IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 January 2023, accepted 21 February 2023, date of publication 6 March 2023, date of current version 17 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3252902

RESEARCH ARTICLE

A Chinese Calligraphy-Writing Robotic System Based on Image-to-Action Translations and a Hypothesis Generation Net

MIN-JIE HSU[1], PO-CHAO YEH[1,2], YI-HSING CHIEN[1], CHENG-KAI LU[1], (Senior Member, IEEE), WEI-YEN WANG[1], (Fellow, IEEE), AND CHEN-CHIEN JAMES HSU[1], (Senior Member, IEEE)

[1]Department of Electrical Engineering, National Taiwan Normal University, Taipei 106, Taiwan
[2]AIMobile Company Ltd., Taipei City 11167, Taiwan

Corresponding author: Cheng-Kai Lu (cklu@ntnu.edu.tw)

This work was financially supported by the "Chinese Language and Technology Center" of National Taiwan Normal University (NTNU) from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan, and National Science and Technology Council (NSTC), Taiwan, under Grants no. 111-2221-E-003-025, 110-2221-E-003-020-MY2, and 111-2634-FA49-013 under the program of AI Thematic Research Program to Cope with National Grand Challenges through Pervasive Artificial Intelligence Research (PAIR) Labs of the National Yang Ming Chiao Tung University.

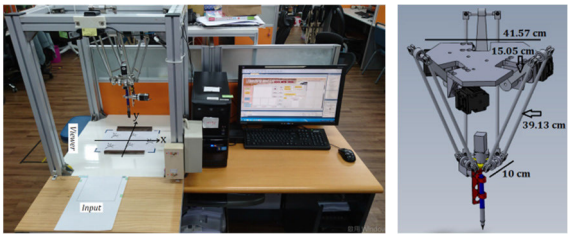
ABSTRACT This paper attempts to use a delta robot's structure and reliable coordinates to develop a self-learning Chinese calligraphy-writing system that requires precise control. Ideally, to achieve human-like behavior, a delta robot can learn stroke trajectories autonomously and present the stroke beauty of calligraphy characters. Unfortunately, state-of-the-art approaches have not yet considered the presentation of stroke beauty resulting from angles of rotation and tilt of the brush. This paper presents an integrated system consisting of a stroke processing module, a hypothesis generation net (HGN) learning model with self-learning capability, a delta robot, and an image capture module. Our approach utilizes both the stroke trajectories from the stroke processing module and angles information from the HGN learning model to automatically produce five degrees of freedom action instructions. Based on the instructions, the delta robot completes calligraphy writing. Then, the image capture module provides feedback to the writing system for error calculation and coordinate correction. We utilize the mean absolute percentage error to verify the performance of the writing results. A correction algorithm and linear regression were used to improve the error correction results (less than 2% error). After several cycles, the written results approached the target sample finally. Consequently, the written results produced by the delta robot prove that our proposed system with learning ability can write Chinese calligraphy aesthetically.

INDEX TERMS Chinese calligraphy, hypothesis generation net, image-to-action translation, robotic calligraphy system.

I. INTRODUCTION

In general, a common method of robot learning is through human demonstrations. After a robot records the trajectory and behavior required to complete a given task, it can perform fixed and repetitive actions through reproducing the given trajectory. To go a step further, a new generation of robots can autonomously complete learning actions through image recognition and human-programmed strategies. However, correcting errors in behaviors during the learning process requires more time to perform multiple tasks accurately. In recent years, the learning system of the Chinese character writing robot has the following approaches: generating the stroke trajectory by using a generative adversarial network (GAN) [1], [2], [3] or by using a control polygon [4], [5], converting the stroke trajectory from the image [6], modeling the stroke trajectory according to the font image [7], [8], [9], [10], and using the perception system to obtain the course of the stroke [11], [12], [13], [14], and so on.

The associate editor coordinating the review of this manuscript and approving it for publication was Shun-Feng Su.



(a) (b)

FIGURE 1. (a) The apparatus of a calligraphy-writing system. (b) Mechanical length of delta robot.

Most Chinese calligraphy robots first model human sample fonts [15] and then acquire data such as stroke features [16], [17], stroke decomposing [18], [19], [20], and stroke recognition [21]. The Chinese calligraphy-writing robots trained by humans execute the writing action according to the coordinates of the stroke trajectory database. According to the accuracy of the robotic arm and the characters that humans want to write, the robotic arm can be adjusted to precise coordinates of the writing position. That is, the robotic arm can write ideal calligraphy, according to the accurate coordinate trajectories set by the human. If the writing coordinates remain unchanged, the writing results remain the same. This learning model is mainly based on artificially established standard operating procedures, allowing the control program to be executed step by step. Unfortunately, both the intrinsic and extrinsic parameters of these written methods must be unchanged. Thus, the learning system cannot actually learn and has no self-learning ability.

A self-learning robot can learn and complete the required target actions by itself according to the tasks given by humans. A learning model is designed regarding the human cognitive system so that the robot has the ability to think, develop and try a new strategy based on its experience. In order to evaluate the strategy, an estimation system is used to compare the target sample and output results. Therefore, the self-learning robot arm requires a model for formulating a strategy generation system and an estimation system, such as hypothesis generation net (HGN) [22], [23]. The HGN is a novel neuron-based learning model which consists of two parts: a hypothesis model and an evaluation model. When these two models interact, the learning system can generate hypotheses to solve complex tasks based on historical experience. Therefore, a robot with HGN possesses a self-learning ability which enables it to learn intrinsic and extrinsic parameters by itself.

The cognition system of a virtual robot allows it to learn how to write Chinese calligraphy in a simulation environment [24], which can observe the process of the virtual robot in a virtual environment. In the process of practicing, humans can analyze the learning results and easily adjust the internal parameters. After these internal parameters are obtained, a delta robot [25] with an HGN learning model can be designed. Fig. 1(a) illustrates the testing environment of the writing system. Fig. 1(b) shows the mechanical length of the delta robot, such that the fixed base equilateral triangle side is 41.57cm, the under platform equilateral triangle side is 10cm, the upper arm length is 15.06cm, and the lower arm parallelogram length is 39.13cm. The control processing of the Chinese calligraphy-writing system comprises seven switches that can manipulate the operation program. Different operation modes are used to implement the Chinese calligraphy-writing robotic system and verify the performance of the written results.

This paper mainly aims to propose self-learning methods, such as an HGN application, a closed-loop structure using two webcams, stroke disassembly, coordinate transformation, and error correction methods. The advantages of these methods include enabling self-learning and an image-to-action translation that can autonomously generate five degrees of freedom (5-DOF) action instructions. In generally, the stroke trajectories of most calligraphy-writing systems are often described as two-dimensional coordinates [8], [9], or three-dimensional coordinates [1], [7], [10], [11]. These represent a relatively simple stroke trajectory planning method. In contrast, our robotic arm considers the posture angle of the brush, which is an important feature of the calligraphy-writing system. The posture of action instruction $\left[Z, \theta, \overline{\theta}\right]$ consists of three elements: the pressure (height), rotation, and tilt of the brush bundle. Through the posture of action instruction, the calligraphy-writing system can precisely control the posture and angle of the brush. The high-dimensional coordinate strategy of stroke writing is the current trend for future calligraphy writing systems.

The contributions of this paper focus on using five-dimensional action instructions D^v to execute stroke trajectories. The action instructions D^v consist of two-dimensional position coordinates T^v and three-dimensional posture coordinates Q^v. T^v denotes $[X, Y]$ and Q^v denotes $\left[Z, \theta, \overline{\theta}\right]$. Through 5-DOF action instructions, the stroke aesthetics of calligraphy can be better presented. Our experimental architecture utilizes two webcams for the input of writing results and feedback, respectively. By designing a closed-loop structure with a webcam, the system can obtain the writing results to calculate errors and perform self-correction. The input calligraphy image is translated from image to action translation, so that the robotic arm can imitate human handwriting and write various calligraphy characters flexibly. Through the self-learning of the proposed method, we can reduce the amount of reprogramming required when the human wants the robot to write different unlearned calligraphic characters. The aesthetic evaluation of the Chinese calligraphy method [17] consists of three parts: stroke coordination, balance, and distribution. Aesthetic evaluation [17] generates or corrects the next stroke trajectory based on the current reference image. It is worth mentioning that the stroke trajectory instructions in [17] only have two-dimensional coordinates, while our proposed system includes five-dimensional coordinates.

This paper is organized as follows: Section II presents the methods of writing processing and architecture of the

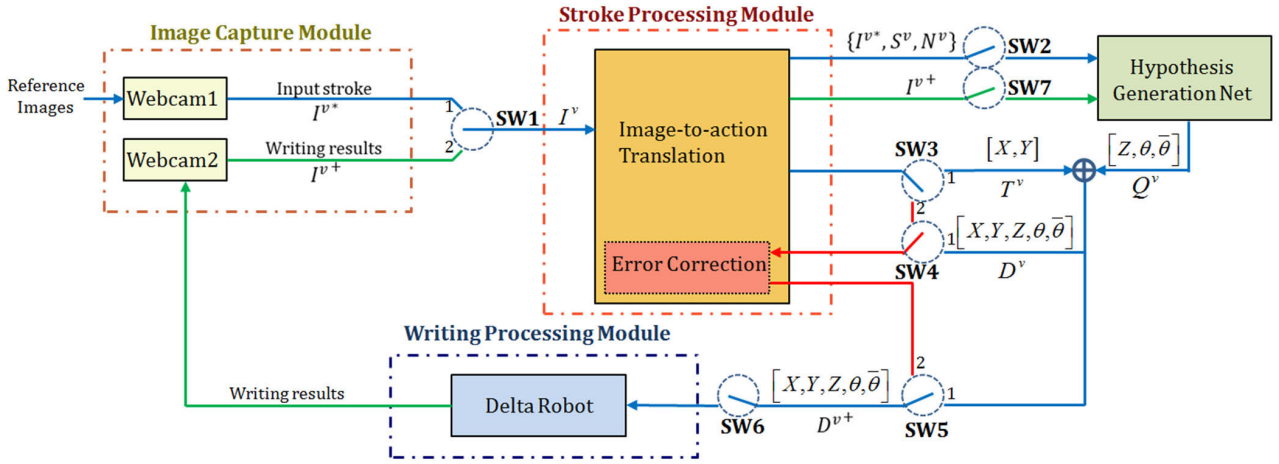


FIGURE 2. The architecture of the Chinese calligraphy-writing system.

Chinese calligraphy-writing system. Section III reveals the experimental results of the delta robot manipulator. The errors, data and aesthetic analysis of the writing results and a comparison of writing systems are provided in Section IV. The conclusions are drawn in Section V.

II. CHINESE CALLIGRAPHY-WRITING SYSTEM

To develop a Chinese calligraphy-writing system, we design a delta robot as a type of robot that effectively performs the proposed method. As shown in Fig. 2, the architecture of the Chinese calligraphy-writing system contains four modules, i.e., (a) the image capture module consisting of two webcams, (b) the stroke processing module mainly dealing with the processes that convert stroke images into action instructions, (c) the writing processing module with a delta robot as the main structure, and (d) the HGN as a learning model of the Chinese calligraphy-writing system.

A. IMAGE CAPTURE MODULE

This module involves pre-processing for capturing stroke images. It consists of two web cameras, one captures the input stroke images I^v, and the other captures the writing results I^{v+} of the delta robot. The output image format is a BMP file, and the region of interest (ROI) image size is W by H pixels. Some strokes comprise one word. We input one stroke and then the next stroke is written. First, this module converts the input image into a binarized image. Then, the binarized image of the stroke can be defined by mathematical equation Eq. (1).

The matrix of the binarized image of the vth image of the input stroke is defined as:

$$M^v = \mathrm{B}(I^v) = \begin{bmatrix} m^v_{0,0} & \cdots & m^v_{0,H-1} \\ \vdots & \ddots & \vdots \\ m^v_{W-1,0} & \cdots & m^v_{W-1,H-1} \end{bmatrix}_{W\times H} = \left[m^v_{i,j}\right]_{W\times H} \quad (1)$$

where $i = 0\ldots W\text{-}1$, $j = 0\ldots H\text{-}1$, $i, j \in \mathbb{N}$, $v, W, H \in \mathbb{N}^+$. The function $\mathrm{B}(\cdot)$ denotes the binarized translation, i and j denote the ith row and jth column of the pixel coordinates, and v denotes the index of the input images. $m^v_{i,j}$ denotes one of the elements of the matrix M^v. The value of $m^v_{i,j}$ is either 1 or 0. If each pixel exceeds the grayscale threshold, the matrix element is set to '1' ($m^v_{i,j} = 1$); otherwise it is '0'. Each '1' element in the matrix represents a stroke pixel in the image. Fig. 3(a) depicts an example of the "L" stroke of the character. The ROI image size is 10 pixels by 10 pixels. For example, Fig. 3(b) presents the matrix of Fig. 3(a).

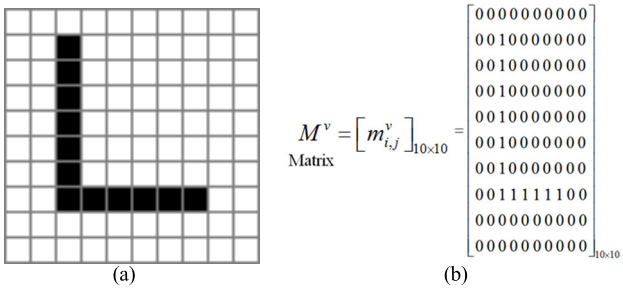


FIGURE 3. Example of the "L" stroke of the character. (a) The image size is 10 pixels x 10 pixels. (b) The matrix of a binarized image is shown in Fig. 3(a).

B. STROKE PROCESSING MODULE

The stroke processing module is an image-to-action translation process as shown in Fig. 4. The purpose of this module is to convert the input stroke images into action instructions. It includes five processing methods: (a) stroke skeleton extraction, (b) stroke disassembling, (c) skeleton coordinate sorting, (d) coordinate conversion, and (e) error correction. The details of the image-to-action translation process are presented in Fig. 4.

1) STROKE SKELETON EXTRACTION

Since calligraphy writing requires the planning of a stroke trajectory, the stroke skeleton must first be extracted from the input stroke image. We utilize the Zhang-Suen thinning algorithm [26] to extract the skeleton of human handwritten

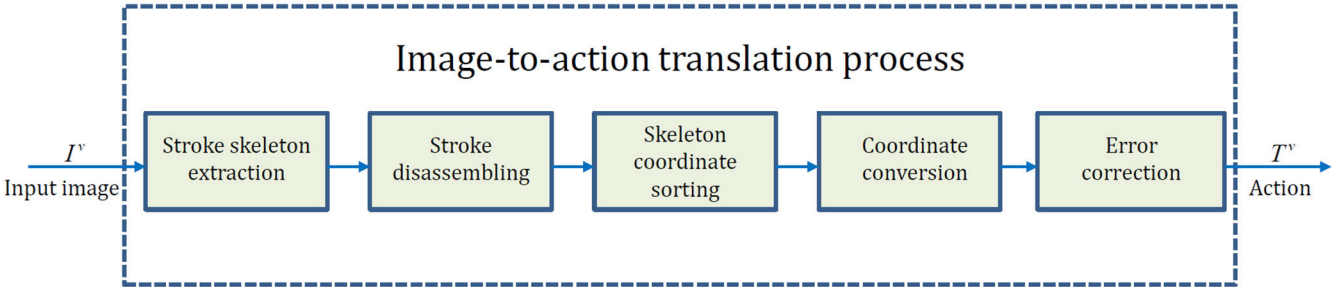


FIGURE 4. Image-to-action translation process.

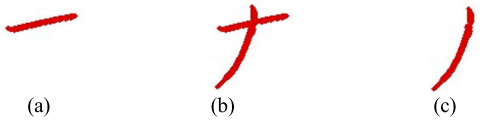


FIGURE 5. Input images of the Chinese calligraphy examples. (a) Previous stroke image. (b) Current stroke image. (c) New single-stroke image.

Chinese calligraphy strokes. The image of each skeleton can be defined as the following mathematical equation.

The matrix of a skeleton of the vth image of input strokes is defined as:

$$S^v = \mathrm{F}(M^v) = \begin{bmatrix} s^v_{0,0} & \cdots & s^v_{0,H-1} \\ \vdots & \ddots & \vdots \\ s^v_{W-1,0} & \cdots & s^v_{W-1,H-1} \end{bmatrix}_{W\times H} = \left[S^v_{i,j}\right]_{W\times H} \tag{2}$$

where $i = 0\ldots W\text{-}1$, $j = 0\ldots H\text{-}1$, $i, j \in \mathbb{N}$, $v, W, H \in \mathbb{N}^+$. The function $\mathrm{F}(\cdot)$ denotes the Zhang-Suen thinning algorithm.

2) STROKE DISASSEMBLING

A human writes the strokes one by one on the paper, as the input I^v of the stroke processing module. When the input image has overlapping strokes (as shown in Fig. 5(b)), the stroke disassembling algorithm can be used to extract a newly separated stroke (see Fig. 5(c)). The process of stroke extraction is described below.

The stroke image I^v can be converted into a stroke matrix M^v, and it can then be converted into a skeleton matrix S^v through the Zhang-Suen thinning algorithm. The elements of the matrix K^v are composed of skeleton coordinates from matrix S^v.

The skeleton matrix of the pixel coordinates of the vth matrix of the input skeleton is defined as:

$$K^v = \psi(S^v) = \begin{bmatrix} j^v_1 & i^v_1 \\ \vdots & \vdots \\ j^v_{p^v} & i^v_{p^v} \end{bmatrix}_{p^v\times 2} = \begin{bmatrix} k^v_1 \\ \vdots \\ k^v_{p^v} \end{bmatrix} \tag{3}$$

where $i, j \in \mathbb{N}, v, p^v \in \mathbb{N}^+$. The function $\psi(\cdot)$ denotes the row and column translation, while p^v denotes the order number of coordinates. The value of p^v depends on the length of the skeleton. $(j^v_{p^v}, i^v_{p^v})$ denotes the p^vth pixel coordinate of the vth skeleton matrix S^v and $j^v_{p^v}$ denotes the jth column of the p^vth pixel coordinate of the elements $(S^v_{i,j} = 1)$ of the vth matrix S^v, $i^v_{p^v}$ denotes the ith row of the p^vth pixel coordinate of the elements $(S^v_{i,j} = 1)$ of the vth matrix S^v, and $k^v_{p^v}$ denotes the p^vth row vector $\left[j^v_{p^v} i^v_{p^v}\right]$ of the vth matrix K^v.

To facilitate the description of the stroke decomposition process. The pseudo-code of the stroke disassembling algorithm is illustrated below:

The circular integral method [27] is described below. As mentioned in Eq. (3), the elements of the skeleton matrix K^v consist of the pixel coordinates obtained from the row and column index of the skeleton matrix S^v. However, we added the element of the radius into the vector $[j^v_{p^v} i^v_{p^v}]$ in the circle matrix C^v.

The circle matrix C^v of the vth image of input strokes is defined as:

$$C^v = \begin{bmatrix} C^v_1 \\ \vdots \\ C^v_{p^v} \end{bmatrix} = \begin{bmatrix} j^v_1 & i^v_1 & r^v_1 \\ \vdots & \vdots & \vdots \\ j^v_{p^v} & i^v_{p^v} & r^v_{p^v} \end{bmatrix}_{p^v\times 3} \tag{4}$$

where $i, j, r \in \mathbb{N}$, $v, p^v \in \mathbb{N}^+$. $C^v_{p^v}$ denotes the p^vth row vector $\left[j^v_{p^v} i^v_{p^v} r^v_{p^v}\right]$ of the vth matrix C^v and $r^v_{p^v}$ denotes the radius of a circle in which the center is at $(j^v_{p^v}, i^v_{p^v})$ as shown in Fig. 8(b). Fig. 8(a) is an image sample of a stroke. The function $\mathrm{O}(\cdot)$ is executed to draw a circle according to $C^v_{p^v}$ and converted to a W-by-H matrix. The function $\mathrm{A}(\cdot)$ executed an area calculation. For example, the count of the pixels of the image is as shown in Eq. (5).

$$\mathrm{A}(M^v) = \sum_{i=0}^{W-1}\sum_{j=0}^{H-1} m^v_{i,j} \tag{5}$$

Fig. 8(c) shows the intersection of the stroke image (Fig. 8(a)) and the circle image (Fig. 8(b)).

$$\frac{\mathrm{A}\left(M^v \cap O\left(C^v_{p^v}\right)\right)}{\mathrm{A}\left(C^v_{p^v}\right)} = U^v_{p^v} \tag{6}$$

If $r^v_{p^v}$ increases gradually, the area of the circle does not exceed the contour of the stroke, and so the value $U^v_{p^v}$ in Eq. (6) is 1 as shown in Fig. 9(a). If the area of the circle exceeds the contour of the stroke, the value $U^v_{p^v}$ in Eq. (6) is less than 1 as shown in Fig. 9(b). The threshold value is set to

Algorithm 1 The Process of Stroke Disassembling

Input: I^{v-1} (Previous stroke) and I^{v} (Current stroke) images.
Output: A new single stroke image O^{v}(as shown in Fig. 5(c)).
1: Step 1
2: Initial parameter $p^{v-1} = p^{v} = 1$, $G = 2$ (Pixel threshold)
3: Input images I^{v-1} and I^{v}.
4: Step 2
5: Execute the binarized translation from function B(I^{v-1}) and B(I^{v}) to binarized matrix M^{v-1} and M^{v}.
6: Step 3
7: Execute stroke skeleton extraction from F(M^{v-1}) and F(M^{v}) to skeleton matrix S^{v-1} and S^{v}.
8: Step 4
9: **for** $i = 0$ **to** W **do**
10: **for** $j = 0$ **to** H **do**
//Execute the matrix type to pixel coordinates translation $\psi(S^{v-1})$ and $\psi(S^{v})$ from skeleton matrix S^{v-1} and S^{v} to pixel coordinates matrix K^{v-1} and K^{v}.
11: **IF** $S^{v-1}_{i,j}$ equal '1'.
12: **Then** Store the row and column index $\left[j^{v-1}_{p^{v-1}} i^{v-1}_{p^{v-1}}\right]$ in vector $k^{v-1}_{p^{v-1}}$
13: $p^{v-1} = p^{v-1} + 1$
14: **IF** $S^{v}_{i,j}$ equal '1'.
15: **Then**Store the row and column index $\left[j^{v}_{p^{v}} i^{v}_{p^{v}}\right]$ in vector $k^{v}_{p^{v}}$
16: $p^{v} = p^{v} + 1$
17: **end for**
18: **end for**
19: Step 5
//Calculate the distance pixel by pixel then delete the pixel coordinates vectors of the previous skeleton. The result is shown in Fig. 6(c).
20: **for** $b = 1$ **to** p^{v-1} **do**
//Parameter p^{v-1} is the count of the vector ($k^{v-1}_{p^{v-1}}$)
21: **for** $q = 1$ **to** p^{v} **do**
//Parameter p^{v} is the count of the vector ($k^{v}_{p^{v}}$)
22: **IF** the two-norm distance $k^{v-1}_{b} - k^{v}_{q}$ less than G.
23: **Then** remove the vector k^{v}_{q} of K^{v}.
24: **end for**
25: **end for**
26: Step 6
27: Restore the stroke by using the circular integral method $R(K^{v})$. The result is shown in Fig. 7.
Output image $O^{v} = \mathrm{R}(K^{v})$.
28: Finally, we get the new single-stroke image as shown in Fig. 5(c).
29: END

0.99. When the value $U^{v}_{p^{v}}$ in Eq. (6) is equal to the threshold value, $r^{v}_{p^{v}}$ is the circle radius that matches the contour of the stroke and the circle radius $r^{v}_{p^{v}}$ corresponds to the contour of the stroke corresponding to each skeleton coordinates. The

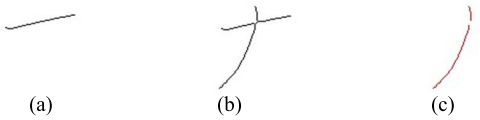


FIGURE 6. Skeleton results of the input images. (a) Previous skeleton image. (b) Current skeleton image. (c) Exclusive-OR result of the images (a) and (b).

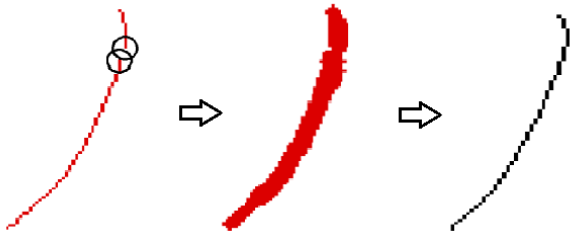


FIGURE 7. Stroke restoration for a new single skeleton.

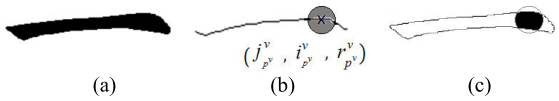


FIGURE 8. (a) The Image of the stroke. (b) A circle with radius $r^{v}_{p^{v}}$ centered at $(j^{v}_{p^{v}}, i^{v}_{p^{v}})$. (c) Intersection of the stroke image.

FIGURE 9. Threshold of circle radius. (a) The value of $U^{v}_{p^{v}}$ is 1. (b) The value of $U^{v}_{p^{v}}$ is less than 1.

method is used to update the radius of the circle matrix C^{v} from r^{v}_{1} to $r^{v}_{p^{v}}$. Therefore, we can obtain the radius $r^{v}_{p^{v}}$ of the circle in which the center is at $(j^{v}_{p^{v}}, i^{v}_{p^{v}})$ and meet the contour of the stroke. The $(j^{v}_{p^{v}}, i^{v}_{p^{v}})$ denotes the coordinates of the skeleton. Then each coordinate point of the skeleton can draw a circle according to the vector $C^{v}_{p^{v}}$ again. Finally, the skeleton can be restored to the original strokes by using the circular integral method.

3) SKELETON COORDINATE SORTING

After the skeleton of a single stroke had been obtained through the previous processing, skeleton coordinate sorting is used to complete the coordinate ordering of the stroke skeleton. Different strokes would have different stroke orders, so the coordinate order of the skeleton must conform to the stroke order of the strokes. First, we convert the skeleton image into a skeleton coordinate matrix and then use the data structure "Graph" to describe the correlation (Edge) of each point (Vertex). Each skeleton coordinate is a Vertex, and the angle corresponding to each coordinate is an Edge. Finally, a single-direction directed graph can be used to describe the coordinate ordering of the entire skeleton.

4) COORDINATE CONVERSION

In addition to the above-mentioned stroke path planning, another important process is the coordinate conversion. The purpose of coordinate conversion is to convert the

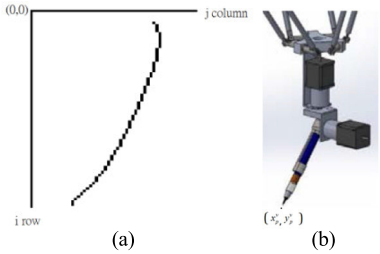


FIGURE 10. The (x_p^v, y_p^v) part of image-to-action translation. (a) A single skeleton of pixel coordinates. (b) Action instruction of Cartesian coordinates.

input stroke I^v or I^{v^+} from pixel coordinates to the actual coordinates T^v that the writing processing module can execute. The conversion method involves converting the length of the stroke image I^v or I^{v^+} captured by the webcam to the stroke length on the actual Cartesian coordinates plane. After the coordinate conversion, the position coordinates corresponding to the actual writing strokes can be obtained. The final coordinate matrix updates each coordinate through the scale value and offset value, and completes the conversion of the image I^v or I^{v^+} into Cartesian coordinates. The pixel coordinates are shown in Fig. 10(a). The origin of the pixel coordinates is in the upper left corner, and it differs from the of Cartesian coordinates. The pixel coordinates correspond to the matrix type, so the horizontal axis is the j column and the vertical axis is the i row. The Cartesian coordinates are shown in Fig. 10(b).

The function of the handwriting stroke skeleton of the vth matrix S^v is defined according to Eq. (3). However, the count of the skeleton coordinates p^v could be normalized into p; sampling normalization as shown in Eq. (7). Subscript $(2 \cdot \frac{p^v}{p})$ denotes the approximation value rounded to a positive integer. We utilized the Eq. (7) to normalize the count of each skeleton coordinate point.

$$\begin{bmatrix} j_1^v & i_1^v \\ j_2^v & i_2^v \\ \vdots & \vdots \\ j_p^v & i_p^v \end{bmatrix}_{p\times 2} = \begin{bmatrix} j_1^v & i_1^v \\ j^v_{(2\cdot\frac{p^v}{p})} & i^v_{(2\cdot\frac{p^v}{p})} \\ \vdots & \vdots \\ j_{p^v}^v & i_{p^v}^v \end{bmatrix}_{p^v\times 2} \tag{7}$$

where $i, j \in \mathbb{N}$, $v, p, p^v, (2 \cdot \frac{p^v}{p}) \in \mathbb{N}^+$.

The function of the handwriting stroke skeleton of the vth matrix S^v is defined as:

$$H^v = \psi(S^v) = \begin{bmatrix} j_1^v & i_1^v \\ \vdots & \vdots \\ j_p^v & i_p^v \end{bmatrix}_{p\times 2} \tag{8}$$

where $i, j \in \mathbb{N}$, $v, p \in \mathbb{N}^+$. (j_p^v, i_p^v) denotes the pixel coordinates of the vth skeleton.

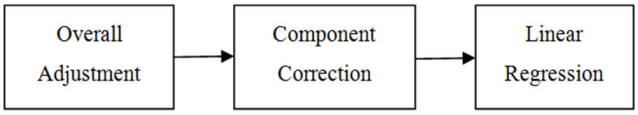


FIGURE 11. Three stages of the error correction process.

The matrix of the writing coordinates of the vth skeleton matrix of input strokes is defined as:

$$W^v = \begin{bmatrix} x_1^v & y_1^v \\ \vdots & \vdots \\ x_p^v & y_p^v \end{bmatrix}_{p\times 2} \tag{9}$$

where $x_p^v, y_p^v \in \mathbb{R}$, $v, p \in \mathbb{N}^+$. (x_p^v, y_p^v) denotes the actual Cartesian coordinates of the vth skeleton.

The function $\chi(\cdot)$ denotes the coordinate conversion from pixel coordinates to Cartesian coordinates.

$$W^v = \chi\left(H^v\right) \tag{10}$$

$$x_p^v = j_p^v \cdot \varphi_x + \tau_x \tag{11}$$

$$y_p^v = i_p^v \cdot \varphi_y + \tau_y \tag{12}$$

where φ_x, φ_y denotes the scale value and τ_x, τ_y denotes the offset value. The scale value is used to adjust the length of the stroke, and the offset value is used to adjust the position of the stroke.

5) ERROR CORRECTION

In the process of converting the image into action, the input stroke image I^v or I^{v^+} is converted into the action instruction T^v of the motor through an image processing series. Finally, the writing processing module writes the Chinese calligraphy according to the action instruction. The result I^+ may not be consistent with the target stroke I^{v^*}. At this time, these methods must be used to calculate and correct the error of the skeleton coordinates T^v, as described below.

There are three stages for error correction: 1. the overall adjustment, 2. the component correction, and 3. linear regression, as shown in Fig. 11.

The target matrix of the handwriting skeleton of the vth matrix of input strokes is defined as:

$$H^{v^*} = \chi\left(H^v\right) = \begin{bmatrix} x_1^{v^*} & y_1^{v^*} \\ \vdots & \vdots \\ x_p{}^{v^*} & y_p{}^{v^*} \end{bmatrix}_{p\times 2} \tag{13}$$

where $x_p^{v^*}, y_p^{v^*} \in \mathbb{R}$, $v, p \in \mathbb{N}^+$. $(x_p^{v^*}, y_p^{v^*})$ denotes the actual Cartesian coordinates of the vth target skeleton from the coordinate conversion of the vth matrix H^v.

The matrix of the writing result skeleton of the vth matrix of input strokes is defined as:

$$R^v = \begin{bmatrix} x_1^v & y_1^v \\ \vdots & \vdots \\ x_p^v & y_p^v \end{bmatrix}_{p\times 2} \tag{14}$$

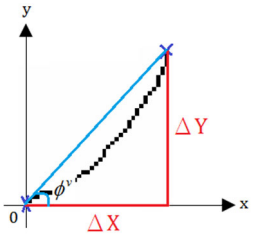


FIGURE 12. Illustration of the x-axis and y-axis components of a skeleton.

where $x_p^v, y_p^v \in \mathbb{R}$, $v, p \in \mathbb{N}^+$. (x_p^v, y_p^v) denote the actual Cartesian coordinates of a skeleton from the coordinate conversion of the vth writing result image.

The function of the error matrix of the vth time calculation is defined as:

$$E^v = \begin{bmatrix} x_1^v & y_1^v \\ \vdots & \vdots \\ x_p^v & y_p^v \end{bmatrix}_{p\times 2} \tag{15}$$

where $x_p^v, y_p^v \in \mathbb{R}$, $v, p \in \mathbb{N}^+$. (x_p^v, y_p^v) denote the Cartesian coordinates of the vth skeleton error.

a: OVERALL ADJUSTMENT

The learning rate L and the error matrix E^v are used to update and correct the coordinate matrix W^v.

$$E^v = R^v - H^{v^*} \tag{16}$$
$$x_p^v = x_p^v - x_p^{v^*} \tag{17}$$
$$y_p^v = y_p^v - y_p^{v^*} \tag{18}$$

Updating the next coordinate matrix for W^v gives

$$W^v = R^v - L \cdot E^v \tag{19}$$
$$x_p^v = x_p^v - L \cdot x_p^v \tag{20}$$
$$y_p^v = y_p^v - L \cdot y_p^v \tag{21}$$
$$Loss\left(x_i^v, y_i^v\right) = \sum_{i=1}^{p} L \cdot x_i^v + \sum_{i=1}^{p} L \cdot y_i^v \tag{22}$$

The learning rate L is defined as less than 1.

b: COMPONENT CORRECTION

The approximation algorithm is used to calculate the X axis and Y axis slope changes of the skeleton at the vth input strokes to obtain the angle ϕ^v, as shown in Fig. 12. Then, the error matrix E^v and the angle ϕ^v are used to take the cosine and sine components to update and correct the coordinate matrix ^{v} of the delta robot for the next writing.

$$\hat{x}_p^v = x_p^v - x_p^v \cdot \eta \cdot \cos\phi^v \tag{23}$$
$$x_p^v = y_p^v - y_p^v \cdot \eta \cdot \sin\phi^v \tag{24}$$
$$Loss\left(x_i^v, y_i^v\right) = \sum_{i=1}^{p} x_i^v \cdot \eta \cdot \cos\phi^v + \sum_{i=1}^{p} y_i^v \cdot \eta \cdot \sin\phi^v \tag{25}$$

where $(\hat{x}_p^v, x_p^v)$ denotes the Cartesian coordinates of the vth next writing matrix W^v. $\eta \cdot \cos\phi^v, \eta \cdot \sin\phi^v$ denotes the x-axis and y-axis components matrix and η denotes a constant of less than 1. The x-axis and y-axis component ($\eta \cdot \cos\phi^v$ and $\eta \cdot \sin\phi^v$) are both less than 1.

c: LINEAR REGRESSION

The linear regression equations can be calculated from the coordinates of each point of the vth target handwriting skeleton matrix H^{v^*} and the coordinates of each point of the delta robot vth writing result skeleton matrix R^v, respectively. The vth angle δ^v, the vth scaling α^v, and the vth offset value β^v can thus be obtained. Finally, these values are utilized to update and correct the coordinate matrix that the delta robot could write next time W^v.

$$\hat{\hat{x}}_p^v = q_x^v + (\hat{x}_p^v - q_x^v) \cdot \cos\delta^v + (y_p^v - q_y^v) \cdot \sin\delta^v \tag{26}$$
$$x_p^v = q_y^v - (x_p^v - q_x^v) \cdot \sin\delta^v + (y_p^v - q_y^v) \cdot \cos\delta^v \tag{27}$$

where (q_x^v, q_y^v) denotes the intersection of two lines and δ^v denotes the angle between two lines.

$$\hat{\hat{x}}_p^v = \hat{\hat{x}}_p^v \cdot \alpha_x^v \tag{28}$$
$$y_p^v = y_p^v \cdot \alpha_y^v \tag{29}$$

where (α_x^v, α_y^v) denotes the scale values.

$$x_p^{v^+} = \hat{\hat{x}}_p^v + \beta_x^v \tag{30}$$
$$y_p^{v^+} = y_p^v + \beta_y^v \tag{31}$$

where (β_x^v, β_y^v) denotes the offset values. $(x_p^{v^+}, y_p^{v^+})$ denotes the Cartesian coordinates of the next vth writing matrix $W_p^{v^+}$.

C. WRITING PROCESSING MODULE

The writing processing module consists of two parts, i.e., an inverse kinematic model [28] of the delta robot and a motor control system. This module can make the writing brush move with 5-DOF. After running the stroke processing program, the stroke image I^v or I^{v^+} input by the image capture module can be converted into a complete set of writing 5-DOF action instructions D^v or D^{v+} (i.e., $\left[X, Y, Z, \theta, \overline{\theta}\right]$) that can be executed by the writing processing module. Then, the inverse kinematics of the delta robot can be used to obtain the output rotation angle required by each motor. The motor completes the movement of the under platform and the change of the posture of the brush according to the input rotation angle. Finally, the delta robot completes the writing action of Chinese calligraphy. The summary of the action instructions of the stroke1 (dian) is shown in Table 1. There are 100 samples on each stroke, i.e., 100 action instructions for a stroke. The unit of distance coordinates is in centimeters (cm). The unit of the angle is in degrees.

The center coordinate of the under platform (x, y) which is shown in Fig. 13. can be extended to the coordinates of the brush nib (X, Y). Eqs. (32) and (33) show the center coordinates of the under platform (x, y). Fig. 13 illustrates the

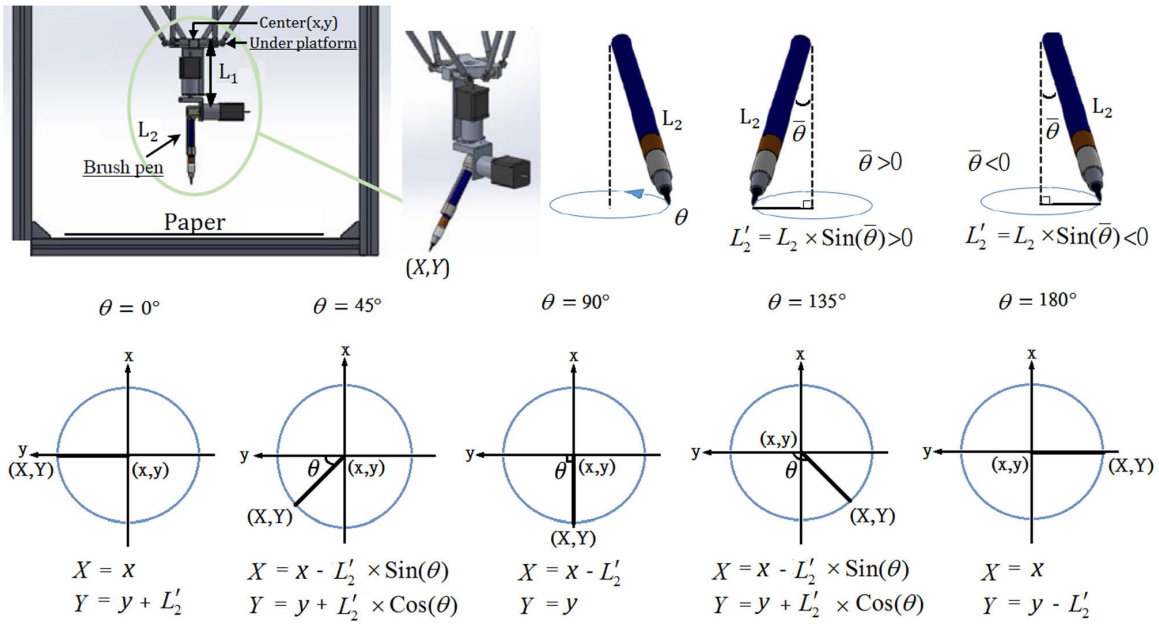


FIGURE 13. Brush nib coordinates of the relationship between (X, Y) and (x, y).

relationship between (X, Y) and (x, y). The Z-axis height is from the paper to the under platform. Then the three rotation angles of the motors are obtained from the inverse kinematics formula of the delta robot [15], [18]. We also define the posture range of the brush. The range of the rotation angle θ is from 0 to 180 degrees and the range of the tilt angle $\bar{\theta}$ is from -3 to 3 degrees. The centeral coordinates of the under platform (x, y) are calculated using the following equations:

$$x = X + L_2' \times \mathrm{Sin}(\theta) \tag{32}$$
$$y = Y - L_2' \times \mathrm{Cos}(\theta) \tag{33}$$

D. SYSTEM CONTROL PROCESSING

The system control processing consists of seven switches as shown in Fig. 2. There are seven switches to control the system processing, i.e., an input image selection switch (SW1), an HGN input switch (SW2 and SW7), a coordinate feedback switch (SW3 and SW4), an instruction switch (SW5), and a write-executing switch (SW6). Switch SW1 includes the human writing input switch (SW1-1) and writing result feedback switch (SW1-2); SW2 is a switch that can input the target image, the target skeleton, and the target stroke number; SW3 includes the instruction generation switch (SW3-1) and error calculation switch (SW3-2); SW4 includes coordinates saving switch (SW4-1) and error calculation switch (SW4-2); SW5 includes the action instruction switch (SW5-1) and error calculation switch (SW5-2); SW6 is a write-executing switch; and SW7 is a writing result feedback switch. According to the different training modes, various switches can be used to adjust the operating procedure and finally complete the required target action. Fig. 14 illustrates these functional switches in two modes. One is an HGN training mode and the other is a Chinese calligraphy writing mode. In Fig. 14, the horizontal axis shows the different stages and the vertical axis shows the different switches. There are three types of stages in each mode, i.e., the input strokes stage, first attempt stage, and robotic writing stage.

TABLE 1. Example of the 5-dof action instructions of the stroke1 "dian".

Order	X axis	Y axis	Z axis	θ rotation	$\bar{\theta}$ tilt
1	1.2593	-0.1232	17.80103	175.8733	-2.3231
2	1.2593	-0.1232	17.74670	168.6974	-2.0543
3	1.2593	-0.1232	17.74907	169.8160	-2.0278
4	1.2593	-0.1232	17.74840	169.7766	-2.0243
5	1.2593	-0.1232	17.74848	169.7887	-2.0235
…	…	…	…	…	…
96	1.9244	-0.6891	17.60033	172.8125	-0.8595
97	1.9251	-0.7120	17.60033	172.8125	-0.8595
98	1.9253	-0.7196	17.60033	172.8125	-0.8595
99	1.9255	-0.7297	17.60033	172.8125	-0.8595
100	1.9257	-0.7356	17.60033	172.8125	-0.8595

1) HGN TRAINING MODE

The main purpose of using the HGN is to help the robot generate appropriate 5-DOF coordinates for brush posture learning, which is a non-trivial task for 3-DOF methods. In the HGN training mode, 5-DOF action instructions are divided into the 2-dimensional coordinates T^v, which are related to the pixel in the image and the complex 3-DOF

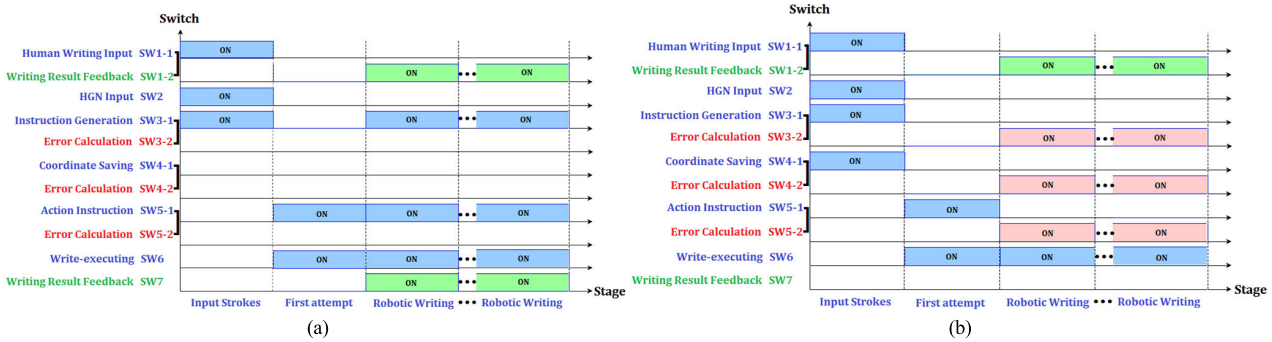


FIGURE 14. System control switches for two training modes. (a) HGN training mode. (b) Chinese calligraphy writing mode.

writing posture Q^v. This mode comprises three stages as shown in Fig. 14(a). Described in detail as follows.

Stage 1. Input strokes

There are three switches involved in this stage, namely SW1-1 human writing input switch, SW2 HGN input switch, and SW3-1 action instruction generation switch. The operating procedure in stage 1 is described below. The handwriting stroke image I^v is obtained from the image capture module through the SW1-1 input to the stroke processing module. After passing the stroke processing module, the target image I^{v^*}, skeleton coordinate S^v, and stroke number N^v are output through SW2 to the HGN. The stroke processing module outputs the actual skeleton position coordinates T^v (i.e., $[X, Y]$) and combines the posture instructions Q^v (i.e., $[Z, \theta, \overline{\theta}]$) of the brush from the HGN, thus resulting in complete action instructions D^v (i.e., $[X, Y, Z, \theta, \overline{\theta}]$).

Stage 2. First attempt

There are two switches involved in this stage, SW5-1 action instruction switch and SW6 write-executing switch. The operating procedure in stage 2 is described below. The complete action instructions are input to the writing processing module through SW5-1 and SW6, and the delta robot completes the calligraphy writing for the first attempt through the writing processing module.

Stage 3. Robotic writing

There are five switches involved in this stage, SW1-2 writing result feedback switch, SW7 writing result feedback switch, SW3-1 instruction generation switch, SW5-1 action instruction switch, and SW6 write-executing switch. The operating procedure in stage 3 is described below.

The image capture module sends the writing result image I^{v^+} back to the stroke processing module through SW1-2, and then sends it back to the HGN through SW7. The stroke processing module continues to output the actual coordinates of the skeleton T^v through SW3-1 and combines with new posture instructions Q^v for the brush which are updated from the HGN, and completes the next action instruction D^v. The complete action instructions are input to the writing processing module through SW5-1 and SW6, and finally, the delta robot completes calligraphy writing through the writing processing module. Stage 3 is repeated until training is finished and the error correction is then executed in HGN [22].

2) CHINESE CALLIGRAPHY WRITING MODE

The purpose of Chinese calligraphy writing mode is to correct the skeleton coordinates and reduce environmental errors such as mechanism errors and coordinate conversion errors. This mode consists of three stages as shown in Fig. 14(b). Described in detail as follows.

Stage 1. Input Strokes

There are four switches involved in this stage, namely SW1-1 human writing input switch, SW2 HGN input switch, SW3-1 action instruction generation switch, and SW4-1 action coordinate saving switch. The operating procedure in stage 1 is described below. The pre-processing is the same as stage 1 of the HGN training mode. Then the stroke processing module saves the action instructions and skeleton coordinates of the handwriting through SW4-1. Finally, the actual Cartesian coordinates of the stroke skeleton and the action instructions for each stroke of the entire Chinese character are stored.

Stage 2. First attempt

There are two switches involved in this stage, the SW5-1 action instruction switch and SW6 write-executing switch. The operating procedure in stage 2 is described below. The complete action instructions are input to the writing processing module through SW5-1 and SW6, and the delta robot completes the calligraphy writing for the first attempt through the writing processing module.

Stage 3. Robotic writing

There are five switches involved in this stage, the SW1-2 writing result feedback switch, SW3-2 skeleton error calculation switch, SW4-2 skeleton error calculation switch, SW5-2 action instruction switch, and SW6 write-executing switch. The operating procedure in stage 3 is described below. The image capture module sends the writing result image I^{v^+} back to the stroke processing module through SW1-2. Then, the stroke processing modules, SW3-2 and SW4-2, complete the error calculation and update the actual coordinates T^v of each stroke skeleton. The action instruction D^{v^+} of the next writing stroke is then completed. Finally, the delta robot completes the calligraphy writing of each stroke of the entire

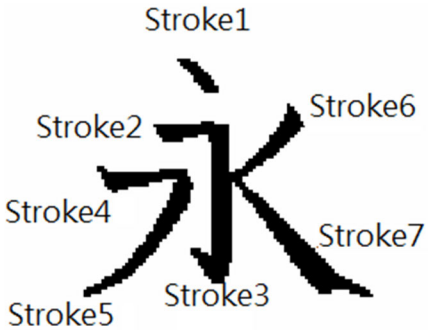


FIGURE 15. Seven fundamental strokes of the Chinese character "永".

Chinese character through SW5-2 and SW6 in this robotic writing stage. To sum up, the posture instructions of the brush of the robotic arm for the first attempt comes from HGN. Afterwards, the writing action instructions can be modified by the writing system itself, repeating stage 3 until the training is finished.

III. EXPERIMENTS AND RESULTS

The Chinese character "永" includes seven fundamental strokes, i.e., stroke1 (dian), stroke2 (heng), stroke3 (shu-gou), stroke4 (ti), stroke5 (wan), stroke6 (pie), and stroke7 (na) as shown in Fig. 15. We utilized this character to verify the performance of the Chinese calligraphy-writing system. There are two critical approaches to writing Chinese calligraphy: Stroke trajectory learning and brush posture learning. Chinese calligraphy writing can only be completed correctly through the correct coordinates of the skeleton. The thickness and angle of the stroke determine the beauty of the word. Therefore, the training process of the Chinese calligraphy-writing system is divided into two modes. One is an HGN training mode and the other is a Chinese calligraphy writing mode.

The purpose of the HGN training mode is to train the writer net [22] of the HGN for every single stroke. In the process of writing calligraphy, there is not only one unique writing posture corresponding to each stroke. At this point, we need to utilize HGN as a posture strategy maker, so that the delta robot can learn the appropriate 3-DOF writing posture through accumulated experience by itself. The purpose of the HGN pre-training process is to train the writing posture of the fundamental strokes. Each fundamental stroke is practiced 200 times. During practice, HGN learns the patterns of calligraphy-writing posture through the learning mechanism of hypothetical generation models in cognitive psychology. HGN generates potential hypotheses through a Hypothesis Net and then evaluates the feasibility of the hypothesis through an Estimator Net to complete the thinking process, which is similar to the psychological human thinking process. Finally, the writer net of the HGN can learn to generate the writing posture of the fundamental strokes.

In the Chinese calligraphy writing mode, the primary purpose is to correct skeleton coordinates and reduce environmental errors. This mode can verify the learning effect

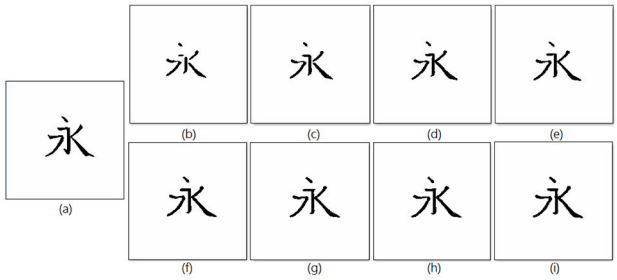


FIGURE 16. Writing results of the eight writing practices of Chinese character "永". (a) Target image. (b)-(i) Writing results from 1st to 8th.

of the writing system. First, the writing system captures the input, i.e., the stroke order of Chinese handwriting, via a webcam. The process of image-to-action translation includes extracting stroke skeletons, disassembling strokes, sorting skeleton coordinates, recognizing strokes, converting coordinates, receiving HGN instructions, and obtaining the angle instructions of motors

through inverse kinematic. The output 5-DOF action instructions include the skeleton coordinates and the posture instructions of a brush. The coordinate matrix of the skeleton is obtained by image processing, and the posture instructions of a brush are obtained through the HGN learning model. The delta robot uses action instructions to complete the Chinese calligraphy writing. Then, the other webcam provides feedback to the writing system for error calculations and coordinate corrections. The writing results finally approach the given sample after self-modifications of the proposed system. After several cycles and modifications, the output results that approximate the target sample can finally be obtained. Figs. 16-19 show the writing results and error rates of the characters "永" and "水" using the self-made delta robot.

IV. ANALYSIS AND DISCUSSION

A. ERROR ANALYSIS

The error sources of this experiment mainly include the mechanical errors of the hardware structure, the visual image difference generated by the webcam, the image size resolution, the coordinate conversion, and the deformation error of the brush bundle.

1) MECHANICAL ERRORS OF THE DELTA ROBOT

The mechanics of the delta robot cause these errors. The mechanical errors in hardware architecture include computational errors from dimensional measurements to motion translation.

2) IMAGE CONVERSION ERROR

The image-to-action translation causes these errors. The difference in the images produced by the webcam is due to the relationship between the external light intensity and the angle of the external light; the images captured at different times were different. Multiple averaged results must be used

TABLE 2. Error rates of strokes(%) of the character "永".

Practice	Stroke 1	Stroke 2	Stroke 3	Stroke 4	Stroke 5	Stroke 6	Stroke 7
1	0.2731	0.3788	1.1184	0.3619	0.9459	0.5200	1.1678
2	0.2881	0.3938	1.2060	0.2427	0.9890	0.6158	0.9930
3	0.3006	0.3507	0.9911	0.1762	0.8633	0.4630	0.6652
4	0.2430	0.3081	0.7887	0.1655	0.5990	0.2678	0.4857
5	0.1630	0.2695	0.6069	0.1523	0.4424	0.1736	0.2532
6	0.1305	0.2701	0.5275	0.1668	0.3210	0.1552	0.2616
7	0.1180	0.2134	0.4293	0.1946	0.1912	0.1446	0.2445
8	0.1180	0.2280	0.3109	0.1498	0.1699	0.1158	0.2710

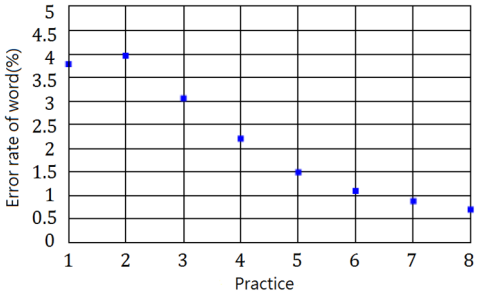


FIGURE 17. Writing results and error rates of the eight writing practices of the character "永".

to reduce this difference. Currently, the size of the ROI image is 200 pixels by 200 pixels. This resolution is difficult to display for small stroke changes and it could affect the results of stroke correction. The image-to-action translation requires the conversion of pixel coordinates to Cartesian coordinates. Therefore, the error ratio of the projection transformation to the actual size measurement would affect the final writing results.

3) DEFORMATION ERROR OF A BRUSH BUNDLE

The deformation of the brush bundle would be changed after writing a character several times. This would affect the difference in stroke thickness and length of the writing result.

B. DATA ANALYSIS

Table 2 and Table 3 show the error correction results for the delta robot using a learning rate of 0.3 during each stroke training process. The calculation method uses the mean absolute percentage error (MAPE) between the target image and the writing result images. The error rate can converge gradually after eight practice sessions.

1) ERROR RATE OF EACH STROKE

Fig. 20 and Fig. 21 show the writing results of the Chinese characters "永" and "水", the MAPE for each stroke, and the relationship between the error percentage and the writing

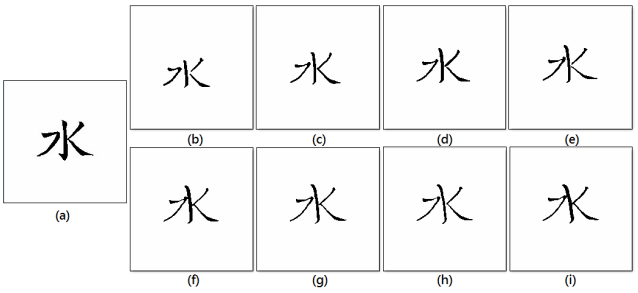


FIGURE 18. Writing results of the eight writing practices of Chinese character "水". (a) Target image. (b)-(i) Writing results from 1st to 8th.

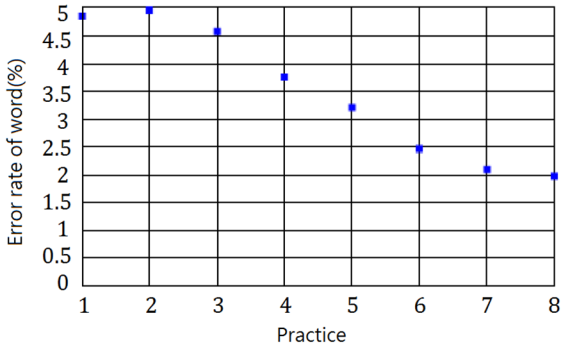


FIGURE 19. Writing results and error rates of the eight writing practices of the character "水".

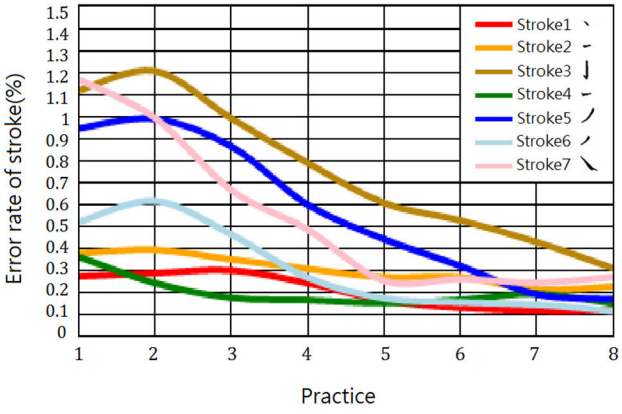


FIGURE 20. Each stroke error rate of the Chinese character "永".

practices, respectively. Some error rates of strokes increase gradually during the first few writing practices. This is because of the error correction, the length and position of the stroke are both adjusted at the same time. When the stroke length is increased but the position is wrong, the stroke error rate may increase slightly.

2) ERROR RATE PER WORD

Table 4 shows the comparison of the writing results of the Chinese characters "永" and "水", and the MAPE with the writing practices of the characters "永" and "水", respectively.

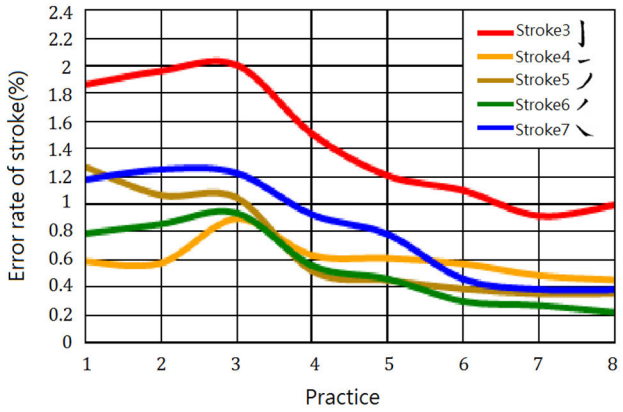


FIGURE 21. Each stroke error rate of the Chinese character "水".

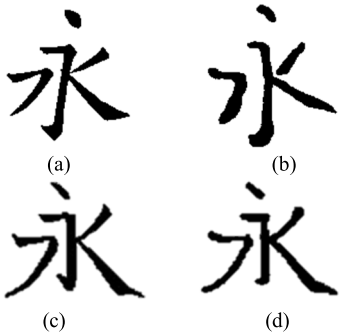


FIGURE 22. Writing result of the Chinese character "永" are shown in (a) as the writing target in [17], (b) as the writing result in [17], (c) as the writing target in the proposed method, and (d) as the writing result in the proposed method.

TABLE 3. Error rates of strokes(%) of the character "水".

Practice	Stroke3	Stroke4	Stroke5	Stroke6	Stroke7
1	1.8651	0.5887	1.2701	0.7897	1.1804
2	1.9643	0.5762	1.0665	0.8674	1.2529
3	2.0076	0.8956	1.0489	0.9376	1.2268
4	1.5088	0.6320	0.5192	0.5571	0.9260
5	1.2071	0.6109	0.4526	0.4622	0.7836
6	1.1007	0.5716	0.3895	0.3001	0.4605
7	0.9174	0.4863	0.3560	0.2694	0.3862
8	0.9958	0.4542	0.3590	0.2206	0.3820

C. AESTHETICS ANALYSIS

In addition to the similarity calculation with human samples, we also performed an aesthetic analysis of the calligraphy writing results. Chinese calligraphy writing can be aesthetically evaluated by three features [17] and described in detail as follows.

The overall aesthetic analysis results of Chinese calligraphy can be evaluated by the weighted sum of the three indexes (Eq. (34)).

$$I_{\mathrm{I}} = k_c I_c + k_b I_b + k_d I_d \tag{34}$$

where I_c denotes the coordination index (Eq. (35)), the aesthetics of any writing result is affected by the relative size of each stroke relative to the others. I_b denotes the balance index (Eq. (36)), the balance of the entire Chinese character is achieved through the different directions and different degrees of inclination of different strokes to achieve aesthetic purposes. I_d denotes the distribution index (Eq. (37)), the relative distribution of the strokes of the entire Chinese character is aesthetically determined by its center of mass and focus point of overall distribution. k_c, k_b, and k_d denote the weighted coefficients.

$$I_c = e^{-\sum_{i=2}^{\mathrm{N}} \left| \frac{l_i}{l_{i-1}} - \frac{l_i^*}{l_{i-1}^*} \right|} \tag{35}$$

$$I_b = e^{-\sum_{i=1}^{\mathrm{N}} |s_i^* - s_i|} \tag{36}$$

$$I_d = e^{-\left| \frac{\mu - \mu^*}{\mu^*} \right|} \tag{37}$$

The coordination index I_c in [17] is used to calculate the relative size ratio of each stroke to other strokes. In Chinese calligraphy characters, the relative length of each stroke affects the beauty of a word and even its meaning. For example, two Chinese characters with completely different meanings "±" and "±" differ only in the length of the strokes in the image. The balance index I_b is used to estimate the writing angle of a stroke in a Chinese character. A stroke that is too slanted or too flat will affect the beauty of the overall calligraphy. The distribution index I_d estimates the relative positions of individual strokes in a calligraphic character. Stroke positions that are too close or too far will make the calligraphy look introverted or indolent. If the reader is interested in the implementation details of aesthetic evaluation, the method in [17] can be referred to.

1) EVALUATION INDEX

There are some problems that affect fairness in the Eq. (35). In [17], an image of a stroke is considered as a unidirectional picture, whereas an image should be bidirectional. In the

TABLE 4. Error rate per word (%).

Practice	Character "永"	Character "水"
1	3.7867	4.8610
2	3.9639	4.9618
3	3.0597	4.5847
4	2.2170	3.7633
5	1.5029	3.2106
6	1.1047	2.4693
7	0.8889	2.1077
8	0.7090	1.9888

TABLE 5. Aesthetics evaluation of Chinese calligraphy-writing "永".

Method	I_c^*	I_b^*	I_d	I_I^*	$MAPE$
The method in [17]	0.8697	0.7972	0.968	0.8604	8.2508
The proposed method	0.7982	0.9360	0.9568	0.8851	0.7090

process of calligraphy writing, the strokes are unidirectional and orderly. However, in terms of judging the beauty of calligraphy writing from images, it is bidirectional and disordered. That is to say, no matter how the stroke order is arranged, it will be the same resulting image. We modify the unidirectional estimate of Eq. (35) to a bidirectional estimate, and then obtain

$$I_c' = e^{-\frac{1}{2}\left(\sum_{i=2}^{N}\left|\frac{l_i}{l_{i-1}}-\frac{l_i^*}{l_{i-1}^*}\right|+\sum_{i=2}^{N}\left|\frac{l_{i-1}}{l_i}-\frac{l_{i-1}^*}{l_i^*}\right|\right)} \tag{38}$$

With the revised bidirectional estimation Eq. (38), we can fairly evaluate the relative proportions of different strokes in Chinese characters. In addition to the directional problem of the image, Eqs. (36) and (38) also have a problem that needs to be improved. Since the evaluation methods of Eqs. (36) and (38) are calculated by summing the errors of all strokes, the scores of characters with more strokes will be lower. We need to modify the summation operations in Eqs. (36) - (38) to average operations, and obtain

$$I_c^* = e^{-\frac{1}{2(N-1)}\left(\sum_{i=2}^{N}\left|\frac{l_i}{l_{i-1}}-\frac{l_i^*}{l_{i-1}^*}\right|+\sum_{i=2}^{N}\left|\frac{l_{i-1}}{l_i}-\frac{l_{i-1}^*}{l_i^*}\right|\right)} \tag{39}$$

$$I_b^* = e^{-(\frac{1}{N})\sum_{i=1}^{N}|s_i^*-s_i|} \tag{40}$$

After obtaining the evaluation methods above Eqs. (37), (39) and (40), we comprehensively estimate the aesthetics of calligraphy characters. We modify Eq. (34) as

$$I_I^* = k_c I_c^* + k_b I_b^* + k_d I_d \tag{41}$$

and then fairly evaluate the aesthetic indices of the proposed method and the writing method in [17].

2) EVALUATION RESULTS

The weighted coefficients are set the same as in [17], i.e., k_c =0.4, k_b =0.4, and k_d =0.2. The results of the aesthetics evaluation are presented in Table 5.

In Table 5, there may be errors between the aesthetic score and personal perception. A slight difference in the score does not mean that its writing result is poor. The indicators of aesthetic analysis are as follows:

(1) Coordination Index: Scored according to the length of the strokes of the target and the result. Higher scores indicate better coordination.

(2) Balance Index: Scored according to the rotation angle of the target and result strokes. Higher scores indicate better balance.

(3) Distribution index: Scoring is based on the distribution of target and result strokes. Higher scores indicate better distribution.

(4) MAPE: Calculate the error percentage based on the strokes of the writing target and the writing result. The larger the ratio indicates the error is bigger.

In summary, the proposed method significantly improves the balance index I_b^* and performs better performance in the overall aesthetic evaluation I_I^*. Therefore, our calligraphy writing method not only has a small error compared to human writing but also has a satisfactory result on aesthetic evaluation. Note that we did not use the indicators of aesthetic analysis as a reference input for designing the loss function, but the aesthetic score of the proposed method is higher than that of [17] which considers these aesthetic indicators.

D. DISCUSSION AND COMPARISON

The experimental results allowed us to analyze and compare the differences, advantages, and disadvantages of calligraphy-writing learning methods.

1) DISCUSSION

The Chinese characters "永" and "水" are composed of similar strokes, but the practice results of each stroke are different. Strokes with significant errors must be adjusted by error correction methods. The current practice does not include all strokes of Chinese characters, and more complex strokes are not easy to disassemble and restore. Thus, the results of the writing would be affected. The stroke order sorting conditions of other strokes must be continuously trained and improved in order to adapt to the writing of all strokes. In addition, the proposed method has some limitations. First, initial parameters are not calculated automatically, making the parameters in HGN very difficult to tune. For example, if the learning rates in HGN are too large, the learning process of calligraphy will be difficult to converge. However, the learning process will be too slow if the learning rates are too small. Therefore, the initial setting of the HGN's learning rates depends on the human experience. Second, the input image captured by a camera has to be sufficiently stable due to no use of image enhancement algorithms.

2) COMPARISON

The Chinese calligraphy-writing system is compared with other writing systems as shown in Table 6. In general writing systems, the stroke trajectories are described as the vector $[X, Y, Z]$ in three dimensions. The position vector $[X, Y]$ is the 2-D skeleton trajectory, and $[Z]$ shows the pressure (height) level in thick or thin strokes. This is a relatively simple stroke trajectory planning method. In contrast, our writing system considers a 5-DOF action instruction as the vector $[X, Y, Z, \theta, \overline{\theta}]$. The angle vector $[\theta, \overline{\theta}]$ represents the rotation and tilt of the brush which requires precise control.

State-of-the-art calligraphy manipulator control techniques require the generation of mechanical coordinates that can write Chinese calligraphy; the proposed method generates complex motion instructions to control the manipulator to perform calligraphy writing tasks. Therefore,

TABLE 6. Summary of comparisons with other writing systems.

Comparison of the Chinese calligraphy writing systems	Mark	Self-learning ability	Writing stability	Writing flexibility	Dimension of the instruction
Method 1: Extracting the trajectory from character [7]	A	No	High	High	3
Method 2: Robotic Chinese calligraphy writing framework [8]	B	No	High	High	2
Method 3: Using corner detection [9]	C	No	Low	High	2
Method 4: Using generative adversarial net [1]	D	Yes	Low	High	3
Method 5: Modeling of the ancient style Chinese character [10]	E	No	High	Low	3
Method 6: Using a touch screen [11]	F	No	High	Low	3
Method 7: The proposed system	G	Yes	Low	High	5

TABLE 7. Dominance analysis of the writing systems.

Aspect	Discussion	Dominance analysis
1	Self-learning ability(**L**)	$\mathbf{L}(D)\,,\mathbf{L}(G) \tilde{\mathrm{N}} \mathbf{L}(A),\mathbf{L}(B),\mathbf{L}(C),\mathbf{L}(E),\mathbf{L}(F)$
2	Writing stability(**S**)	$\mathbf{S}(A)\,,\mathbf{S}(B)\,,\mathbf{S}(E)\,,\mathbf{S}(F) \tilde{\mathrm{N}} \mathbf{S}(C),\mathbf{S}(D),\mathbf{S}(G)$
3	Writing flexibility(**X**)	$\mathbf{X}(A)\,,\mathbf{X}(B)\,,\mathbf{X}(C)\,,\mathbf{X}(D)\,,\mathbf{X}(G) \tilde{\mathrm{N}} \mathbf{X}(E),\mathbf{X}(F)$
4	Dimension of the instruction (**M**)	$\mathbf{M}(G) \tilde{\mathrm{N}} \mathbf{M}(A),\mathbf{M}(B),\mathbf{M}(C),\mathbf{M}(D),\mathbf{M}(E),\mathbf{M}(F)$
	Conclusion	$\mathrm{G}\{3\}+ \tilde{\mathrm{N}} \mathrm{A}\{2\},\mathrm{B}\{2\},\mathrm{D}\{2\} \tilde{\mathrm{N}} \mathrm{C}\{1\},\mathrm{E}\{1\},\mathrm{F}\{1\}$

we utilize mechanical coordinate generation methods (e.g., [1], [7], [8], [9], [10], [11]) to compare the performance in the experimental results, as shown in Table 6. These methods are described below.

Method 1: Extracting the trajectory of the writing brush from the thinned-center-line of the stroke [7].

Method 2: Matching and decomposing strokes of character automatically to the corresponding robot trajectories [8].

Method 3: Using corner detection to decompose characters and matching the decomposed strokes to robotic writing trajectories [9].

Method 4: Using a generative adversarial net (GAN) based calligraphic robotic framework to produce trajectories [1].

Method 5: Modeling of ancient-style Chinese characters by B-splines for robotic calligraphy writing [10].

Method 6: Using a capacitive touch screen to obtain touch point positions, strokes, width, writing speed, acceleration and other characteristics [11].

These methods are used for discussion in Table 6. This table compares four aspects: (a) Self-learning ability, (b) Writing stability, (c) Writing flexibility, and (d) Dimensions of the instruction. They are defined as follows.

(a) Self-learning ability

We can confirm whether the writing system has self-learning ability through conditions such as writing result feedback, error calculation, and coordinate correction. Our system comprises two webcams for the input of writing results and feedback, respectively. Through this closed-loop structure, the system can calculate errors and perform self-correction. In addition, the use of these webcams is inexpensive relative to using other expensive perception sensors.

(b) Writing stability

The stability of writing is defined by examining calligraphic writing results for stroke fluency, proportional font, and word correctness. If the strokes are jittery, not smooth enough, have a wrong aspect ratio, or are misspelled, then the system is considered unstable. Our system utilizes an image-to-action process for calligraphic writing. As mentioned in the previous section, some errors may occur during the conversion process, resulting in unstable output results. Other systems use font databases for calligraphic writing for more stable results.

(c) Writing flexibility

Writing flexibility implies that the robotic arm can write in a variety of calligraphic fonts. Writing different fonts requires generation of different stroke trajectories. If the writing system cannot generate new stroke trajectories, the program needs to be modified to plan new stroke trajectories or add new font trajectories to the database. Through self-learning, our writing system can avoid program modification and shorten stroke modeling times, making writing more flexible.

(d) Dimensions of the instruction

Practical operations in real environments must consider factors such as kinematic transformations, noise, and reliability, so the writing results of real robotic arms are used to check the robustness and fault tolerance of the writing system. Therefore, high writing dimensions are needed to improve the writing results. In contrast, our robotic arm considers the posture angle of the brush, which is a unique feature of our writing system. The experimental results of the robotic arm demonstrate the effectiveness and feasibility of our proposed Chinese calligraphy-writing system.

The dominance analysis uses a multiple comparison process to determine the order of the strengths and weaknesses of the various writing systems, thereby determining their relative importance. As shown in Table 6 and Table 7, the marks A, B, C, D, E, F, and G denote the various writing systems. L(·), S(·), X(·), and M(·) denote the dominance analysis at four system discussion levels. Points are awarded to writing systems that meet the competency criteria. If a writing system has a higher cumulative score, it means that the system has an advantage over the others. For example, L(·) determines whether the writing system has the self-learning ability. $L(A) \ni L(B)$ denotes that the writing

system "A'' has an advantage over writing system "B''. The accumulated point for the writing system "A'' is 1, which is expressed as $A\{1\}$. The + symbol indicates that the writing system considers the posture angle of the brush, which helps render better writing results. We utilize dominance analysis to obtain the results shown in Table 7. The conclusion is that the proposed writing system can autonomously generate 5-DOF action instructions better than other systems. This also highlights the direction in which our writing system will need continuous improvement in the future.

V. CONCLUSION

This paper presents a self-learning Chinese calligraphy-writing system which was designed and trained using an HGN learning model. The proposed writing system is based on the 5-DOF action instructions, which combine stroke trajectories and angle information from the HGN, to accomplish Chinese calligraphy writing. The action instructions of the robotic arm are automatically generated and corrected by the writing system through an image-to-action translation. In addition, the proposed Chinese calligraphy writing system can obtain the instructions from HGN for the pressure (height), rotation, and tilt of the brush bundle to precisely control the posture and angle of the brush for the presentation of stroke beauty resulting. The MAPE method is chosen to verify the performance of the writing results. The correction algorithm and linear regression were used to improve the error correction results. The writing results demonstrated in this paper finally approach the given human writing sample after self-modifications of the proposed writing system, proving that our proposed robotic writing system is capable of self-learning and correction. Further research is required to generate proper parameters of HGN automatically and to both evaluate and analyse the computational complexity of the proposed method.

ACKNOWLEDGMENT

The authors are grateful to the National Center for High-Performance Computing for the computer time and facilities to conduct this research.

REFERENCES

[1] F. Chao, J. Lv, D. Zhou, L. Yang, C.-M. Lin, C. Shang, and C. Zhou, "Generative adversarial nets in robotic Chinese calligraphy," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1104–1110.

[2] X. Gao, C. Zhou, F. Chao, L. Yang, C.-M. Lin, and C. Shang, "A robotic writing framework–learning human aesthetic preferences via human–machine interactions," *IEEE Access*, vol. 7, pp. 144043–144053, 2019.

[3] F. Chao, F. Chao, J. Lyu, R. Wu, X. Gao, C. Zhou, L. Yang, C.-M. Lin, and C. Shang, "Robotic Chinese calligraphy with human preference," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Aug. 2019, pp. 360–366.

[4] H. Fujioka, H. Kano, H. Nakata, and H. Shinoda, "Constructing and reconstructing characters, words, and sentences by synthesizing writing motions," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 36, no. 4, pp. 661–670, Jul. 2006.

[5] H. Fujioka and H. Kano, "Design of cursive characters using robotic arm dynamics as generation mechanism," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2006, pp. 3195–3200.

[6] R. Wu, W. Fang, F. Chao, X. Gao, C. Zhou, L. Yang, C.-M. Lin, and C. Shang, "Towards deep reinforcement learning based Chinese calligraphy robot," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2018.

[7] F. Yao, G. Shao, and J. Yi, "Extracting the trajectory of writing brush in Chinese character calligraphy," *Eng. Appl. Artif. Intell.*, vol. 17, no. 6, pp. 631–644, Sep. 2004.

[8] L. Gan, W. Fang, F. Chao, C. Zhou, L. Yang, C.-M. Lin, and C. Shang, "Towards a robotic Chinese calligraphy writing framework," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2018.

[9] F. Chao, Y. Huang, C.-M. Lin, L. Yang, H. Hu, and C. Zhou, "Use of automatic Chinese character decomposition and human gestures for Chinese calligraphy robots," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 1, pp. 47–58, Feb. 2019.

[10] F. Yao and G. Shao, "Modeling of ancient-style Chinese character and its application to CCC robot," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, Aug. 2006, pp. 72–77.

[11] J. Li, W. Sun, M. Zhou, and X. Dai, "Teaching a calligraphy robot via a touch screen," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2014, pp. 221–226.

[12] S. Mueller, N. Huebel, M. Waibel, and R. D'Andrea, "Robotic calligraphy—Learning how to write single strokes of Chinese and Japanese characters," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2013, pp. 1734–1739.

[13] X. Zhang, Y. Li, Z. Zhang, K. Konno, and S. Hu, "Intelligent Chinese calligraphy beautification from handwritten characters for robotic writing," *Vis. Comput.*, vol. 35, pp. 1193–1205, May 2019.

[14] Y. Sun, H. Qian, and Y. Xu, "Robot learns Chinese calligraphy from demonstrations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 4408–4413.

[15] J. H. M. Lam and Y. Yam, "Stroke trajectory generation experiment for a robotic Chinese calligrapher using a geometric brush footprint model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 2315–2320.

[16] D. Zhou, J. Ge, R. Wu, F. Chao, L. Yang, and C. Zhou, "A computational evaluation system of Chinese calligraphy via extended possibility-probability distribution method," in *Proc. 13th Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery (ICNC-FSKD)*, Jul. 2017, pp. 884–889.

[17] Z. Ma and J. Su, "Aesthetics evaluation for robotic Chinese calligraphy," *IEEE Trans. Cogn. Develop. Syst.*, vol. 9, no. 1, pp. 80–90, Mar. 2017.

[18] Y. Sun, H. Qian, and Y. Xu, "A geometric approach to stroke extraction for the Chinese calligraphy robot," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. Hong Kong: Hong Kong Convention and Exhibition Center, May 2014.

[19] X. Gao, F. Yang, T. Chen, and J. Si, "Chinese character components segmentation method based on faster RCNN," *IEEE Access*, vol. 10, pp. 98095–98103, 2016.

[20] X. Wu, Q. Chen, Y. Xiao, W. Li, X. Liu, and B. Hu, "LCSegNet: An efficient semantic segmentation network for large-scale complex Chinese character recognition," *IEEE Trans. Multimedia*, vol. 23, pp. 3427–3440, 2021.

[21] X. Liu, B. Hu, Q. Chen, X. Wu, and J. You, "Stroke sequence-dependent deep convolutional neural network for online handwritten Chinese character recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4637–4648, Nov. 2020.

[22] W.-Y. Wang, M.-J. Hsu, L.-A. Yu, Y.-H. Chien, and C.-C. Hsu, "Deep learning-based hypothesis generation model and its application on virtual Chinese calligraphy-writing robot," *IEEE Access*, vol. 8, pp. 87243–87251, 2020.

[23] M.-J. Hsu, Y.-H. Chien, W.-Y. Wang, and C.-C. Hsu, "Virtual Chinese calligraphy-writing robot with memory system," in *Proc. ICPAI*, Dec. 2020, p. 1.

[24] Y.-H. Chien, M.-J. Hsu, L.-A. Yu, W.-Y. Wang, and C.-C. Hsu, "Robotic calligraphy system using delta-like robot manipulator and virtual brush model," *iRobotics*, vol. 2, no. 4, pp. 1–6, Dec. 2019.

[25] M.-J. Hsu, Y.-H. Chien, Y.-T. Wu, W.-Y. Wang, and C.-C. Hsu, "Mechanical design and kinematics analysis of robotic calligraphy system using a delta-like robot manipulator," in *Proc. iFUZZY*, Nov. 2019, pp. 1–3.

[26] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital pattern," *Commun. ACM*, vol. 27, no. 3, pp. 236–239, 1984.

[27] M.-J. Hsu, Y.-H. Chien, W.-Y. Wang, and C.-C. Hsu, "Design and implementation of robotic calligraphy system," in *Proc. Int. Conf. Adv. Robot. Intell. Syst.*, Sep. 2017, pp. 60–63.
[28] C.-Y. Tzou, M.-J. Hsu, J.-Z. Jian, Y.-H. Chien, W.-Y. Wang, and C.-C. Hsu, "Mathematical analysis and practical applications of a serial-parallel robot with delta-like architecture," *Int. J. Eng. Res. Sci.*, vol. 2, no. 5, pp. 80–91, May 2016.

MIN-JIE HSU was born in Taipei, Taiwan, in 1993. He received the B.S. degree in electrical engineering from the National Taiwan Normal University, Taipei, in 2015, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. His research interests include artificial intelligence, fuzzy logic systems, neural networks, and reinforcement learning.

PO-CHAO YEH received the B.S. degree in electrical engineering from Tamkang University, Taipei, Taiwan, in 1999, the first M.S. degree in communications, navigation, and control engineering from the National Taiwan Ocean University, Keelung, Taiwan, in 2016, and the second M.S. degree in electrical engineering from the National Taiwan Normal University, Taipei, in 2022. He is currently a Senior Engineer with the Inventec Group, AIMobile Company. His research interests include reinforcement learning, robust adaptive control, and artificial intelligence.

YI-HSING CHIEN was born in Taipei, Taiwan, in 1978. He received the M.S. degree in electrical engineering from Fu Jen Catholic University, Taipei, in 2007, and the Ph.D. degree in electrical engineering from the National Taipei University of Technology, Taipei, in 2012. He is currently a Postdoctoral Researcher with the Department of Electrical Engineering, National Taiwan Normal University, Taipei. His current research interests and publications include the areas of fuzzy logic control, robust adaptive control, machine learning, and neural networks.

CHENG-KAI LU (Senior Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Fu Jen Catholic University, Taipei, Taiwan, in 2001 and 2003, respectively, and the Ph.D. degree in engineering from The University of Edinburgh, U.K., in 2012. After graduation, he worked as the Director of the Research and Development Division, Chyao Shiunn Electronic Industrial Company (Shanghai), before joining the Science and Technology Policy Research and Information Centre, National Applied Research Laboratories, Taiwan. He is currently a Faculty Member of the Department of Electrical Engineering, National Taiwan Normal University (NTNU), Taipei. Before he joined NTNU, he was a Faculty Member of the Electrical and Electronic Engineering Department, Universiti Teknologi Petronas (UTP), Malaysia, from 2016 to 2021. Apart from academic experience, he has more than eight years of industrial work experience. He has not only published his research works in peer-reviewed papers (book chapters, journal articles, conferences, and reports) but also has filed a couple of patents. His research interests include medical imaging, embedded systems, artificial intelligence and their applications, and clinical decision support systems. He served as an Executive Member for the IEEE EMBS Malaysia Chapter, from January 2017 to February 2018, and Penang Chapter, since 2018.

WEI-YEN WANG (Fellow, IEEE) received the Diploma degree in electrical engineering from the National Taipei Institute of Technology, Taipei, Taiwan, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from the National Taiwan University of Science and Technology, Taipei, in 1990 and 1994, respectively.

From 1990 to 2006, he worked concurrently as a Patent Screening Member of the National Intellectual Property Office, Ministry of Economic Affairs, Taiwan. In 1994, he was appointed as an Associate Professor in the Department of Electronic Engineering, St. John's and St. Mary's Institute of Technology, Taiwan. From 1998 to 2000, he worked with the Department of Business Mathematics, Soochow University, Taiwan. From 2000 to 2004, he was with the Department of Electronic Engineering, Fu Jen Catholic University, Taiwan. In 2004, he became a Full Professor with the Department of Electronic Engineering, Fu Jen Catholic University. In 2006, he was a Professor and the Director of the Computer Center, National Taipei University of Technology, Taiwan. From 2007 to 2014, he was a Professor with the Department of Applied Electronics Technology, National Taiwan Normal University, Taiwan. From 2011 to 2013, he was the Director of the Information Technology Center, National Taiwan Normal University. Since 2003, he has been certified as a Patent Attorney in Taiwan. He is currently a Professor with the Department of Electrical Engineering, National Taiwan Normal University. His current research interests and publications include the areas of fuzzy logic control, robust adaptive control, neural networks, computer-aided design, digital control, and CCD camera-based sensors. He has authored or coauthored over 200 refereed conference and journal papers in these areas.

Dr. Wang is a fellow of IET. He was a recipient of the Best Associate Editor Award for IEEE TRANSACTIONS ON CYBERNETICS. He is currently serving as an Associate Editor for the IEEE TRANSACTIONS ON CYBERNETICS and an Associate Editor for the *International Journal of Fuzzy Systems*.

CHEN-CHIEN JAMES HSU (Senior Member, IEEE) was born in Hsinchu, Taiwan. He received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1987, the M.S. degree in control engineering from the National Chiao Tung University, Hsinchu, in 1989, and the Ph.D. degree from the School of Microelectronic Engineering, Griffith University, Brisbane, QLD, Australia, in 1997.

He was a Systems Engineer with IBM Corporation, Taipei, for three years, where he was responsible for information systems planning and application development, before commencing his Ph.D. studies. He joined the Department of Electronic Engineering, St. John's University, Taipei, as an Assistant Professor, in 1997, and was appointed as an Associate Professor, in 2004. From 2006 to 2009, he was with the Department of Electrical Engineering, Tamkang University, Taipei. He is currently a Professor with the Department of Electrical Engineering, National Taiwan Normal University, Taipei. He is the author or coauthor of more than 200 refereed journals and conference papers. His current research interests include digital control systems, evolutionary computation, vision-based measuring systems, sensor applications, and mobile robot navigation. He is a fellow of IET.