Sensors and Materials, Vol. 38, No. 3 (2026) 1675–1693
MYU Tokyo

S & M 4400

Handwriting Optimization System Using Pressure Sensor Array for Touch-type Digital Painting

Yu Zhu[1*] and Jiabin Hu[2]

[1]Wuhan Textile University, Wuhan 430077, China
[2]School of Computer Science and Technology, Hubei Business College, Wuhan 430079, China

(Received January 29, 2026; accepted March 19, 2026)

Keywords: pressure sensor array, touch type, digital painting, handwriting optimization

Conventional handwriting recognition systems often fail to capture complex hand-device interactions, such as grip force distribution and subtle posture changes, but remain susceptible to hand-tremor noise. To overcome such inherent limitations of traditional single-point pressure-sensing systems, we developed a pressure-array-based manuscript optimization (PAMO) system with an intelligent stylus integrated with a 128-unit flexible piezoresistive sensor array (16 × 8 units) that captures a high-dimensional manifold of grip force data at a 120 Hz sampling frequency. The PAMO system conducts pressure-adaptive handwriting optimization using a dual-layer optimization framework and includes a second-order Kalman filter for real-time denoising and position prediction and Cubic B-spline interpolation to ensure G^2 and C^2 continuity. The PAMO system significantly improves trajectory smoothness, achieving a 42.7% reduction in the standard deviation of curvature for straight lines (0.201 vs 0.351 for the baseline system) and a 42.0% reduction for circles (0.283 vs 0.488 for the baseline system). The one-step prediction function of the PAMO system reduced end-to-end latency to 16.9 ms, showing an 8.5 ms improvement over the baseline system. By mapping dynamic grip features, such as total grip force, center of pressure, and pressure gradient, to morphological stroke parameters, the PAMO system represents a significant advancement toward approximating the nuanced physical reality of traditional artistic expression.

1. Introduction

With the advancement of digital technology, digital painting and handwriting have become indispensable in artistic creation, industrial design, online education, and mobile offices.[1,2] Devices such as Wacom styluses and Apple Pencils have enhanced the naturalness of human–computer interaction. An essential technology underlying these devices is handwriting simulation based on single-point pressure sensing.[3,4] By detecting the vertical pressure exerted by the pen tip on the screen, these devices present corresponding line thickness and color depth, providing the experience of traditional calligraphy and painting. However, such a single-point pressure-sensing system presents inherent limitations.

*Corresponding author: e-mail: 2013024@wtu.edu.cn
https://doi.org/10.18494/SAM6237

ISSN 0914-4935
https://myukk.org/

First, it does not capture the information associated with pen-holding posture and grip technique. In traditional calligraphy and painting, artists sophisticatedly control brush-tip shape, ink intensity, and line texture by adjusting grip tightness, tilt angle, and finger force distribution. For example, to produce a willow-leaf stroke that transitions from thin to thick and back to thin, vertical pressure must be delicately changed with perfect coordination between the wrist and fingers. Such multidimensional controls cannot be conducted using a single pressure sensor.[5,6]

Second, hand tremors result in jagged edges or ripples during slow, fine strokes, which are regarded as high-frequency noises for an algorithm processing single-pressure sensor data. This degrades line smoothness and aesthetic quality. However, existing algorithms such as moving average may have smooth lines, but they can affect response speed and handwriting details, resulting in significant delays and blurry strokes.[7,8]

Third, most handwriting rendering algorithms rely on preset brush textures and simplified physical systems. This makes it difficult to reproduce the complex visual effects of ink on paper, such as gaps in strokes (flying white), rough textures (dry brush), and changes in darkness or moisture, which are determined by variations in pen pressure, drawing speed, and the amount of ink applied.[9]

To address these challenges, various algorithms have been proposed. Inertial measurement units (IMUs) are used to sense pen tilt and azimuth variations to increase input dimensions. However, this method does not detect the distribution of pen-grip force.[10] Advanced systems with deep-learning-based handwriting beautification algorithms exhibit limited generalization and immediacy, although they effectively reproduce specific styles. They also require expanded training data, but still suffer from poor interpretability.[11]

To address the challenges of reproducing calligraphic skills in digital environments, we developed a touch-type pressure-array-based manuscript optimization (PAMO) system based on a pressure sensor array. The system was developed to overcome the limitations stemming from insufficient front-end perceptual data. The developed system precisely captures hand–device interaction in writing and drawing to enable the generation of handwriting that reflects physical reality. The PAMO system enhances the fidelity of digital handwriting and painting and contributes to the advancement of the pressure sensor technology used in capturing complex grip forces, tilt variations, and dynamic pressure distributions. The integration of the pressure sensor technology leads to the development of sophisticated input devices capable of supporting nuanced artistic expression and improving the naturalness of human–computer interaction.

2. Methods

2.1 Hardware architecture

Coordinate data were collected using a commercial digitizer (Wacom Intuos Pro) following the standard Windows Ink protocol. Wacom Intuos Pro was used in this study because it is widely regarded as the industry standard for professional digital painting, offering high-precision performance that provides a rigorous benchmark for evaluating the PAMO system.[12] The hardware architecture of the developed system comprises the intelligent stylus, the data acquisition and processing module, and the display and interaction software (Fig. 1).

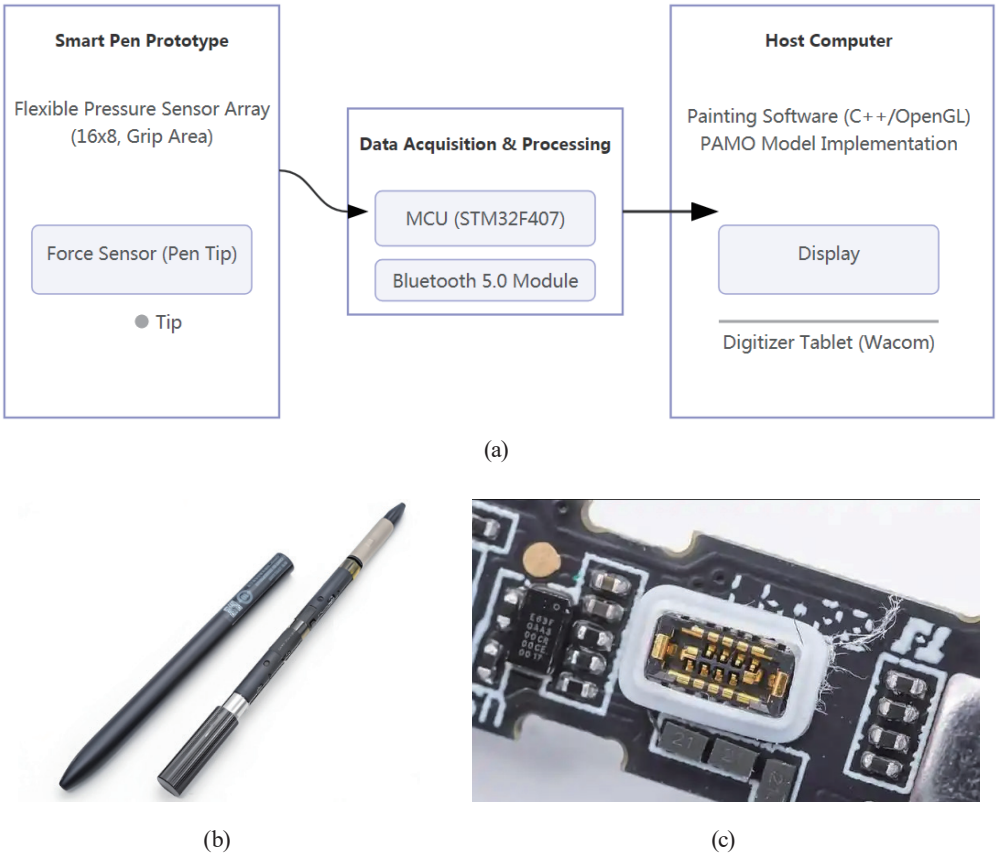


Fig. 1. (Color online) (a) System architecture and composition, (b) digital painting pen, and (c) piezoelectric sensor in the pen.

The stylus barrel is designed for ergonomic use and fabricated from 3D-printed acrylonitrile butadiene. The pen has a total length of 165 mm and a maximum diameter of 14 mm at the grip area, with a total weight of 28.5 g. A flexible pressure sensor array (128 units, 16 × 8 units) is embedded 3–7 cm from the pen tip. The internal electronics consist of an STM32F103 microcontroller unit (MCU) for data acquisition and an nRF52832 Bluetooth 5.0 module for wireless transmission. For ground-truth trajectory validation during initial calibration, a high-speed camera (Casio Exilim EX-ZR1500) was used to capture handwriting at 240 frames per second.[13]

The data acquisition and processing module is integrated into the barrel on an MCU. The MCU scans and records analog signals from the pressure sensor array and the pen-tip sensor at a sampling frequency of 120 Hz, utilizing a multiplexer and an analog-to-digital converter. The acquired raw data are wirelessly transmitted to a computer through a Bluetooth 5.0 module. Coordinate data are collected using a commercial digitizer (Wacom Intuos Pro) following the standard Windows Ink protocol. To synchronize and process the inputs, a customized drawing application was developed on the Windows 10 platform using C++ and OpenGL. This software integrates coordinate data from the digitizer with the pressure data stream from the stylus. The algorithm of the PAMO system is implemented, enabling real-time data processing and the generation and display of optimized handwriting on the screen.

2.2 Pressure sensor array

A flexible piezoresistive thin-film sensor array was employed in this study. The array measures 40 mm in height and 20 mm in width, and comprises 128 independent sensing units. Each unit measures 2 × 2 mm, with an interunit spacing of 0.5 mm. The sensor comprises two layers of flexible polyimide films enclosing a piezoresistive material layer. When external pressure is applied, the resistance of the piezoresistive layer changes nonlinearly. After calibration, the array achieves a measurement range of 0–10 N, an accuracy of 0.05 N, and a response time of less than 1 ms, satisfying the requirements for real-time interaction. The array is wrapped around the triangular grip region of the pen barrel, ensuring effective coverage of the contact points of the thumb, index finger, and middle finger. The detailed specifications are provided in Table 1.

Before experiments, each sensing unit was separately calibrated. A mapping relationship between the applied force and the analog-to-digital converter reading (V) was established, followed by a nonlinear correction to minimize unit-to-unit variability and improve measurement consistency.

2.3 Real-time pressure distribution

At each sampling moment, t_k, the pressure values of the 128 sensing units collected by the MCU form a 16 × 8 pressure matrix P_k as below.

$$P_k = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} & p_{1,4} & p_{1,5} & p_{1,6} & p_{1,7} & p_{1,8} \\ p_{2,1} & p_{2,2} & p_{2,3} & p_{2,4} & p_{2,5} & p_{2,6} & p_{2,7} & p_{2,8} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{16,1} & p_{16,2} & p_{16,3} & p_{16,4} & p_{16,5} & p_{16,6} & p_{16,7} & p_{16,8} \end{bmatrix}_k \tag{1}$$

To extract meaningful features from the original pressure matrix, the following indicators are calculated as follows.

Table 1
Specifications of flexible pressure sensor array used in stylus.

Parameter	Specification
Array layout	16 × 8 mm^2
Number of sensing units	128
Sensing unit size	2 × 2 mm^2
Active area	40 × 20 mm^2
Pressure range	0–10 N
Precision	0.05 N
Response time	<1 ms
Operating voltage	3.3 V
Substrate material	Polyimide

- Total grip force (*TGF*): The sum of all pressure values within the sensor matrix, representing the overall tightness of the user's grip on the pen.

$$TGF_k = \frac{1}{16}\sum_{i=1}^{16} P_{ik} \tag{2}$$

- Center of pressure (*CoP*): The location of the resultant force application point within the array coordinate system, indicating the primary area of hand contact and force concentration.

$$CoP_{xk} = \frac{\sum_{i=1}^{N_k}\sum_{j=1}^{I_i} p_{ij}^k t_{ij}^k}{TGr_k} \tag{3}$$

$$CoP_{yk} = \frac{\sum_{i=1}^{I_k}\sum_{j=1}^{J} j \cdot p_{ijk}}{TGf_k} \tag{4}$$

- Pressure gradient (PG): The differences in the pressure matrix along the two principal axes. PG reflects the rate of change in pressure, which is associated with pen-tip rotation and variations in applied force.[14]

Using the extracted features, *CoP* and *TGF* are visualized and compared with the original pressure heat map, as shown in Fig. 2.

2.4 Noise from hand tremor

Hand tremors are involuntary rhythmic movements with frequencies ranging from 8 to 12 Hz. In digital painting, hand tremors introduce high-frequency noise into the coordinate

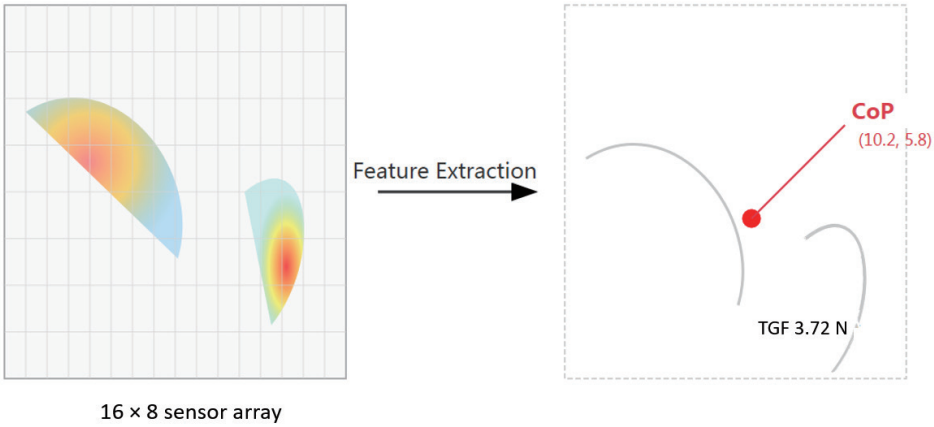


Fig. 2. (Color online) (a) Original pressure heat map generated on the sensor array when the user holds the pen and (b) the calculated *CoP* and *TGF*.

sequences collected from the pen tip.[15] We modeled this effect as an additive Gaussian white noise process as follows. Let the true, smooth pen-tip position be (x_k, y_k), then the observed position is expressed as

$$x_k^0 = x_k + v_{xk}, \tag{5}$$

$$y_k^0 = y_k + v_{y_k}. \tag{6}$$

Here, v_{xk} and v_{yk} are Gaussian noises with a mean of 0 and a standard deviation (SD) of σ_v. In this study, we collected data by having users slowly draw straight lines and circles on the screen, and estimated the statistical characteristics of the noise by using high-pass filtering on the data. In the developed model in this study, σ_v is between 0.3 and 0.5 pixel units.

2.5 Position prediction based on Kalman filtering

To address the issues of noise and delay, we adopted the Kalman filter (Table 2). We modeled the movement of the pen tip as a second-order (uniform acceleration) dynamic system. At time step k, the state vector S_k of the system is defined as position (x_k, y_k), velocity $(\dot{x}_k, \dot{y}_k)$, and acceleration $(\ddot{x}_k, \ddot{y}_k)$.

$$S_k = [x_k, y_k, \dot{x}_k, \dot{y}_k, \ddot{x}_k, \ddot{y}_k]^T \tag{7}$$

2.5.1 Prediction

On the basis of the state of $k - 1$ at the previous moment, the state of k is predicted at the current moment using

$$\overline{S}_k = A \cdot \hat{S}_{k-1}. \tag{8}$$

Here, A is the state transition matrix. For the uniform acceleration model, at Δt, the following is constructed.

Table 2
Parameters of Kalman filter used in position prediction (diag: diagonal matrix).

Parameter	Value	Description
Sampling interval (Δt)	1/120	Inverse of system sampling frequency (ms)
Process noise covariance (Q)	diag(0.01, 0.01, 0.1, 0.1, 1, 1)	Empirically tuned, reflecting model uncertainty
Measurement noise covariance (R)	diag(0.4^2, 0.4^2)	Based on noise modeling
Initial state covariance (P_0)	diag(1, 1, 1, 1, 1, 1)	Assuming high initial uncertainty

$$A=\begin{bmatrix} 1 & 0 & \Delta t & 0 & \frac{1}{2}\Delta t^2 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & \frac{1}{2}\Delta t^2 \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

Meanwhile, the covariance matrix of prediction errors is updated using

$$\overline{P}_k = AP_{k-1}A^T + Q. \tag{10}$$

Here, Q is the process noise covariance matrix, representing the uncertainty of the model.

2.5.2 Update

Observed values at the current moment are used to correct the predicted state.[16] First, the Kalman gain is calculated as

$$K_k = \overline{P}_k H^T (H\overline{P}_k H^T + R)^{-1}, \tag{11}$$

where H is the observation matrix presented as

$$H=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{12}$$

Then, the status estimate is updated as

$$\hat{S}_k = \overline{S}_k + K_k\left(Z_k - H\overline{S}_k\right). \tag{13}$$

Finally, the error covariance matrix is updated as

$$P_k = \left(I - K_k H\right)\overline{P}_k. \tag{14}$$

Through this recursive process, the position component is obtained as the smoothed coordinate after using Kalman filtering. In the prediction, the pen-tip position at the subsequent time instant is estimated. The predicted position is used to initiate rendering calculations in advance, thereby effectively compensating for system latency.

2.6 Cubic B-spline interpolation

In curve design, position continuity (G^0) ensures that the segments meet at the same point, while tangent continuity (G^1) ensures that their directions align. G^2 continuity requires one further step so that the curvature, the rate at which the curve bends, matches at the junction, and the curve transitions smoothly without any change in bending. Cubic B-spline interpolation provides G^2 continuity for trajectory smoothing. By maintaining a continuous curvature, cubic B-splines generate paths that are visually smooth and physically realistic, making them particularly suitable for handwriting, motion planning, and computer graphics applications.[17]

Kalman filtering enables the noise removal of the point sequence, but the points are still linearly connected.[18] To generate a smooth curve with G^2 continuity, we adopted cubic B-spline interpolation. The coordinates of Point $Q(u)$ on the curve are calculated as follows.

$$Q(u) = \sum_{i=0}^{n} N_{i,p}(u) C_i \tag{15}$$

Here, $N_{i,p}$ is a weight determined by the De Boor–Cox recursive formula and indicates how much influence each control point has on a specific part of the curve. For cubic B-splines ($p = 3$), it ensures C^2 parametric continuity, meaning that the curve is smooth and its curvature changes gradually, preventing sharp elbows in the digital line.

In the PAMO system developed in this study, the smoothed point sequence produced by Kalman filtering is not directly used as the control points of the B-spline. Instead, these points serve as interpolation targets for the curve. A fast local interpolation algorithm is applied to compute the four control points of each cubic B-spline segment from every four consecutive data points. This method is employed to ensure that the curve passes strictly through all data points while preserving local controllability and computational efficiency. The overall process is illustrated in Fig. 3.

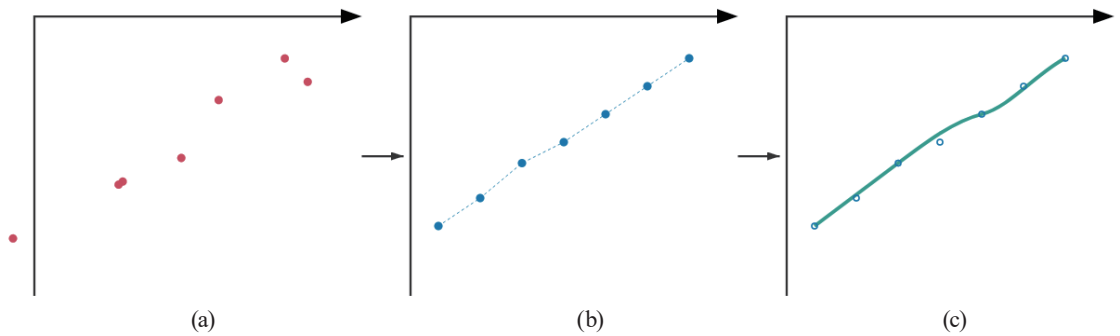


Fig. 3. (Color online) Process of digital ink refinement: (a) raw noisy input coordinates, (b) point sequence denoised via Kalman filtering (uniform acceleration model), and (c) smooth curve generation using cubic B-spline interpolation to achieve G^2 and C^2 continuity.

2.7 Stroke rendering of pressure adaptive control

Traditional stroke rendering methods map pen-tip pressure linearly to stroke width. In contrast, the method developed in this study incorporates the grip pressure distribution matrix into stroke-shape control. Specifically, when generating the B-spline curve segment between P_k and P_{k+1}, the result of fixed geometric calculation of control points is replaced using that obtained through a pressure-adaptive approach. The positions of the control points are influenced by the immediate pressure characteristics.

We define the rendering width W_k and a morphological control parameter α_k as follows.

$$W_k = w_{base} + w_{tip} \cdot f\left(p_{tip_k}\right) + w_{TGF} \cdot g\left(TGF_k\right) \tag{16}$$

Here, w_{base}, w_{tip}, and w_{TGF} are weight coefficients, p_{tip_k} is the amount of physical force the user applies to the stylus tip, TGF_k is the secondary force measurement likely related to the interaction between the pen and the tablet surface, and f and g are configurable nonlinear mapping functions used to transform raw pressure and force data into a more natural, aesthetic range for stroke width.

$$\alpha_k = \alpha_{base} + \alpha_{vel} \cdot \sqrt{\dot{x}_k^2 + \dot{y}_k^2} \tag{17}$$

Here, α_{base} is the baseline shape of the brush when the pen is stationary, α_{vel} is the coefficient that determines how much the brush shape deforms from the speed of the stroke, and $\dot{x}_k^2$ and $\dot{y}_k^2$ are the first-order derivatives from S_k. $\sqrt{\dot{x}_k^2 + \dot{y}_k^2}$ represents the resultant speed of the pen tip. α_k is jointly governed by *CoP* and PG. We used α_k to adjust the positions of the two internal control points $C_{k,1}$ and $C_{k,2}$ of the B-spline segment. When the force application point of the user's finger changes or the force application is uneven, α_k causes $C_{k,1}$ and $C_{k,2}$ to shift to one or both sides of the normal curve direction (Fig. 4). The adjustment is conducted using the following equations.

$$C_{k,1} = C_{k,1} + \alpha_k \cdot W_k \cdot \vec{n}_k \tag{18}$$

$$\dot{C}_{k,2} = C_{k,2} - \alpha_k \cdot W_{k+1} \cdot \dot{n}_{k+1} \tag{19}$$

Here, n_k is the normal vector of the curve at point P_k. This adjustment introduces asymmetry into the generated B-spline segments, producing naturalistic effects such as brushstrokes, side strokes, and gaps in strokes during rendering. For example, when the index finger exerts force and *CoP* shifts upward, the resulting deformation simulates calligraphic phenomena such as pause or fold, which cannot be reproduced only through pen-tip pressure. The final stroke outline is formed by these adjusted B-spline segments.

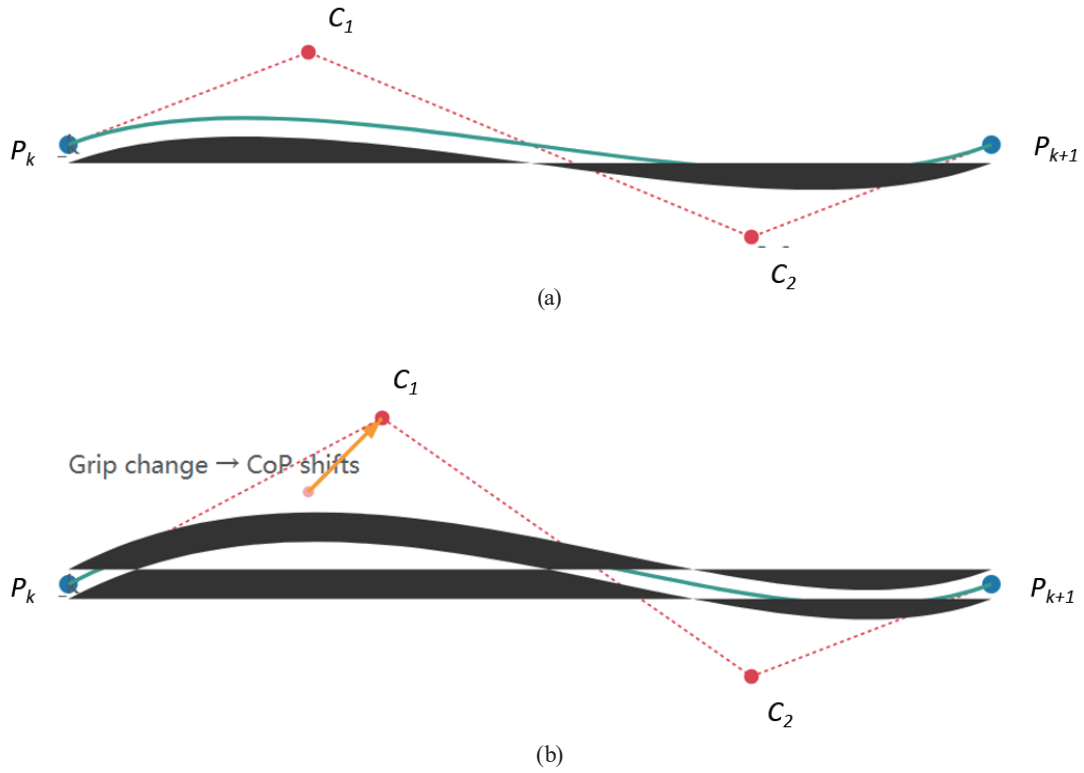


Fig. 4. (Color online) Pressure-adaptive stroke rendering. (a) Symmetrical strokes generated by standard B-spline interpolation and (b) the change in grip pressure causing *CoP* shift and control points *Cs* shifts, forming a natural side edge (chisel tip).

3. Results

3.1 Trajectory smoothness and denoising

We experimented with the PAMO system to evaluate its performance. Fifteen participants were involved in this study to evaluate the developed sensing framework and user interface. The participants included eight males and seven females, ensuring a balanced gender representation. The age of the participants ranged from 23 to 29 years, with a mean age of 25.8 years. To ensure the validity of the results across different levels of expertise and professions, the participants were selected from diverse backgrounds within the digital arts industry, including freelancers, game concept artists, user interface/user experience designers, and graduate students in specialized art programs. Professional experience in digital painting ranged from 2.8 to 8.0 years (4.77 years on average), indicating that all participants possessed appropriate technical proficiency to provide expert feedback on the system's performance and utility (Table 3).

Wacom Intuos Pro provides a built-in driver, and the default settings of Adobe Photoshop were used. The participants drew straight lines and circles on the screen at a low speed of 2 cm/s. From the drawings, we collected the raw coordinate data, the data processed by the baseline system, and the data processed by the PAMO system.

Table 3
Demographic profiles and professions of the study participants.

Participant number	Gender	Age (years)	Profession	Digital painting experience (years)
1	Male	24	Freelancer	3.5
2	Female	28	Freelancer	6
3	Male	25	Game content creator	4.2
4	Female	27	Graduate student majored in animation	5
5	Male	23	Communication design	2.8
6	Female	29	User interface/user experience designer	7.5
7	Male	26	Freelancer	5.3
8	Female	25	Master's student (digital art education)	4
9	Male	24	Film storyboard creator	3
10	Female	27	Independent manga artist	6.8
11	Male	26	Freelancer	3.2
12	Female	25	Freelancer	4.5
13	Male	28	Digital watercolor instructor	8
14	Female	24	Graphic designer	3.7
15	Male	26	Graphic designer	4

Smoothness was quantified using the SD of trajectory curvature. The SD of the ideal curvature of a straight line is zero, while that of a circle is constant. A smaller SD indicates smoother lines and more regular shapes.

The results in Tables 4 and 5 demonstrate that the PAMO system significantly improves the smoothness of digital ink trajectories by reducing curvature variation. According to the experimental data, the SD of curvature for the PAMO system is consistently lower than both the raw data and the baseline system. For slow straight line drawing, the PAMO system showed an SD of curvature of 0.201, representing a 42.7% improvement over the baseline system. Similarly, in slow circle drawing, the PAMO system attained an SD of 0.283, which is a 42.0% improvement over the baseline. These reductions in curvature fluctuation show that the Kalman filter of the PAMO system effectively suppresses hand-tremor noise. The results also prove that the system produces geometrically smoother trajectories than the smoothing algorithms embedded in the PAMO system.

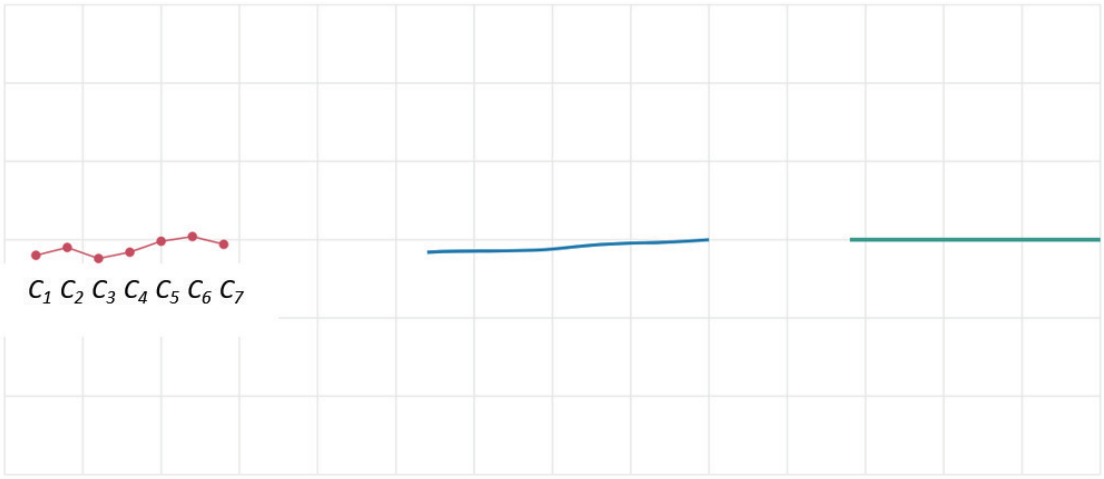
Figure 5 shows the results presented in Tables 3 and 4 regarding the smoothness of pen trajectories. The raw data shows jitter and noise arising from hand tremor and hardware measurement errors, as described in Table 2 [Fig. 5(a)]. The baseline system showed the smoothing performance of the commercial software. Although this output is smoother than the raw data, it displays subtle wobbles and variations in curvature 9 [Fig. 5(b)]. The trajectory generated by the PAMO system closely approximates a perfect straight line. This result demonstrates that the combined use of the Kalman filter for noise suppression and cubic B-spline interpolation for maintaining C^2 continuity effectively eliminates hand tremor and produces geometrically superior trajectories.

Table 4
SD of trajectory curvature for slow straight line drawing.

Data	SD of curvature	Standard error	Improvement over baseline (%)
Raw data	0.892	0.045	—
Baseline system	0.351	0.021	0 (Reference)
PAMO system	0.201	0.015	42.7

Table 5
SD of trajectory curvature for slow circle drawing.

Data	SD of curvature	Standard error	Improvement over baseline (%)
Raw data	1.124	0.058	—
Baseline system	0.488	0.033	0 (Reference)
PAMO system	0.283	0.024	42.0



(a) (b) (c)

Fig. 5. (Color online) Stroke smoothing performance (C_i: control points; x- and y-axes: horizontal and vertical positions of pen tip on tablet surface). (a) Raw data and (b) baseline and (c) PAMO systems.

3.2 Immediacy and response

Table 6 shows that system latency significantly affects user experience. We used high-speed cameras to capture the moment when the user's pen tip touches the screen and the handwriting appears on the screen, and measured the end-to-end latency through frame-by-frame analysis. The latency is compared for the baseline system, the PAMO system with prediction not enabled, and the PAMO system with prediction using Kalman filtering. Because of the higher algorithmic complexity of the PAMO system, its latency (28.9 ms) is higher than that of the baseline system (25.4 ms) when prediction is not enabled. However, when the one-step prediction function of the Kalman filter is enabled, the mean system latency was reduced to 16.9 ms with an SD of 1.8 ms, which was 8.5 ms faster than the baseline system (an increase by 33.5%). This reduction is

Table 6
End-to-end model latencies.

Model	Mean latency	SD
Baseline system	25.4	2.1
PAMO (without prediction)	28.9	2.5
PAMO (with prediction)	16.9	1.8

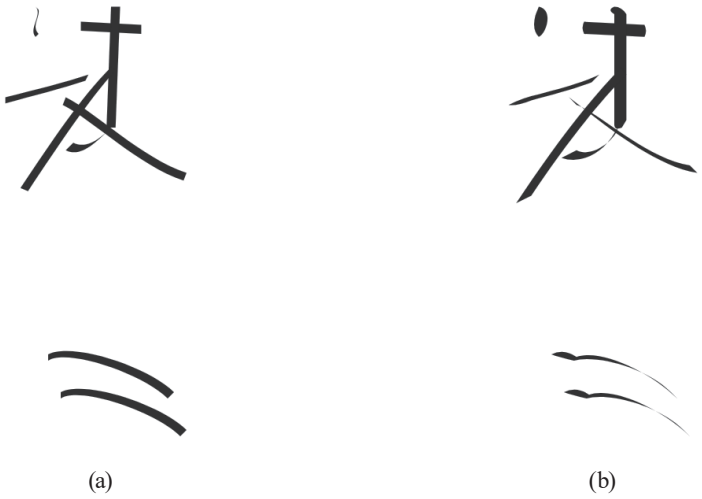
attributed to the predicted location points, which enable rendering to start working one sampling period (8.3 ms) in advance, effectively compensating for computational and communication delays. The appearance time point of the ink marks in the PAMO system with prediction is earlier than that of the other two systems.

3.3 Quality assessment of strokes

We asked the participants to draw Chinese characters and painting elements with complex stroke variations using the baseline and PAMO systems. Figure 6 shows the character 龙 and bamboo leaves drawn by a participant with a professional art background. The PAMO system provides richer and more natural details than the baseline system. In the character 龙 [simplified Chinese letter for dragon (龍)], the transitions of thickness at the starting, running, and ending points of dots, horizontal strokes, vertical strokes, hooks, lifts, and left-falling and right-falling strokes, as well as the shape of the brush tip, are closer to calligraphy drawn with a brush [Fig. 6(a)]. At the end of the stroke, the PAMO system simulates a natural tip effect by sensing the user's gradual relaxation and lifting of the finger (*TGF* decreases, while *CoP* changes). When drawing bamboo leaves [Fig. 6(b)], the PAMO system creates a cadence with a bamboo joint and a change in force from light to heavy, with a full and elastic form. Although the strokes of the baseline system's forms are monotonous, with linear variations in thickness, lacking vivid details and charm, it responds to pressure.

While the PAMO system is intuitive and requires no specialized technical pretraining for general use, the high-level calligraphic effects shown in Fig. 6 leverage the existing skills of a trained artist. The system acts as a high-fidelity bridge, ensuring that the professional's nuanced grip transitions are accurately reflected in the digital medium. For novice users, the PAMO system provides a more forgiving and smoother trajectory than does the baseline system, although artistic mastery of the pressure-adaptive features remains a skill-based process.

To assess the participants' perceptions of system performance, we employed a 7-point Likert scale, which is widely used for measuring subjective attitudes and experiences. This scale provides a balanced range of responses, allowing participants to express varying degrees of agreement or satisfaction. Each point on the scale corresponds to a qualitative category: 1 = strongly disagree, 2 = disagree, 3 = somewhat disagree, 4 = neutral, 5 = somewhat agree, 6 = agree, and 7 = strongly agree. The 7-point format was selected to capture subtle differences in user evaluations while maintaining ease of use and statistical robustness. In this study, the participants rated four dimensions of system performance: line smoothness, response, pressure expression, and realism. The subjective scoring results (Table 7) are highly consistent with the quantitative analysis, showing a significant preference for the PAMO system across all metrics.



(a) (b)

Fig. 6. Calligraphic and artistic rendering using the PAMO system: the character '龙' rendered in cursive (Caoshu) style and bamboo leaves drawn by participant using (a) baseline and (b) PAMO systems.

Table 7
Usage assessment results of participants.

Dimension	Baseline system	PAMO system	*p*-value (paired *t*-test)
Line smoothness	5.1 ± 0.8	6.5 ± 0.6	<0.001
Reponse	5.5 ± 0.7	6.7 ± 0.5	<0.001
Pressure expression	4.8 ± 0.9	6.6 ± 0.7	<0.001
Realism	4.9 ± 1.0	6.6 ± 0.5	<0.001

The PAMO system scored higher than the baseline system in all four dimensions, with a *p*-value of less than 0.001 in paired *t*-tests. The PAMO system outperformed the baseline system in line smoothness and response. The advantages of the PAMO system lie in pressure expression and realism, where the average scores increased from 4.8 and 4.9 to 6.6, an improvement of 35%. The grip pressure sensor array enables the PAMO system to capture users' creative intentions and transform them into expressive and realistic visual handwriting.

4. Discussion

4.1 Comparison with other systems

Compared with existing digital handwriting optimization technologies, the PAMO system provides an advantage owing to its perception–processing–rendering ability (Table 8). While traditional single-point pressure sensing, the current baseline system, is cost-effective and widely adopted, it suffers from limited expressiveness owing to low-dimensional input. Regardless of software filtering complexity, the other methods are constrained to a three-dimensional optimization space of time-position-single-point pressure. By introducing a 128-dimensional pressure array, the PAMO system expands the input space to a high-

Table 8
Features of PAMO system and different technical methods.

System	Sensory input	Advantage	Disadvantage
Single-point pressure sensing (baseline system)	Tip pressure, coordinates	Low cost, widely adopted	Low-dimensional input, limited expressiveness
IMU-enhanced	Pen tilt/azimuth angles	Can simulate chisel tips	Insensitive to grip force, poor fine finger control
Learning-based AI system	Coordinate sequence, images	Excellent results for specific styles	Poor real-time performance, low controllability
PAMO system	Pressure array, tip pressure, coordinates	High-dimensional input, real-time control, high expressiveness, and good generality	Higher hardware cost, calibration required

dimensional manifold to distinguish between maintaining a grip while increasing tip pressure and adjusting finger posture to change brush direction. In existing systems, these interactions might produce identical effects. However, the PAMO system utilizes changes in *CoP* and PG to generate unique, realistic pen-tip effects.

Alternative systems, such as IMU-enhanced models, utilize pen tilt and azimuth angles to simulate chisel-tip effects. However, these are insensitive to grip force and lack fine finger control. In calligraphy, lifting, pressing, pausing, and stopping techniques involve minute inclination changes made by precisely applied finger force, movements that an IMU cannot accurately sense. While learning-based AI models such as convolutional neural networks and generative adversarial networks produce realistic style transfers, they show limitations in real-time performance and controllability.

While these post processing models lack the immediacy required for continuous user control or the generality to adapt to multiple styles, the PAMO system is a general-purpose physical model. Rather than learning a specific artistic style, the PAMO system is dedicated to reflecting the physical process in real time. A next-generation smart pen needs to combine the PAMO's pressure array with an IMU to reflect the full-body posture and local force application.

We compared the performances of the PAMO system and other optimization systems that include a long short-term memory (LSTM) network and a multi-channel Savitzky–Golay (S–G) filter (Table 9).[19] While the LSTM system excels in predicting complex patterns, its computational overhead leads to a latency of 45 ms, which is perceptible in digital painting. The S–G filter, although fast, lacks the predictive capabilities of the PAMO system's Kalman layer, resulting in higher curvature variance during rapid strokes. As shown in Table 9, PAMO reduces curvature SD by 15.4% compared with the S–G filter and 8.2% compared with the LSTM system, while maintaining a sub-10 ms processing time.[20]

4.2 Sensor density

In the PAMO system, an appropriate pressure sensor array density must be chosen to balance between sensory resolution and practical system constraints. As illustrated in Fig. 7, lower densities, such as a 4 × 2 array in which the sensor unit size exceeds the fingertip contact area, cannot capture subtle movements of *CoP* or meaningful variations in PG. In the array, the

Table 9
Performances of PAMO system and other optimization systems.

System	Average curvature SD (line)	Average curvature SD (circle)	Processing latency
S–G	0.238	0.285	4 ms
Learning-based (LSTM)	0.219	0.252	45 ms
PAMO (this study)	0.201	0.224	8 ms

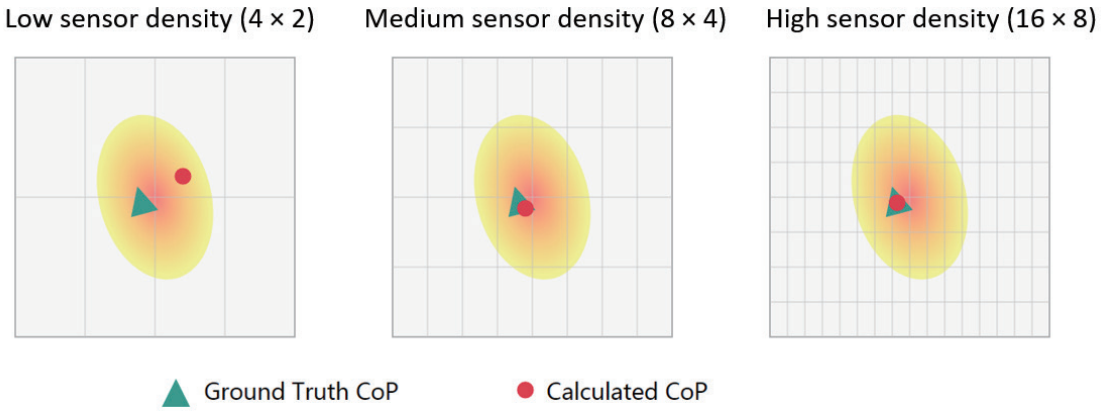


Fig. 7. (Color online) Accuracy of *CoP* in same pressure distribution under different sensor densities.

system degenerates into simple multipoint pressure sensing and loses the ability to achieve fine morphological control. Although increasing density theoretically yields better results, it might introduce challenges. First, increasing the number of sensors incurs high costs. Second, the expanded data volume increases computational overhead, which degrades real-time system performance. Third, higher densities of sensors reduce power efficiency, which is particularly challenging for wireless mobile devices.

In this study, the 16 × 8 sensor array effectively covers the finger contact area. With a unit size of 2 mm, smaller than the core region of a typical fingertip pressure distribution, the array is sufficient to capture the essential *CoP* and PG. Because the marginal benefit of increasing density for stroke-shape control is negligible, the 16 × 8 array is an optimal balance of effectiveness, cost, and performance. However, it is necessary to systematically quantify the relationship between sensor density and the enhancement of stroke expressiveness.

4.3 Limitations and future research

Although the PAMO system has demonstrated promising results, limitations remain, necessitating further research.

Since the current system employs uniformly defined pressure-mapping functions and weight coefficients, pen-holding habits, palm sizes, and force application strategies might vary considerably among users. For optimal performance, the system must adopt a brief personalized calibration procedure, enabling users to adjust these parameters to their individual style by

drawing a set of sample strokes. The pen is relatively large and operates in a wired/Bluetooth dual mode, which limits its applicability as a commercial product. Integrating high-density flexible sensors, processing chips, batteries, and wireless communication modules into a consumer-grade stylus at reduced cost is still challenging. Advancements in flexible hybrid electronics must be pursued to overcome such a limitation.

The PAMO system emphasizes stroke contours. A more comprehensive system needs to be developed to reflect the physical properties of ink and pigment, including humidity, viscosity, diffusion, and interactions with paper texture. High-dimensional control parameters generated by the PAMO system can be used as inputs for advanced fluid dynamics simulations or real-time texture synthesis algorithms. The integration of machine learning into the PAMO system requires further investigation. A lightweight neural network can be trained to learn improved mappings from pressure-matrix features to B-spline control-point offsets, replacing the manually designed functions. Such hybrid methods, combining physical modeling with data-driven learning, enhance system scalability and performance.

While the PAMO system enhances digital expressiveness, further research on the integration of active haptic feedback mechanisms is required to reach the level of traditional calligraphy.[20] By providing real-time tactile resistance that correlates with grip intensity and pen tilt, the system simulates the physical interaction between a brush and paper texture. Additionally, the high-dimensional data captured by the 128-unit array must be utilized as input for advanced fluid dynamics models to simulate complex ink effects such as viscosity and diffusion.[21] Transitioning from fixed mapping functions to lightweight, personalized neural networks enables automatic calibration for the diverse holding habits of different users.

While the current PAMO system with standard piezoresistive materials demonstrates high effectiveness, the adoption of nanomaterials can enhance the performance. Specifically, the integration of Max phase-derived two-dimensional material-based flexible sensors can enhance sensitivity by several orders of magnitude, enabling the detection of sub-gram grip variations.[22] Multimodal sensor arrays capable of simultaneously detecting normal and shear forces also enable the PAMO system to provide data on tangential friction.[23] Such new materials and sensor arrays can be used to reproduce different paper textures or the drag of a viscous brush. The adoption of microstructured active layers can also improve the linearity and response speed of the sensor array, refining the trajectory smoothing and pressure mapping algorithms.[24]

5. Conclusions

We developed and validated the PAMO system, which addresses the limitations of the present digital painting technology. Traditional digital styluses rely on single-point pressure sensing, a limitation that prevents them from capturing the multidimensional nuances of human grip and posture. As a result, they lack artistic expressiveness and are highly susceptible to hand-tremor noise, resulting in jagged trajectories and a monotonous “digital” feel that fails to replicate the aesthetic qualities of traditional calligraphy.

To overcome these challenges, we designed an intelligent stylus equipped with a 128-unit flexible pressure sensor array (16 × 8) capable of capturing high-dimensional grip force

distributions at 120 Hz. The system processes the data through a collaborative perception–processing–rendering chain. First, a second-order Kalman filter (uniform acceleration model) is used to suppress hand-tremor noise and predict future positions to compensate for latency. Next, cubic B-spline interpolation ensures G^2 and C^2 continuity, producing visually and physically realistic trajectories. Extracted features, including *TGF*, *CoP*, and PG, are used to adjust B-spline control points to simulate natural brush effects such as chisel tips.

The PAMO system improved trajectory smoothness by 42.7% for straight lines and 42.0% for circles compared with those of the baseline commercial system (Wacom Intuos Pro). By leveraging one-step prediction, it reduced mean end-to-end latency to 16.9 ms, representing a 33.5% improvement in responsiveness. Sensor analysis revealed the 16 × 8 array density (2 mm unit size) to be the optimal configuration for capturing fingertip pressure characteristics while balancing cost and power efficiency. Subjective assessments by the participants in the experiment with the PAMO system rated the system higher in pressure expressiveness and realism, with mean Likert scores of 6.6/7.0, corresponding to a 35% increase over traditional systems.

The PAMO system demonstrates that integrating high-dimensional tactile sensing with physical modeling enables digital devices to capture subtle creative intentions. The PAMO system provides a technical foundation for the next generation of professional digital art tools, bridging the gap between digital stylus technology and the expressive qualities of traditional calligraphy.

Acknowledgments

This research was supported by the ‘2024 Hubei Province Higher Education Laboratory Research Project: Research on the Design of a Short Video Fusion Matrix for Standardized Operations and Safety Management in Experimental Teaching (Project Number: HBSY2024-51).’

References

1 Y. Tang, Y. Wang, T. Hu, R. Yi, X. Tan, and L. Ma: IEEE Trans. Vis. Comput. Graph. **31** (2025) 10897. https://doi.org/10.1109/TVCG.2025.3618184
2 L. Han, B. Pan, Y. Chen, and J. tang: Neural Comput. Appl. **37** (2024) 13309. https://doi.org/10.1007/s00521-024-09955-w
3 K. Madono and E. Simo-Serra: Comput. Graph. Forum **42** (2023) e14695. https://doi.org/10.1111/cgf.14965
4 M. Bigerelle, R. Guibert, A. Mironova, F. Robache, R. Deltombe, L. Nys, and C. A. Brown: Metrol. Prop. **11** (2023) 015001. https://doi.org/10.1088/2051-672X/acbe53
5 C. Ma, P. Zhang, L. Pan, X. Li, C. Yin, A. Li, R. Zong, and Z. Zhang: Expert Syst. Appl. **203** (2022) 117400. https://doi.org/10.1016/j.eswa.2022.117400
6 P. Alexander and S. Daniel: Front. Artif. Intell. **5** (2022) 787179. https://doi.org/10.3389/frai.2022.787179
7 A. Heimann-Steinert, A. Latendorf, A. Prange, D. Sonntag, and U. Müller-Werdan: Psychol. Res. **85** (2020) 3075. https://doi.org/10.1007/s00426-020-01452-8
8 M. Shinji and T. Hiroki: Proc. 82nd Annu. Meeting of the Japanese Psychological Association (JPA, 2018). 3PM-056. https://doi.org/10.4992/pacjpa.82.0_3pm-056 (in Japanese).
9 D. Prattichizzo, L. Meli, and M. Malvezzi: IEEE Trans. Haptics **8** (2015) 356. https://doi.org/10.1109/toh.2015.2434812

10 F. Lamberti, A. Sanna, and G. Paravati: EURASIP J. Image Video Process. **2014** (2014) 53. https://doi.org/10.1186/1687-5281-2014-53
11 R. Hirabayashi, E. A. Draffan, and R. Newman: Assist. Technol. Res. Ser. **33** (2013) 386. https://doi.org/10.3233/978-1-61499-304-9-386
12 S. Amershi, D. Weld, M. Vorvoreanu, and A. Fourney: Proc. CHI Conf. Human Factors in Computing Systems (ACM, New York, 2013) 1243. https://doi.org/10.1145/3290605.3300233
13 Casio Computer: https://support.casio.com/storage/en/manual/pdf/EN/001/EXZR1500_FB_EN.pdf (accessed February 2026).
14 J.-S. Wang and F.-C. Chuang: IEEE Trans. Ind. Electron. **59** (2012) 299. https://doi.org/10.1109/TIE.2011.2167895
15 A. Cai, C. Yao, H. Liu, C. Li, Y. Zhang, Y. Guo, X. Wang, and X. Ban: Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. **9** (2025) 1. https://doi.org/10.1145/3749538
16 A. Omorkulov, A. Maripov, M. Eshbaev, K. Shakirov, T. Alieva, E. Toktorbaeva, T. Kadyrbaeva, and Z. Abdullaeva: ADR **9** (2021) https://doi.org/10.4236/adr.2021.91002
17 Y.-F. Zhou, C.-M. Zhang, and S.-S. Gao: Lect. Notes Comput. Sci. **5359** (2008) 1096. https://doi.org/10.1007/978-3-540-89646-3_109
18 A. Prange and D. Sonntag: Front. Artif. Intell. **5** (2022) 71. https://doi.org/10.3389/frai.2022.787179
19 H. Schwellnus, H. Carnahan, A. Kushki, H. Polatajko, C. Missiuna, and T. Chau: Aust. Occup. Ther. J. **59** (2012) 180. https://doi.org/10.1111/j.1440-1630.2012.01014.x
20 D. Prattichizzo, L. Meli, and M. Malvezzi: IEEE Trans. Haptics **8** (2015) 356. https://doi.org/10.1109/TOH.2015.2434812
21 R. Kosara, S. Miksch, and H. Hauser: IEEE Comput. Graph. Appl. **22** (2002) 22. https://doi.org/10.1109/38.974515
22 Z. Hu, F. Xie, Y. Yan, H. Lu, J. Cheng, X. Liu, and J. Li: RSC Adv. **14** (2024) 9547. https://doi.org/10.1039/D3RA07772A
23 E. Choi, S. Hwang, Y. Yoon, H. Seo, J. Lee, S. Yeom, G. Ryu, H. Yang, S. Kim, O. Sul, and S.-B. Lee: Sensors **19** (2019) 1300. https://doi.org/10.3390/s19061300
24 A. H. Anwer, N. Khan, M. Z. Ansari, S.-S. Baek, H. Yi, S. Kim, S. M. Noh, and C. Jeong: Sensors **22** (2022) 4460. https://doi.org/10.3390/s22124460