., L-1\}$ as a set of L possible labels. Then, we define two different distributions over $X \times Y$ for the pre-collected underlying dataset and the target user dataset, called the underlying domain D_U and the target domain D_T respectively. Our transfer learning algorithm is then provided with a large number of the underlying samples U drawn *i.i.d* from D_U and a slightly small amount of the target user samples T drawn *i.i.d* from D_T.

$$U = \{u_i = (s_i, l_i)\}_{i=1}^{n} \sim (D_U)^n \tag{7}$$

$$T = \{t_i = (s_i, l_i)\}_{i=n+1}^{N} \sim (D_T)^{n'} \tag{8}$$

with $N = n + n'$ being the total number of training samples. The goal of our transfer learning algorithm is to build a classifier $c : X \rightarrow Y$ with a low *target risk*

$$R_{D_T}(c) = \Pr_{(s_i,l_i)\sim D_T}(c(s_i) \neq l_i) \tag{9}$$

5.2.3 Network architectures. The architecture is shown in Fig. 9. Our network tasks a 3-channel human capacitance signal sequence of length S_L as input. Here, each channel data of input sequence s_i is obtained from the capacitive sensors. The feature extractor uses a bidirectional long short-term memory [30] (LSTM) and a

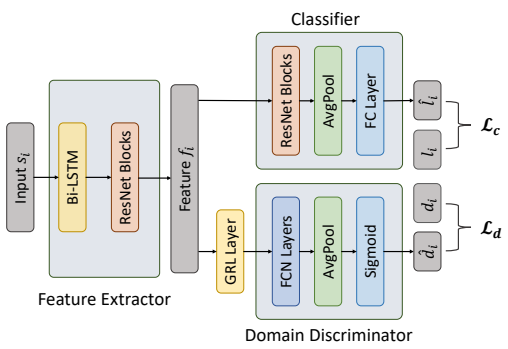


Figure 9: Architecture of our recognition neural network based DANN.

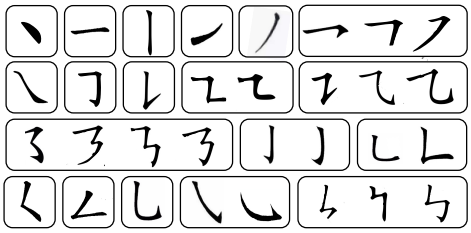


Figure 10: The Chinese character strokes.

resnet block [29] to obtain the features f_i extracted from the sequence s_i. Meanwhile, the classifier composed of resnet block and Multilayer Perceptron layer takes feature f_i as input and outputs an L-dimensional vector $\hat{l}_i$ corresponding to the L possible labels, while the domain discriminator based on FCN structure outputs a $P \times Q$ matrix $\hat{d}_i$ to distinguish signals on different domains D_U and D_T. Then, We employ a *domain − adversarial neural network* (DANN) [21, 37] based transfer learning approach to train the aforementioned network model.

5.2.4 Chinese Character Recognition. Unlike the smaller number of categories, such as letters and numbers, the number of Chinese characters poses a challenge for input recognition [5]. *HandPad* splits Chinese characters into corresponding stroke sequences according to the writing process, and then the Chinese character recognition can be split into two parts: stroke recognition and stroke sequence matching. There are 32 different strokes in Chinese characters, and certain pictorial similarities exist between the different strokes, such as vertical and vertical hooks, apostrophes, and tilts. Vision-based Chinese character recognition methods classify strokes into five categories [5].

HandPad can mine other handwriting trajectory information from sequence signals based on human capacitance. For example, it can distinguish between apostrophes and tilts based on the direction of finger movement. Thus, we can perform more subtle recognition of strokes and thus optimize the stroke sequence matching of Chinese characters. We divide the strokes into 19 categories, as shown in Fig 10. The system merges the different strokes based on the following factors: similarity between strokes, the limitations of the three-channel sensor in terms of imaging, writing speed, and the effect of stroke trajectory deformation on data acquisition. The recognition of Chinese strokes is similar to the recognition of letters and numbers, and we implement the recognition of Chinese characters based on sequence matching.

6 EVALUATION

6.1 Experimental Setup

The *HandPad* system consists of a capacitance sensor (TI FDC2214 [20] with the excitation frequency of $10MHz$) and an STM32 microcontroller (sampling rate of $30Hz$), where the capacitance sensor is used to measure the human capacitance in real-time. The capacitive sensor is the mutual capacitance sensor and the sensor operates in

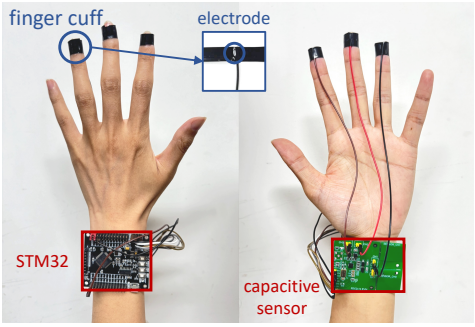
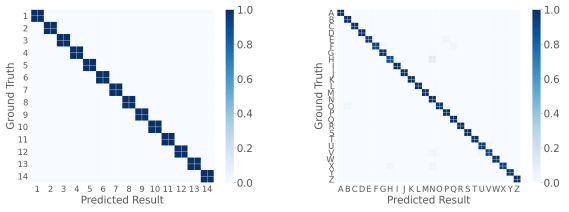


Figure 11: Experimental setup: TI FDC2214 sensor and STM32 microcontroller are used to measure human capacitance, easily integrated into wearable devices; the finger cuff-based connection is easy to wear.

shunt mode (passive sensing). The software part is the sampling of the sensor output using the STM32, the processing and recognition of the sensor data on the PC side (Lenovo LEGION Y7000). For security considerations, the operational parameters of the FDC2214 (voltage range: 2.7-3.1V, current: 2.1mA) are below safety thresholds (10V and 20mA for continuous contact [2]). The system acquires the human capacitance signal from the three fingertips of the left hand. The user touches different positions of the left hand for keystroke input or slides on the three fingers for handwriting input.

Keystroke. The keystroke interaction system consists of dataset collection for finger modeling and keystroke tests. The system sets the 14 keys on the back of the left hand (3×4 *keyboard*, *"Delete"* and *"Enter"*). The user touches these positions successively with the right-hand finger to build the dataset for finger modeling. Moreover, the user touches the same positions to collect the dataset for the keystroke test. For each key, the user collected data for 30 touches and each touch lasted for about 1*s*.

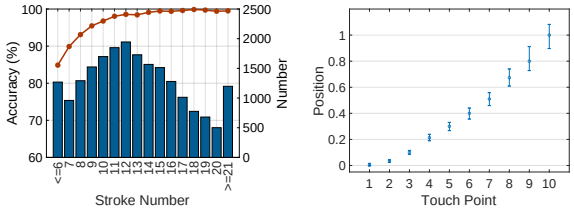
Handwriting Recognition. The handwriting recognition system consists of the following steps: First, we invited 15 additional volunteers, comprising 9 males and 6 females aged between 19 and 33 (average age of 24.6, SD of 6.53), to collect the underlying dataset, which includes handwritten letters, numbers, and strokes of Chinese characters. Then, we invited 10 participants as target users to evaluate *HandPad*, containing 4 males and 6 females with ages ranging from 22 to 29 (average age of 26.2). These volunteers and target users were asked to write the letters, numbers, and strokes



(a) Accuracy of keystrokes. (b) Accuracy of letters.

(c) Accuracy of numbers. (d) Accuracy of Chinese strokes.

Figure 12: Confusion matrix for the accuracy of keystrokes and handwriting inputs.



(a) Accuracy of Chinese character. (b) Localization error of touch point.

Figure 13: Micro Benchmarks of the *HandPad* system: (a) stroke distribution of Chinese characters and the corresponding recognition accuracy; (b) positioning errors for different positions of the hand;

20 times each. As shown in Fig. 11, the electrodes are held in place at the fingertip by a finger cuff to measure body capacitance, securing the electrode-skin contact point and minimizing the effect of cable movement. The volunteers and target users wrote on the left hand with the finger of the right hand without restriction on the writing trajectory and speed. Based on the underlying dataset, we trained the handwriting recognition model for the target users using transfer learning.

Furthermore, we made the stroke sequence dataset of Chinese characters for the stroke sequence matching, which contains 20795 different Chinese characters and the corresponding stroke sequences. For each stroke number, the target users wrote 80 Chinese characters selected from the Chinese character dataset to build a test set of Chinese characters.

Training details. The model was trained on one NVIDIA TESLA V100 for one hundred epochs with a batch size of 256. The sequence length of each channel was limited to 108, i.e. approximately 3.6s of data sampled at a sampling rate of 30HZ. The optimizer used is Adam with a learning rate of 1e-3, $\beta_1 = 0.9$ and $\beta_2 = 0.999$, a weight decay of 0. A learning rate schedular is used to decay the learning rate of each parameter group by $\gamma = 0.2$ every 50 epoch.

Metrics. *HandPad* was assessed using the following metrics:

Recognition Accuracy: The probability that input is correctly recognized in all relevant samples.

Confusion Matrix: Each row of the matrix represents the predicted result, and each column represents the actual label. The ith row and jth column of the matrix indicate the proportion of the ith input identified as the jth input.

6.2 Micro Benchmarks

We first evaluated the recognition accuracy of *HandPad* for both keystroke and handwriting inputs. As shown in Fig. 12, the recognition accuracy of the keystroke is 100%, while the recognition accuracy of handwriting letters, numbers, and Chinese character strokes are 99.1%, 97.6%, and 94.6%. It can be seen that different interactions based on human capacitance are distinguishable.

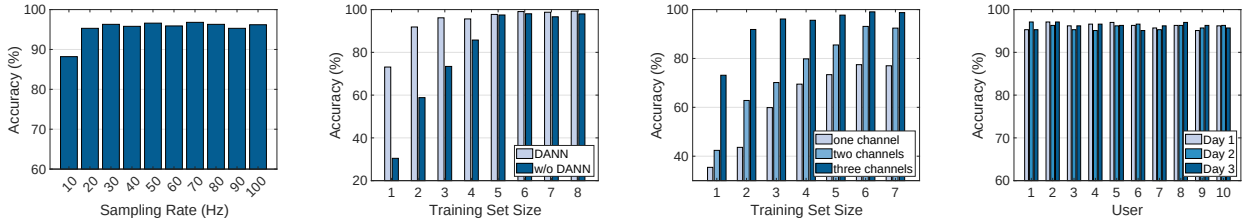
Fig. 13(a) shows the number of Chinese characters with different numbers of strokes (histogram) and the recognition accuracy of Chinese characters with different numbers of strokes (scatterplot). Compared to stroke recognition, the recognition accuracy of Chinese characters has improved significantly, and the average accuracy is 97.9%. Furthermore, we evaluated the accuracy of the finger model for touch point localization. Fig. 13(b) shows the localization error of the samples in different contact points (1 to 10 in Fig. 4). To further guarantee the accuracy of the keystrokes, the proposed keystrokes are chosen as 1, 3, 4, and 6 positions of three fingers to build a 3×4 keyboard, and 8 and 10 positions as *"Delete"* and *"Enter"*. It can be seen that there is no deviation in the positioning results of these positions.

6.3 Influence of System Parameters

6.3.1 Influence of Sampling Rate. A high sampling rate raises the cost of hardware devices and the threshold for system deployment. In this experiment, we verified the impact of sampling rate on keystroke and handwriting recognition. As shown in Fig. 14(a), we evaluated the system performance in various cases from 10Hz to 100Hz. As the sample rate increases from 10Hz to 30Hz, the recognition accuracy increases from 88.2% to 96.3%. As the sample rate continues to increase, the recognition accuracy plateaus. Thus, *HandPad* has a low sampling rate requirement and can be easily integrated into other devices as an interactive system.

6.3.2 Influence of Training Set Size. To improve the universality of the system, *HandPad* reduces the burden of dataset collection for the target users with transfer learning. The dataset size of the target user still has an impact on the performance of the system. In this experiment, we verify the effect of the training set size on the recognition accuracy of the system in transfer learning.

As shown in Fig. 14(b), the recognition accuracy of the system improves with the increase of the training set size. When the training set size increases from 1 to 3 for each input, the handwriting recognition accuracy for three types of handwriting interaction increases from 73.1% to 96.2%. However, when the sample exceeds 3, system performance improvement is weakened. Considering the burden caused by dataset collection and the demand for high-precision recognition, we set the training set size to 3 for each input.



(a) Influence of sampling rate. (b) Influence of training set size. (c) Influence of channel size. (d) Influence of different users.

Figure 14: Performance of the *HandPad* system under various factors includes: (a)-(c) the influence of different system parameters; and (d) the impact of distinct user identities.

In contrast, without transfer learning, users are required to collect 5 samples for each handwriting input to achieve similar accuracy, resulting in a 66.7% increase in the time cost of dataset collection. Although the number of samples collected for each category is reduced from 5 to 3. For the three handwriting inputs, the overall training set samples can be reduced by 110 samples.

6.3.3 Influence of Channel Size. To minimize the hardware requirement of the system, *HandPad* uses the capacitive sensor of three channels to measure the human capacitance at three fingertips to mine the trajectory of finger movement. In this experiment, we attempt to further reduce the channel size of the capacitive signal. Specifically, we evaluated the performance in recognizing handwriting input using the signal of one and two channels.

Fig. 14(c) shows the best performance of the system in both cases. It can be seen that channel size has a significant impact on system performance, and the recognition accuracy of one channel signal will seriously affect the efficiency of the interaction system. This is because the situation of one channel severely restricts access to information about lateral movement. As the training set increased, the recognition accuracy for the three types of handwritten input under different channels stabilized at 77.1%, 92.4%, and 97.6%. To ensure high recognition accuracy, we used three channels to acquire the capacitance signal.

6.4 Influence of Other Factors

6.4.1 Influence of Different Users. Recent researches [16, 60, 67] confirmed the differences in human capacitance between users, mainly in the response of the human body to different frequency signals. Therefore, we need to evaluate the effect of different users on the recognition performance to demonstrate the system's ubiquitousness. For the 10 target users, we tested the system at three different times, spaced two days apart.

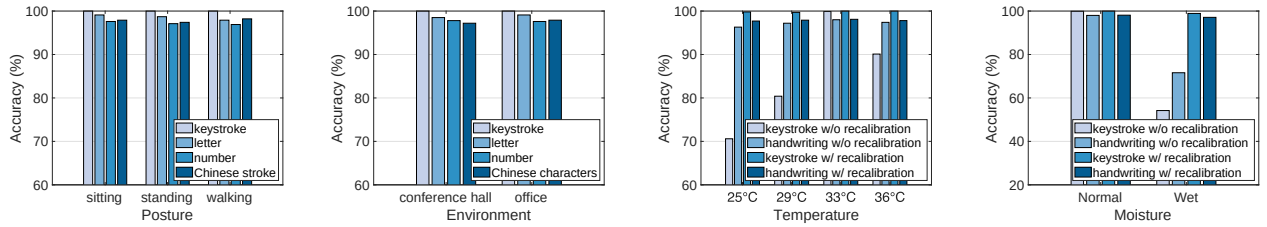
Although different users usually differ in terms of writing trajectory, writing speed, etc., the system builds recognition models for these target users. As shown in Fig. 14(d), we compare the recognition accuracy of keystrokes and handwriting input for these users. It can be seen that the training process based on transfer learning can fit the handwriting styles of different users with a limited training set size. Moreover, the different time points do not affect the recognition accuracy of the system. First, the writing habits of different users are relatively stable; second, *HandPad* utilizes human capacitance at different contact points for recognition, and overall changes in skin properties do not affect the differences in measurements between different contact points.

6.4.2 Influence of Different Postures. We evaluate the accuracy of handwriting and keystroke recognition for the target users in various postures: walking, standing, and sitting. Notably, these users did not require any supplementary calibration while changing their motion postures. The experimental results, which are shown in Fig 15(a), illustrate that the *HandPad* system is resilient to variations in user motion. In this way, it can be ensured that the change in capacitance is uniquely derived from the method hair of our proposed modulation of human capacitance. Although the user's movement does cause a change in the extrinsic capacitance, it is very weak compared to the magnitude of the modulation of the intrinsic capacitance, which does not affect the handwriting recognition accuracy of the system.

6.4.3 Influence of Different Environment. The recognition accuracy of *HandPad* in two different environments is shown in Fig. 15(b) and the system does not need to be retrained for different environments. The conference room contains a 70-inch TV and one central air conditioner, while the office has 42 computer-screen pairs, 4 microwaves, 11 laptops, and 6 central air conditioners, leading to higher EM noise. The effect of environmental changes on input accuracy is minimal in both cases. Moreover, temperature affects human capacitance due to shifts in body tissue permittivity [54]. Handwriting input, based on normalized capacitance, remains stable despite temperature changes. Keystroke input, using absolute capacitance, functions properly after a quick recalibration when experiencing significant temperature changes. The calibration mechanism requires only 2 seconds. Experimental results indicate that when hand temperature varies from $25^{\circ}C$ to $36^{\circ}C$, our system achieves a minimum keystroke accuracy of 71.2% and handwriting accuracy of 96.3% without recalibration. Following recalibration, the minimum accuracies improve to 99.8% for keystrokes and 97.7% for handwriting as shown in Fig. 15(c).

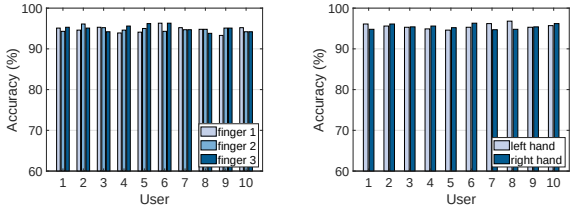
6.4.4 Influence of Different Fingers and Hands. Besides the temperature, hand moisture significantly influences human capacitance. Calibration ensures high-precision input even after handwashing. As shown in Fig. 15(d), without recalibration, accuracy declines to 54.2% for keystrokes and 71.6% for handwriting when hands are wet. Recalibration restores accuracy to 98.9% for keystrokes and 97.1% for handwriting.

Furthermore, we verified that the trend of the human capacitance with touch position is similar between different fingers in the feasibility experiment. In this experiment, we further verified the accuracy of handwriting input with different fingers. We invited



(a) Influence of different postures (b) Influence of different environment. (c) Influence of different temperatures. (d) Influence of hand moisture.

Figure 15: Performance of the *HandPad* system under various factors includes: (a) the influence of different postures (b) the influence of different postures (c) the influence of different temperatures. (d) the influence of hand moisture.



(a) Influence of different fingers. (b) Influence of different hands.

Figure 16: Impacts on system performance when deploying *HandPad* on different fingers or different hands.

all the target users for the evaluation, and each target user was instructed to use the index (Finger 1), middle (Finger 2), and ring (Finger 3) fingers of their right hand to touch the back of their left hand while writing. As shown in Fig. 16(a), stable handwriting interaction can be achieved by using different fingers. In addition, we experimented the handwriting recognition on different hands and verified the influence on system performance. For different handwriting inputs, target users were asked to collect the training data on the left and right hand respectively. As shown in Fig. 16(b), the system maintained a stable handwriting recognition accuracy. Therefore, the keystroke and handwriting interaction system based on human capacitance is robust and can adapt to different users.

7 DISCUSSION

Limitation. The recognition of Chinese characters in *HandPad* is achieved through stroke sequence matching, and the method based on sequence matching cannot distinguish between Chinese characters with the same stroke sequence. The language processing model based on transformer [61] can combine contextual information to identify the correct handwriting input in the current context.

Furthermore, we developed distinct models for the recognition of letters, numbers, and Chinese characters, achieving high accuracies of 99.1%, 97.6%, and 97.9%, respectively. While a unified inference pipeline could potentially reduce storage requirements, training a single model for the combined task of recognizing numbers, letters, and strokes resulted in an accuracy of 92.6%, representing a significant decrease in performance.

Future Work. We verified the positioning of touch points based on the finger model, and the human capacitive signal of three channels can reflect the trajectory of finger movements. Moreover, the slight deformation of the trajectory does not affect the recognition of handwriting letters, numbers, and strokes. Therefore, in future work, we will further extend the system to other handwriting inputs, such as Japanese, Korean, and other languages. Furthermore, PIN-based password [46] input can also be implemented on three fingers based on the finger model. On the other hand, the three channels limit the accuracy of horizontal localization and the system is only able to identify which finger is touched. However, combined with contextual information, it is worth trying to reason about the movement trajectory from the coarse-grained 2D position. We will attempt to generate fine-grained trajectories using methods such as diffusion model [13] in future work.

Moreover, we believe a wireless version of our system is feasible. Using short-range wireless technologies like Bluetooth, capacitive data can be transmitted from fingertips to wearable devices such as smartwatches. Advances in technology have produced compact Bluetooth chips, like the nrf52832 [57](4mmx4mm), and small capacitive sensor chips, like the TI FDC2214 (3mmx3mm). By employing flexible circuit board technology, we can integrate the capacitance chip, electrode sheet, Bluetooth SoC chip, and a miniature arc battery [24] into a fingertip cot design. Insights from prior work [48, 72] on sensor integration into wearables inform our move to develop a wireless smart finger-cap device, a focus for future research.

Meanwhile, temperature and moisture sensors are highly effective for monitoring the system environment and prompting users to recalibrate, thereby ensuring accuracy and reliability, which is an important aspect of future work.

8 CONCLUSION

This paper presents a human-computer interaction system based on human capacitance. The proposed *HandPad* system uses the capacitive sensor of three channels to measure the human capacitance at the fingertips and enables the interaction with finger touch on the left hand. The system mines the movement information of the touch point from the human capacitance signal to achieve the keystroke and handwriting input. The system uses transfer learning to build a recognition model for the target user, which reduces the burden of training set collection. The experiments demonstrate the feasibility of *HandPad* for human-computer interaction, and the system can meet the text input requirements in different scenarios.

ACKNOWLEDGMENTS

We are grateful to all anonymous reviewers for their constructive comments. This work is supported in part by the NSFC (61936015, 62072306).

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. https://doi.org/10.1109/TVCG.2021.3067776

[2] A Ahlbom, U Bergqvist, JH Bernhardt, JP Cesarini, M Grandolfo, M Hietanen, AF Mckinlay, MH Repacholi, David H Sliney, J AJ Stolwijk, et al. 1998. Guidelines for limiting exposure to time-varying electric, magnetic, and electromagnetic fields (up to 300 GHz). *Health physics* 74, 4 (1998), 494–521.

[3] Karan Ahuja, Paul Streli, and Christian Holz. 2021. TouchPose: Hand Pose Prediction, Depth Estimation, and Touch Classification from Capacitive Images. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 997–1009. https://doi.org/10.1145/3472749.3474801

[4] Tobias Boceck, Sascha Sprott, Huy Viet Le, and Sven Mayer. 2019. Force Touch Detection on Capacitive Sensors Using Deep Neural Networks *(MobileHCI '19)*. Article 42, 6 pages.

[5] Jingye Chen, Bin Li, and Xiangyang Xue. 2021. Zero-Shot Chinese Character Recognition with Stroke-Level Decomposition. *CoRR* abs/2106.11613 (2021). arXiv:2106.11613 https://arxiv.org/abs/2106.11613

[6] Mingshi Chen, Panlong Yang, Jie Xiong, Maotian Zhang, Youngki Lee, Chaocan Xiang, and Chang Tian. 2019. Your Table Can Be an Input Panel: Acoustic-Based Device-Free Interaction Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article 3 (mar 2019), 21 pages. https://doi.org/10.1145/3314390

[7] Taizhou Chen, Tianpei Li, Xingyu Yang, and Kening Zhu. 2023. Efring: Enabling thumb-to-index-finger microgesture interaction through electric field sensing using single smart ring. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–31.

[8] Wenqiang Chen, Lin Chen, Yandao Huang, Xinyu Zhang, Lu Wang, Rukhsana Ruby, and Kaishun Wu. 2019. Taprint: Secure Text Input for Commodity Smart Wristbands *(MobiCom '19)*. Article 17, 16 pages.

[9] Wenqiang Chen, Maoning Guan, Yandao Huang, Lu Wang, Rukhsana Ruby, Wen Hu, and Kaishun Wu. 2018. ViType: A Cost Efficient On-Body Typing System through Vibration. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. 1–9. https://doi.org/10.1109/SAHCN.2018.8397098

[10] Frederick Choi, Sven Mayer, and Chris Harrison. 2021. 3D Hand Pose Estimation on Conventional Capacitive Touchscreens. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction* (Toulouse amp; Virtual, France) *(MobileHCI '21)*. Association for Computing Machinery, New York, NY, USA, Article 3, 13 pages. https://doi.org/10.1145/3447526.3472045

[11] Gabe Cohn, Daniel Morris, Shwetak N. Patel, and Desney S. Tan. 2011. Your Noise is My Command: Sensing Gestures Using the Body as an Antenna. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 791–800. https://doi.org/10.1145/1978942.1979058

[12] Christian Corsten, Simon Voelker, Andreas Link, and Jan Borchers. 2018. Use the Force Picker, Luke: Space-Efficient Value Input on Force-Sensitive Mobile Touchscreens *(CHI '18)*. 1–12.

[13] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. 2023. Diffusion Models in Vision: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023), 1–20. https://doi.org/10.1109/TPAMI.2023.3261988

[14] Dian Ding, Lanqing Yang, Yi-Chao Chen, and Guangtao Xue. 2021. VibWriter: Handwriting Recognition System based on Vibration Signal. In *2021 18th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. 1–9. https://doi.org/10.1109/SECON52354.2021.9491615

[15] Dian Ding, Lanqing Yang, Yi-Chao Chen, and Guangtao Xue. 2022. Handwriting Recognition System Leveraging Vibration Signal on Smartphones. *IEEE Transactions on Mobile Computing* (2022), 1–1. https://doi.org/10.1109/TMC.2022.3148172

[16] Dian Ding, Lanqing Yang, Yi-Chao Chen, and Guangtao Xue. 2022. Leakage or Identification: Behavior-Irrelevant User Identification Leveraging Leakage Current on Laptops. *PACM IMWUT.* 5, 4, Article 152 (dec 2022), 23 pages.

[17] David Dobbelstein, Christian Winkler, Gabriel Haas, and Enrico Rukzio. 2017. PocketThumb: A Wearable Dual-Sided Touch Interface for Cursor-Based Control of Smart-Eyewear. *PACM IMWUT.* 1, 2, Article 9 (jun 2017), 17 pages. https://doi.org/10.1145/3090055

[18] Fengyi Fang, Hongwei Zhang, Lishuang Zhan, Shihui Guo, Minying Zhang, Juncong Lin, Yipeng Qin, and Hongbo Fu. 2023. Handwriting Velcro: Endowing AR Glasses with Personalized and Posture-Adaptive Text Input Using Flexible Touch Sensor. *PACM IMWUT.* 6, 4, Article 163 (jan 2023), 31 pages.

[19] 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* (<conf-loc>, <city>Honolulu</city>, <state>HI</state>, <country>USA</country>, </conf-loc>) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–7.

[20] TI FDC2214. 2022. https://www.ti.com/product/FDC2214

[21] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[22] Oliver Glauser, Shihao Wu, Daniele Panozzo, Otmar Hilliges, and Olga Sorkine-Hornung. 2019. Interactive Hand Pose Estimation Using a Stretch-Sensing Soft Glove. *ACM Trans. Graph.* 38, 4, Article 41 (jul 2019), 15 pages. https://doi.org/10.1145/3306346.3322957

[23] Alix Goguey, Sylvain Malacria, and Carl Gutwin. 2018. Improving Discoverability and Expert Performance in Force-Sensitive Text Selection for Touch Devices with Mode Gauges. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3174051

[24] Grepow. [n. d.]. https://www.grepow.com/wearables/smart-ring.html

[25] Zhengxin Guo, Fu Xiao, Biyun Sheng, Huan Fei, and Shui Yu. 2020. WiReader: Adaptive Air Handwriting Recognition Based on Commercial WiFi Signal. *IEEE Internet of Things Journal* 7, 10 (2020), 10483–10494. https://doi.org/10.1109/JIOT.2020.2997053

[26] Aakar Gupta, Muhammed Anwar, and Ravin Balakrishnan. 2016. Porous Interfaces for Small Screen Multitasking Using Finger Identification *(UIST '16)*. New York, NY, USA, 145–156.

[27] Aakar Gupta and Ravin Balakrishnan. 2016. DualKey: Miniature Screen Text Entry via Finger Identification. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 59–70. https://doi.org/10.1145/2858036.2858052

[28] Chris Harrison, Julia Schwarz, and Scott E. Hudson. 2011. TapSense: Enhancing Finger Interaction on Touch Surfaces *(UIST '11)*. 627–636.

[29] 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*. 770–778.

[30] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991* (2015).

[31] Kaori Ikematsu, Masaaki Fukumoto, and Itiro Siio. 2019. Ohmic-Sticker: Force-to-Motion Type Input Device That Extends Capacitive Touch Surface *(UIST '19)*. 1021–1030.

[32] Kaori Ikematsu and Shota Yamanaka. 2020. ScraTouch: Extending Interaction Technique Using Fingernail on Unmodified Capacitive Touch Surfaces. *PACM IMWUT.* 4, 3, Article 81 (sep 2020), 19 pages.

[33] Yan Jiang, Xiaoyu Ji, Kai Wang, Chen Yan, Richard Mitev, Ahmad-Reza Sadeghi, and Wenyuan Xu. 2022. WIGHT: Wired Ghost Touch Attack on Capacitive Touchscreens. In *2022 IEEE Symposium on Security and Privacy (SP)*. 984–1001. https://doi.org/10.1109/SP46214.2022.9833740

[34] Niels Jonassen. 1998. Human body capacitance: static or dynamic concept?[ESD]. In *Electrical Overstress/Electrostatic Discharge Symposium Proceedings. 1998 (Cat. No. 98TH8347)*. IEEE, 111–117.

[35] Wolf Kienzle, Eric Whitmire, Chris Rittaler, and Hrvoje Benko. 2021. ElectroRing: Subtle Pinch and Touch Detection with a Ring. 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 3, 12 pages. https://doi.org/10.1145/3411764.3445094

[36] Jiwan Kim and Ian Oakley. 2022. SonarID: Using Sonar to Identify Fingers on a Smartwatch *(CHI '22)*. Article 287, 10 pages.

[37] Maxime W Lafarge, Josien PW Pluim, Koen AJ Eppenhof, Pim Moeskops, and Mitko Veta. 2017. Domain-adversarial neural networks to address the appearance variability of histopathology images. In *Third International Workshop, DLMIA 2017*. Springer, 83–91.

[38] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. PalmTouch: Using the Palm as an Additional Input Modality on Commodity Smartphones *(CHI '18)*. 1–13.

[39] Huy Viet Le, Sven Mayer, and Niels Henze. 2018. InfiniTouch: Finger-Aware Interaction on Fully Touch Sensitive Smartphones. 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, 779–792. https://doi.org/10.1145/3242587.3242605

[40] Huy Viet Le, Sven Mayer, and Niels Henze. 2019. Investigating the Feasibility of Finger Identification on Capacitive Touchscreens Using Deep Learning *(IUI '19)*. Association for Computing Machinery, New York, NY, USA, 637–649.

[41] Chang-Ju Lee, Jong Kang Park, Canxing Piao, Han-Eol Seo, Jaehyuk Choi, and Jung-Hoon Chun. 2018. Mutual Capacitive Sensing Touch Screen Controller for Ultrathin Display with Extended Signal Passband Using Negative Capacitance. *Sensors* 18, 11 (2018). https://doi.org/10.3390/s18113637

[42] Yijie Li, Yi-Chao Chen, Xiaoyu Ji, Hao Pan, Lanqing Yang, Guangtao Xue, and Jiadi Yu. 2021. Screenid: Enhancing qrcode security by fingerprinting screens. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 1–10.

[43] Yijie Li, Juntao Zhou, Dian Ding, Yi-Chao Chen, Lili Qiu, Jiadi Yu, and Guangtao Xue. 2024. MuDiS: An Audio-independent, Wide-angle, and Leak-free Multi-directional Speaker. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*. 263–278.

[44] Yihao Liu, Kai Huang, Xingzhe Song, Boyuan Yang, and Wei Gao. 2020. MagHacker: Eavesdropping on Stylus Pen Writing via Magnetic Sensing from Commodity Mobile Devices *(MobiSys '20)*. 148–160.

[45] Zongjian Liu, Jieling He, Jianjiang Feng, and Jie Zhou. 2023. PrinType: Text Entry via Fingerprint Recognition. *PACM IMWUT*. 6, 4, Article 174 (jan 2023), 31 pages.

[46] Philipp Markert, Daniel V. Bailey, Maximilian Golla, Markus Dürmuth, and Adam J. Aviv. 2020. This PIN Can Be Easily Guessed: Analyzing the Security of Smartphone Unlock PINs. In *IEEE Symposium on Security and Privacy (SP)*. 286–303. https://doi.org/10.1109/SP40000.2020.00100

[47] Sven Mayer, Huy Viet Le, and Niels Henze. 2017. Estimating the Finger Orientation on Capacitive Touchscreens Using Convolutional Neural Networks *(ISS '17)*. 220–229.

[48] Akihito Miyamoto, Sungwon Lee, Nawalage Florence Cooray, Sunghoon Lee, Mami Mori, Naoji Matsuhisa, Hanbit Jin, Leona Yoda, Tomoyuki Yokota, Akira Itoh, et al. 2017. Inflammation-free, gas-permeable, lightweight, stretchable on-skin electronics with nanomeshes. *Nature nanotechnology* 12, 9 (2017), 907–913.

[49] Hyoungsik Nam, Ki-Hyuk Seol, Junhee Lee, Hyeonseong Cho, and Sang Won Jung. 2021. Review of Capacitive Touchscreen Technologies: Overview, Research Trends, and Machine Learning Approaches. *Sensors* 21, 14 (2021). https://doi.org/10.3390/s21144776

[50] Ian Oakley, Carina Lindahl, Khanh Le, DoYoung Lee, and MD. Rasel Islam. 2016. The Flat Finger: Exploring Area Touches on Smartwatches *(CHI '16)*. New York, NY, USA, 4238–4249.

[51] Hao Pan, Yi-Chao Chen, Qi Ye, and Guangtao Xue. 2021. MagicInput: Training-Free Multi-Lingual Finger Input System Using Data Augmentation Based on MNISTs *(IPSN '21)*. New York, NY, USA, 119–131.

[52] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2010), 1345–1359.

[53] Maicon D. Pereira, Germán A. Alvarez-Botero, and Fernando Rangel de Sousa. 2015. Characterization and Modeling of the Capacitive HBC Channel. *IEEE Transactions on Instrumentation and Measurement* 64, 10 (2015), 2626–2635. https://doi.org/10.1109/TIM.2015.2420391

[54] Ronald Pethig. 1987. Dielectric properties of body tissues. *Clinical Physics and Physiological Measurement* 8, 4A (1987), 5.

[55] Simon Rogers, John Williamson, Craig Stewart, and Roderick Murray-Smith. 2011. AnglePose: Robust, Precise Capacitive Touch Tracking via 3d Orientation Estimation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 2575–2584. https://doi.org/10.1145/1978942.1979318

[56] Juan-Yao Ruan, Paul C.-P. Chao, and Wei-Dar Chen. 2010. A multi-touch interface circuit for a large-sized capacitive touch panel. In *SENSORS, 2010 IEEE*. 309–314. https://doi.org/10.1109/ICSENS.2010.5689881

[57] Nordic Semiconductor. [n. d.]. https://www.nordicsemi.com/Products/nRF52832

[58] System Usability Scale (SUS). [n. d.]. https://www.usability.gov/how-to-andtools/methods/system-usability-scale.html.

[59] NASA task load Index (NASA-TLX). [n. d.]. https://humansystems.arc.nasa.gov/groups/tlx/downloads/TLXScale.pdf

[60] Edward Jay Wang, Jake Garrison, Eric Whitmire, Mayank Goel, and Shwetak Patel. 2017. Carpacio: Repurposing Capacitive Sensors to Distinguish Driver and Passenger Touches on In-Vehicle Screens *(UIST '17)*. 49–55.

[61] Shaolei Wang, Baoxin Wang, Jiefu Gong, Zhongyuan Wang, Xiao Hu, Xingyi Duan, Zizhuo Shen, Gang Yue, Ruiji Fu, Dayong Wu, Wanxiang Che, Shijin Wang, Guoping Hu, and Ting Liu. 2020. Combining ResNet and Transformer for Chinese Grammatical Error Diagnosis. In *Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications*. 36–43.

[62] Eric Whitmire, Mohit Jain, Divye Jain, Greg Nelson, Ravi Karkar, Shwetak Patel, and Mayank Goel. 2017. DigiTouch: Reconfigurable Thumb-to-Finger Input and Text Entry on Head-Mounted Displays. *PACM IMWUT*. 1, 3, Article 113 (sep 2017), 21 pages.

[63] Mathias Wilhelm, Daniel Krakowczyk, and Sahin Albayrak. 2020. PeriSense: ring-based multi-finger gesture interaction utilizing capacitive proximity sensing. *Sensors* 20, 14 (2020), 3990.

[64] WK Wong, Filbert H Juwono, and Brendan Teng Thiam Khoo. 2021. Multi-features capacitive hand gesture recognition sensor: A machine learning approach. *IEEE sensors journal* 21, 6 (2021), 8441–8450.

[65] Kaishun Wu, Qiang Yang, Baojie Yuan, Yongpan Zou, Rukhsana Ruby, and Mo Li. 2021. EchoWrite: An Acoustic-Based Finger Input System Without Training. *IEEE Transactions on Mobile Computing* 20, 5 (2021), 1789–1803. https://doi.org/10.1109/TMC.2020.2973094

[66] Robert Xiao, Julia Schwarz, and Chris Harrison. 2015. Estimating 3D Finger Angle on Commodity Touchscreens *(ITS '15)*. 47–50.

[67] Zhenyu Yan, Qun Song, Rui Tan, Yang Li, and Adams Wai Kin Kong. 2019. Towards Touch-to-Access Device Authentication Using Induced Body Electric Potentials *(MobiCom '19)*. Article 23, 16 pages.

[68] Shanhe Yi, Zhengrui Qin, Ed Novak, Yafeng Yin, and Qun Li. 2016. GlassGesture: Exploring head gesture interface of smart glasses. In *IEEE INFOCOM 2016*. 1–9. https://doi.org/10.1109/INFOCOM.2016.7524542

[69] Huanpu Yin, Anfu Zhou, Guangyuan Su, Bo Chen, Liang Liu, and Huadong Ma. 2020. Learning to Recognize Handwriting Input with Acoustic Features. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 2, Article 64 (jun 2020), 26 pages.

[70] Yongsang Yoo and Byong-Deok Choi. 2021. Readout Circuits for Capacitive Sensors. *Micromachines* 12, 8 (2021).

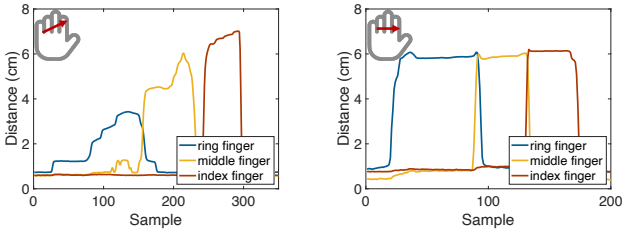
[71] Yongzhao Zhang, Wei-Hsiang Huang, Chih-Yun Yang, Wen-Ping Wang, Yi-Chao Chen, Chuang-Wen You, Da-Yuan Huang, Guangtao Xue, and Jiadi Yu. 2020. Endophasia: Utilizing Acoustic-Based Imaging for Issuing Contact-Free Silent Speech Commands. *PACM IMWUT*. 4, 1, Article 37 (mar 2020), 26 pages.

[72] Hao Zhou, Taiting Lu, Yilin Liu, Shijia Zhang, Runze Liu, and Mahanth Gowda. 2023. One ring to rule them all: An open source smartring platform for finger motion analytics and healthcare applications. In *Proceedings of the 8th ACM/IEEE Conference on Internet of Things Design and Implementation*. 27–38.

APPENDIX

A LINE TEST

The line drawing test is a common method used in 2D localization. To verify the feasibility of the writing pad, the volunteers were asked to slide on the surface of three fingers in two directions. As *HandPad* only uses a three-channel capacitive sensor, the three-channel results cannot be directly compared to the two-dimensional spatial position. We tried to visualize the trajectory of the touch-points by adding a timeline. As shown in Fig. 17(a) and Fig. 17(b), we tested two movements, an oblique upward swipe and a horizontal swipe. It can be seen that when sliding obliquely upwards, the position of the contact point gradually increases in the three channels; when sliding horizontally, the position of the contact point remains stable. Based on the above experiments, we attempted to extend the interaction system with handwriting input, including letters, numbers, and Chinese characters.



(a) Position the finger slide. (b) Position the finger slide.

Figure 17: The line test: (a) and (b) show the capacitance change curves captured by the three electrodes during cross-finger writing.

B NETWORK ARCHITECTURES

We detail each network module of our architecture in the subsequent paragraphs.

Feature Extractor. For the input sequence s_i, we first use a bidirectional long short-term memory (LSTM) to find and exploit long-range dependencies in the data. The number of features in the hidden state is 256, and the number of layers is 2. Next, after the $BatchNorm - Dropout - ReLU$ layers, we use two fully convolutional network-based (FCN-based) residual network (ResNet) blocks to further extract sequence features. The ResNet block consists of three convolution blocks with a structure of $Conv - BN - ReLU$ where the first $Conv$ downsamples by a factor of 2. The convolutional layers within the first ResNet block have 128 filters with a kernel size of $1 \times 1, 3 \times 3, 1 \times 1$. The convolutional layers within the second ResNet block have 256, 256, 64 filters.

A separate convolutional layer is applied to the input of the ResNet layer (the number of 1×1 filter for two ResNet blocks is 128 and 64), and the output of the ResNet block is the sum of the above two components. These ResNet blocks weaken the degradation problem caused by over-deepening of the network during feature extraction, allowing more essential features of the sequence to be preserved. Then, we can obtain the features f_i extracted from the sequence s_i.

Classifier. The classifier is a network consisting of two ResNet blocks, a pooling layer, and a fully connected layer. The structure of the ResNet block is the same as in the feature extractor, and the f_i is further integrated and processed by these ResNet blocks.

Eventually, the fully connected layer with modules of the form $Linear - ReLU$ outputs the possibility $\hat{l}_i$ that the sequence belongs to each label. Specifically, the layer is a Multilayer Perceptron with one hidden layer with 128 units and an output layer with the same number of units as labels.

Domain Discriminator. Domain discriminator is an FCN-based structure with three $Conv - BN - ReLU$ layers, where the convolutional layers have 32 1×1,16 3×3, 8 1×1 filters and stride 2, 2, 2 respectively. After the last convolutional layer, a pooling layer is applied to downsampling, followed by a *Sigmoid* function. The *Sigmoid* function is applied to all elements of input and will re-scale them so that every element in $\hat{d}_i$ lies in the range $[0, 1]$. For the domain discriminator, the output $\hat{d}_i$ converges to 1 when the input feature $f_i = G_f(s_i, \theta_f), s_i \in u_i$, while the output $\hat{d}_i$ converges to 0 when the input feature $f_i = G_f(s_i, \theta_f), s_i \in t_i$.

C MODEL OPTIMIZATION AND INFERENCE

We use a model based on DANN to accomplish the above goal. As shown in Fig. 9, the model has three main parts: **(i)** Feature Extractor. **(ii)** Classifier. **(iii)** Domain Discriminator.

Namely, let $G_f(\cdot, \theta_f)$ be the feature extractor that maps each input human capacitance signal sequential s_i to a deep feature representation f_i, with parameters θ_f. Also, let $G_c(\cdot, \theta_c)$ be the classifier that receives the output of the feature extractor f_i and computes the network's label prediction $\hat{l}_i$, with parameters θ_c, while $G_d(\cdot, \theta_d)$ corresponds to the discriminator that is responsible for the computation of domain predictions $\hat{d}_i$, with parameters θ_d. Meanwhile, we use d_i to define whether a training sample u_i or t_i belongs to the underlying domain or the target domain.

We define the label prediction loss and the domain prediction loss respectively by the $CrossEntropy$ loss

$$\begin{aligned} \mathcal{L}_c^i(\theta_f, \theta_c) &= \mathcal{L}_c(\hat{l}_i, l_i) = \mathcal{L}_c(G_c(f_i; \theta_c), l_i) \\ &= \mathcal{L}_c(G_c(G_f(s_i; \theta_f); \theta_c), l_i) \end{aligned} \tag{10}$$

and the $BCEWithLogits$ loss

$$\begin{aligned} \mathcal{L}_d^i(\theta_f, \theta_d) &= \mathcal{L}_d(\hat{d}_i, d_i) = \mathcal{L}_d(G_d(f_i; \theta_d), d_i) \\ &= \mathcal{L}_d(G_d(G_f(s_i; \theta_f); \theta_d), d_i) \end{aligned} \tag{11}$$

The training of our recognition model is performed by optimizing the overall loss function:

$$\begin{aligned} \mathcal{L}(\theta_f, \theta_c, \theta_d) = \sum_{i=1}^{n} (\mathcal{L}_c^i(\theta_f, \theta_c) + \lambda \mathcal{L}_d^i(\theta_f, \theta_d)) + \\ \beta \sum_{i=n+1}^{N} (\mathcal{L}_c^i(\theta_f, \theta_c) + \lambda \mathcal{L}_d^i(\theta_f, \theta_d)) \end{aligned} \tag{12}$$

where the parameter $\beta \in [0, 1]$ is scale-related to the size of the training underlying dataset and target user dataset.

The parameters of the whole network are optimized via gradient back-propagation during an iterative training process that consists of three successive update rules:

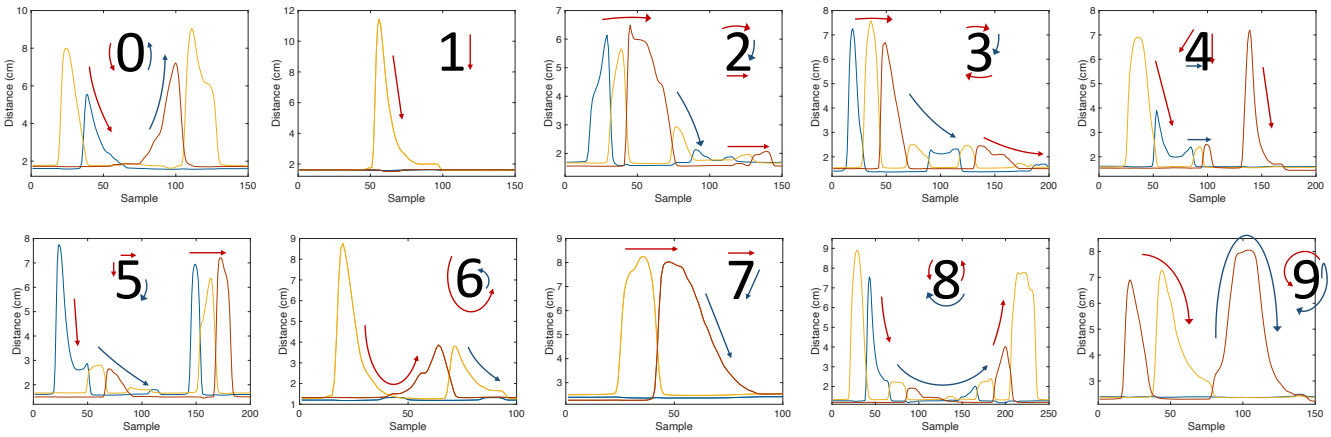


Figure 18: Visualization of handwriting numbers: the blue, yellow, and red lines indicate the distance from the contact point to the root of the ring finger, middle finger, and index finger of the left hand respectively.

- Optimization of the feature extractor with learning rate μ_f:

$$\theta_f \leftarrow \theta_f - \mu_f\left(\frac{\partial \mathcal{L}_c^i}{\partial \theta_f} - \lambda \frac{\partial \mathcal{L}_d^i}{\partial \theta_f}\right) \tag{13}$$

- Optimization of the classifier with learning rate μ_c:

$$\theta_c \leftarrow \theta_c - \mu_c \frac{\partial \mathcal{L}_c^i}{\partial \theta_c} \tag{14}$$

- Optimization of the domain discriminator with learning rate μ_d:

$$\theta_d \leftarrow \theta_d - \mu_d \lambda \frac{\partial \mathcal{L}_d^i}{\partial \theta_d} \tag{15}$$

The update rules Eq.13 and Eq.15 work in an adversarial way: with Eq.15, the parameters θ_d are updated for the domain prediction task (by minimizing $\mathcal{L}_d$). By finding optimized θ_d, the domain discriminator has a solid capability to distinguish the features f_i extracted from the sequences s_i from domain D_U and domain D_T. Furthermore, with Eq.13, the parameters θ_f are updated to ensure that the features extracted by the feature extractor over the two domains D_U and D_T have similar distributions (by maximizing $\mathcal{L}_d$). This optimization process makes the extracted features f_i as indistinguishable as possible for the domain discriminator, thus resulting in the domain-invariant features. Meanwhile, as shown in Fig. 9, such a gradient reversal optimization process can be accomplished by a *gradient reversal layer* (GRL) which requires no parameter update. The parameter $\lambda \in [0, 1]$ controls the intensity of the adversarial component.

D STROKE SEQUENCE MATCHING ALGORITHM

The target user writes a single Chinese character and a series of stroke sequences $S = \{s_j\}_{j=1}^k$ is captured by the capacitive sensor. Then, the models mentioned above (feature extractor and classifier) calculate the stroke label probability vector $\hat{l}_j$ for each signal sequence $s_j \in S$. In the database of the *HandPad* system, Chinese characters are stored as a tuple (h_i, C_i) where h_i corresponds to the $UTF-8$ encoding of the i-th character and $C_i = \{c_{ij}\}_{j=1}^k$ represents its stroke sequence vector. With all the stroke label possibility vector $I = \{\hat{l}_j\}_{j=1}^k$ as input, *HandPad* used the stroke sequence matching algorithm to find the most accurate matching of Chinese characters from the database as shown in Algo. 1.

Algorithm 1 Stroke Sequence Matching Algorithm

Input: $I = \{\hat{l}_i\}_{i=1}^k$
Output: Chinese characters h

```
Initialize Φ with all tuples (h_i, C_i) that satisfy the condition that length(C_i) = k.
Obtain the most likely stroke input sequence Ĉ = {c_j}_{j=1}^k of target user where c_j = argmax(l̂_j).
Initialize Ψ of the same length as Φ to record the matching degree between each C_i ∈ Φ and Ĉ.
for all (h_i, C_i) ∈ Φ do
    Compare c_ij ∈ C_i and ĉ_j ∈ Ĉ_j case-by-case.
    Ψ[i] ← (MN_i, p_i) where MN_i is the number of matching elements and p_i = Σ_{c_ij != ĉ_j} l̂_j[c_ij]
    In particular,
    if MN_i = k then
        return h_i
    end if
end for
return h_i whose corresponding Ψ[i] has the highest MN_i and has the maximum p_i if others have the same MN_i as it.
```

E USER STUDY

In this section, we examine the usability and workload of *HandPad*. We invited 40 participants to use each application for 20 minutes per day for one week. We used a standard methodology based on the System Usability Scale (SUS) [58] to study the user experience with 11 questions. Specifically, the questions in the questionnaire have five response options ranging from "strongly agree" to "strongly disagree". The questionnaire is as follows:

	Strongly Disagree (-2)	Disagree (-1)	Not sure (0)	Agree (1)	Strongly Agree (2)	Average Rating
Q1	24	9	4	3	0	-1.35
Q2	1	3	5	10	21	1.175
Q3	0	4	3	8	25	1.35
Q4	0	2	5	11	21	1.275
Q5	1	3	7	10	19	1.075
Q6	7	26	4	1	2	-0.875
Q7	0	1	3	9	27	1.55
Q8	27	5	5	2	1	-1.375
Q9	0	2	4	14	20	1.3
Q10	4	16	10	8	2	-0.7
Q11	1	2	6	14	17	1.1

Table 1: The result of questionnaire.

About you

In this section, you will be presented with a number of questions about yourself and your opinions and attitudes.

1 Your gender:
- □ Female
- □ Male

2 How old are you?
- □ Below 18
- □ 18-24
- □ 25-34
- □ 35-44
- □ 45-54
- □ 55-64
- □ Above 65

About the user experience

In this section, you will be presented with a number of questions about your opinions and attitudes towards *Handpad*.

After completing the registration and login tasks, please rate your opinions toward *Handpad* with the following aspects.

1 I find the system very uncomfortable to wear
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

2 I think I would like to use the system regularly
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

3 I think the system is easy to use
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

4 I think the system would meet most people's writing needs
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

5 I think the system can accommodate the writing habits of different users
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

6 I would need a long time of training to use this system
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

7 I think most people learn to use the system quickly
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

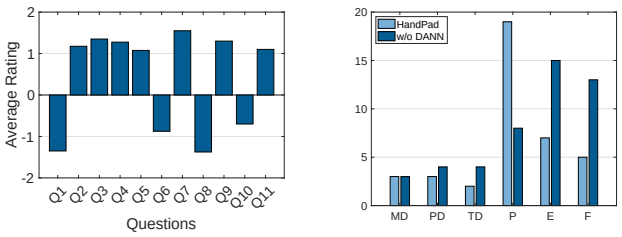
8 I find the system very cumbersome to use
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

9 I think it's acceptable to collect 3 sample for each input
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

10 I think it's acceptable to collect 5 sample for each input
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

11 I think it is beneficial to reduce the amount of data collected.
Strongly Disagree (-2) □—□—□—□—□ Strongly Agree (2)

Please give your advice here.



(a) System Usability Scale. (b) NASA Task Load Index.

Figure 19: User study of the *HandPad* system.

The results of the questionnaire are presented in the Tab. 1 and Fig. 19(a). The majority of participants in the questionnaire expressed positive opinions about the system, highlighting its ease of use.

For the workload indicator, we asked all participants to fill out the NASA Task Load Index [59] to evaluate *HandPad*, including Mental Demand (MD), Physical Demand (PD), Temporal Demand (TD), Performance (P), Effort (E), and Frustration (F). As shown in Fig. 19(b), both systems scored low (below 5) in terms of mental, physical, and time demands. However, *HandPad* performed better overall than the system without DANN (19 VS 8). 21 participants indicated that the advantage of transfer learning was not significant

when handwriting input was limited, which means that the symbols to be input included only numbers. The users only need to collect the dataset of numbers 0-9 for training, which is acceptable for most users. However, when multiple input modalities were involved, such as letters and Chinese character strokes, the total number of samples that needed to be collected by the system without DANN was significantly higher, and the excessively long sample collection time was frustrating.