Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A virtual 3D interactive painting method for Chinese calligraphy and painting based on real-time force feedback technology*

Chao GUO‡, Zeng-xuan HOU, You-zhi SHI, Jun XU, Dan-dan YU

(*School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, China*)

E-mail: 358826947@qq.com; hou@dlut.edu.cn; 43972049@qq.com; 1017184024@qq.com; 642484580@qq.com

Received May 24, 2016; Revision accepted Jan. 23, 2017; Crosschecked Nov. 20, 2017

Abstract: A novel 3D interactive painting method for Chinese calligraphy and painting based on force feedback technology is proposed. The relationship between the force exerted on the brush and the resulting brush deformation is analyzed and a spring-mass model is used to build a model of the 3D Chinese brush. The 2D brush footprint between the brush and the plane of the paper or object is calculated according to the deformation of the 3D brush when force is exerted on the 3D brush. Then the 3D brush footprint is obtained by projecting the 2D brush footprint onto the surface of the 3D object in real time, and a complete 3D brushstroke is obtained by superimposing 3D brush footprints along the painting direction. The proposed method has been successfully applied in a virtual 3D interactive drawing system based on force feedback technology. In this system, users can paint 3D brushstrokes in real time with a Phantom Desktop haptic device, which can effectively serve as a virtual reality interface to the simulated painting environment for users.

Key words: 3D brush model; 3D brushstroke; 3D interactive painting; Real-time force feedback technology
https://doi.org/10.1631/FITEE.1601283 **CLC number:** TP391.9

1 Introduction

Virtual reality 3D painting has been a major area of interest in computer graphics for many years. Numerous approaches have been proposed to realistically represent a brush and model its behavior and interaction with a 3D object surface.

These methods can generally be divided into three categories: texture mapping-based methods (Hanrahan and Haeberli, 1990; Johnson *et al*., 1999; Igarashi and Cosgrove, 2001; Foskey *et al*., 2002; Lin *et al*., 2002), empirical brush models (Agrawala *et al*., 1995; Gregory *et al*., 2000; Kim *et al*., 2003; 2004; Fu and Chen, 2008; Fu *et al*., 2010), and physical brush-based models (Adams *et al*., 2004; Otsuki *et al*., 2009; 2010). A physical brush model has not been constructed in the texture mapping-based and empirical brush-modeling simulation methods yet. Hence, 3D painting could not be simulated in real time. Force feedback technology can provide a sense of reality in the 3D painting process although it was not referred to in the above methods. Many research works (Adams *et al*., 2004; Otsuki *et al*., 2009; 2010) have proposed physical brush-based models. However, these methods are inappropriate for Chinese calligraphy and Chinese painting. Chinese calligraphy and painting are usually created using Chinese brushes, which are different from western brushes. Such brushes are used to create the brushs-trokes typical of Chinese art, which convey the artist's deep feelings when he/she paints on a 3D object.

Based on the force feedback technology, we propose a new method to simulate the 3D drawing process for Chinese calligraphy and painting. The advantages are the following:

1. Based on the force feedback technology, we designed a 3D Chinese brush model. Using this

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (No. 51175058)
ORCID: Chao GUO, http://orcid.org/0000-0003-2610-9613

model, users can draw 3D brushstrokes in real time.

2. It provides a natural and efficient method of human-computer interaction. Users virtually paint in real time with a Phantom Desktop haptic device and a 3D mouse (two-handed interaction), which can effectively enhance the reality of the interactive painting process.

3. A local mapping technique is suggested instead of global surface parameterization vertices for the geometric model, which can effectively reduce the computational complexity.

2 Related work

Igarashi and Cosgrove (2001) proposed a method for dynamically generating an efficient texture bitmap and its associated UV mapping (which is the 3D modeling process of projecting a 2D image to a 3D model's surface for texture mapping) in an interactive texture painting system for 3D models. Nevertheless, the repeated reprojection of the existing texture gradually degrades the quality of the bitmap and, hence, their system is suitable for simple paintings with a limited number of brushstrokes.

ArtNova offers the capability of interactively applying textures onto 3D surfaces directly by brushstrokes, with the orientation of the texture determined by the stroke (Foskey *et al*., 2002; Lin *et al*., 2002). However, a 3D brush model is not yet constructed in both methods, which have impacts on the reality of interactive painting.

Fu and Chen (2008) presented a haptic-based 3D painting system, which supports painting directly on the CAD model as well as the scanned physical objects. Their painting system provides virtual brushes based on a haptic device with an input of three degrees of freedom (DOFs). Since a real brush has six DOFs when painting in real life, their system cannot provide various brush effects.

Based on an implicit haptic technique, the haptic decoration technique allows the user to paint directly on 3D models, and then sense the variations in thickness due to the added paint (Kim *et al*., 2003; 2004). However, a 3D brush model has not yet been constructed in the methods of Kim *et al*. (2003; 2004), which is the same problem found in the works by Foskey *et al*. (2002) and Lin *et al*. (2002).

Gregory *et al*. (2000) proposed an intuitive 3D interface for interactively editing and painting a polygonal mesh using a force feedback device. Their system allows users to naturally create complex forms and patterns aided not only by visual feedback, but also by their sense of touch. A six-DOF haptic device was not integrated into their system and the painting is created without two-handed interaction.

Adams *et al*. (2004) proposed a sample-based approach that represents the geometry and appearance of the 3D object as well as the brush surface. The generalization of 2D pixel-based paint models to sampling points allows users to simulate the paint transfer for 3D objects. However, their method fails to mimic the real Chinese brush features of brush flattening and bristle spreading when the real-time force is exerted on the Chinese brush.

Chu and Tai (2002; 2004; 2005) constructed brush models to simulate interactive painting on a 2D painting surface. They represented the brush surface as a swept surface defined by the spine and a varying elliptic cross section, and incorporated lateral spine nodes to allow brush flattening. They modeled the brush dynamics based on energy optimization. This modeling method can simulate small-scale deformations of the brush geometry instead of large-scale bending or stretching due to restriction of the constrained energy minimization. They adopted a Wacom tablet as the input device rather than a haptic device, which reduces the sense of reality for users.

A western brush (Baxter *et al*., 2004) is modeled as a spring-mass system. Using an approximated implicit integration method, Baxter *et al*. were able to produce a real-time system for creating acrylic-like painting. However, no attention was paid to the brush flattening and bristle spreading, which plays an important role during the Chinese calligraphic and drawing process.

Baxter and Lin (2004) proposed a multiple spine method to mimic western brushes. Because they adopted the energy minimization method to simulate brush deformations, they encountered the same difficulties as Chu and Tai (2002; 2004; 2005).

Chen *et al*. (2015) proposed a real-time painting simulation system that models the interactions among the brush, paint, and canvas at the bristle level. However, their method does not mention the force feedback technology so the sense of reality is reduced during the virtual interactive painting process.

In this paper, a 3D interactive painting method for Chinese calligraphy and painting based on force feedback is proposed. We research the relationship between real time force and brush deformation, and adopt a spring-mass model to mimic a Chinese brush. In the model, we compute the force exerted on the brush through a perpendicular virtual spring, which deforms in the normal plane direction. Then we simulate both small- and large-scale brush bending and stretching according to the force in real time. We also compute a 2D brush footprint in accordance with the brush deformation at the sampling time. Along the painting direction, a complete 3D brushstroke is obtained by superimposing 3D brush footprints, which are obtained by projecting 2D brush footprints in the plane onto the 3D object surface in real time. At the same time, users feel the force on the brush, which simulates touch during the Chinese calligraphic and painting process. With our method, users can paint on the surface of a 3D object in real time, in the manner similar to a real drawing process.

3 Modeling a 3D Chinese brush based on force feedback technology

In a virtual Chinese calligraphic and painting process, the key component is an expressive 3D Chinese brush model. Inspired by our previous modeling method (Guo *et al*., 2015), we build a Chinese brush model using a two-layer structure incorporating geometry and dynamics to take into consideration real Chinese brush characteristics.

3.1 Geometry

Geometry and dynamics are closely related. A well-structured geometry as the basis for simulating different kinds of brush deformations can improve real-time performance. Inspired by the method from Chu and Tai (2002; 2004), we describe the geometry (Fig. 1) as a skeleton and a surface.

The skeleton controls the brush deformation: it can be described as a connected line segment (spinal segment) sequence that gradually becomes shorter in the front. The brush head bends more than the root does due to its being softer. The brush head and abdomen are commonly used for painting, so for modeling efficiency, progressively shorter segments are used toward the brush head, so as to prompt higher resolution. Assuming that the skeleton has m+1 nodes, $P_0, P_1, \ldots, P_m$, where P_0 and P_m are the head node and root node, respectively. The length representation of $P_{j-1}P_j$ is le_j, $\{\mathrm{le}_j\}$ constituting an arithmetic progression, and the computational method is described as follows:

$$\begin{cases}\mathrm{le}_j = \mathrm{le}_1 + (j-1)d, \\ d = 2(\mathrm{LE} - m\cdot \mathrm{le}_1)/\left[m(m-1)\right],\end{cases} \tag{1}$$

where j (a positive integer) is less than or equal to m, LE and d are the whole brush skeleton length and common difference, respectively, and le_1 refers to the softness of the brush: when the brush bears the same force, the softer brush bristles around the brush head are more easily deformed, and le_1 is smaller, and vice versa. We obtain le_1 based on experiments, and then we obtain d according to LE and le_1.

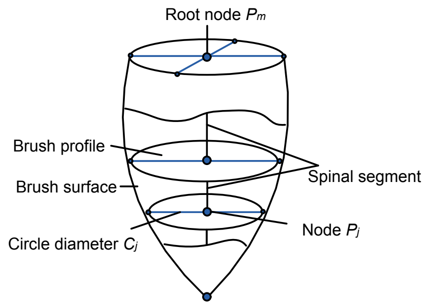


Fig. 1 Brush geometry

The whole spinal segment portion is located in the same plane (Fig. 2) when the brush is bent. θ ($\theta\in(0, \pi)$) and β_j ($j\in[1, n]$) are the angle of the brush holder to the plane, and the angle of the $P_{j-1}P_j$ to the plane, respectively. β_j is initially equal to θ. For the purpose of controlling the brushstroke outline, the cross section, the center of which is P_j ($j\in[1, n]$), is denoted as the brush profile. The two adjacent spinal segments are divided by the brush profile, which is also perpendicular to the plane of the spinal segments. Various positions and brush profile sizes are used for simulating the brush deformation.

The skeleton and various brush profiles are used for defining the brush surface, which is a triangular mesh surface. Initially, the brush profiles are circles, because the Chinese brush shape is cone. The circle diameters are predefined to mimic the kinds of brushes that are commonly used. During the painting process, the circle diameter, the center of which is P_m, stays the same, while the brush profile circle is flattened to form an ellipse when force is exerted on the brush (Fig. 3). This expression is an effective simulation of a real brush deformation.

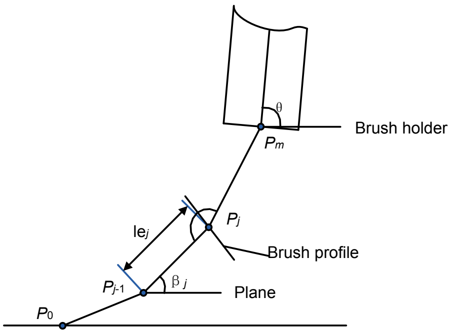


Fig. 2 Skeleton deformation

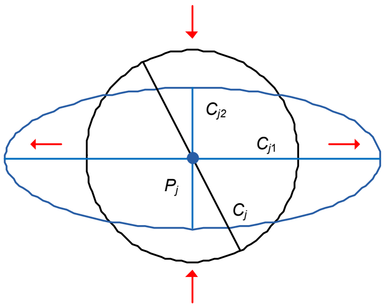


Fig. 3 Brush profile deformation, the center of which is node P_j

In Fig. 3, C_j, C_{j1}, and C_{j2} are the circle diameter, ellipse major diameter and minor diameter, respectively, and C_{j1} is described in Eq. (2).

$$C_{j1} = C_j \cdot (1 + d \cdot \text{pe} + h \cdot f_c). \quad (2)$$

Dividing the pressure F by the maximum output force of the force feedback device, we obtain the pressure coefficient (pe). As the maximum output force of our haptic device is 7.9 N, the pe (value range [0, 1]) is equal to F/7.9. When drawing in real life, users often need to paint with brushes of varying softness. Thus, for purposes of simulation, the adjustment coefficients d and h are set for adjusting the ellipse sizes based on real painting experiments. The coefficient of friction influence (f_c) is equal to $\mu\cdot$pe, where μ is the friction coefficient of the 3D object surface and brush.

C_{j2} is given in Eq. (3) according to the conservation of area:

$$C_{j2} = C_j^2 / C_{j1}. \quad (3)$$

3.2 Dynamics

We apply the spring-mass model (Fig. 4) to mimic the dynamic deformation process for brush flattening and bristle spreading when real-time force is exerted on the brush. During the virtual interactive painting process, a vertical spring which deforms in the normal plane is set between P_m and its projection P'_m. In the initial painting condition (Fig. 4a), P'_m and P_0 overlap each other. When the pressure is exerted on the brush (Fig. 4b), the location of every spinal segment changes, but the length of every spinal segment is invariant. The feedback pressure (F_p) is proportional to the downward displacement of the brush. F_p is given as follows:

$$F_p = \lambda_c \cdot S \cdot D_s. \quad (4)$$

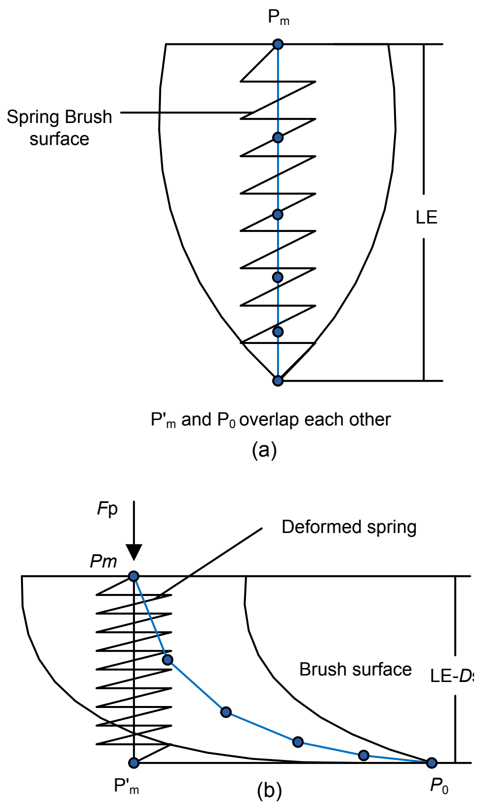


Fig. 4 Brush spring-mass model: (a) in the initial painting condition; (b) when the pressure is exerted on the brush

For the purpose of simulating an appropriate output force transferred to the user, a feedback force coefficient (λ_c) is adopted to control the force magnitude. The λ_c (N/mm) is determined experimentally according to our hardware components of Phantom Desktop haptic device and HP Xw8600 workstation. The value range for the brush softness coefficient (S) is (0, 1). The brush is much softer with the decrease in S, so the downward displacement is much larger when the same force is exerted on the brush. D_s (mm) refers to the amount of deformation in the spring, which is also the downward displacement of the brush.

Friction can help the user feel the roughness of the 3D object surface and control the 3D Chinese brush. We compute the friction (F_f), which is proportional to F_p (N) according to Eq. (5):

$$F_f = \mu \cdot F_p. \tag{5}$$

4 Three-dimensional brushstroke

3D painting systems often employ polygonal meshes to represent the underlying 3D geometry. By establishing some mapping between a 2D plane and the 3D surface, the appearance attributes, resulting from paint operations, can be obtained. In this study, a local mapping method is proposed to map the 2D brush footprint onto the 3D object surface in real time. The 3D brushstroke is obtained by superimposing the 3D brush footprints along the painting direction. Then a Kubelka-Munk (KM)-based method is applied to render the 3D brushstroke in real time.

4.1 Two-dimensional brush footprint

The brush footprint on the plane is similar to a raindrop shape (Fig. 5a), and we simulate the footprint profile by the B-spline fitting (Fig. 5b).

In Fig. 5, the coordinate values of the skeleton nodes and profile points for the brush footprint are calculated through a geometric relationship. The mathematical expressions of the related points are given as follows.

α ($\alpha \in [0, 2\pi)$) is the angle between the brush head direction ($P'_m P_0$) and the X axis. The 3D coordinate values of node P_j are given in Eq. (6):

$$\begin{cases} x_{P_j} = \cos\alpha \sum_{t=j+1}^{m} \left(\mathrm{le}_t \cdot \cos\beta_t\right), \\ y_{P_j} = \sin\alpha \sum_{t=j+1}^{m} \left(\mathrm{le}_t \cdot \cos\beta_t\right), \\ z_{P_j} = \sum_{t=1}^{j} \left(\mathrm{le}_t \cdot \sin\beta_t\right). \end{cases} \tag{6}$$

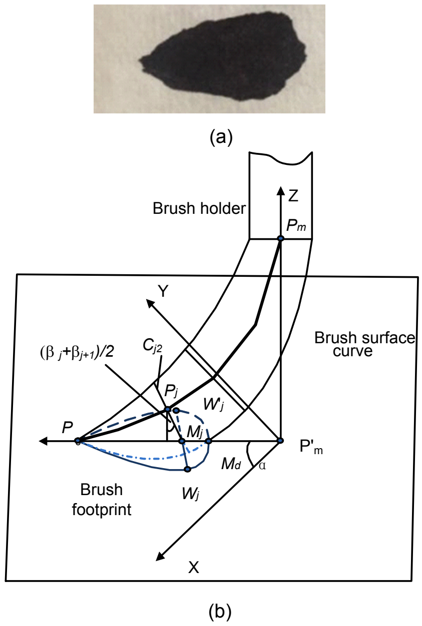


Fig. 5 Real (a) and simulated (b) 2D brush footprint

On the brush profile, the center of which is node P_j, the profile point W_i of the brush footprint is determined as follows:

$$\begin{cases} x_{W_j} = x_{P_j} - z_{P_j} \tan\left(\frac{\beta_j + \beta_{j+1}}{2}\right)\cos\alpha + \left|W_j M_j\right| \sin\alpha, \\ y_{W_j} = y_{P_j} - z_{P_j} \tan\left(\frac{\beta_j + \beta_{j+1}}{2}\right)\sin\alpha - \left|W_j M_j\right| \cos\alpha, \\ z_{W_j} = 0. \end{cases} \tag{7}$$

We compute the $W_j M_j$ length in Eq. (8).

$$\left|W_j M_j\right| = \frac{C_{j1}}{2C_{j2}} \sqrt{C_{j2}^2 - 4z_{P_j}^2 \Big/ \cos^2\left(\frac{\beta_j + \beta_{j+1}}{2}\right)}. \tag{8}$$

As the brush footprint is axially symmetric, we compute W_j' in Eq. (9).

$$\begin{cases} x_{W_j'} = x_{P_j} - z_{P_j} \tan\left(\dfrac{\beta_j + \beta_{j+1}}{2}\right)\cos\alpha - \left|W_j M_j\right|\sin\alpha, \\ y_{W_j'} = y_{P_j} - z_{P_j} \tan\left(\dfrac{\beta_j + \beta_{j+1}}{2}\right)\sin\alpha + \left|W_j M_j\right|\cos\alpha, \\ z_{W_j'} = 0. \end{cases} \tag{9}$$

M_d is the intersection between the brush surface curve and P_0P_m', and then the B-spline fitting is based on points P_0, W_j, M_d, and W_j'.

4.2 Three-dimensional brush footprint

When a collision between the brush and the 3D object surface is detected, the paint is transferred from the brush to the surface. A local mapping technique is proposed to describe the paint transfer process. First, the smallest enclosing sphere is computed from the current position of the bent brush. Then the plane position is determined, and the 3D brush footprint profile is obtained by mapping the 2D brushstroke onto the 3D object surface in real time. A real-time algorithm for the ink quantity in the 3D brush footprint is also advanced to simulate the ink transfer process between the brush and the 3D object surface.

4.2.1 Three-dimensional brush footprint profile

Inspired by the smallest enclosing ball algorithm (Larsson and Källberg, 2013), we compute the bounding sphere of the bent brush according to an effective ball expanding operation until all vertices of the surface of the bent brush are in the bounding sphere.

In Fig. 6, vertex A is an arbitrary vertex of the bent brush. On the surface of the bent brush, the farthest vertex from vertex A is denoted by vertex B which is obtained by traversing all the vertices of the surface of the bent brush. In the same way, the farthest vertex from vertex B is denoted by vertex C. The midpoint and half-length of line segment BC are denoted by D_0 and r_0, respectively. The initial sphere is defined as (D_0, r_0).

The length of the line segment D_0E, which is obtained by connecting an outlier vertex E and D_0, is denoted by l_1. D_1 is positioned along the line segment D_0E at distance r_1 from vertex E, and the updated sphere is defined as (D_1, r_1). In the initial sphere, the diameter FG is perpendicular to the line segment D_0E. Vertex E and diameter FG form an isosceles triangle EFG with base $2r_0$, and the radius of the updated sphere r_1 is directly given by

$$r_1 = \frac{r_0^2 + l_1^2}{2 \cdot l_1}. \tag{10}$$

The length of line segment D_0D_1 (ld_1) is then computed simply as

$$\mathrm{ld}_1 = \sqrt{r_1^2 - r_0^2}. \tag{11}$$

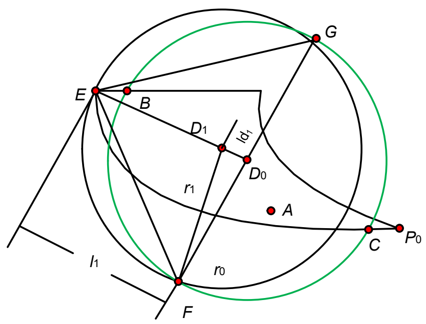


Fig. 6 The bounding sphere of the bent brush

The coordinate values of the updated center of the sphere are calculated according to ld_1 and the coordinate values of D_0.

The above steps are repeated until all vertices of the surface of the bent brush are in the bounding sphere, and the smallest enclosing sphere of the bent brush is obtained.

In Fig. 7, the projection point of the root node P_m' is on the surface of the 3D object, and the direction of the vector $P_m'P_m$ is defined as the normal of vertex P_m'. When the distance between the object surface vertex and the center of the smallest enclosing sphere of the bent brush is less than or equal to the radius of the smallest enclosing sphere of the bent brush, these object surface vertices are defined as effective vertices at this sampling time. The average normal of these effective vertices are defined as the normal of the 2D plane, and then the plane position is determined according to P'_m and the normal of the 2D plane. Then the 2D brush footprint on the 2D plane is mapped onto the 3D object surface according to each normal of the effective vertex, and the 3D brush footprint profile is obtained.

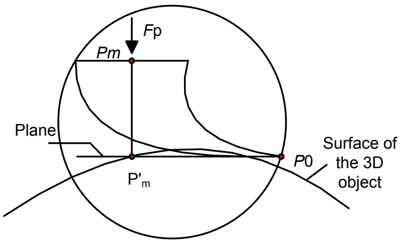


Fig. 7 The determination of the plane position

4.2.2 Ink quantity of the 3D brush footprint

The ink quantity transfers between the brush and the virtual 3D object surface during the 3D painting process, and then forms the 3D brush footprint. In general, the ink quantity of the 3D brush footprint at a sampling point is proportional to the pressure exerted on the brush and the current ink quantity of the brush. Suppose the initial brush ink quantity is denoted by Z, and the pressure factor of the ith sampling point is denoted by pe_i. The brush footprint ink quantity at the first sampling time (Q_1) is determined by

$$Q_1 = \eta \cdot \mathrm{pe}_1 \cdot Z. \tag{12}$$

To control the ink quantity of the 3D brush footprint under different painting conditions (for example, the painting is undertaken with different brushes and 3D object surfaces), we set the ink quantity factor η, which is determined by painting experiments to simulate the most realistic ink transfer process between the brushes and the 3D object surfaces.

The ink quantity of the 3D brush footprint at the ith sampling time (Q_i) is determined by

$$Q_i = \eta \cdot \mathrm{pe}_i \cdot \left(Z - \sum_{t=1}^{i-1} Q_t \right), \tag{13}$$

where $i \geq 2$, and i is a positive integer.

4.3 Generation and display of the 3D brush stroke

During the virtual 3D painting process, we simulate the 3D brushstroke by superimposing 3D brush footprints along the sampling points. In each vertex of a 3D brushstroke, a new KM-based method (Hou *et al*., 2015) is applied to transform ink quantities into color intensities (RGB) which are commonly used to display color in computer-aided painting, and can be easily stored for accessing and displaying the painting in real time.

In this study, the storage data for the painting are stored as follows: the initial position of the painting, the position, normal, and RGB color intensities of each vertex of the 3D surface. An example for a stored painting is given as follows:

InitialPosition={0, 0, 0}
{172, 205, 309, 22, 30, 42, 39}.

5 Experiment and analysis

During interactive human-computer drawing, in order to create the most authentic painting experience possibly, two features are important when choosing an input device to simulate real Chinese brush behaviors: (1) given that a real Chinese brush has six degrees of freedom, the number of input DOFs should be as close to the six DOFs of a real Chinese brush as possible; (2) the force and play in the brush felt in the painter's hand through the brush is a critical indicator of the position and state of the brush at each moment of the painting process, thus the input device should provide a haptic sensation which is very similar to a physical painting experience with real brushes. Table 1 shows some input devices frequently used during an interactive painting process. From Table 1 we can conclude that the Phantom Desktop device has more DOFs, and the output force can be controlled through the computer program.

Table 1 Three input devices which are frequently used during the interactive painting process

Input device	Number of input DOFs	Number of output DOFs	Haptics
Mouse	2	0	None
Wacom Intuos Tablet	5	0	Static
Phantom Desktop	6	3	Programmable

In our virtual 3D interactive drawing system with force feedback, we use the Qt and Visual Studio

2005 as the graphical user interface (GUI) and the integrated development environment (IDE), respectively. To handle the visual display and user interface, we apply Open Inventor as the graphical kernel library. We also adopt standard modules from the OpenHaptics library of SensAble Technologies, Inc. to simulate real-time haptic sensation. The hardware components of our system (Fig. 8) include an HP Xw8600 workstation for graphics and haptics rendering, an HP SpaceBall 5000 3D mouse (held in the left hand) for moving and rotating the 3D object (for example, a bowl), and a Phantom Desktop haptic device (held in the right hand) for manipulating the 3D brush. The update frequency of the haptic device is 1 kHz to provide stable real-time interaction. The other operations are done in the painting loop which runs at a 30-Hz frequency for the visual display.

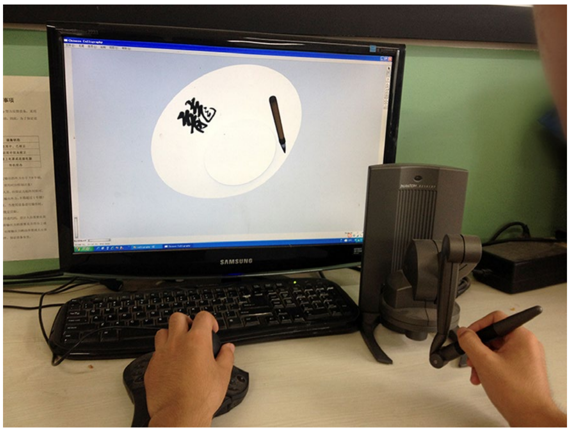


Fig. 8 The hardware setup for the virtual 3D interactive drawing system with force feedback

Users can realize interactive drawing on the virtual 3D object surface by realistically wielding the force feedback device. In the schematic diagram of the system (Fig. 9), the haptic device provides the brush motion and location information in order to compute the force exerted on the brush. The deformed brush interacts with the 3D object surface, which forms the 3D brushstroke. At the same time, the calculated force is sent back to the Phantom Desktop device as the feedback force, which is controlled by the users by adjusting the 3D brushstroke to the desired stroke.

Reasonable parameters are used to simulate haptic sensation. The parameters can be characterized as geometric parameters, dynamic parameters, and

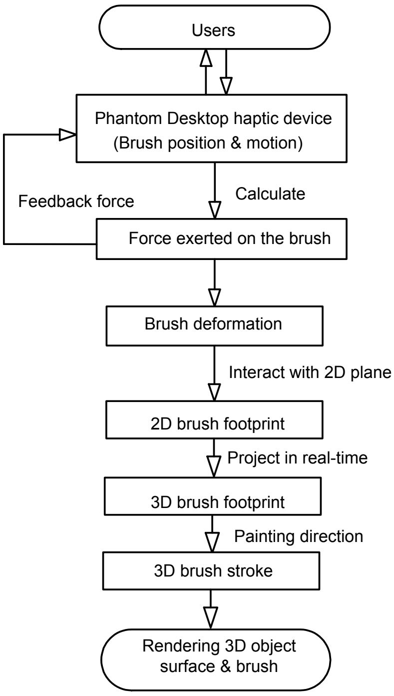


Fig. 9 A schematic diagram of the virtual 3D interactive drawing system based on force feedback

3D object surface parameter (friction coefficient μ). The geometric parameters refer to the P_0P_1 length (le_1), the number of nodes in the brush skeleton (m), the length of the whole skeleton (LE), and circle diameters. The dynamic parameters of the brush include a brush softness coefficient (S), and adjustment coefficients d and h. The LE and circle diameters are predefined to simulate different kinds of Chinese brushes. In general, the more skeleton nodes there are, the better the performance of the brush will be; however, more nodes will also increase the computational cost, and vice versa. We set the value range for m as 8–13. S (value range 0.3–0.7) is related to the magnitude of force when the Chinese brush is bent at 90°. The value ranges for le_1, d, and h are 0.8–1.4, 1.3–1.8, and 0.6–0.8, respectively. However, μ (value range 0.2–0.3) is related to the Chinese brush and the 3D object surface.

To model the Chinese brush, we applied the parameters in Table 2, and exerted different pressures (F_p). The brush deformation, 3D footprints, and strokes are shown in Fig. 10.

Table 2 Main parameters in the painting experiment

Parameter	S	le_1 (mm)	m	μ	d	h	LE (mm)
Value	0.7	1	9	0.24	1.5	0.75	35

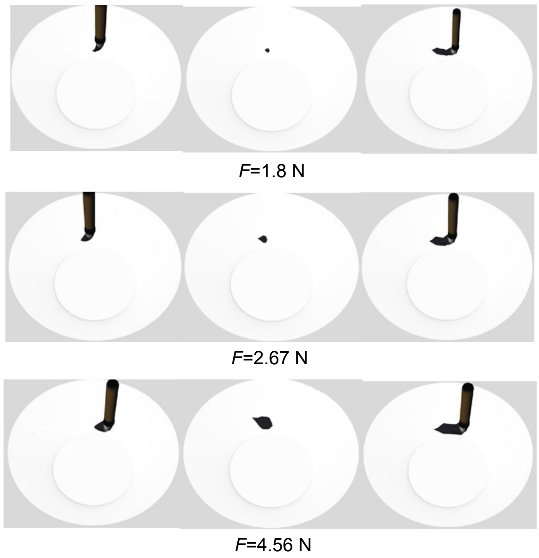


Fig. 10 Brush deformation, 3D footprints and strokes (from left to right) when different pressures are exerted on the brush

Painting was carried out exerting the same force on brushes of different sizes, and the deformations and brushstrokes are shown in Fig. 11.

Fig. 11 Brush deformation and different-sized brushstrokes (painted with the same force exerted on the brushes)

In traditional painting methods, one user uses a 2D mouse and keyboard to control a 3D brush and object. In our system, the HP SpaceBall 5000 3D mouse provides users with six DOFs for a 3D object, and it can move and rotate the object. In a 3D painting with the Phantom Desktop haptic device and HP SpaceBall 5000 3D mouse, the force information is input through the haptic device. During the drawing process, users can also adjust the force exerted on the brush to paint the desired 3D brushstroke by observing its action in real time.

To demonstrate the viability of the two-handed interaction (HP SpaceBall 5000 3D mouse and Phantom Desktop haptic device), we invited six volunteers (two art students and four non-art students) in our school to use the system in an experiment. We provided minimal training in handling the haptic device to the six volunteers before drawing. The volunteers used two devices randomly (I: keyboard and mouse; II: HP SpaceBall 5000 3D mouse and Phantom Desktop haptic device) for drawing the Chinese character ‘目’. The drawing time with device II was nearly half of the time with device I as shown in Fig. 12.

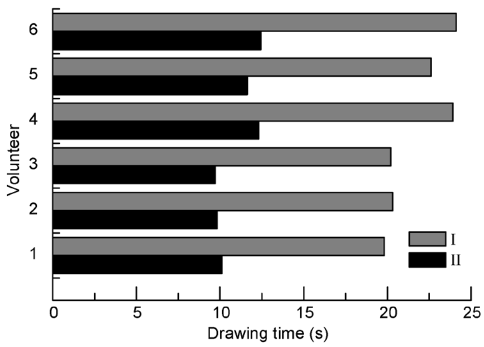


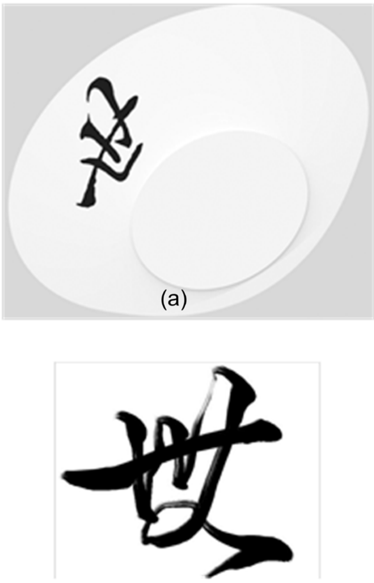
Fig. 12 Drawing time with the two devices (I and II)

We investigated the volunteers’ responses regarding these two devices. Of the volunteers, 5/6 of them expressed their preferences for device II. They commented that force feedback was useful in detecting and maintaining contact with the 3D object surfaces. On the other hand, they said it was quite difficult to create the painting in a 3D virtual reality space using only visual guidance. Based on the responses from the six volunteers, we are currently planning an extensive and formal user study to carefully measure and evaluate how our system helps to enhance reality for a painter in human-computer interaction 3D painting.

Some examples of Chinese calligraphy and paintings, which were created on the bottom of a bowl using our system, are shown in Fig. 13.

Figs. 14a and 15a show the Chinese characters ‘世’ and ‘漫步人生’, respectively, which were painted on the side of a bowl with our system. Figs. 14b and 15b are related Chinese characters created on a 2D painting surface with the ‘MoXi’ system (Chu and Tai, 2005). Comparing Figs. 14a with 14b and Figs. 15a with 15b, we can conclude that users can do a better job simulating Chinese character features with our method (for example, note the ink diffusion effects in Fig. 15a), and they can also draw Chinese calligraphic works on the surface of a 3D object with our system.

Fig. 13 Examples of Chinese calligraphy and paintings created on the bottom of a bowl using our system



(a)

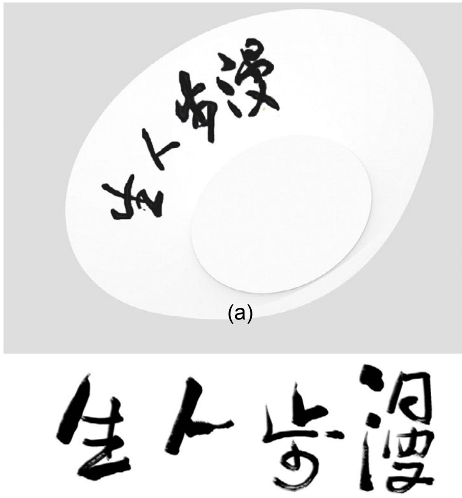
(b)

Fig. 14 The Chinese character '世' created with our system (a) and with the 'MoXi' system (b)

6 Conclusions

In this paper, a new 3D interactive painting method for Chinese calligraphy and painting has been proposed using force feedback technology to simulate the painting process.

First, we constructed a 3D model of a Chinese brush to simulate the deformation of the brush based on the force in real time, and then we computed the2D brush footprint. A real-time projection algorithm



(a)

(b)

Fig. 15 The Chinese characters '漫步人生' created with our system (a) and with the 'MoXi' system (b)

is also presented to map the 2D brush footprint into the 3D brush footprint, and then the 3D brushstroke is obtained. Our method has been successfully applied to a virtual 3D interactive drawing system based on the force feedback technology. With our system, the deformations and strokes of different 3D Chinese brushes were simulated in real time, and the visual feedback and force feedback were also simulated during the virtual interactive painting process. Compared with other painting systems which adopt a mouse and keyboard as the input devices, 3D input devices, such as the Phantom Desktop haptics and the HP SpaceBall 3D mouse, provide users with a high degree of freedom to directly control the brush and the movement of a 3D object in the 3D space, which can enhance reality for users, and allows the artists and designers to freely express their creativities through the sense of touch.

There are several possible future research directions. These include improved physical models for the 3D Chinese brush, more realistic 3D brush deformations integrated with tactile cues, and haptic sensations of non-physical attributes (such as anticipation of a collision).

References

Adams, B., Wicke, M., Dutré, P., *et al.*, 2004. Interactive 3D painting on point-sampled objects. Proc. 1st Eurographics Conf. on Point-Based Graphics, p.57-66. https://doi.org/10.2312/SPBG/SPBG04/057-066

Agrawala, M., Beers, A.C., Levoy, M., 1995. 3D painting on scanned surfaces. Proc. Symp. on Interactive 3D Graphics, p.145-150. https://doi.org/10.1145/199404.199429

Baxter, W., Wendt, J., Lin, M.C., 2004. IMPaSTo: a realistic, interactive model for paint. Proc. 3rd Int. Symp. on Non-photorealistic Animation and Rendering, p.45-56, 148. https://doi.org/10.1145/987657.987665

Baxter, W.V., Lin, M.C., 2004. A versatile interactive 3D brush model. Proc. 12th Pacific Conf. on Computer Graphics and Applications, p.319-328. https://doi.org/10.1109/PCCGA.2004.1348363

Chen, Z.L., Kim, B., Ito, D., *et al.*, 2015. Wetbrush: GPU-based 3D painting simulation at the bristle level. *ACM Trans. Graph.*, **34**(6), Article 200. https://doi.org/10.1145/2816795.2818066

Chu, N.S.H., Tai, C.L., 2002. An efficient brush model for physically-based 3D painting. Proc. 10th Pacific Conf. on Computer Graphics and Applications, p.413-421. https://doi.org/10.1109/PCCGA.2002.1167885

Chu, N.S.H., Tai, C.L., 2004. Real-time painting with an expressive virtual Chinese brush. *IEEE Comput. Graph. Appl.*, **24**(5):76-85. https://doi.org/10.1109/MCG.2004.37

Chu, N.S.H., Tai, C.L., 2005. MoXi: real-time ink dispersion in absorbent paper. *ACM Trans. Graph.*, **24**(3):504-511. https://doi.org/10.1145/1073204.1073221

Foskey, M., Otaduy, M.A., Lin, M.C., 2002. ArtNova: touch-enabled 3D model design. Proc. IEEE Virtual Reality, p.119-126. https://doi.org/10.1109/VR.2002.996514

Fu, C.W., Xia, J.Z., He, Y., 2010. LayerPaint: a multi-layer interactive 3D painting interface. Proc. SIGCHI Conf. on Human Factors in Computing Systems, p.811-820. https://doi.org/10.1145/1753326.1753445

Fu, Y.X., Chen, Y.H., 2008. Haptic 3D mesh painting based on dynamic subdivision. *Comput. Aid. Des. Appl.*, **5**(1-4): 131-141. https://doi.org/10.3722/cadaps.2008.131-141

Gregory, A.D., Ehmann, S.A., Lin, M.C., 2000. inTouch: interactive multiresolution modeling and 3D painting with a haptic interface. Proc. IEEE Virtual Reality, p.45-52. https://doi.org/10.1109/VR.2000.840362

Guo, C., Hou, Z.X., Yang, G.Q., *et al.*, 2015. The simulation of the brushstroke based on force feedback technology. *Math. Prob. Eng.*, Article 164821. https://doi.org/10.1155/2015/164821

Hanrahan, P., Haeberli, P., 1990. Direct WYSIWYG painting and texturing on 3D shapes. Proc. 17th Annual Conf. on Computer Graphics and Interactive Techniques, p.215-223. https://doi.org/10.1145/97880.97903

Hou, Z.X., Guo, C., Chen, G.Z., *et al.*, 2015. The methods of the pigment mixing and color storage in the virtual painting. *J. Inform. Comput. Sci.*, **12**(18):6853-6861.

Igarashi, T., Cosgrove, D., 2001. Adaptive unwrapping for interactive texture painting. Proc. Symp. on Interactive 3D Graphics, p.209-216. https://doi.org/10.1145/364338.364404

Johnson, D., Thompson, T.V., Kaplan, M., *et al.*, 1999. Painting textures with a haptic interface. Proc. IEEE Virtual Reality, p.282-285. https://doi.org/10.1109/VR.1999.756963

Kim, L., Sukhatme, G.S., Desbrun, M., 2003. Haptic editing of decoration and material properties. Proc. 11th Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Systems, p.213-220. https://doi.org/10.1109/HAPTIC.2003.1191280

Kim, L., Sukhatme, G.S., Desbrun, M., 2004. A haptic-rendering technique based on hybrid surface representation. *IEEE Comput. Graph. Appl.*, **24**(2):66-75. https://doi.org/10.1109/MCG.2004.1274064

Larsson, T., Källberg, L., 2013. Fast and robust approximation of smallest enclosing balls in arbitrary dimensions. *Comput. Graph. Forum*, **32**(5):93-101. https://doi.org/10.1111/cgf.12176

Lin, M.C., Baxter, W., Foskey, M., *et al.*, 2002. Haptic interaction for creative processes with simulated media. IEEE Int. Conf. on Robotics and Automation, p.598-604. https://doi.org/10.1109/ROBOT.2002.1013424

Otsuki, M., Tsukadaira, M., Kimura, A., *et al.*, 2009. Development of brush device facilitating painting operation in 2D/3D space. Proc. ICROS-SICE Int. Joint Conf., p.4323-4326

Otsuki, M., Sugihara, K., Kimura, A., *et al.*, 2010. MAI painting brush: an interactive device that realizes the feeling of real painting. Proc. 23rd Annual ACM Symp. on User Interface Software and Technology, p.97-100. https://doi.org/10.1145/1866029.1866045