PalmPen: A Palmprint Positioning Pen

JINYANG YU, Department of Automation, BNRist, Tsinghua University, China
ZHAOGUO WANG, Department of Automation, BNRist, Tsinghua University, China
ZHIYU PAN, Department of Automation, BNRist, Tsinghua University, China
JIANJIANG FENG*, Department of Automation, BNRist, Tsinghua University, China
JIE ZHOU, Department of Automation, BNRist, Tsinghua University, China

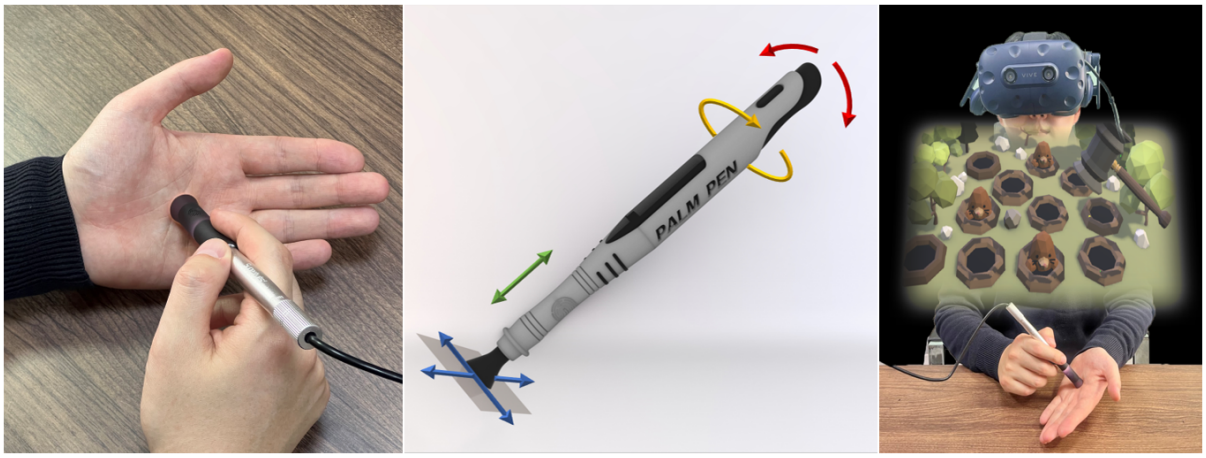

Fig. 1. Left: PalmPen is pen-shaped camera device that enables absolute positioning and continuous tracking on the palm. Middle: Apart from 2D touch positions, PalmPen supports extra input parameters, such as touch pressure, rotation angles, and inclination angles. Right: PalmPen is a versatile and portable input device for VR/AR systems and large displays.

Palm-based interaction offers unique advantages in tasks requiring minimal visual attention and low cognitive load, thanks to the sensitive tactile feedback of the palm and the body's proprioceptive abilities. However, current research in this field faces challenges like limited detection precision and complex equipment. This paper presents PalmPen, a novel pen-shaped camera device that relies on computer vision algorithms to enable absolute positioning and continuous tracking on the palm. We created a training dataset containing full and partial palmprints with corresponding positional data using cameras and an

*Corresponding author

Authors' Contact Information: Jinyang Yu, jy-yu20@mails.tsinghua.edu.cn, Department of Automation, BNRist, Tsinghua University, Beijing, China; Zhaoguo Wang, wangzg24@mails.tsinghua.edu.cn, Department of Automation, BNRist, Tsinghua University, Beijing, China; Zhiyu Pan, pzy20@mails.tsinghua.edu.cn, Department of Automation, BNRist, Tsinghua University, Beijing, China; Jianjiang Feng, jfeng@tsinghua.edu.cn, Department of Automation, BNRist, Tsinghua University, Beijing, China; Jie Zhou, jzhou@tsinghua.edu.cn, Department of Automation, BNRist, Tsinghua University, Beijing, China.

ACM 2474-9567/2018/7-ART
https://doi.org/XXXXXXX.XXXXXXX

optical tracker. A deep learning network was created to predict the location of partial palmprints within a full palmprint. PalmPen achieved a mean positioning error of 2.74 mm in the experiments. A user study with 12 participants demonstrated that PalmPen offers superior positioning accuracy and input efficiency over other methods. Additionally, properties such as touch pressure and rotation angles can be inferred from palmprint sequences, extending PalmPen's application range.

CCS Concepts: • **Human-centered computing** → **Pointing devices**; **Pointing**.

Additional Key Words and Phrases: Palm-based Input, Pen-shaped Device, On-skin Input

ACM Reference Format:
Jinyang Yu, Zhaoguo Wang, Zhiyu Pan, Jianjiang Feng, and Jie Zhou. 2018. PalmPen: A Palmprint Positioning Pen. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 1 (July 2018), 27 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Traditional input devices, such as the mouse and trackpad, have been the most commonly used input methods for several decades. However, their application has been limited in the context of VR/AR devices and large displays due to their inability to provide an immersive and intuitive experience [55]. Additionally, physical constraints—such as the need for a flat surface and a restricted range of motion—render them unsuitable for the dynamic and spatial interactions required in these environments [47]. Although products like pen-shaped mice[1], ring mice[2], and mini touchpads[3] have improved the portability of traditional input devices to some extent, their usability and input efficiency in real-world scenarios remain unclear. Furthermore, traditional mice and trackpads support only two-dimensional relative input, which is inadequate for complex tasks, such as drawing or VR gaming, that require absolute positioning and additional input dimensions.

On-skin input, a readily available input method, has garnered significant attention from researchers [10, 16, 37, 38, 51]. Humans possess the ability to sense their body's position, movement, and orientation in space through proprioception [48]. This innate proprioceptive sense allows the skin to function as an interface for eyes-free interaction. The skin's rich and precise tactile feedback, combined with the brain's proprioceptive abilities, enables individuals to perceive surface information—such as the specific location and movement of tactile signals—without visual input, thus facilitating various input tasks. As a result, on-skin input methods, grounded in proprioceptive tactile perception, hold significant potential for applications in low-vision or vision-free scenarios, as well as in operations requiring minimal cognitive or attentional effort [10, 12, 13, 39, 51]. Among the various on-skin input locations, the palms and fingers are the most commonly used due to their flexibility, sensitivity, and accessibility [4].

The palm, which approximates a two-dimensional plane, is inherently suited for use as a two-dimensional interaction interface. Additionally, the widespread use of palmprints in forensics and law enforcement demonstrates their uniqueness and permanence. Distinct ridge flow patterns have been identified in various regions of the palm [9], allowing for absolute positioning by analyzing and matching specific palmprint patches. However, there has been limited research on utilizing palmprints for two-dimensional input with absolute positioning.

We propose PalmPen, a novel pen-shaped camera device that enables absolute positioning and continuous tracking on the palm. The system begins by capturing a full palmprint image using a phone camera, which serves as a reference. A camera embedded in the pen tip then continuously captures partial palmprint images, which are automatically matched against the full palmprint to determine their positions. These positions are mapped to corresponding interaction points on a screen.

To develop PalmPen, we created a training dataset containing full and partial palmprints with their positional information, using cameras and an optical tracker. A deep learning network was designed to predict the location

[1]https://direct.sanwa.co.jp/ItemAttr/400-MAWBT202
[2]https://direct.sanwa.co.jp/ItemPage/400-MABT156BK
[3]https://www.ergonomictouchpad.com/mini_touchpad.php

of a partial palmprint within the full palmprint, thereby identifying the corresponding interaction points. Experiments demonstrated PalmPen's positioning accuracy, achieving a mean positional error of 2.74 mm. A user study with 12 participants evaluated the device's performance in both controlled laboratory settings and real-world scenarios. In the laboratory, we assessed the device's absolute positioning accuracy with a target tapping task and its input efficiency with a target tracking task. In real-world settings, we tested PalmPen's performance in large-screen display and VR applications. Comparisons with a pen-shaped mouse and a phone-based touchpad confirmed the superiority of our approach.

Additionally, our findings indicate that properties such as touch pressure, rotation angles, and inclination angles can be inferred from sequences of partial palmprints, expanding the range of potential input parameters. Incorporating these properties, we developed several real-world applications to showcase the full potential of PalmPen.

The main contributions of this work are:

- The introduction of PalmPen, a novel pen-shaped camera device that enables absolute positioning and continuous tracking on the palm, enhanced with input parameters such as touch force, rotation, and inclination estimation.
- The development of a deep learning network that accurately locates a partial palmprint within a full palmprint, achieving a positioning error of 2.74 mm.
- A user study evaluating PalmPen's positioning accuracy and input efficiency in four target selection tasks, demonstrating its superiority over a pen-shaped mouse and a phone-based touchpad.

2 Related Work

2.1 On-Skin Input Techniques

Despite employing different skin locations, on-skin input techniques use several common sensing technologies, including external optical sensing, signal propagation through the skin, and touch sensors that make direct contact with the skin [4]. External optical sensing is the most widely used technology for on-skin input. Motion capture systems track reflective markers placed on the skin surface and fingers to determine their positions [5, 13]. Systems using infrared (IR) sensors emit light that reflects off the inputting finger; the finger's location is then estimated based on the angle, amplitude, and other signal characteristics of the reflection, as well as its travel time. For instance, watch-based IR sensors have been directed toward the knuckles to track taps and touch gestures on the back of the hand [23, 27, 51]. Computer vision algorithms are also frequently used to analyze images captured by cameras, determining the touch state and contact position of the finger on the skin [10, 40, 42]. Some on-skin input techniques rely on sensing signals propagated through the skin, such as sound, mechanical vibrations, and electromyography (EMG) [16, 37, 45]. Touch sensors, which directly detect the touching state and position, include capacitive touch sensors, piezoelectric sensors, and pressure sensors [38, 52].

Various body locations have been utilized to provide natural user interfaces, including the palms [10, 13, 22, 51], fingers [8, 26, 30], forearms [5, 25, 45], laps [36], and ears [21]. The palms are among the most frequently used locations for on-skin input due to their flexibility and accessibility. Kohli and Whitton [24] proposed using the non-dominant hand as a user interface to provide haptic feedback in virtual environments. Their system employs magnetic trackers to monitor the palm of the non-dominant hand and the index finger of the dominant hand. Results from their user study demonstrated the feasibility of the proposed method; however, finger touch detection was not accurate due to the non-planar nature of the palm surface, and calibration was required for each user because of variations in palm shape and size. Dezfuli et al. [10] introduced the PalmRC technique, which leverages the palm as an interactive surface for television remote control. In a controlled experiment, they used optical tracking systems to monitor the palm of the non-dominant hand and the index finger of the dominant hand. The results demonstrated the feasibility and effectiveness of this technique, and usability was

further improved by replacing the optical tracking system with a depth camera, eliminating the need for reflective markers. Similarly, Harrison et al. [15] proposed OmniTouch, a wearable depth-sensing and projection system that enables interactive multitouch applications on everyday surfaces, including hands, arms, and legs. Results indicated that reliable operation on the hands requires buttons to be at least 22.5 mm in diameter. Gustafson et al. [12] argued that users inadvertently learn an interface when using a physical device and can transfer that knowledge to an imaginary interface. They proposed an "Imaginary Phone" prototype, allowing users to interact by mimicking the use of a physical phone by tapping and sliding on their empty non-dominant hand without visual feedback. Pointing on the hand is tracked using a depth camera, and touch events are sent wirelessly to an actual phone to invoke corresponding actions. Their results showed that the minimum button size for effective interaction on the palm is 17.7 mm in diameter, significantly smaller than on an empty space interface. In a follow-up study, the authors further investigated the role of visual and tactile cues in palm-based imaginary interfaces and found that tactile cues sensed by the palm, rather than the pointing finger, were most effective for user orientation. The study also demonstrated that eyes-free interfaces on the skin outperform those on physical devices [13]. Beyond touch state and contact position, researchers have explored other properties, such as touch force. Ono et al. [40] used an RGB camera to capture the appearance of the palm and the touching finger. They tracked palm deformation by calculating optical flow, which was then used to estimate the magnitude and direction of the applied force. Research on palm-based interactions using external optical sensing has shown the feasibility and efficiency of using the palm as an interaction interface. However, the main drawback of external optical sensing technologies is their limited portability, which restricts usability in mobile environments.

To improve the portability of palm-based input methods, researchers have proposed various wearable devices. Wang et al. [51] developed EyeWrist, a wrist-mounted prototype that detects gestures on the palm using an infrared (IR) camera and a laser-line projector. This device allows users to interact with other devices by drawing stroke gestures on their palms without needing to look at them. Wang et al. [50] also proposed PalmType, which utilizes the palm as an interactive keyboard for smart wearable displays. Finger positions and taps are detected by a wrist-worn array of 15 IR proximity sensors. Hajika et al. [14] introduced RadarHand, a wrist-worn wearable equipped with millimeter-wave radar to detect on-skin touch gestures. Prätorius et al. [41] proposed Skinteract, a system for on-body interaction that leverages the unique texture of human skin. A capacitive fingerprint sensor captures images and locates the corresponding area within a pre-mapped palm. However, this approach only captured images from 11 regions and evaluated the recognition rate within those regions, limiting it to discrete input like button clicking. Additionally, the proposed matching algorithm has a high computational cost, with a processing time of approximately 3 seconds to match a full palm area of 100 cm^2, which is impractical for real-time interactions. While these wearable devices improve the portability of palm-based input, they primarily focus on discrete input scenarios, such as typing and gesture recognition. In contrast, PalmPen aims to provide accurate absolute positioning and continuous tracking capabilities.

In summary, on-skin input techniques based on external optical sensing deliver high precision but suffer from limited portability and degraded performance in challenging environments. Conversely, wearable on-skin input devices improve portability but lack the capability for continuous, accurate position tracking. The PalmPen addresses these limitations by exploiting the unique patterns of palmprints to enable precise position tracking without relying on external systems. Its lightweight, pen-shaped design ensures high portability, making it suitable for diverse real-world applications.

2.2 Pen-shaped Input Devices

Pen interaction is increasingly becoming a standard input method for precision tasks such as writing, annotating, drawing, and 2D manipulation. Commercial digital pens typically work with sensors integrated into tablet devices. The most affordable pens use passive capacitive tracking, where passive styli function as simple conductors

detected by the touch sensor. In contrast, active capacitive styli (e.g., Microsoft Surface Pen[4]) emit an electrostatic signal that is detected by the touchscreen, providing higher precision. Electromagnetic (EM) styli (e.g., WACOM pen[5]) operate using electromagnetic induction technology. A grid of sensing coils beneath the tablet generates a low-frequency electromagnetic field. When the pen is brought near the tablet surface, the coil inside the pen is powered by this electromagnetic field, allowing the stylus to transmit its signal back to the tablet. The sensing coils in the tablet detect the signal returned by the stylus and calculate its precise location. The pen's position on the 2D plane is determined by analyzing the strength and phase differences of the signals detected by each coil.

In addition to the extra features provided by commercial digital pens, such as pressure sensitivity, inclination, and rotation angles, researchers are exploring new input parameters. Matulic et al. [34] proposed mounting a downward-facing camera at the top end of a digital tablet pen to expand input capabilities. By analyzing images captured by the camera, the device can recognize dominant and non-dominant hand poses, detect tablet grips and hand gestures, capture physical content from the environment, and identify users and the pen. Fang et al. [11] suggested mounting an upward-facing camera at the top of a digital tablet pen to use facial gestures as command triggers. Hwang et al. [19] introduced MagPen, a magnetically driven pen interface that operates both on and around mobile devices. The magnetic field generated by the stylus is used to analyze pen actions and gestures, such as detecting pen orientation, performing dragging motions along the device frame, and identifying pen-spinning gestures. While these additional parameters improve input efficiency, they still rely on a mobile device or tablet to function, limiting their feasibility in truly mobile environments.

On the other hand, some studies focus on pen positioning on unmodified physical surfaces. Lüthi et al. [31] introduced DeltaPen, a pen device that operates on passive surfaces without requiring external tracking systems or active sensing surfaces. DeltaPen integrates two adjacent lens-less optical flow sensors at its tip to accurately reconstruct directional motion and yaw rotation. It also supports tilt interaction using a built-in inertial sensor, and includes a pressure sensor and high-fidelity haptic actuator for pressure sensitivity and haptic feedback. However, this approach only enables relative positioning. Romat et al. [44] developed Flashpen, a digital pen designed for accurate and fluid handwriting and drawing in VR environments. Flashpen utilizes a gaming mouse sensor for precise relative input and optical tracking systems for absolute tracking. It demonstrated similar speed and accuracy to that of a professional drawing tablet in tracing tasks. Maierhöfer et al. [33] proposed TipTrack, a system that tracks the position of an IR-emitting pen tip on a planar surface using two cameras equipped with infrared filters. The touch states are detected based on the light patterns from the captured images of the pen tip. While both Flashpen and TipTrack achieved high tracking accuracy, they still rely on external tracking systems for absolute positioning. In contrast, PalmPen aims to provide high portability and accurate absolute positioning without the need for an external tracking system.

In summary, most pen-shaped input devices depend on external tracking systems or dedicated sensing tablets to achieve continuous position tracking, preventing their use as standalone solutions. While existing standalone pen devices offer relative positioning capabilities, this limitation significantly reduces both input efficiency and potential application scenarios. The PalmPen addresses these challenges by functioning as a fully portable standalone device that transforms the user's palm into an interactive surface. This innovative approach enables absolute positioning while simultaneously capturing detailed pen pose properties.

2.3 Palmprint Recognition

Recently, contactless palmprint identification has gained popularity due to its advantages in user convenience, hygiene, and security [29]. Most contactless palmprint recognition methods focus on full-to-full matching [20]. Key steps in contactless palmprint recognition, such as key point detection, region of interest (ROI) extraction,

[4]https://support.microsoft.com/en-US/surface-pen
[5]https://www.wacom.com/en-us/products/pen-tablets/wacom-intuos-pro

and alignment, typically require full palmprints [1]. Feature extraction and matching are usually performed on these aligned palmprints. However, the feasibility of using a pen tip camera, which captures only a small patch of the palmprint, makes this contactless recognition pipeline unsuitable for PalmPen.

In contrast, contact-based palmprint recognition has been utilized in forensics and law enforcement for decades [28]. Partial palmprints collected from crime scenes can be matched against a database of full palmprints. Most contact-based palmprint recognition algorithms rely on minutiae due to their discriminative power and persistence [6, 9, 20, 28]. However, the limited size of the palmprint patch captured by PalmPen means that the number of minutiae is constrained, which can impact recognition performance. Additionally, minutiae and ridge lines may not be clearly visible in the full palmprint captured by a phone camera due to lighting conditions and camera angle, making minutiae an unreliable feature for recognition.

While traditional palmprint recognition algorithms are designed to verify matches between two palmprints, they are not optimized for small palmprint patch alignment. In contrast, the PalmPen's absolute positioning algorithm is specifically engineered to perform two key functions: (1) precisely localizing a palmprint patch within a full palmprint image, and (2) maintaining real-time processing speeds to ensure fluid interaction. This dual capability enables accurate, instantaneous positioning that supports natural pen input on the palm surface.

3 PalmPen Concept

The concept behind PalmPen is similar to that of the Anoto pen [18], which uses a tiny camera to capture a non-repetitive pattern on a surface and, by analyzing this pattern, determines its absolute position. PalmPen leverages the unique characteristics of the palmprint to provide precise absolute positioning on the palm. The following sections detail the rationale behind PalmPen's template registration and sensing technology.

3.1 Template Registration

Absolute positioning on the palm is achieved by comparing images captured by the pen with a pre-registered palmprint template. One potential registration method is the stitching approach proposed by Skinteract [41]. This method involves sequentially adding new images to an intermediate stitching result by matching scale-invariant feature transform (SIFT) features between pairs of images and computing a homography matrix using random sample consensus (RANSAC). However, stitching a large number of images can introduce cumulative distortion due to the projection of non-planar soft tissue onto a two-dimensional space. Given that the quality of the registered template significantly impacts the pen's positioning accuracy, the stitching approach is unsuitable for full palm registration. An alternative method involves creating a template library consisting of unstitched palmprint patches with their corresponding locations on the palm. However, this approach requires precise tracking of each patch's location, which depends on an external tracking system, thereby significantly increasing the complexity of the registration process. Furthermore, both patch-based registration methods require continuous image capture, which is time-consuming and prone to missing regions, resulting in incomplete templates. To address these issues, we propose using a single photograph of the full palm, taken with a mobile phone camera, as the registered template. This simplifies the registration process from capturing numerous palmprint patches to taking a single photo. This approach ensures the completeness and quality of the template while significantly reducing the registration time.

3.2 Input Device

The input device for PalmPen should have a small form factor and a high image capture rate. Capacitive and optical fingerprint sensors are the most commonly used for capturing fingerprints. Capacitive fingerprint sensors are thin, lightweight chips suitable for portable and wearable devices. However, their frame rates are typically below 10 frames per second (FPS), which is inadequate for interaction purposes. In contrast, optical fingerprint

sensors can achieve higher frame rates, but they require a camera, light source, and prism, making them bulky. Both types of contact-based sensors are also susceptible to poor skin conditions, such as dryness or moisture, and the images they capture differ in modality from the full palmprint captured by a phone camera, complicating the matching process. Alternatively, a camera can be used to capture palmprint patches. Cameras have the potential to be compact while offering a high frame rate. The images captured by the phone camera and the pen tip camera share the same modality, with only slight differences in style, making the matching process more straightforward. For these reasons, a camera is chosen as the sensor of PalmPen.

4 Data Collection

To train the palmprint positioning algorithm, we collected full palmprints, palmprint patches, and their corresponding locations. A data collection system was built using a phone camera, a pen-tip camera, and an optical tracker (see Figure 2a) . A custom program was developed to synchronize the captured data.

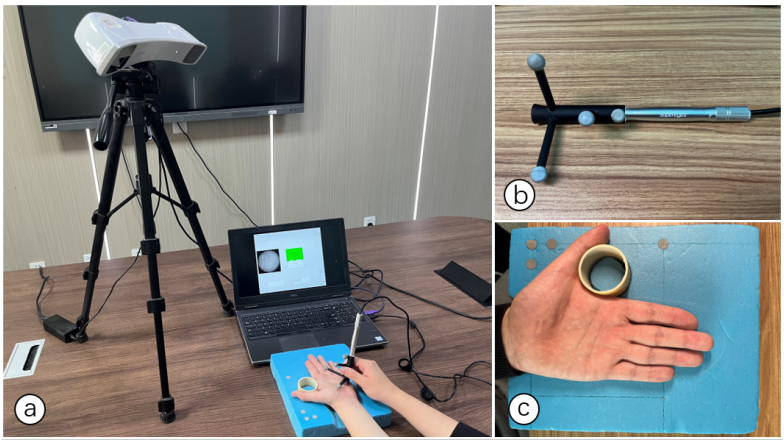

Fig. 2. (a) The setup for the data collection system. (b) The pen-shaped camera with optical tracking markers. (c) The calibration board.

4.1 Apparatus

We employed a PST Iris optical tracking system[6] to track both the palm and the pen. This system is capable of capturing six degrees of freedom (DOF) for registered reflective markers, including 3D spatial positions and 3D orientations. This data is used to calculate the pen's relative position and angles with respect to the palm. The main camera of an iPhone 13 was used to capture full palmprints. The ProCam application was used to set the focus manually at a fixed value. The captured images have a resolution of 4032 × 3024 pixels. The phone was mounted on the adjustable arm of a tripod.

We used a Supereyes B005 handheld digital microscope[7] as the input device for PalmPen (see Figure 2b). This pen-shaped device is 126 mm long and 11 mm in diameter, with an embedded camera equipped with adjustable LED lights. The camera features an adjustable focal length and captures images at 30 frames per second (FPS) with a resolution of 640 × 480 pixels. We designed a 3D-printed black sleeve, 70 mm long with a 14 mm diameter

[6]https://www.ps-tech.com/products-pst-iris/
[7]https://www.supereyes-store.com/collections/microscopes/products/supereyes-b005-200x-handheld-digital-microscope-otoscope-magnifier-w-led-tripod

opening, with four branches to hold reflective markers. The sleeve blocks external light, ensuring that images are illuminated solely by the LED lights in the pen. When installed on the pen, the distance from the camera to the sleeve's opening is fixed at 35 mm, matching the camera's focal length.

The calibration board used for data acquisition measures approximately 200 × 250 × 30 mm (see Figure 2c). A cylindrical guide helps position the palm on the board. This design allows us to indirectly determine the spatial coordinates of the palm area by capturing the spatial coordinates of the calibration board. Four reflective markers are also attached to the board to facilitate a capture model within the optical tracking system. The board's groove design keeps the palm stable and nearly parallel to the board's surface when placed naturally. Although the palm has thickness variations, subsequent image-matching tasks focus only on the coordinates within the 2D plane.

We implemented a program to synchronize data collected from the optical tracker and the pen-shaped camera (see Figure 3). The user interface consists of three columns: the first displays the palmprint patches currently being captured, the second contains an area capture indicator, and the third provides control buttons and text boxes for information recording. The system records the areas of the palm that have been captured based on real-time camera position coordinates, marking these areas with green squares to guide participants. This approach ensures comprehensive data acquisition, avoiding the omission of valuable information.

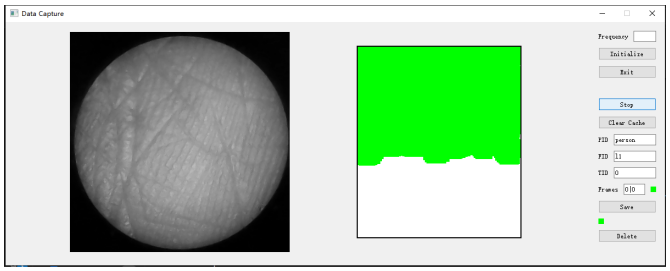

Fig. 3. The data collection program consisting of the cropped palmprint patches, area capture indicator, and the control buttons.

4.2 Participants

For the data collection phase, we recruited 10 participants (8 males, 2 females) aged between 20 and 34 years (M = 22.9, SD = 4.0) from the campus. All participants were right-handed and had prior experience using a mouse and touchpad. Informed consent was obtained from each participant, and their personal information was handled confidentially. None of the participants had any medical conditions or upper limb injuries that could affect their ability to complete the data collection tasks.

4.3 Procedure

Upon welcoming the participants, we introduced the study's objectives and demonstrated the apparatus and tasks. Participants were asked to complete a consent form and provide basic information, including age, gender, dominant hand, and experience with different input devices. Each participant was seated comfortably in front of a desk, with the seat height adjusted so their forearms were level with the desk. Participants were then instructed to place their non-dominant hand in the groove of the calibration board with the palm facing up.

First, a reference image was captured using the phone camera with the pen placed on the participant's hand. The optical tracker recorded the positions of the reflective markers on both the pen and the calibration board. This position data was later used to calibrate the phone camera. The pen was then removed, and three additional photos were taken under different lighting conditions created by an external flashlight. Since the phone camera

remained fixed while capturing these four photos, the palms in the images were perfectly aligned. The pen was then returned to the participant's hand, and the phone camera was moved to a different location. This process was repeated three times to capture the palms from various angles. The camera flash remained on throughout to provide adequate illumination. In total, we collected 90 full palmprints (10 participants × 3 angles × 3 lighting conditions).

After capturing sufficient full palm images, participants were instructed to pick up the pen and begin the collection process for the palmprint patches. As participants moved the pen, its position was automatically tracked by the optical tracker. Participants were asked to keep the pen perpendicular to the palm's surface and to glide it in an S-shaped trajectory across the palm. Throughout the process, participants monitored green area indicators in the collection software to ensure comprehensive coverage of the palm. The pen tip was to lightly touch the palm's surface without applying pressure to avoid skin deformation. After excluding images where the pen was not in contact with the palm, a total of 27,317 palmprint patches were collected from the 10 participants.

4.4 Pre-processing

4.4.1 Coordinate Alignment. To accurately and automatically determine the relative positions of the palmprint patches within the full palmprints, we aligned the spatial coordinates with the image coordinates of the reflective markers. First, the phone camera's internal parameters were determined using Zhang's chessboard calibration method [54]. The positions of the reflective markers in the reference images were then manually annotated. The camera's external parameters were obtained using a Perspective-n-Point (PnP) algorithm, which utilized the spatial and image coordinates of the markers. Finally, the position of each palmprint patch was calculated by projecting the spatial coordinates of the pen tip onto the image plane.

4.4.2 Image Processing for Palmprint Patches. In practice, the pen tip camera often exhibited radial distortion due to the shape of the camera lens. This distortion was minimal for light rays passing near the lens center but increased with distance from the center, resulting in a fisheye-like effect in the localized palmprint images. This effect deviated significantly from the true modality of the palmprint in the global image. To correct this, we calibrated the camera and removed the distortion using Zhang's chessboard calibration method [54]. To reduce computational cost, the resolution of the palmprint patches was halved to 200 × 200 pixels (363 points per inch, PPI). Since there were unavoidable rotations of the pen during the data collection process, each image had a rotation angle. We corrected this by rotating each image back to 0 degrees based on the rotation angle recorded by the optical tracker. Finally, a circular mask was applied to remove invalid areas from the images.

4.4.3 Image Processing for Full Palmprints. Since the full palmprints were captured from different angles, a perspective transformation was applied to each image. The rectangular area of the calibration board was transformed and cropped using the four corner points as references. The resolution of the cropped full palmprints was adjusted to match the PPI of the palmprint patches, and the corresponding positions of the palmprint patches were updated accordingly.

5 PalmPen Algorithm

The goal of the PalmPen algorithm is to achieve absolute positioning and continuous tracking on the palm while offering additional input parameters for richer interactions. The algorithm consists of the following processes: (1) Absolute positioning on the palm, (2) Continuous movement tracking, (3) Touch state detection, and (4) Relative pen pose estimation. The overall workflow of the PalmPen is illustrated in Figure 4.

5.1 Overview

To enhance the portability and usability of the PalmPen, we employ an algorithm that relies exclusively on palm images. During registration, a user captures a full palm image and selects a rectangular region as the template. This template is then processed by a feature extraction module to generate a global deep feature. To optimize efficiency, the registration phase is performed offline. During online interaction, the PalmPen captures a real-time image sequence. Users can interact with the palm by performing various actions—such as sliding, rotating, pressing, or tilting the pen. Touch detection is achieved by analyzing grayscale variations, which also identify the first post-touch image frame. This frame, along with the precomputed global deep feature, is input into an absolute positioning network to estimate the touched location. Continuous movement tracking and relative pen pose estimation are then performed by processing adjacent frames.

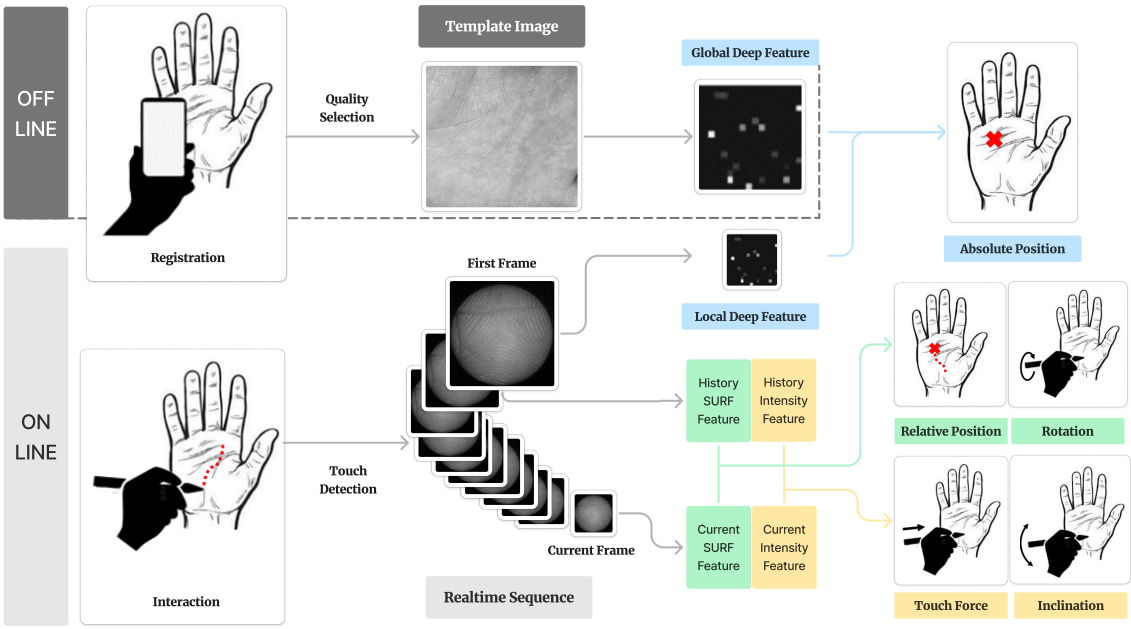

Fig. 4. The overall workflow of PalmPen.

5.2 Absolute Positioning on the Palm

We developed a deep learning network with a two-stage training strategy to estimate the position of a palmprint patch within a full palmprint. In the following sections, we first provide a comprehensive explanation of the data preprocessing steps applied to the input images. We then describe the structure of the deep learning network used in this study. Finally, we detail the two-stage training strategy adopted for this framework.

5.2.1 Data Preprocessing. Contrast Limited Adaptive Histogram Equalization (CLAHE) is an image processing technique that improves image contrast while minimizing noise amplification [43]. Amrouni et al. [2] demonstrated that applying CLAHE enhances the performance and robustness of contactless palmprint recognition algorithms. Therefore, we applied CLAHE to both the full palmprints and the palmprint patches. During the training phase, we augmented the images with random intensity variations and added random Gaussian noise. Random zooming and rotation were applied exclusively to the palmprint patches.

5.2.2 Model Design. Most palmprint recognition algorithms rely on minutiae matching; however, due to the size and resolution of the captured images, minutiae are not a reliable feature for positioning. To address this, we designed a texture-based deep learning network for the absolute palmprint positioning task. The proposed network consists of three modules: a feature extraction module, a feature fusion module, and a position estimation module (see Figure 5).

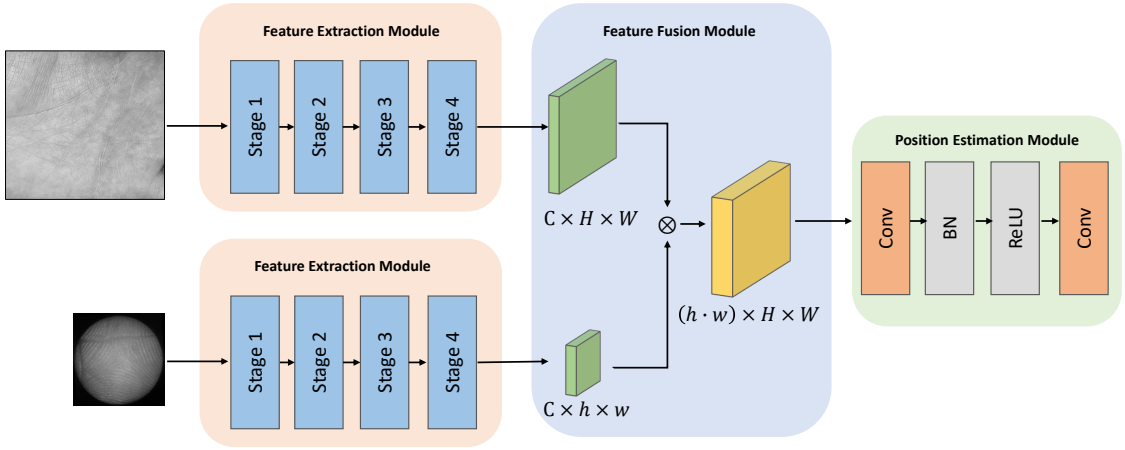

Fig. 5. The proposed network for absolute palmprint positioning.

Two modified High-Resolution Networks (HRNet) [49] were employed to extract features separately from the full palmprint and the palmprint patches. HRNet is a deep learning architecture specifically designed for tasks requiring high-resolution representations, such as human pose estimation, semantic segmentation, and object detection. Unlike traditional networks that downsample input images to a lower resolution and then upsample to generate predictions, HRNet maintains high-resolution feature maps throughout the network. This design preserves fine-grained spatial information, which is critical for the palm positioning task where precise localization is essential. HRNet also learns multi-scale representations by maintaining multiple branches with different resolutions that continuously exchange information. This enables the network to capture both fine-grained details, such as ridge lines in the palmprint, and broader contextual information. To reduce computational costs, we introduced an additional downsampling layer at the beginning of the original HRNet W32, resulting in extracted features with a height and width that are $\frac{1}{8}$ of the input image size.

The extracted features have C channels, set to 32 in this study. The features are fused by calculating a coherent feature as follows:

$$\mathbf{F}_{\text{coh}} = \mathbf{F}_{\text{patch}}^{T} \mathbf{F}_{\text{full}},$$

where $\mathbf{F}_{\text{coh}} \in \mathbb{R}^{(h\times w)\times(H\times W)}$ is the coherent feature, $\mathbf{F}_{\text{full}} \in \mathbb{R}^{C\times(H\times W)}$ is the feature of the full palmprint, and $\mathbf{F}_{\text{patch}} \in \mathbb{R}^{C\times(h\times w)}$ is the feature of the palmprint patch. The coherent feature is then reshaped to $(h \cdot w) \times H \times W$.

This coherent feature is fed into the position estimation module, where the first convolutional layer has 8 output channels and the final convolutional layer has 1 output channel. The output of the position estimation module is a 2-dimensional heatmap, which is $\frac{1}{8}$ the size of the input full palmprint.

The absolute positioning network has a model size of 517 MB and requires 2.18×10^{11} floating point operations (FLOPs). However, since the palmprint feature extraction is performed offline, the operational computational demand is reduced to 9.94×10^{9} FLOPs.

5.2.3 Model Training. We employed heatmaps as the regression target rather than using actual coordinates. Each pixel in the heatmap represents the probability or confidence that a palmprint patch is present at that location. Consequently, heatmaps not only capture local features but also incorporate information from the entire image, enhancing overall robustness. Our pilot tests examined non-zero heatmap region radii ranging from 100 to 300 pixels, revealing that a 200-pixel radius provided the optimal balance of accuracy and robustness. We used the mean squared error (MSE) loss function to supervise the estimation results. Figure 6 shows the predicted heatmap on top of the palmprint. The region in the black circle indicates the actual position of the palmprint patch.

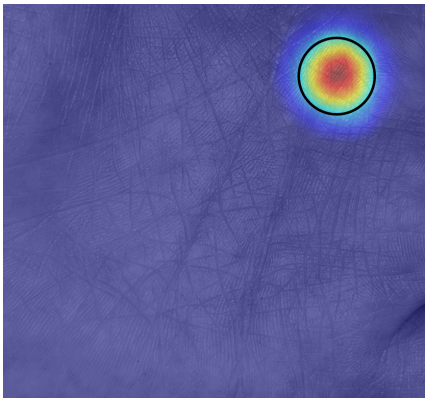

Fig. 6. The predicted heatmap on top of the palmprint. The region in the black circle indicates the actual position of the palmprint patch.

Despite instructing participants to collect image patches that cover as much of the palm as possible, coverage remains limited, particularly near the edges. To address this, we implemented a two-stage training strategy. In the first stage, we extracted palm patches from full palmprints using a sliding window with a 50-pixel step, substituting these patches for images captured by the pen tip camera. The second stage involves domain transfer, where the actual palmprint patches are used for training. During this phase, we froze the weights of the feature extraction module for full palmprints and reduced the learning rate to one-tenth of its original value. We employed a ten-fold leave-one-out cross-validation approach for training and validating the network models.

5.3 Continuous Movement Tracking

During the data collection process, we observed that applying excessive pressure with the pen on the palm could cause skin distortion, significantly affecting the accuracy of position estimation. Additionally, deep learning networks are both power-intensive and computationally costly. To address these issues, we propose a continuous movement tracking approach based on a non-learning relative positioning method. We employed the Speeded-Up Robust Features (SURF) algorithm to extract feature points from palmprint patches. Given that the changes between adjacent frames in a palmprint sequence are minimal, SURF feature points enable rapid and accurate movement tracking. After extracting feature points, we used the Fast Library for Approximate Nearest Neighbors (FLANN) to identify matching points, and applied Lowe's ratio test to filter out unreliable matches. Finally, the Random Sample Consensus (RANSAC) algorithm was used to determine the optimal transformation matrix and eliminate outliers. Once the feature point pairs were established, the mean displacement of all pairs was calculated to represent the relative movement of the pen. Figure 7 shows the matched key points in adjacent frames of palmprint patches.

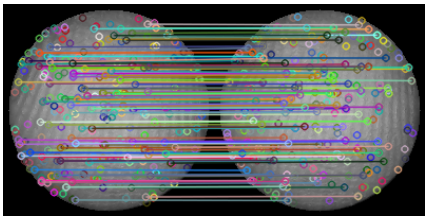

Fig. 7. Matched key points in adjacent frames of palmprint patches.

5.4 Touch State Detection

Given that the focal length of the pen tip camera is 35 mm, the mean intensity of the image is extremely low when no object is positioned near the pen tip opening. As the pen tip camera is housed within a black, non-transparent sleeve, external light is effectively excluded when the pen is pressed against the palm, and the illumination of the palm skin relies entirely on the integrated LED lights. We observed that the mean intensity in the image's central region remains stable when the pen is in contact with the palm. Thus, the touch state can be reliably detected using a threshold value, which in this study is set at 130.

During the development of PalmPen, we also noted that the palm skin inside the pen tip opening tends to bulge when the pen is pressed against the palm. The mean intensity in the central region of the image increases as the skin gets closer to the LED lights, indicating that image intensity is proportional to the applied pressure. Consequently, we use the difference between the mean intensity and the touch detection threshold as a rough estimate of the touch force. Leveraging this relationship, clicking can be performed by slightly increasing the touch force. Figure 8 illustrates the intensity changes of palmprint patches in different touch states, with the black rectangle marking the central area used for calculating the mean intensity.

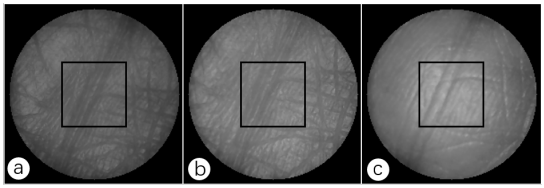

Fig. 8. Intensity changes of the palmprint patches in different touch states: (a) hover; (b) touch (b) press with force. The black rectangle marks the central area used for calculating the mean intensity.

5.5 Relative Pen Pose Estimation

In addition to position, the pose of the pen can also be utilized as an input parameter. Since the transformation matrix between adjacent frames has already been computed during the continuous movement tracking step, the relative rotation can be derived from this matrix.

We propose a straightforward algorithm for estimating the pen's inclination angle. In this study, the upright position is defined as the zero inclination angle. When the pen is tilted, one side of the opening's edge is elevated, causing the mean intensity on the lifted side to be lower than on the opposite side. To estimate the inclination angle, we use the following equation:

$$\text{Inclination} = a \times \left((\text{Intensity}_{\text{lower}} - \text{Intensity}_{\text{upper}}), (\text{Intensity}_{\text{right}} - \text{Intensity}_{\text{left}}) \right),$$

where a is a scaling factor, set to 0.6 in this study. Inclination represents the angle relative to the upright position in the longitudinal and lateral directions. Here, $\text{Intensity}_{\text{lower}}$, $\text{Intensity}_{\text{upper}}$, $\text{Intensity}_{\text{right}}$, and $\text{Intensity}_{\text{left}}$ denote the mean intensities in the lower, upper, right, and left halves of the image, respectively. Figure 9 shows the intensity changes of the palmprint patches at different tilt directions.

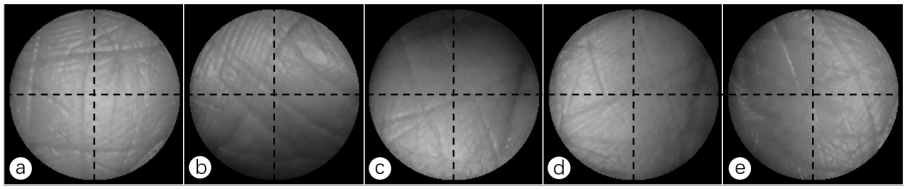

Fig. 9. intensity changes of the palmprint patches at different tilt directions: (a) upwright; (b) tilt forward; (c) tilt backword; (d) tilt left; (e) tilt right.

6 Experiments

In this section, we evaluate the absolute positioning performance of PalmPen. We use the SIFT-based matching algorithm employed by Skinteract [41] as a baseline for comparison. Additionally, we assess the impact of palmprint patch size, the effectiveness of the two-stage training strategy, and the influence of training with both hands through ablation studies. We address following research questions:

RQ1: How does the PalmPen's absolute positioning accuracy compare to conventional SIFT-based matching algorithms?

RQ2: Do the two-stage training and palmprint flip strategies improve the absolute positioning accuracy?

RQ3: What level of accuracy can be achieved by different palmprint patch sizes?

6.1 Data-splitting

As previously mentioned, we utilized a ten-fold leave-one-out cross-validation approach to evaluate both PalmPen and the baseline method. In each fold, data from nine individuals were used for training, while data from the remaining individual were used for testing. This evaluation protocol ensures that every individual is treated as a new user in at least one fold of the validation process. Results are reported as the average across all folds.

6.2 Evaluation Metrics

We use the mean absolute error (MAE) and standard deviation (SD) as evaluation metrics. The distance between the estimated position and the ground truth position is first calculated in pixels. This result is then converted to millimeters by multiplying by the pixel size.

6.3 Results

The experimental results for both the PalmPen and the baseline method are presented in Table 1. For RQ1, the PalmPen demonstrated significantly better performance, with a mean absolute error (MAE) of 2.74 mm and a standard deviation (SD) of 1.83 mm. The images captured in this study have lower contrast compared to the contact-based palmprint patches used in Skinteract [41]. The SIFT algorithm, which relies on detecting keypoints in areas of high contrast and detail, generally performs better with sharper images, as they provide more reliable keypoints. In contrast, PalmPen's absolute positioning network learns the texture patterns of the entire palmprint, allowing for accurate estimations even when parts of the image are blurred.

Table 1. Quantitative results of absolute palm positioning: PalmPen and SIFT-based method. Errors are reported in mm.

	MAE	SD
PalmPen	**2.74**	**1.83**
SIFT	37.25	16.66

The results of the ablation studies are presented in Table 2. For RQ2, the PalmPen model without the two-stage training strategy achieved a mean absolute error (MAE) of 4.9 mm, which is 78.83% higher than the model with the two-stage training strategy. This improvement is attributed to the denser sampling of image patches, particularly in covering a larger area of the palm, including regions near the edges.

Table 2. Ablation studies results for PalmPen: impact of palmprint patch size, the effectiveness of the two-stage training strategy, and the influence of training with both hands. Errors are reported in mm.

	Patch Size	MAE	SD
PalmPen	200	**2.74**	**1.83**
PalmPen w/o two-stage training	200	4.9	3.63
PalmPen w/o palmprint flip	200	7.82	5.11
PalmPen	144	7.43	4.56
PalmPen	104	14.858	7.55

The PalmPen system allows users to register either their left or right palm as a template. Given the mirror-symmetrical geometry of human hands, our default implementation flips right-hand palmprint data during processing and employs a model trained exclusively on left-hand data. To evaluate the efficacy of this approach, we conducted a comparative study by training an additional model using both left and right-hand data. For RQ2, the results revealed that the dual-hand model achieved a mean absolute error (MAE) of 7.82 mm, representing a 185.4% increase in error compared to the left-hand-only model (2.74 mm MAE). We attribute this performance degradation to the model's attempt to learn location-pattern relationships from mirrored data. When trained on both hands, each physical location corresponds to two distinct palmprint patterns (original and mirrored), resulting in less concentrated heatmap distributions and consequently reduced positional accuracy. For practical implementation with right-hand users, our solution maintains computational efficiency by: (1) horizontally flipping right-hand input images, (2) processing them through the left-hand model, and (3) mirroring the output heatmap coordinates to restore proper right-hand spatial references.

The design of the pen tip opening significantly impacts the pen's usability. In this ablation study, we explored a balance between accuracy and comfort by cropping the input image to $\frac{3}{4}$ and $\frac{1}{2}$ of its original size. For RQ3, the results indicate a substantial decrease in accuracy as the input image size is reduced. This decrease is due to the loss of unique pattern information in the image patches, with patches from different regions potentially exhibiting similar patterns.

7 User Study

We conducted a user study to assess the device's performance in both controlled laboratory conditions and real-world environments. In the lab setting, we measured the device's absolute positioning accuracy through a target-tapping task and its input efficiency using a target-tracking task. For real-world evaluation, we tested the

PalmPen's performance in applications involving large screen displays and virtual reality (VR). We addressed the following research questions:

RQ4: How does PalmPen's input accuracy and efficiency compare to baseline methods under controlled laboratory conditions?

RQ5: How does PalmPen's input accuracy and efficiency compare to baseline methods in real-world environments, including VR and large screen display?

RQ6: Does PalmPen demonstrate superior user preference ratings and lower perceived workload compared to existing baseline methods?

7.1 Participants

We recruited 12 participants (8 males, 4 females) aged between 20 and 28 years (M = 22.3, SD = 2.4) from the campus. All participants were right-handed and had prior experience using a mouse and touchpad. However, no participant had prior experience using a pen-shaped mouse. Informed consent was obtained from each participant, and their personal information was handled confidentially. None of the participants had any medical conditions or upper limb injuries that could affect their ability to complete the data collection tasks. All participants were given unrestricted time to practice with different techniques until they were satisfied with their ability to use each technique.

7.2 Input Methods

Unlike the setup in the data collection phase, the user study did not utilize the optical tracking system, and the reflective markers were removed from the PalmPen prototype (see Figure 10a). Consequently, coordinate alignment was not performed on the palmprints. Each participant was required to capture a full palm image as a registered template, which was then cropped according to the effective area they selected. To optimize efficiency, the global deep feature of each template was extracted offline prior to the absolute positioning algorithm. Apart from the PalmPen, we used two additional input devices as baseline methods. A pen-shaped Puluomi wireless mouse (see Figure 10b) was used to evaluate the performance of relative input; participants operated the mouse on the palm of their non-dominant hand. A Tecno Spark Pro 20+ smartphone[8], measuring 164.65 mm in height and 75.04 mm in width, was used as a touchpad (see Figure 10c). Participants held the phone with their non-dominant hand and operated it with the index finger of their dominant hand. Both the PalmPen and the phone-based touchpad adopted the interaction method proposed in ARC-Pad [35], where the first touch initiates absolute positioning, causing the cursor to jump to the touched location, and subsequent movements of the finger and PalmPen use relative positioning.

Fig. 10. The input devices used in the user study: (a) PalmPen; (b) a pen-shaped mouse; (c) a phone-based touchpad.

[8]https://www.tecno-mobile.com/phones/tech-specs/tecspecs/spark-20-pro-+/

7.3 Apparatus

The first two tasks in the laboratory setting were performed on a 15.6-inch Dell laptop with a screen resolution of 1920 × 1080. The large-screen interaction task was conducted on a 65-inch Maxhub EC65CAC display, with the program running on the laptop and projected onto the display via an HDMI cable. The VR interaction task was performed using an HTC Vive headset.

7.4 Tasks

7.4.1 Task 1: Target Tapping. This task aims to evaluate the absolute positioning performance of the PalmPen and the touchpad in an eyes-free setting (see Figure 11a). Participants were seated in front of a laptop, where a red circle was displayed on the screen. They were instructed to tap on their palm using the PalmPen or on the touchpad using their finger. The tapped position was mapped to the laptop screen, and the distance between the tapped location and the target location was recorded. During this task, participants were required to focus on the screen without looking at the input devices to simulate a scenario with minimal visual attention. A total of 480 trials were conducted: 12 participants × 20 repetitions × 2 input methods.

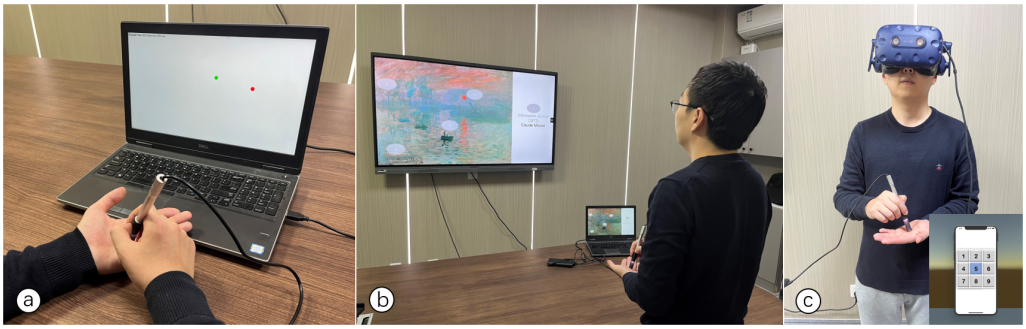

Fig. 11. The task setups for the user study: (a) the target tapping and target tracking tasks; (b) the large display interaction task; (c) the button click task in VR.

7.4.2 Task 2: Target Tracking. This task evaluates the overall input efficiency of the three input devices (see Figure 11a). During the task, a red circle appeared on the screen. Once participants began the task, a timer was initiated, and a green circle appeared, indicating the cursor's current position. Participants were required to move the green circle to the red circle's location as quickly as possible. The time taken to reach the red circle was recorded. A total of 720 trials were completed: 12 participants × 20 repetitions × 3 input methods.

7.4.3 Task 3: Interaction on a Large Display. This task assesses the performance of the three input devices in a real-world large screen interaction scenario (see Figure 11b). Participants stood 3 meters away from a large display, simulating a presentation setting, similar to the scenario described in [47]. They were instructed to select 5 targets on each page of the display. Upon selecting a target, a short audio cue was played. The time taken to select each target was recorded. In total, 720 trials were performed: 12 participants × 4 pages × 5 targets × 3 input methods.

7.4.4 Task 4: Button Click in VR. This task evaluates button-clicking accuracy in a virtual reality (VR) environment (see Figure 11c). Participants wore a VR headset and stood in the center of the room, where a keypad with nine buttons was displayed. They were instructed to click each button following specific instructions. The participant's palm and the touchpad were divided into nine regions, each corresponding to a button on the keypad. To aid in

memorization, the numbers 1 to 9 were displayed on the buttons. Each time a participant clicked a button, the target button and the actual clicked button were recorded. A total of 432 trials were conducted: 12 participants × 18 clicks × 2 input methods.

7.5 Procedure

Upon arrival, participants were introduced to the study's objectives and given a demonstration of the apparatus and tasks. They completed a consent form and provided basic demographic information, including age, gender, dominant hand, and prior experience with different input devices. A reference palm image was captured by the phone camera for the PalmPen for each participant. They were allowed unlimited time to practice with each input device until they were confident in their ability to use them effectively. A pool of target positions for each task was generated in advance to ensure consistency across all input devices. The order of input methods for each task was counterbalanced across participants using a Latin square design to mitigate any bias from learning effects. Following task completion, participants completed post-study questionnaires, including a NASA-TLX (raw TLX) [17] questionnaire for workload assessment, and a custom questionnaire measuring fatigue levels and subjective preferences across the evaluated interaction techniques.

7.6 Performance Analysis

To assess the normality of the data in all tasks and subjective ratings, we employed the Shapiro-Wilk test [46], which indicated that all time-related tasks (Tasks 2-3) violated the assumption of normality. Consequently, we used violin plots to visualize the actual distribution of the original data and applied non-parametric Wilcoxon signed-rank tests [53] with Benjamini-Hochberg corrections [3] to evaluate the significance of completion time differences between methods. For subjective ratings, we plotted the means and 95% confidence intervals (CIs) and performed paired t-tests with Benjamini-Hochberg corrections to determine statistical significance.

7.6.1 Task 1: Target Tapping. Table 3 illustrates the performance results for the target tapping task. The mean absolute error (MAE) for PalmPen and touchpad were 9.87 mm and 13.71 mm, respectively. For RQ4, PalmPen demonstrated a significantly lower error compared to the touchpad ($p < 0.005$), achieving a 28.02% reduction in MAE. It was observed that the error increased as the target was positioned further from the center of the palm. This phenomenon is consistent with findings reported in PalmRC [10], likely due to a decrease in proprioceptive feedback as the target moves farther from the body.

Table 3. Quantitative results of target tapping task: PalmPen and phone-based touchpad. Errors are reported in mm.

	MAE	SD
PalmPen	**9.87**	**4.59**
Phone-based touchpad	13.71	8.76

7.6.2 Task 2: Target Tracking. Figure 12 presents the task completion times for the target tracking tasks. The average completion times for PalmPen, touchpad, and mouse were 2.18 seconds, 3.02 seconds, and 3.23 seconds, respectively. For RQ4, PalmPen's completion time was significantly shorter than that of both the touchpad ($p < 0.001$) and the mouse ($p < 0.001$), with reductions of 27.81% and 32.51% compared to the touchpad and mouse, respectively. Both PalmPen and the touchpad were faster than the mouse, likely due to the advantage of absolute positioning. According to Fitts' law [32], the time required to complete a pointing task is proportional to the distance between the pointer's initial position and the target. Consequently, relative positioning can reduce interaction efficiency when the target is far from the cursor. Additionally, because of the physical limitations of

the interaction space, clutching is necessary to move the cursor, which can degrade performance [7], especially on large displays. In contrast, absolute positioning allows for quick jumps to locations near the target, significantly reducing target selection time. On the other hand, the absolute positioning approach could have negative impact on the input efficiency when the target is close to the cursor.

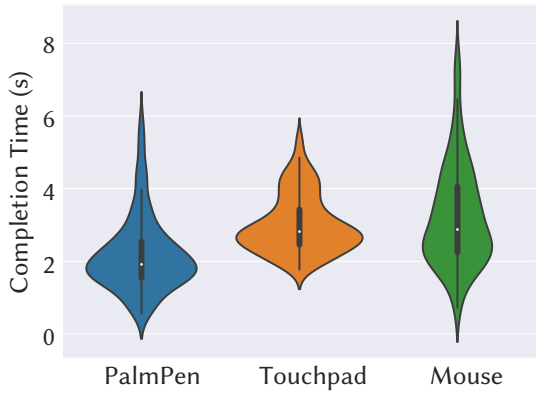

Fig. 12. Completion times (in seconds) for the target tracking task using different input devices.

7.6.3 Task 3: Interaction on a Large Display. Figure 13 displays the mean time for target selection on a large display using the three input devices. The average times required for target selection were 1.73 seconds, 2.80 seconds, and 2.86 seconds for the PalmPen, touchpad, and mouse, respectively. For RQ5, PalmPen required significantly less time than both the touchpad ($p < 0.001$) and the mouse ($p < 0.001$), with reductions of 38.21% and 39.51% compared to the touchpad and mouse, respectively. Similar to task 2, the absolute positioning approach adopted by PalmPen and the touchpad showed better performance than the relative positioning approach adopted by the mouse.

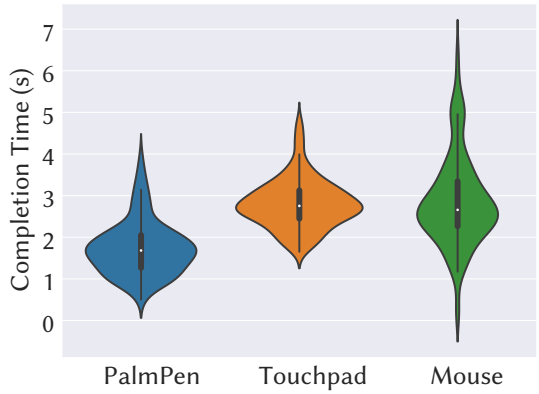

Fig. 13. Completion times (in seconds) for the target selection task on a large display using different input devices.

7.6.4 Task 4: Button Click in VR. Table 4 presents the button-clicking accuracies in virtual reality for both PalmPen and the touchpad. For RQ5, the overall accuracies were 94.10% for PalmPen and 90.28% for the touchpad. Given that there are only nine buttons, each occupies a relatively large area, making the task relatively straightforward for both input methods. Notably, the touchpad demonstrated significantly lower accuracy for buttons located near the fingers, whereas PalmPen exhibited consistent accuracy across all regions. This discrepancy may be attributed to the way participants held the phone; most held the left half of the phone, causing the right half to be positioned farther from their hands and resulting in reduced proprioceptive feedback. The confusion matrices are illustrated in Figure 14.

Table 4. Button clicking accuracy in VR using: (a) PalmPen; (b) touchpad; (c) mouse.

	1	2	3	4	5	6	7	8	9	Overal
PalmPen	87.50%	96.88%	100.00%	87.50%	90.63%	96.88%	100.00%	100.00%	87.50%	94.10%
touchpad	93.75%	96.88%	87.5%	90.63%	84.38%	87.50%	100.00%	87.50%	84.38%	90.28%

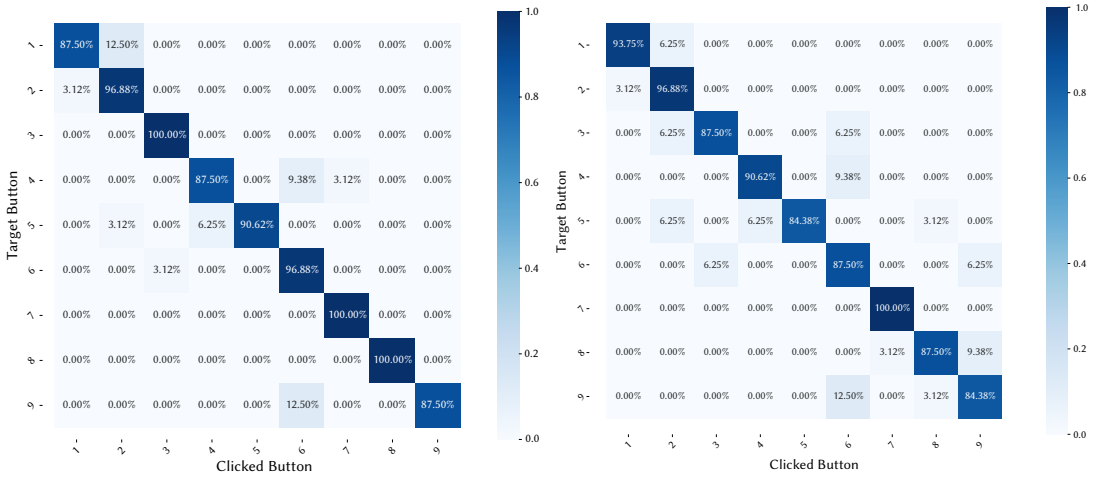

Fig. 14. Confusion matrix for button clicking accuracy in VR using: a) PalmPen; (b) touchpad.

7.6.5 Subjective Ratings. Based on the responses in the questionnaires, we depicted the subjective ratings in Figure 15 and Figure 17.The workload results, as assessed by the NASA-TLX, are shown in Figure 15. For RQ6, PalmPen exhibited a significantly lower total workload than both the touchpad ($p < 0.01$) and the mouse ($p < 0.01$). According to the measurements, the PalmPen resulted in the lowest workload across all evaluation criteria except for mental workload. Fatigue assessment showed PalmPen to be the least fatiguing input method, achieving statistically significant improvement over the mouse ($p < 0.005$) though not against the touchpad ($p > 0.05$). For RQ6, in terms of user preference, PalmPen received significantly higher ratings than both comparison devices (mouse: $p < 0.05$; touchpad: $p < 0.005$). Some participants noted that while the absolute positioning approach used by both the PalmPen and the touchpad is more efficient, it required additional focus to locate the cursor

after repositioning. When the actual cursor position deviated significantly from the perceived touch point due to proprioceptive errors, participants reported a high level of frustration.

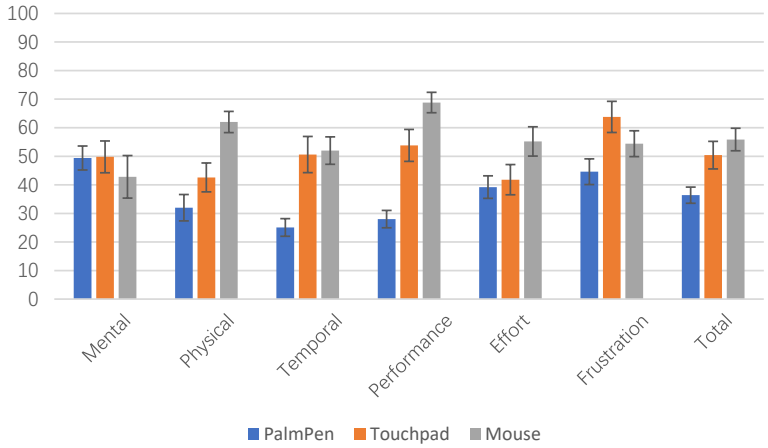

Fig. 15. Workload measurement in NASA-TLX units. Error bars: 95% CI.

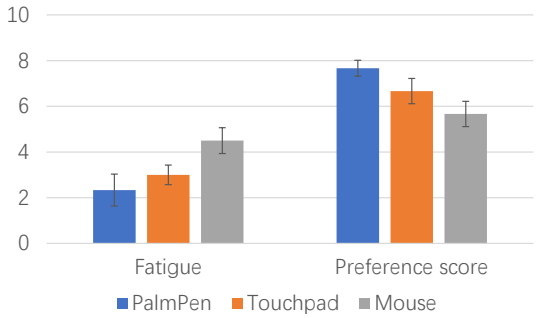

Fig. 16. Fatigue level and user preference score. Error bars: 95% CI.

7.7 Discussion

The absolute positioning algorithm demonstrates high precision with a mean absolute error (MAE) of 2.74 mm. However, the observed MAE for PalmPen in Task 1 was 9.87 mm, suggesting that proprioceptive inaccuracy constitutes the primary source of error. This finding confirms that the algorithm's intrinsic accuracy is sufficient for vision-free interaction scenarios.

In target tapping tasks (Tasks 1 and 4), the touchpad exhibited significantly lower accuracy than PalmPen. We attribute this performance gap to the enhanced proprioceptive feedback of direct palm interaction compared to handheld touchpad operation. For target tracking tasks (Tasks 2 and 3), both devices employed a hybrid approach: initial absolute positioning followed by relative adjustment. The touchpad's inferior tapping accuracy frequently resulted in larger initial cursor displacements. According to Fitts' law, these greater initial offsets

directly increased subsequent correction times, explaining PalmPen's significantly shorter task completion times ($p < 0.01$).

The hybrid absolute-relative positioning method yielded distinct advantages for both PalmPen and touchpad, manifesting in reduced mean completion times and lower variance compared to the mouse. While mouse performance depended entirely on the initial-to-target distance, the hybrid approach only required correction of the residual error after the initial absolute positioning phase. This fundamental difference in interaction mechanics accounts for the observed efficiency improvements.

User assessments further reinforced PalmPen's advantages. The system achieved the lowest overall workload scores across all NASA-TLX dimensions except mental demand. Participants also reported lowest fatigue levels and awarded PalmPen the highest preference ratings.

8 Application

PalmPen is a portable standalone input device that uses the user's palm as the interaction surface. It provides absolute positioning, continuous tracking and additional input parameters, such as touch pressure, rotation angles, and inclination angles. To use the device, users are only required to take a photo of the palm as a registered template without the need for calibration. During input, only a natural grip and minimal hand movements are required, significantly reducing muscle strain and fatigue. We have identified two real-world scenarios for PalmPen and implemented a range of applications highlighting its usability and efficiency.

8.1 Large Screen Displays

The PalmPen's absolute positioning capability renders it an excellent candidate for use as a large screen display controller (see Figure 17a). This approach enables more natural and efficient pointing and target selection compared to conventional methods, as users can directly interact with their palms as the control surface. The system combines absolute positioning with continuous tracking to significantly enhance input efficiency, while its portable design allows for unconstrained movement during operation. Additionally, PalmPen is well-suited for complex tasks, such as sketching and 3D object manipulation, on large screens (see Figure 17b). The PalmPen's pressure-sensitive input enables precise control over stroke thickness during sketching tasks, while its pen pose tracking capability provides comprehensive control for manipulating 3D objects with multiple degrees of freedom. These combined features facilitate sophisticated interaction techniques that surpass conventional input methods for complex workflows on large displays.

8.2 VR/AR Systems

The PalmPen's proprioceptive tactile perception offers substantial potential for applications in low-vision or vision-free scenarios, such as virtual reality (VR) and augmented reality (AR) systems. Unlike conventional VR controllers or vision-based AR hand tracking systems that require extensive arm movements, the PalmPen's compact operation minimizes hand displacement, thereby substantially reducing musculoskeletal strain and fatigue during prolonged use. We have implemented a "Whack-a-Mole" game in VR (see Figure 17c), where users can strike the virtual moles by touching the palm with the PalmPen. Additionally, the PalmPen's extra pose properties allow it to function as a joystick; the device's tilt angles can be used to control the movement of an object within the VR game (see Figure 17d).

9 Limititions

While the feasibility and performance of the PalmPen were demonstrated through experiments and a user study, certain limitations in this study should be acknowledged.

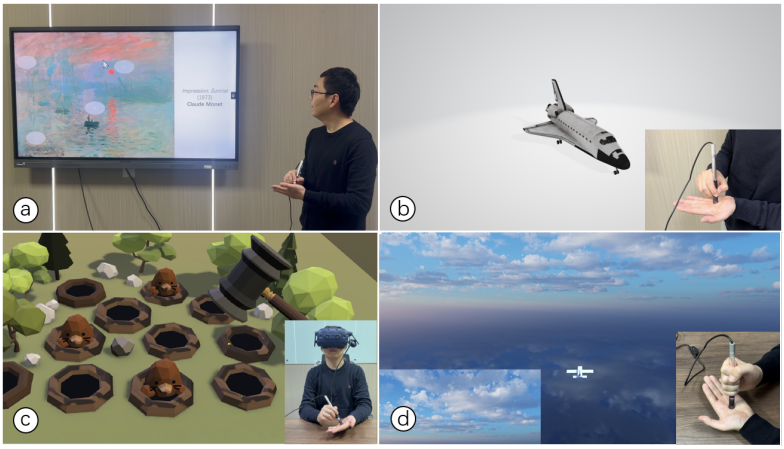

Fig. 17. Applications of PalmPen: (a) large display controller; (b) 3D object manipulation; (c) VR game controller using absolute positioning; (d) VR game joystick using inclination angle.

Visual focus. The absolute positioning approach used by the PalmPen required additional visual focus to locate the cursor after repositioning, particularly when the actual cursor position was distant from the perceived touch point. Enhancing positioning accuracy and incorporating visual aids, such as adjusting cursor size, could improve ease of cursor location for users.

Power consumption. The PalmPen relies on a camera and LED lights for operation, both of which contribute to significant power consumption and consequently limit the device's operational duration. The prototype consumes 0.45 W during operation—approximately three times the power consumption of a standard mouse. The current implementation utilizes an off-the-shelf pen-shaped camera with a 640×480 resolution, though only the central 400×400 pixel region is processed and subsequently downsampled to 200×200 pixels. Adopting a dedicated low-power image sensor with reduced resolution could substantially decrease power usage. Furthermore, integrating supplementary sensors—such as an inertial measurement unit (IMU) [47] or optical mouse sensors [44]—could further optimize efficiency. These sensors could manage touch detection and continuous tracking, enabling the camera to operate solely during initial touch events for absolute positioning.

Ergonomics. The PalmPen also faces several ergonomic challenges. For instance, the larger-than-normal pen tip opening impairs proprioceptive accuracy, and the edge of the pen affects the smoothness of sliding on the palm. One potential solution is to reposition the camera, such as by placing it on the top or side of the pen. Future iterations should explore various hardware designs to address these ergonomic concerns.

Long-term usability. One limitation of this study is the absence of a long-term usability evaluation. While the current findings provide insights into short-term performance and user experience, factors such as user adaptability, sustained efficiency, and fatigue over extended use were not assessed. These aspects are critical for determining the practical viability of the PalmPen in real-world, prolonged usage scenarios. Future work should include longitudinal studies to better understand how users adapt to the PalmPen over time and how long-term use affects physical and cognitive workload.

10 Conclusion

This paper introduced PalmPen, a novel pen-shaped camera device that enables absolute positioning and continuous tracking on the palm, offering a unique and intuitive input method for various applications. The device

utilizes a deep learning network to accurately predict the location of partial palmprints captured by the pen tip camera, achieving a mean positioning error of 2.74 mm. User studies demonstrated the superiority of PalmPen over pen-shaped mice and phone-based touchpads in terms of both accuracy and efficiency for various tasks, including target tapping, tracking, large-screen interaction, and button clicking in VR. Furthermore, PalmPen's ability to infer additional input parameters, such as touch pressure, rotation angles, and inclination angles, extends its application range and opens possibilities for more nuanced and complex interactions. PalmPen, as a versatile and portable input device for mobile environments, offers a user-friendly and efficient alternative to traditional input methods.

11 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62376132 and 62321005.

References

[1] Dele W. S. Alausa, Emmanuel Adetiba, Joke A. Badejo, Innocent Ewean Davidson, Obiseye Obiyemi, Elutunji Buraimoh, Abdultaofeek Abayomi, and Oluwadamilola Oshin. 2022. Contactless Palmprint Recognition System: A Survey. *IEEE Access* 10 (2022), 132483–132505. https://doi.org/10.1109/ACCESS.2022.3193382

[2] Nadia Amrouni, Amir Benzaoui, Rafik Bouaouina, Yacine Khaldi, Insaf Adjabi, and Ouahiba Bouglimina. 2022. Contactless Palmprint Recognition Using Binarized Statistical Image Features-Based Multiresolution Analysis. *Sensors* 22, 24 (2022). https://doi.org/10.3390/s22249814

[3] Yoav Benjamini and Yosef Hochberg. 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal statistical society: series B (Methodological)* 57, 1 (1995), 289–300.

[4] Joanna Bergström and Kasper Hornbæk. 2019. Human–Computer Interaction on the Skin. *ACM Comput. Surv.* 52, 4, Article 77 (Aug. 2019), 14 pages. https://doi.org/10.1145/3332166

[5] Joanna Bergstrom-Lehtovirta, Kasper Hornbæk, and Sebastian Boring. 2018. It's a Wrap: Mapping On-Skin Input to Off-Skin Displays. 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, 1–11. https://doi.org/10.1145/3173574.3174138

[6] Raffaele Cappelli, Matteo Ferrara, and Dario Maio. 2012. A Fast and Accurate Palmprint Recognition System Based on Minutiae. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 42, 3 (2012), 956–962. https://doi.org/10.1109/TSMCB.2012.2183635

[7] Géry Casiez, Daniel Vogel, Qing Pan, and Christophe Chaillou. 2007. RubberEdge: Reducing Clutching by Combining Position and Rate Control with Elastic Feedback. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology* (Newport, Rhode Island, USA) *(UIST '07)*. Association for Computing Machinery, New York, NY, USA, 129–138. https://doi.org/10.1145/1294211.1294234

[8] Liwei Chan, Rong-Hao Liang, Ming-Chang Tsai, Kai-Yin Cheng, Chao-Huai Su, Mike Y. Chen, Wen-Huang Cheng, and Bing-Yu Chen. 2013. FingerPad: Private and Subtle Interaction Using Fingertips. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 255–260. https://doi.org/10.1145/2501988.2502016

[9] Jifeng Dai, Jianjiang Feng, and Jie Zhou. 2012. Robust and Efficient Ridge-Based Palmprint Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 8 (2012), 1618–1632. https://doi.org/10.1109/TPAMI.2011.237

[10] Niloofar Dezfuli, Mohammadreza Khalilbeigi, Jochen Huber, Florian Müller, and Max Mühlhäuser. 2012. PalmRC: Imaginary Palm-Based Remote Control for Eyes-Free Television Interaction. In *Proceedings of the 10th European Conference on Interactive TV and Video* (Berlin, Germany) *(EuroITV '12)*. Association for Computing Machinery, New York, NY, USA, 27–34. https://doi.org/10.1145/2325616.2325623

[11] Xinrui Fang, Chengshuo Xia, and Yuta Sugiura. 2021. FacialPen: Using Facial Detection to Augment Pen-Based Interaction. In *Proceedings of the Asian CHI Symposium 2021* (Yokohama, Japan) *(Asian CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3429360.3467672

[12] Sean Gustafson, Christian Holz, and Patrick Baudisch. 2011. Imaginary Phone: Learning Imaginary Interfaces by Transferring Spatial Memory from a Familiar Device. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 283–292. https://doi.org/10.1145/2047196.2047233

[13] Sean G. Gustafson, Bernhard Rabe, and Patrick M. Baudisch. 2013. Understanding Palm-Based Imaginary Interfaces: The Role of Visual and Tactile Cues when Browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 889–898. https://doi.org/10.1145/2470654.2466114

[14] Ryo Hajika, Tamil Selvan Gunasekaran, Chloe Dolma Si Ying Haigh, Yun Suen Pai, Eiji Hayashi, Jaime Lien, Danielle Lottridge, and Mark Billinghurst. 2024. RadarHand: A Wrist-Worn Radar for On-Skin Touch-Based Proprioceptive Gestures. *ACM Trans. Comput.-Hum. Interact.* 31, 2, Article 17 (Jan. 2024), 36 pages. https://doi.org/10.1145/3617365

[15] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 441–450. https://doi.org/10.1145/2047196.2047255

[16] Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: Appropriating the Body as an Input Surface. 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, 453–462. https://doi.org/10.1145/1753326.1753394

[17] Sandra G. Hart. 2006. Nasa-Task Load Index (NASA-TLX); 20 Years Later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 50, 9 (2006), 904–908. https://doi.org/10.1177/154193120605000909

[18] Ramon Linus Hofer and Andreas Kunz. 2010. Digisketch: Taming Anoto Technology on LCDs. In *Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems* (Berlin, Germany) *(EICS '10)*. Association for Computing Machinery, New York, NY, USA, 103–108. https://doi.org/10.1145/1822018.1822034

[19] Sungjae Hwang, Andrea Bianchi, Myungwook Ahn, and Kwangyun Wohn. 2013. MagPen: Magnetically Driven Pen Interactions on and around Conventional Smartphones. In *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Munich, Germany) *(MobileHCI '13)*. Association for Computing Machinery, New York, NY, USA, 412–415. https://doi.org/10.1145/2493190.2493194

[20] Anil K. Jain and Jianjiang Feng. 2009. Latent Palmprint Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 6 (2009), 1032–1047. https://doi.org/10.1109/TPAMI.2008.242

[21] Takashi Kikuchi, Yuta Sugiura, Katsutoshi Masai, Maki Sugimoto, and Bruce H. Thomas. 2017. EarTouch: Turning the Ear into an Input Surface. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Vienna, Austria) *(MobileHCI '17)*. Association for Computing Machinery, New York, NY, USA, Article 27, 6 pages. https://doi.org/10.1145/3098279.3098538

[22] Seokhwan Kim, Shin Takahashi, and Jiro Tanaka. 2010. New Interface Using Palm and Fingertip without Marker for Ubiquitous Environment. In *2010 IEEE/ACIS 9th International Conference on Computer and Information Science*. 819–824. https://doi.org/10.1109/ICIS.2010.110

[23] Jarrod Knibbe, Diego Martinez Plasencia, Christopher Bainbridge, Chee-Kin Chan, Jiawei Wu, Thomas Cable, Hassan Munir, and David Coyle. 2014. Extending Interaction for Smart Watches: Enabling Bimanual around Device Control. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI EA '14)*. Association for Computing Machinery, New York, NY, USA, 1891–1896. https://doi.org/10.1145/2559206.2581315

[24] Luv Kohli and Mary Whitton. 2005. The Haptic Hand: Providing User Interface Feedback with the Non-Dominant Hand in Virtual Environments. In *Proceedings of Graphics Interface 2005* (Victoria, British Columbia) *(GI '05)*. Canadian Human-Computer Communications Society, Waterloo, CAN, 1–8.

[25] Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. 2014. Skin Buttons: Cheap, Small, Low-Powered and Clickable Fixed-Icon Laser Projectors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 389–394. https://doi.org/10.1145/2642918.2647356

[26] Chen Liang, Chi Hsia, Chun Yu, Yukang Yan, Yuntao Wang, and Yuanchun Shi. 2023. DRG-Keyboard: Enabling Subtle Gesture Typing on the Fingertip with Dual IMU Rings. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 170 (Jan. 2023), 30 pages. https://doi.org/10.1145/3569463

[27] Soo-Chul Lim, Jungsoon Shin, Seung-Chan Kim, and Joonah Park. 2015. Expansion of Smartwatch Touch Interface from Touchscreen to Around Device Interface Using Infrared Line Image Sensors. *Sensors* 15, 7 (2015), 16642–16653. https://doi.org/10.3390/s150716642

[28] Eryun Liu, Anil K. Jain, and Jie Tian. 2013. A Coarse to Fine Minutiae-Based Latent Palmprint Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 10 (2013), 2307–2322. https://doi.org/10.1109/TPAMI.2013.39

[29] Yang Liu and Ajay Kumar. 2020. Contactless Palmprint Identification Using Deeply Learned Residual Features. *IEEE Transactions on Biometrics, Behavior, and Identity Science* 2, 2 (2020), 172–181. https://doi.org/10.1109/TBIOM.2020.2967073

[30] Zongjian Liu, Jieling He, Jianjiang Feng, and Jie Zhou. 2023. PrinType: Text Entry via Fingerprint Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 174 (Jan. 2023), 31 pages. https://doi.org/10.1145/3569491

[31] Guy Lüthi, Andreas Rene Fender, and Christian Holz. 2022. DeltaPen: A Device with Integrated High-Precision Translation and Rotation Sensing on Passive Surfaces. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 57, 12 pages. https://doi.org/10.1145/3526113.3545655

[32] I. Scott MacKenzie and William Buxton. 1992. Extending Fitts' Law to Two-Dimensional Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Monterey, California, USA) *(CHI '92)*. Association for Computing Machinery, New York, NY, USA, 219–226. https://doi.org/10.1145/142750.142794

[33] Vitus Maierhöfer, Andreas Schmid, and Raphael Wimmer. 2024. TipTrack: Precise, Low-Latency, Robust Optical Pen Tracking on Arbitrary Surfaces Using an IR-Emitting Pen Tip. In *Proceedings of the Eighteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Cork, Ireland) *(TEI '24)*. Association for Computing Machinery, New York, NY, USA, Article 18, 13 pages. https://doi.org/10.1145/3623509.3633366

[34] Fabrice Matulic, Riku Arakawa, Brian Vogel, and Daniel Vogel. 2020. PenSight: Enhanced Interaction with a Pen-Top Camera. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376147

[35] David C. McCallum and Pourang Irani. 2009. ARC-Pad: Absolute+Relative Cursor Positioning for Large Displays with a Mobile Touchscreen. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology* (Victoria, BC, Canada) *(UIST '09)*. Association for Computing Machinery, New York, NY, USA, 153–156. https://doi.org/10.1145/1622176.1622205

[36] Tzu-Wei Mi, Jia-Jun Wang, and Liwei Chan. 2023. LapTouch: Using the Lap for Seated Touch Interaction with HMDs. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 3, Article 114 (Sept. 2023), 23 pages. https://doi.org/10.1145/3610878

[37] Adiyan Mujibiya, Xiang Cao, Desney S. Tan, Dan Morris, Shwetak N. Patel, and Jun Rekimoto. 2013. The Sound of Touch: On-Body Touch and Gesture Sensing Based on Transdermal Ultrasound Propagation. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces* (St. Andrews, Scotland, United Kingdom) *(ITS '13)*. Association for Computing Machinery, New York, NY, USA, 189–198. https://doi.org/10.1145/2512349.2512821

[38] Aditya Shekhar Nittala, Anusha Withana, Narjes Pourjafarian, and Jürgen Steimle. 2018. Multi-Touch Skin: A Thin and Flexible Multi-Touch Sensor for On-Skin Input. 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/10.1145/3173574.3173607

[39] Uran Oh and Leah Findlater. 2015. A Performance Comparison of On-Hand versus On-Phone Nonvisual Input by Blind and Sighted Users. *ACM Trans. Access. Comput.* 7, 4, Article 14 (Nov. 2015), 20 pages. https://doi.org/10.1145/2820616

[40] Ryosuke Ono, Shunsuke Yoshimoto, and Kosuke Sato. 2013. Palm+Act: Operation by Visually Captured 3D Force on Palm. In *SIGGRAPH Asia 2013 Emerging Technologies* (Hong Kong, Hong Kong) *(SA '13)*. Association for Computing Machinery, New York, NY, USA, Article 14, 3 pages. https://doi.org/10.1145/2542284.2542298

[41] Manuel Prätorius, Aaron Scherzinger, and Klaus Hinrichs. 2015. SkInteract: An On-Body Interaction System Based on Skin-Texture Recognition. In *Human-Computer Interaction – INTERACT 2015*, Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, and Marco Winckler (Eds.). Springer International Publishing, Cham, 425–432.

[42] Manuel Prätorius, Dimitar Valkov, Ulrich Burgbacher, and Klaus Hinrichs. 2014. DigiTap: An Eyes-Free VR/AR Symbolic Input Device. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology* (Edinburgh, Scotland) *(VRST '14)*. Association for Computing Machinery, New York, NY, USA, 9–18. https://doi.org/10.1145/2671015.2671029

[43] Ali M Reza. 2004. Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for Real-Time Image Enhancement. *Journal of VLSI signal processing systems for signal, image and video technology* 38 (2004), 35–44.

[44] Hugo Romat, Andreas Fender, Manuel Meier, and Christian Holz. 2021. Flashpen: A High-Fidelity and High-Precision Multi-Surface Pen for Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 306–315. https://doi.org/10.1109/VR50410.2021.00053

[45] T. Scott Saponas, Desney S. Tan, Dan Morris, Ravin Balakrishnan, Jim Turner, and James A. Landay. 2009. Enabling Always-Available Input with Muscle-Computer Interfaces. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology* (Victoria, BC, Canada) *(UIST '09)*. Association for Computing Machinery, New York, NY, USA, 167–176. https://doi.org/10.1145/1622176.1622208

[46] Samuel Sanford Shapiro and Martin B Wilk. 1965. An Analysis of Variance Test for Normality (Complete Samples). *Biometrika* 52, 3-4 (1965), 591–611.

[47] Xiyuan Shen, Chun Yu, Xutong Wang, Chen Liang, Haozhan Chen, and Yuanchun Shi. 2024. MouseRing: Always-Available Touchpad Interaction with IMU Rings. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 412, 19 pages. https://doi.org/10.1145/3613904.3642225

[48] Charles S Sherrington. 1907. On the Proprio-ceptive System, Especially in Its Reflex Aspect. *Brain* 29, 4 (1907), 467–482.

[49] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[50] 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* (Copenhagen, Denmark) *(MobileHCI '15)*. Association for Computing Machinery, New York, NY, USA, 153–160. https://doi.org/10.1145/2785830.2785886

[51] Cheng-Yao Wang, Min-Chieh Hsiu, Po-Tsung Chiu, Chiao-Hui Chang, Liwei Chan, Bing-Yu Chen, and Mike Y. Chen. 2015. PalmGesture: Using Palms as Gesture Interfaces for Eyes-Free Input. 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, 217–226. https://doi.org/10.1145/2785830.2785885

[52] Martin Weigel, Tong Lu, Gilles Bailly, Antti Oulasvirta, Carmel Majidi, and Jürgen Steimle. 2015. iSkin: Flexible, Stretchable and Visually Customizable On-Body Touch Sensors for Mobile Computing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors*

in Computing Systems (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 2991–3000. https://doi.org/10.1145/2702123.2702391

[53] Frank Wilcoxon. 1992. *Individual Comparisons by Ranking Methods*. Springer New York, New York, NY, 196–202. https://doi.org/10.1007/978-1-4612-4380-9_16

[54] Z. Zhang. 2000. A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 11 (2000), 1330–1334. https://doi.org/10.1109/34.888718

[55] Qian Zhou, George Fitzmaurice, and Fraser Anderson. 2022. In-Depth Mouse: Integrating Desktop Mouse into Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 354, 17 pages. https://doi.org/10.1145/3491102.3501884