Hindawi
Scientific Programming
Volume 2023, Article ID 9891427, 1 page
https://doi.org/10.1155/2023/9891427

Retraction

Retracted: Analysis on the Artistic Presentation Effect of 3D Rendering Ink Painting Based on the Evaluation of Deep Learning Model

Scientific Programming

Received 18 July 2023; Accepted 18 July 2023; Published 19 July 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Pan, "Analysis on the Artistic Presentation Effect of 3D Rendering Ink Painting Based on the Evaluation of Deep Learning Model," *Scientific Programming*, vol. 2022, Article ID 9259389, 12 pages, 2022.

Hindawi
Scientific Programming
Volume 2022, Article ID 9259389, 12 pages
https://doi.org/10.1155/2022/9259389

Research Article

Analysis on the Artistic Presentation Effect of 3D Rendering Ink Painting Based on the Evaluation of Deep Learning Model

Yue Pan

School of Art, Southeast University, Nanjing 211189, Jiangsu, China

Correspondence should be addressed to Yue Pan; 230169274@seu.edu.cn

Received 30 March 2022; Revised 28 April 2022; Accepted 10 May 2022; Published 29 May 2022

Academic Editor: Lianhui Li

At present, with the popularity of multimedia, people's appreciation and interest in ink painting art have been deepened. In the research of art works, this paper proposes a rendering method of three-dimensional ink painting effect based on deep learning model, which can simulate the painting skills of ink painting in terms of intensity and distance. Specifically, first construct a two-dimensional ink texture, then map the horizontal component of the texture using the contour information of the model, and finally adjust the ink tone by controlling the coordinates of the vertical component of the texture through the attribute function. Linear attribute function, quadratic attribute function, and attribute function with light source factor are used for simulation. The results show that this method can be used for ink and wash simulation of three-dimensional scene, making the image more layered and more in line with the artist's actual technology.

1. Introduction

The use of computer programs to simulate hand-made artworks is one of the key goals of nonrealistic drawing techniques. To date, researchers in the field of computer graphics have developed many nonrealistic algorithms to simulate Western artworks, such as oil paintings, watercolours, drawings, and prints.

In the last decade, many scholars in China have also started to work on nonrealistic algorithms for traditional Chinese ink painting styles. One approach is to extract the contour lines of the model [1, 2], then stylize the contour lines [3], and then control the stylized parameters to simulate the drawing of the brush. This approach is more complex to implement, and the extraction of contour lines in the graphics space and the visible judgement of stylisation can affect the real-time rendering speed. The second method is based on the traditional cartoon rendering algorithm [4, 5], which controls the ink texture mapping by designing a specific lighting function to achieve the ink style rendering effect, which is simple, realistic, and has good real-time performance.

In 1997, Curtis pioneered the use of Kubelka-Munk (KM) theory in the simulation of watercolours [6]. In that paper, the user selects the reflection values of the pigments against a black and white background so that the K and S values can be calculated, and the final colour output is then calculate using the superimposition and mixing formulae of the KM colour model. In 2003, Rudolf used the same theory for the simulation of crayon paintings based on Curtis, who used KM theory to solve the problems that arose when composing two layers of images. 2004 saw Yamamoto use the KM theory for the simulation of watercolours. In 2004, Yamamoto also used KM colour theory in his study of the conversion of digital images into pencil drawings, but it allowed for user-defined colour transparency, i.e., assuming that the value of the colour parameter would not change on a white background, but setting a parameter to a certain scale on a black background, thus controlling the transparency of the colour. The above are all simulation of Western painting, in the case of ink painting [7], Suarez [8] tried to introduce the KM colour model during the final colouring.

The study of diffusion of different pigments was presented by [9], but did not really address the diffusion and superimposition effects of different pigments. Only a single KM colour model was used to simulate the colour of different pigments. Since there is no substantial improvement of the diffusion model, the simulation of the final effect cannot be satisfactorily obtained.

The literature [10] gives an interactive 3D virtual brush model that uses subdivision surfaces around a skeleton based on a spring system to represent the geometry of the brush; this method is not suitable for generating the bifurcation effect of the brush. [11] proposed calculating the deformation of the brush hair based on the physical properties of the springs based on the theory of elasticity, which is difficult to achieve in real time when the number of hair is large. [12] proposed a model based on the bristle entity, which simulates the bifurcation using replication parameters to generate bristles with the same structure resulting in a lack of close correlation between the bristles and an unrealistic shape of the bifurcated bristles. The quill model constructed by [13] constrains the spring deformation by means of energy minimisation. This method is only applicable to small-scale deformations and works better for simulating wet brushes without bifurcation.

The first paper model of 2D fibre structure was proposed by [14] that plane was proposed to be divided into small regions, and a large number of curves with random orientations were distributed within each region to simulate fibre bundles. Then [15–17] and others proposed their own paper models. This type of paper model achieves better results to some extent, but there are still several fundamental problems: first, it does not consider the effect of different types of rice paper on the diffusion effect and is an uncontrollable paper model that lacks scalability; second, it also does not take into account how to represent the paper texture.

In Pingping [18], based on a summary of previous work, the smallest unit of the paper plane is partitioned into paper cells (one pixel corresponds to multiple paper cells), while multiple adjacent paper cells containing paper fibres form fibre clusters, which are connected by fibre bundles, and the transport of ink and water takes place between the fibre clusters through the fibre bundles.

The first one-dimensional brush depiction models were proposed by [19], where a brush is defined as a one-dimensional array of brush hairs; in a method based on the contour of a brush path [20], the path is described as a number of continuous Bezier or B-spline curves. [21] modeled the brush as consisting of a number of bristles randomly distributed over a circular area, with each bristles drawing a separate curve as the brush moves, and all the curves together forming a single stroke. [22] illustrates the principle of the generation of the black edge phenomenon in ink painting. [23] proposed an automatic cell-based computational model to simulate the movement of water and ink particles. [24] used a fluctuation rule to represent the flow of ink particles. [25] proposed a water absorption force balance algorithm to calculate the diffusion amount of adjacent paper cells. [5] proposed a pseudo-Brownian motion method based on a particle system to simulate the diffusion process, taking into account the effects of water absorption, humidity, evaporation rate, and background texture. [26] complemented the algorithm by proposing an exchange threshold value as an exchange condition and an exchange rate to control the amount of exchange. However both values are static in the diffusion process. [27] used a local equilibrium model and a three-layer paper model to simulate the effect of ink diffusion. This method can simulate simple stacking effects, and Young gave a graph of the stacking effect. [28] proposed a diffusion model with concentration difference as the driving force for diffusion, but did not present a theoretical basis, [29] proposed to simulate the diffusion of ink particles with Fick's first law of steady-state diffusion [30], but in reality the diffusion process is a nonsteady-state process, [31, 32] proposed to use a fluid based on LBE (Lattice Boltzmann Equation) to simulate ink diffusion. The diffusion model is easy to parallelize, has good real-time performance, and can produce very realistic ink and wash wet brush painting effects. However, by treating both water and ink particles as equally sized particles, the particle diffusion model is not a true particularised model, but simply a diffusion model under the rule of concentration correspondence. This method does not allow for differences in the diffusion of multiple pigments.

As can be seen, KM colour models have been chosen to solve the problem of colour setting, both in colour simulation for Western painting and in colour setting for ink painting. The problem of colour calculation is complicated by the diffusion of pigments in ink painting, especially when multiple pigments are diffused together. How to apply the KM colour model's superimposition and mixing formulae through layering based on the characteristics of diffusion is one of the main focuses of this paper. Based on the second type of method, this paper proposes an ink rendering algorithm based on two-dimensional texture mapping, which can simulate the traditional ink painting technique of intensity and distance. This paper uses attribute functions to control the huge changes in the vertical component of ink textures to simulate the shadow and distance techniques of ink painting. Compared with the existing algorithms, the 2D texture mapping of the attribute function provides richer rendering effects. In the ink rendering of the 3D scene, the effects of light and shade and distance are added, which makes the image more layered and closer to the painter's painting skills.

2. Principle of the Algorithm

The main objective of the algorithm in this paper is to achieve a simulation of the intensity and distance technique while obtaining a traditional ink and wash rendering. The difference from traditional 3D ink rendering is the inclusion of the processing and use of 3D depth information. By constructing an attribute function with the 3D depth information as the independent variable for texture mapping, and then controlling the colour tones of the textures to express the changes in the 3D depth information, the effect can be obtained in terms of the intensity of the ink strokes in the near distance and the lightness of the strokes in the far distance. This idea is based on traditional cartoon rendering techniques [6] and the X-TOON system [7].

Bella developed the X-TOON system based on the traditional cartoon rendering algorithm, which extends the

1D cartoon texture to a 2D texture, using the vertical component of the texture to express the colour tone changes and then controlling the texture mapping to obtain a richer cartoon effect.

This paper uses the X-TOON systematic approach to produce two-dimensional ink textures. The hue of the horizontal component of the texture (first dimension) is associated with the model contour edge information, and the texture coordinates are controlled by calculating *N*-*V*, where *N* is the unit normal vector of the spatial surface and *V* is the unit vector of view. The vertical component of the texture (second dimension) is associated with the colour hue (represented in the article by the attribute function *D*), and the texture coordinates are controlled by the attribute function *D*. Each value in the vertical component is associated with its own 1*D* ink texture, so the whole 2*D* texture can be seen as a superposition of 1*D* ink textures of progressively increasing hue. The ink texture generate is shown in Figure 1.

2.1. Chafing Renderer. There is a wide variety of shops in ink and wash work, and the differences in expression between the various crops are large, while the painter is more spontaneous in using this technique it is not possible to propose an algorithm for each of these techniques. It is therefore not possible to propose an algorithm for each chap. The texture map is then drawn based on a two-step texture mapping method.

The regularised chunking set has been obtained after chunking feature analysis $S_k^{\text{texture}} = \{s_i \mid i = 1, 2, 3, \ldots\}$ 1itexturek first determines the number of interpolated textures. Multiple textures are now connected by line blending [11] to form a large mapped texture map. Item N, let chapped textures μ_A and μ_B, is $|N = |\mu_A - \mu_B|/k$ averaged over the H information. The smaller the coefficient, the greater the degree of variation between individual chops. Then k is used to control the degree of jump in the transition between the chapped textures; the larger k is. The smaller N is, and the greater the degree of variation between the individual chappings. The text uses $k = 5$. The interpolated image between A and B is then I_{between}.

$$I_{\text{between}} = \alpha \cdot A + (1 - \alpha). B, \alpha = 1, \frac{N-1}{N}, \frac{N-2}{N}, \ldots, 0. \tag{1}$$

Figure 2 shows the final composite mapping texture after the difference.

2.2. S-O Two-Step Texture Mapping. To optimise the rendering results, the S-O two-step texture mapping method is used to chisel the model. This method was proposed by [13]. The basic idea is that instead of directly texturing the 3D model surface, an intermediate transition surface is introduced and the mapping process is decomposed into a two-step operation: the first step is the mapping from the textured plane to the transition surface, i.e., $T(u, v) \longrightarrow S(x', y', z')$, which becomes the *S* mapping; the first step is the mapping from the transition surface to the 3D model surface, i.e., $S(x', y', z') \longrightarrow O(x, y, z)$, which becomes the *O* mapping. The transition surfaces in *S* mapping are usually simple 3*D* surfaces such as spheres, cylinders, and cubes, as shown in Figure 3.

There are 12 combinations of *S*-mapping and *O*-mapping. In this paper, we take a spherical surface as the transition surface for *S*-mapping and modify the third *O*-mapping method for chaffing.

The sphere of the transition surface is divided into two parts symmetrical along the *x*-plane, and the texture is mapped to the two parts symmetrical to the sphere using a mapping algorithm with an area-equivalence constraint [14]. As in Figure 4, gave a 2D texture coordinate system v, the centre of the texture plane coincides with the origin of the coordinate system at any point $p(u, v)$, which can be expressed in polar coordinates (p, α). Given a three-dimensional coordinate system x, y, z, the centre of a spherical planet of radius R coincides with the origin, and any point Q (x, y, x) on the spherical plane is expressed in spherical coordinates as (R, y). Due to the symmetry, the sphere can be divided into two hemispheres by the xy plane, and the mapping method is the same in both hemispheres.

Let a point *P* on the texture plane be mapped to a point on the hemisphere. In order to effectively reduce texture distortion, an area-equivalence constraint *Q* [4] is used here to construct the mapping relationship between the two points. The area equiproportionality constraint, i.e., the ratio of the area of the texture plane to the area of the corresponding mapped area on the sphere is transversely equal to the constant *K*:

$$K = \frac{\pi \cdot \rho^2}{2\pi R^2 (1 - \cos \phi)}. \tag{2}$$

This gives $\rho = 2\pi R^2 (1 - \cos \phi)$ another $\alpha = \theta$. The texture coordinates can be calculated as

$$\begin{cases} u = \rho \cdot \ \cos \alpha = \sqrt{2KR^2 (1 - \cos \phi)}. \cos \theta, \\ v = \rho \cdot \ \sin \alpha = \sqrt{2KR^2 (1 - \cos \phi)}. \sin \theta. \end{cases} \tag{3}$$

When $10 \leq \phi \leq \pi/2$, the mapping results in the upper hemisphere; when $\pi/2 \leq \phi \leq \pi$, the mapping results in the lower hemisphere.

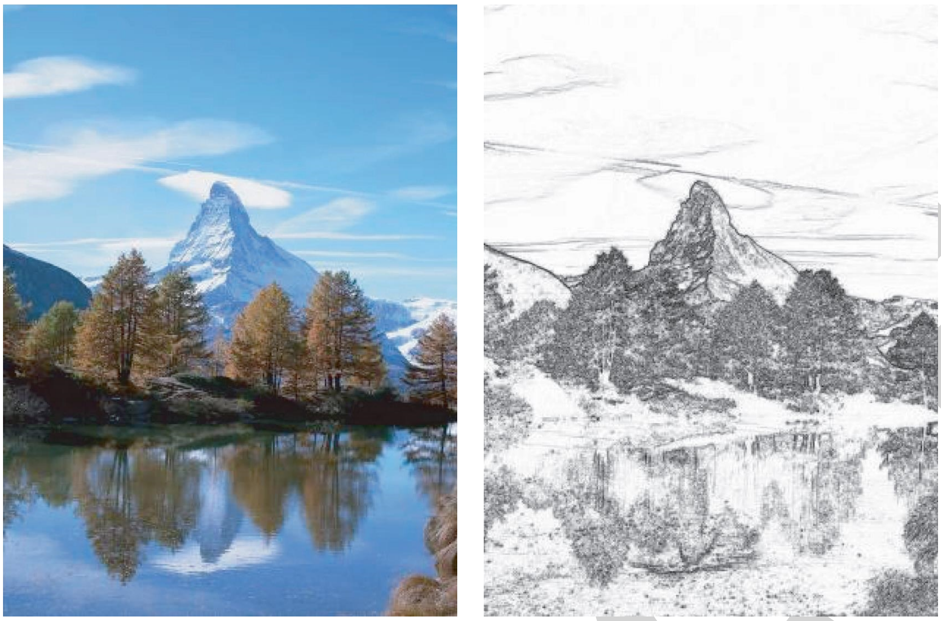
Since the origin of the texture coordinates is generally located in the lower left corner. The above equation is modified:

$$\begin{cases} u = \rho \cdot \ \cos \alpha = 0.5 + \sqrt{2KR^2 (1 - \cos \phi)}. \cos \theta, \\ v = \rho \cdot \ \sin \alpha = 0.5 + \sqrt{2KR^2 (1 - \cos \phi)}. \sin \theta. \end{cases} \tag{4}$$

Experimentally, when using a mapped texture map of 512 × 512 size, take $R = 1.0$, $K = 0.12$.

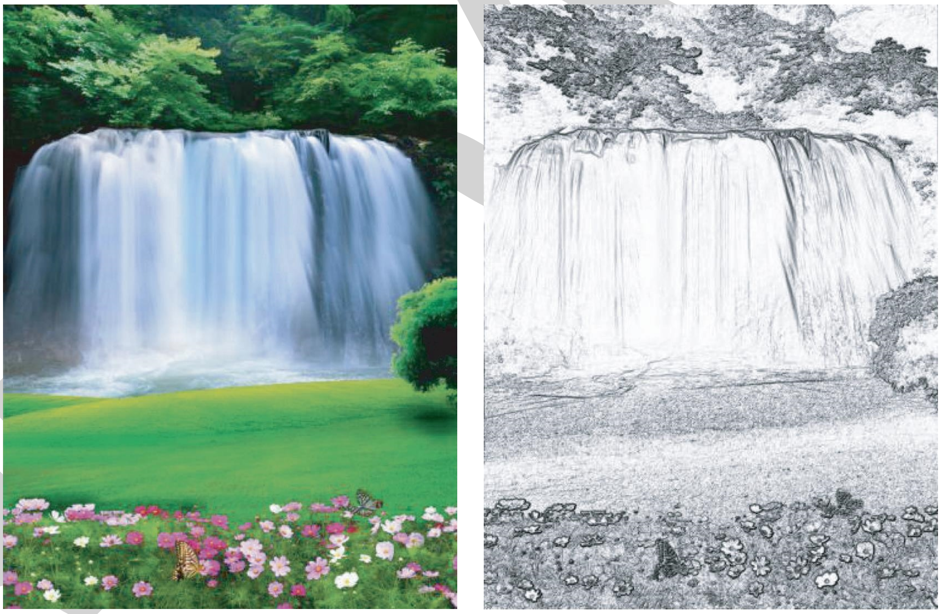
The third mapping method is used for the O mapping, i.e., the texture at the vertex $V(x, y, z)$ on the surface of the model, taking the ray formed by the line from the centre of the object to this vertex, and the texture at the intersection with the transition surface $V'(x', y', z')$.

The centre point of a model is usually complex, the algorithm for finding the centre point is as follows: let the



(a) (b)

Figure 1: Ink painting with two texture.



(a) (b)

Figure 2: Synthetic results after image interpolation. (a) Lotus leaf chap and (b) phi chap.

coordinates of a vertex in the *xyz* three-dimensional coordinate space be $P(v_x, v_y, v_z)$, calculate the sum of all the vertex coordinate vector values, and then find the average of all the coordinate vectors, then get the location of the object's centre point coordinates $P'(c_x, c_y, c_z)$. By the centre point and a vertex direction vector is $PP' = (v_x - c_x, v_y - c_y, v_z - c_z)$, and then determine it is in the sphere and thus the coordinates of *uv* are calculated from the above equation.

3. Ink and Watercolor Rendering

The horizontal component of the texture is first mapped using the contour area information to sketch the shape of the model, and then the vertical component of the texture is mapped using the attribute functions to control the colour tone information to achieve the ink and wash rendering effect.

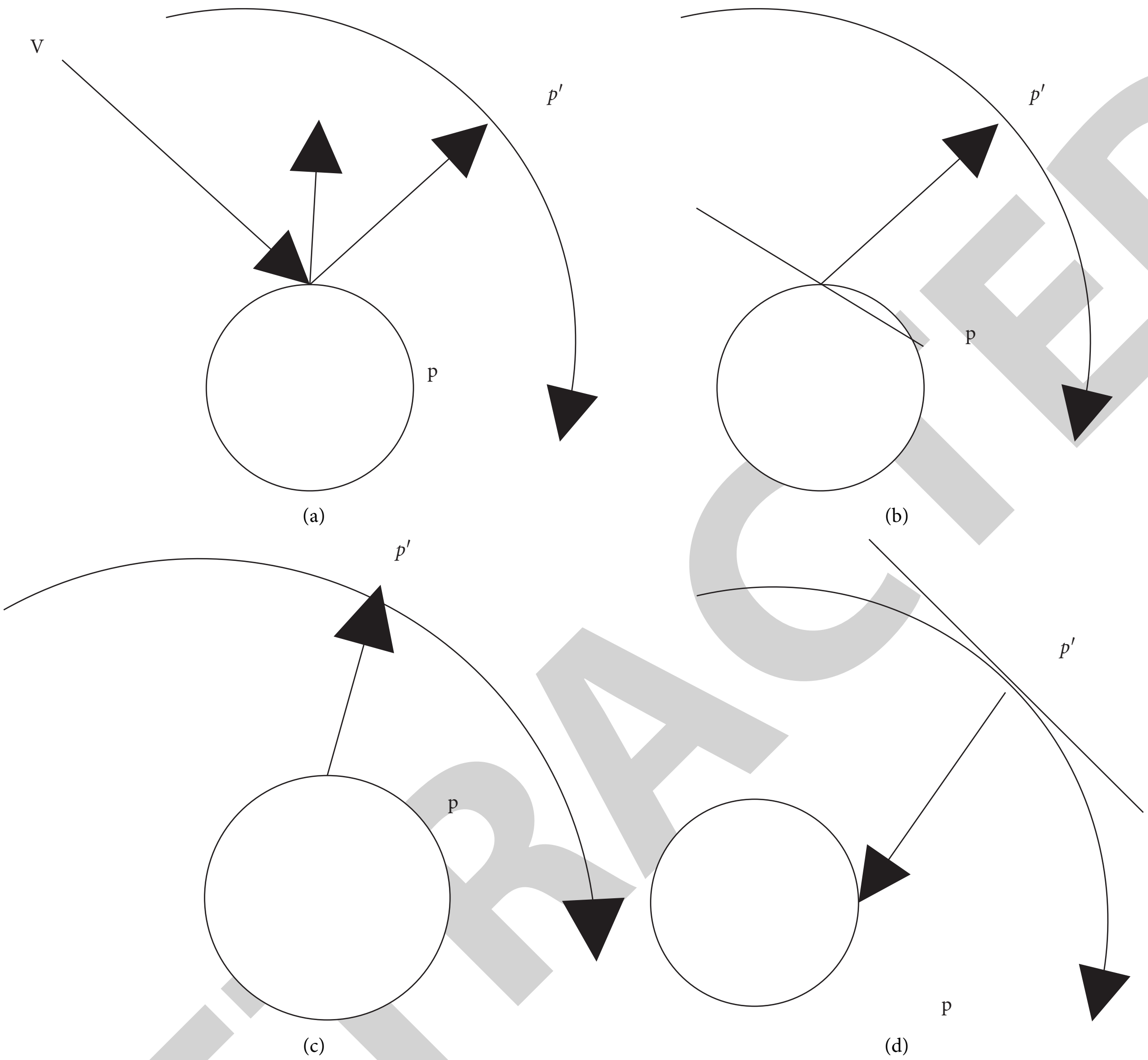


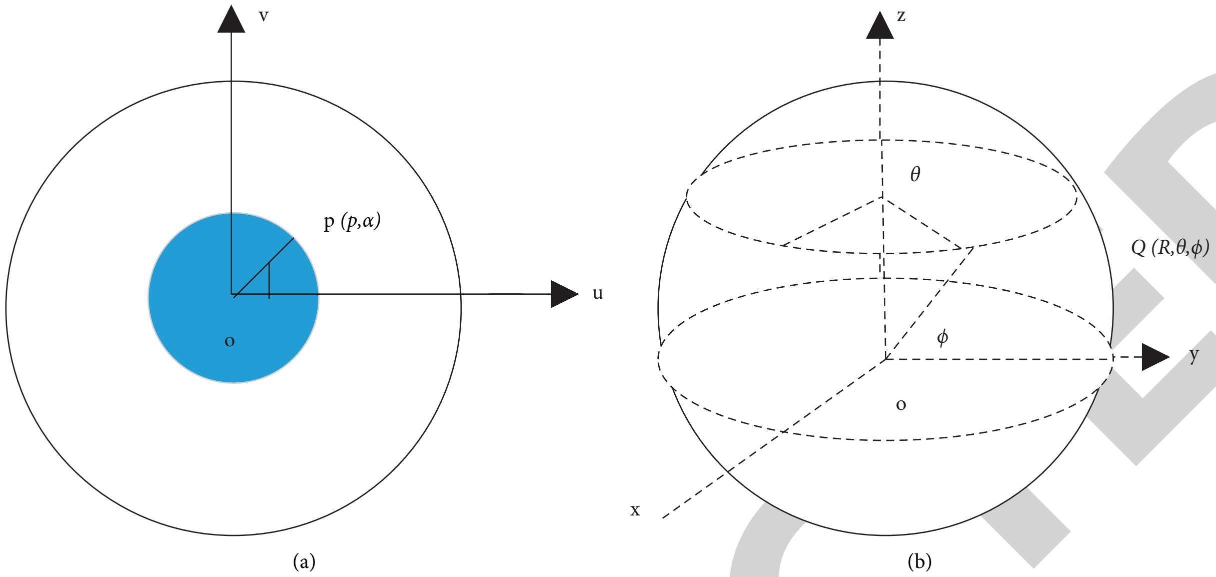
Figure 3: Four ways of O mapping.

3.1. Drawing Model Shapes. In ink painting, when the artist uses ink to paint a scene, it often spreads and soaks through the rice paper, giving the brushstrokes that depict the outline of the object in a certain area. For this reason, instead of extracting single-pixel contour lines from the model, the shape of the model is outlined by mapping the texture around the contour lines to a certain area.

The simulation process is implemented in the OpenGL shading language. In the vertex shading, the absolute value a of the dot product of the unit normal vector and the unit apparently vector is calculated separately for each vertex, and a is passed to the pixel shading as a mutable variable. In the pixel shading, an is used to control the texture coordinates of the horizontal component of the texture. In order to achieve the effect of white space in an ink painting, a threshold λ can be set so that when $a>\lambda$, the pixels of the slice are set to the colour of the white space in the ink texture. The larger value of the threshold λ, the smaller the white area. Figure 5 shows the rendering effect on a fixed vertical component texture coordinates. With λ values of 0.5 in Figure 5(a) and 0.7 in Figure 5(b).

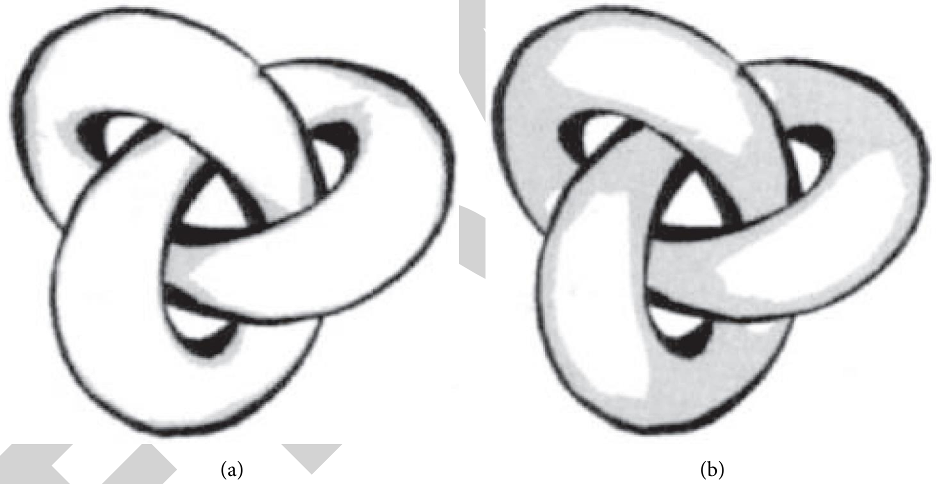
3.2. Control of Colour Tone Information. This step performs the texture mapping of the vertical component and is the heart of the algorithm. In order to simulate the fade technique, elements closer to the point of view are assigned textures with stronger tone. In contrast, elements that are further away from the viewpoint are assigned lighter textures.

To do this, an attribute function is designed for the vertical component texture mapping, starting with the depth information in 3D space as the independent variable. For the selection of depth information, two methods can be used: (i) the Euclidean distance from the 3D point to the viewpoint is used as the depth information (e.g., Figure 6(a)); (ii) the distance from the 3D point to the viewpoint along the focus axis is used as the depth information (e.g., Figure 6(b). The



(a) (b)

Figure 4: Schematic diagram of the spherical S mapping.



(a) (b)

Figure 5: Ink rendering with fixed vertical component texture coordinates.

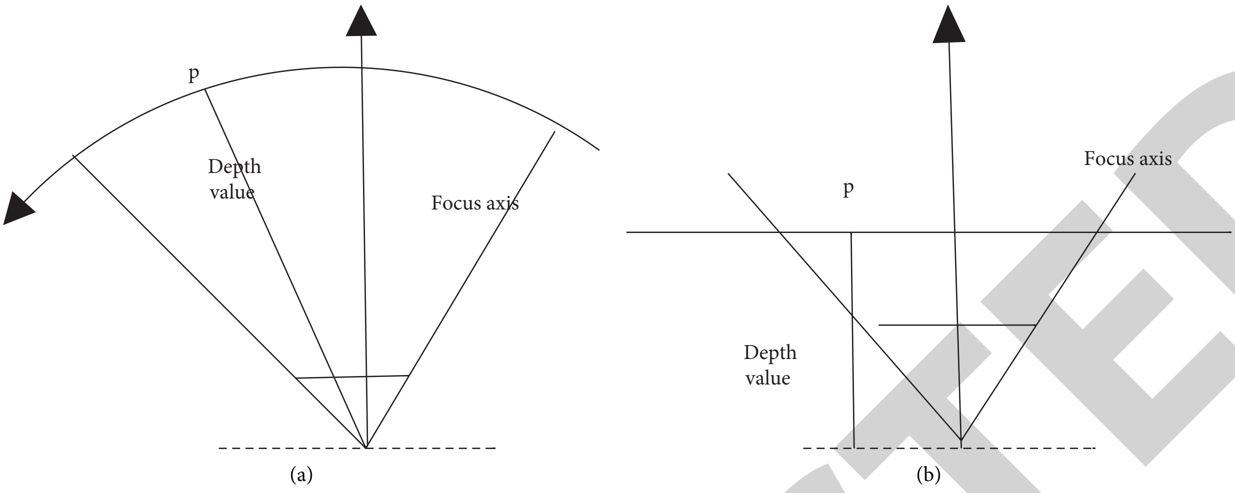
second calculation method was found to be better for simulating the effect of ink and wash painting in terms of intensity and distance.

For the attribute function D, the following constraints are imposed: ① the range of D is between 0 and 1, thus corresponding to the range of texture coordinates; ② set the threshold $d_{\min}$, when the distance from the point on the model to the viewpoint is less than or equal to $d_{\min}$, the value of D is 1, and when the distance is greater than $d_{\min}$, the value of D gradually decreases; ③ set the threshold $d_{\max} = r \times d_{\min}$, when the distance from the model point to the viewpoint is equal to when the distance from the model point to the viewpoint is equal to $d_{\max}$, the value of D is 0, where the value of r is greater than 1.

Different types of functions can be constructed to meet the above requirements. Line attribute functions, quadratic attribute functions, and attribute functions that take into account the effects of illumination are designed to simulate the ink effect in 3D models. The simplest method is to use linear property functions:

$$\begin{cases} D = 1, \quad d \leq d_{\min}, \\ D = \dfrac{1}{((1-r) \times d_{\min} \times d + r/(1-r))}, \quad d \geq d_{\min}, \end{cases} \tag{5}$$

where d is the current depth property value of the 3D point and r and $d_{\min}$ are adjustable parameters. In the slice shader, the result is calculated as the coordinate value of the vertical component of the ink texture, and the function value decreases linearly when the current depth value of the model exceeds the threshold $d_{\min}$. Using the point of view as a reference, the larger vertical component texture coordinates are taken in the near area, corresponding to the darker tones



(a) (b)

Figure 6: Calculation of depth attributes.

in the ink texture, and the smaller vertical component texture coordinates are taken in the far area, corresponding to the lighter tones in the ink texture, thus obtaining the effect of intensity and distance.

If you want a more intense contrast, you can use the secondary attribute function:

$$\begin{cases} D = 1, \quad d \leq d_{\min}, \\ D = \dfrac{\left(d - r \times d_{\min}\right)^2}{\left(\left(1 - r\right)^2 \times {d_{\min}}^2\right)}, \quad d \geq d_{\min}. \end{cases} \tag{6}$$

The light source direction is introduced as a weighting factor to influence the attribute function, which can obtain the effect of the intensity and distance of the white left in the highlights of the illuminated surface and the thick ink of the shadows on the backlit surface. The direction factor constructed in the experiment is $(N - L)\gamma$, where N is the unit normal vector of 3D points, L is the normalised light source direction vector, and γ is the light factor. The attribute functions are

$$\begin{cases} D = 1 - \log 0^{d/d_{\min}}, \\ D = \alpha \times \dfrac{\left(d - r \times d_{\min}\right)^2}{\left(\left(1 - r\right)^2 \times {d_{\min}}^2\right)} + (1 - \alpha) \times (N.L)^{\gamma}, \ (N.L)^{\gamma} \leq \beta, \end{cases} \tag{7}$$

where α is a weighting factor that controls the weight of light and depth information on the hue and β is an adjustable parameter that controls the area of shadows.

4. Simulation of Ink and Wash Painting Materials and Their Physical Processes

Ink painting is a complex process due to the special nature of the painting material and the use of various painting and calligraphy techniques. The study is based on a simulation of the natural medium and its physical processes. The simulation method of simulating the natural medium and its physical process is based on the mechanism of ink painting, and the main drawing tools and materials such as brushes, rice paper, ink, and pigments are studied. The method is based on the mechanism of ink painting, modelling the main drawing tools and materials such as brushes, rice paper, ink and pigments, and simulating their inherent physical properties and dynamic interaction behaviour. The aim is to simulate realistic ink and wash effects. The research results can be broadly divided into three parts: Xuan paper. The simulation of Xuan paper, the simulation of brushes, the simulation of ink diffusion, and the simulation of colour modelling.

4.1. Xuan Paper. Xuan paper is in contact with the brush on the one hand, accepting the ink from the bristles, and on the other hand acting as a carrier for the ink and wash that eventually becomes the painting. The modelling of the paper is therefore important, as it is directly related to the quality of the ink and wash effect simulation. The most important aspect of modelling rice paper is its fibre structure and its role in guiding and blocking the diffusion of ink and water. The advantage of this method is that the resolution is improved, and the irregular shape of the fibre masses will achieve better results on the boundary, as shown in Figure 7, but at the same time, the randomness of the fibre mass composition and the irregularity of the shape make it more difficult to describe and process the ink in it, making it difficult to use graphics hardware resources.

4.2. Hairbrush. With the development of NPR, virtual brush research has attracted more and more researchers. The 2*D* brushstroke model approach [13, 30], although able to simulate many of the effects of ink painting, reduces the freedom of the artist to create because of the overemphasis on the final simulation effect, reducing the unique writing effect of ink painting, as in Figure 8. The authors of [10] use a parametric gradient “ raindrops” to simulate the contact area

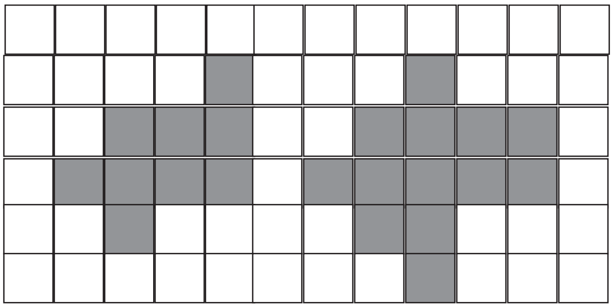


Figure 7: Model of the remaining paper fibre mass.

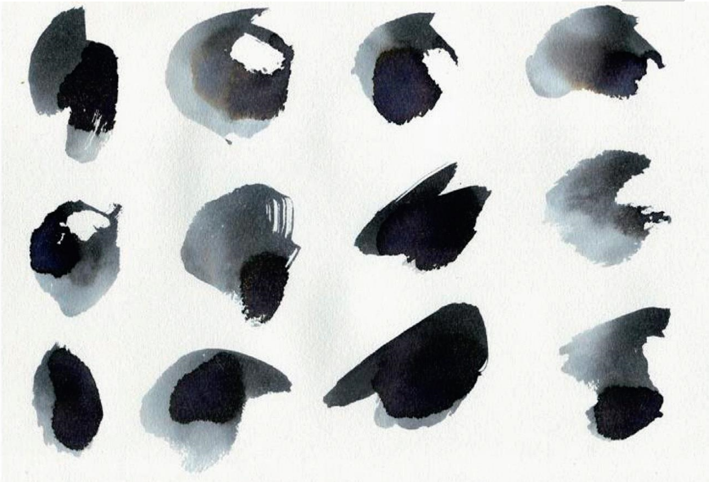


Figure 8: Schematic diagram of the two-dimensional brush stroke model.

between the paper surface and the brush, simplifying the parameter settings and operations, but with a single brush stroke, many artistic effects cannot be realized.

The use of alpha map mapping on the hair bundle surface after bifurcation limits the richness of the variation, as shown in Figure 9.

The focus of the brush simulation is to simulate the interaction behaviour between the brush, the rice paper, and the ink. Most of the above studies have ignored the effect of the structure of the rice paper on the ink and water transfer process. There is a need to reduce the computational complexity to ensure real time. There is also much room for improvement in describing the behaviour of the brush accurately and efficiently. [3] proposed a practical A practical 3D brush modelling system is proposed in [3] to accurately simulate the geometry of the brush and its dynamic interaction behaviour such as movement and bending. The method improves computational efficiency while effectively simulating the dynamic changes of the brush during the artistic creation process, as shown in Figure 10.

4.3. Diffusion Processes. The special feature of ink painting is that the brush bristles contain a large amount of water, and the pigment diffuses within the rice paper by the action of the water. The study of diffusion is indispensable in order to fundamentally simulate the effect of drawing ink paintings. [15] proposes to analyse ink and wash from the viewpoint of

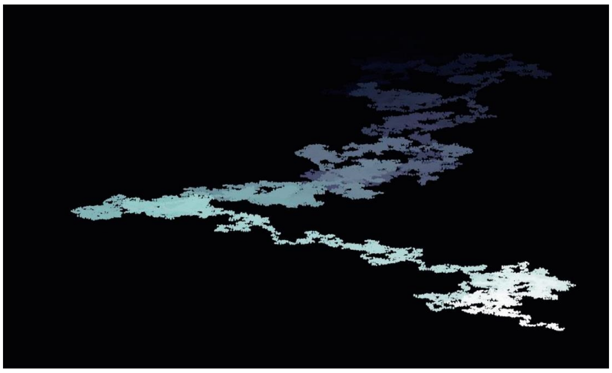


Figure 9: Brush surface mapping.

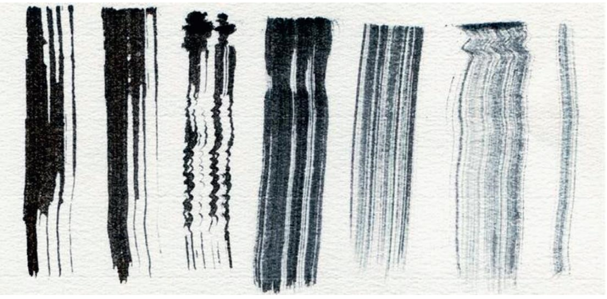
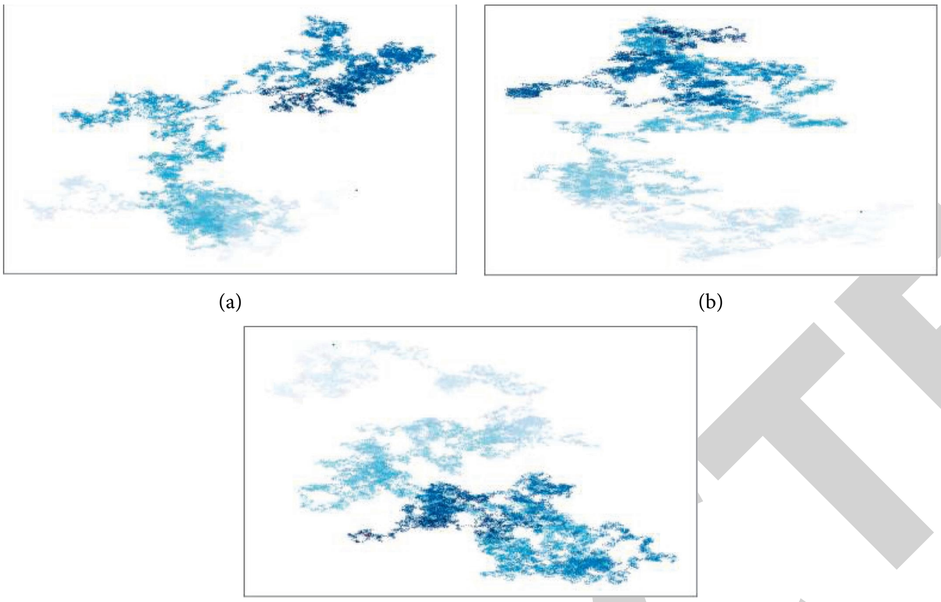


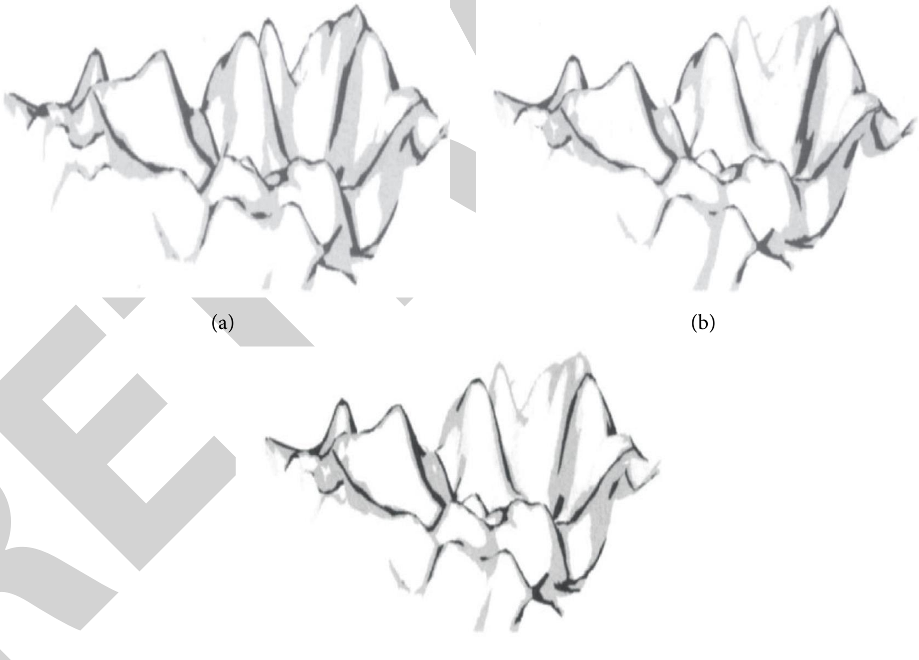
Figure 10: Potential energy-threshold controlled pen movement.


(a)

(b)

(c)

Figure 11: Linearly varying ink and wash effects of shade and distance (a-c).


(a)

(b)

(c)

Figure 12: Comparison of three ink rendering effects. (a) Ink rendering without the depth attribute function (b) Ink rendering with linear attribute functions. (c) Ink rendering with quadratic attribute function.

particle systems, they are all composed of tiny particles that satisfy the condition of doing Brownian motion, and they undergo a complex movement in the fibre structure of the paper until the ink dries, finally forming an ink painting work. Throughout the process, this includes the diffusion of ink particles, the collision of particles with each other, the deposition of ink particles, and the evaporation of water particles, among other forms. The task of ink diffusion modelling is to study the mechanism of movement and simulates these movement patterns to obtain the final ink distribution information and finally display the output.

The diffusion of ink and pigments is the most complex part of the ink painting simulation. As can be seen from the above analysis, existing algorithms are not yet able to reproduce this process comprehensively, accurately and in real time, and previous simulations of diffusion

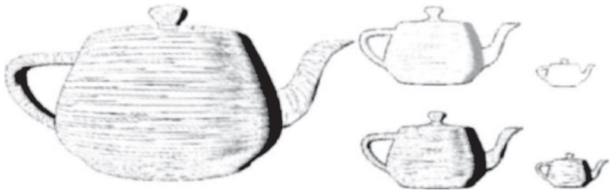
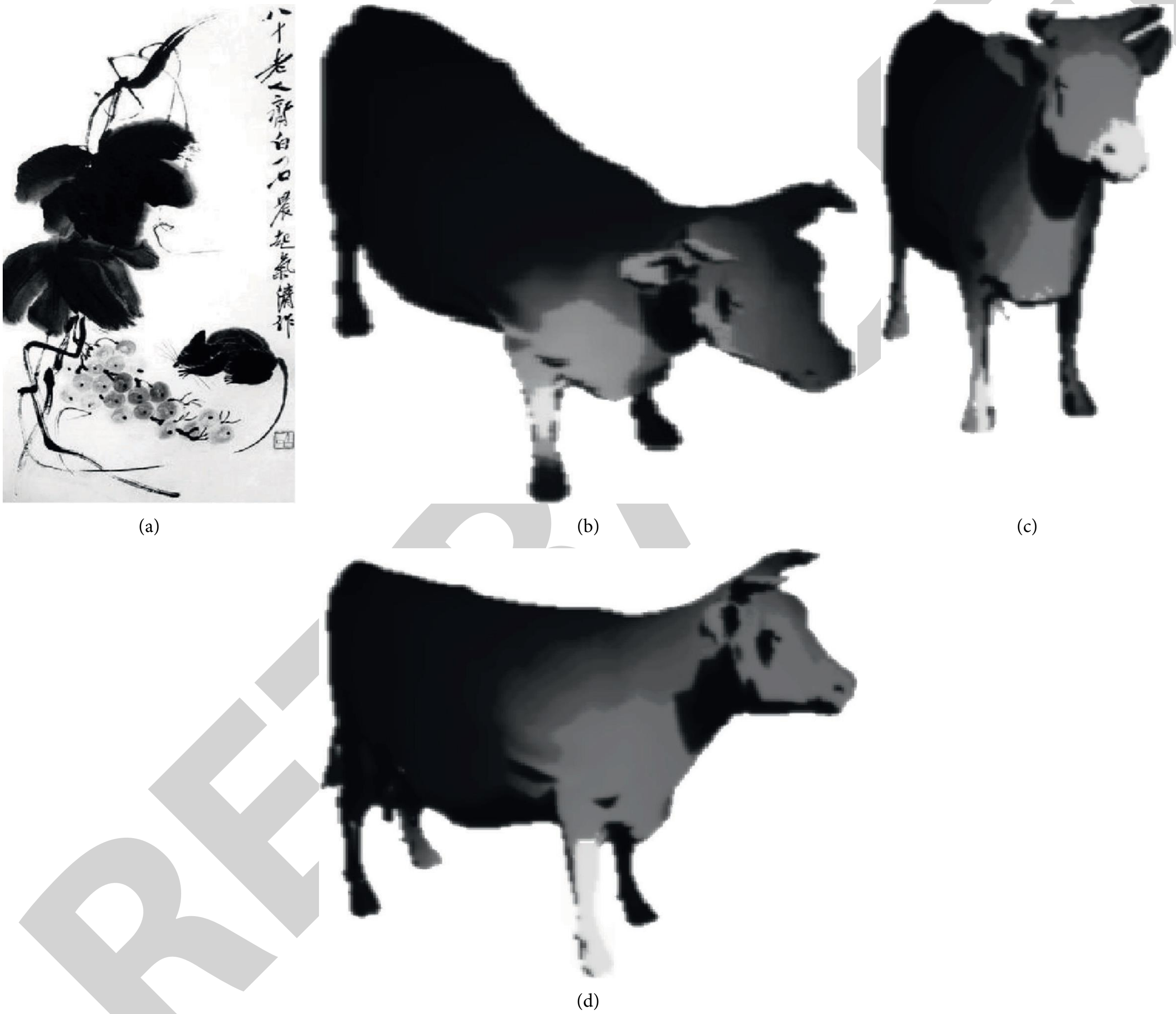


Figure 13: Effect of pencil drawing with abstract detail.



(a)

(b)

(c)

(d)

Figure 14: Qi Baishi's Mouse Grapes (a) and the results of this rendering (b)–(d). (a) The Mouse Grapes. (b) Rendering results of this paper 1. (c) Rendering result of this paper 2. (d) Rendering result of this paper 3.

superimpositions are limited to two aspects: first, particle-based diffusion has been proposed, but none of them are particle-based diffusion in the true sense. Second, they are limited to the diffusion of a single ink particle and do not include the diffusion of all the essential properties of the various pigments. Third, there is no in-depth study of the reality of the superposition of different pigments.

5. Results and Discussion

5.1. Algorithm Effect. The algorithm presented above has been implemented in VC+ + 6.0 and the OpenGL graphics engine programming environment and has been tested on a PC with a 1.8 G Hz CPU, DDR 1G RAM, and an ATIRHD 2400 graphics card.

In the experiments, the equations were verified separately as property functions, and the same parameters and sampling points were set for both experiments in order to compare the two. Figure 11 shows the effect of linking with increasing values of depth d. Figure 11(a) shows the effect when $d < d_{\min}$, which is closer to the point of view and the ink is thicker; 11(b) and 11(c) show the effect when $d > d_{\min}$ and d is gradually increased, which shows that the ink is getting lighter. Figure 12 shows a comparison between the traditional ink simulation without the depth attribute function, the ink simulation with the linear attribute function and the ink simulation with the quadratic attribute function when drawing the mountain model, with the parameter $d_{\min}$ at 100 and r at 10. The traditional 3D ink rendering algorithm does not consider the effect of the depth attribute on the rendering effect, so no matter how the depth value of the model vertices is changed, it is difficult to produce a strong or light. The results obtained by the algorithm in this paper vary with the depth value, as shown in Figures 12(b) and 12(c), with a stronger sense of hierarchy using the secondary attribute function. The reason for the different degrees of intensity is the variation in the value of the attribute function.

From the experimental results, it can be seen that the algorithm of this paper can be used to simulate the technique of intensity and distance better, and by changing the attribute function model and the relevant parameters of the function, the rendering effect with different contrast of intensity and distance can be obtained, which is a simple and practical method and provides an effective way for the 3D ink rendering of large-scale scenes.

Other attempts have been made to apply the attribute function. In artwork, there is often a technique of detail abstraction, where the details of something are detailed in the near distance, and only the general outline is drawn in the far distance, while the internal details are sketchy or not even portrayed. It is worth noting that the detail abstraction simulation here is different from the LOD technique, which only uses property functions and texture mapping techniques to obtain a sensory abstraction effect, without any processing of the 3D model mesh.

The traditional pencil drawing simulation requires a set of pencil drawing textures, and then uses the lighting function to control the weight of the different textures, and finally mixes the textures. The depth value is also taken into account, and the depth value is calculated by adding a white texture to the original pencil texture, and then using a linear property function to control the weighting factor of the texture, the final result is shown in Figure 13. It can be seen that the method can be used to simulate a detailed abstraction technique.

5.2. Drawing Effect. Figure 14 shows work by Qi Baishi when he was 80 years old, which mainly uses the technique of writing, using the intensity of ink colour to represent the object. From the experimental results, it can be seen that for different styles of input work, the algorithm is able to pass the basic stylistic features to the 3D model and render the ink accordingly, resulting in output images of similar styles.

6. Conclusions

The paper uses attribute functions to control the huge changes of the vertical components of the ink textures to simulate the ink painting technique of shade and distance. Compared with the existing algorithms, the 2D texture mapping by the attribute function provides a richer rendering effect, and in the ink rendering of 3D scenes, the addition of the effect of shade and distance makes the image more layered and closer to the painter's painting technique.

In order to provide a richer surface detail of the 3D model, the radial curvature of the model surface can be calculated, which is one of the directions for the future work.

Data Availability

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

Acknowledgments

The authors would like to thank Qi Baishi for provides the painting.

References

[1] Z.-z. Zhang, Z.-j. Wu, T. Chen, and T. Lei, "Analysis on intensity zonation in thick loess region," *Journal of Shanghai Jiaotong University*, vol. 18, no. 6, pp. 724–728, 2013.

[2] J. Lee, "Diffusion rendering of black ink paintings using new paper and ink models," *Computers & Graphics*, vol. 25, no. 2, pp. 295–308, 2001.

[3] N. S.-H. Chu and C.-L. Tai, "MoXi," *ACM SIGGRAPH 2005 Sketches on - SIGGRAPH '05*, vol. 24, no. 3, pp. 504–511, 2005.

[4] C. H. Cao, Y. N. Tang, and D. Y. Huang, "IIBE: An Improved Identity-Based Encryption Algorithm for WSN Security," *Security and Communication Networks*, vol. 2021, Article ID 8527068, 8 pages, 2021.

[5] F. Li and S. Li, "real-time rendering of 3D animal models in Chinese ink painting style," in *Proceedings of the 2020 International Conference on Culture-Oriented Science & Technology (ICCST)*, pp. 284–287, IEEE, Beijing, China, October 2020.

[6] M. Kaplan, B. Gooch, and E. Cohen, "Interactive artistic rendering," in *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering*, pp. 67–74, Annecy France, June 2000.

[7] D. Wu, C. Zhang, L. Ji, R. Ran, H. Wu, and Y. Xu, "Forest fire recognition based on feature extraction from multi-view images," *Traitement du Signal*, vol. 38, no. 3, pp. 775–783, 2021.

[8] J. Suarez, F. Belhadj, and V. Boyer, "Real-time 3D rendering with hatching," *The Visual Computer*, vol. 33, no. 10, pp. 1319–1334, 2017.

[9] W. Qian, D. Xu, K. Yue, Z. Guan, Y. Pu, and Y. Shi, "Gourd pyrography art simulating based on non-photorealistic rendering," *Multimedia Tools and Applications*, vol. 76, no. 13, Article ID 14559, 2017.

[10] A. Radwan, K. M. S. Huq, S. Mumtaz, K. F. Tsang, and J. Rodriguez, “Low-cost on-demand C-RAN based mobile small-cells,” *IEEE Access*, vol. 4, pp. 2331–2339, 2016.

[11] A. R. Timming, D. Nickson, D. Re, and D. Perrett, “What do you think of my ink? Assessing the effects of body art on employment chances,” *Human Resource Management*, vol. 56, no. 1, pp. 133–149, 2017.

[12] J. Wang, “The integration of ink art and modern advertising design--analysis of the effect of ink dots in advertising design in the teaching of art history,” in *Proceedings of the 8th International Conference on Education, Management, Information and Management Society (EMIM 2018)*, pp. 793–796, Atlantis Press, Shenyang, China, August 2018.

[13] L. Wang, C. Zhang, and Q. Chen, “A Communication Strategy of Proactive Nodes Based on Loop Theorem in Wireless Sensor Networks,” in *Proceedings of the 2018 Ninth International Conference on Intelligent Control and Information Processing (ICICIP)*, pp. 160–167, IEEE, Wanzhou, China, November 2018.

[14] S. Du, W. Li, H. Wu, P. Y. Abel Chuang, M. Pan, and P.-C. Sui, “Effects of ionomer and dispersion methods on rheological behavior of proton exchange membrane fuel cell catalyst layer ink,” *International Journal of Hydrogen Energy*, vol. 45, no. 53, Article ID 29430, 2020.

[15] A. Al Galil, F. Mohammed, S. Chandra, P. Zambare, F. A. Al-Mekhlafi, and A. Al-Keridis, “Effect of dimethoate on the developmental rate of forensic importance Calliphoridae flies,” *Saudi Journal of Biological Sciences*, vol. 28, no. 2, pp. 1267–1271, 2021.

[16] T. Van Cleve, G. Wang, M. Mooney et al., “Tailoring electrode microstructure via ink content to enable improved rated power performance for platinum cobalt/high surface area carbon based polymer electrolyte fuel cells,” *Journal of Power Sources*, vol. 482, Article ID 228889, 2021.

[17] H. Li, D. Zeng, L. Chen, Q. Chen, M. Wang, and C. Zhang, “Immune Multipath Reliable Transmission with Fault Tolerance in Wireless Sensor Networks,” in *Proceedings of the International Conference on Bio-Inspired Computing: Theories and Applications*, pp. 513–517, Springer, Penang, Malaysia, September 2016.

[18] Y. Pingping, “The concept of beauty in painting by Qi Baishi,” *Laplage em Revista*, vol. 7, no. Extra-A, pp. 265–273, 2021.

[19] T. Hoi-chiu, “The art of Chinese painting,” in *The Routledge Encyclopedia of Traditional Chinese Culture*, pp. 210–220, Routledge, Oxfordshire, UK, 2019.

[20] L. Yang, T. Xu, J. Du, H. Zhang, and E. Wu, “Brushwork master: Chinese ink painting synthesis for animating brushwork process,” *Computer Animations and Virtual Worlds*, vol. 31, no. 4-5, p. e1949, 2020.

[21] Y. Chen, “Development and implementation strategy of students’ concept innovation in Chinese painting teaching based on big data,” in *Journal of Physics: Conference Series*, vol. 1992, no. 4, IOP Publishing, Article ID 042063, 2021.

[22] X. Tang, P. Zhang, J. Du, and Z. Xu, “Painting and calligraphy identification method based on hyperspectral imaging and convolution neural network,” *Spectroscopy Letters*, vol. 54, no. 9, pp. 645–664, 2021.

[23] F. Wang and X. Zheng, “The comparison of the hedonic, repeat sales, and hybrid models: evidence from the Chinese paintings market,” *Cogent Economics & Finance*, vol. 6, no. 1, Article ID 1443372, 2018.

[24] B. Zhang, J. Chen, X. Huang, and W. Xu, “Chinese flower and bird painting: a new form of art therapy for depression,” *Sage Open*, vol. 11, no. 2, Article ID 215824402110021, 2021.

[25] L.-T. Chiu, “A new page of literati painting from Singapore and Malaysia: a study of chen wen hsi and chung chen sun,” *Translocal Chinese: East Asian Perspectives*, vol. 15, no. 1, pp. 93–130, 2021.

[26] P. Jing, “The influence of sichuan opera art on chang dai-chien’s works,” in *Proceedings of the 2nd International Conference on Contemporary Education, Social Sciences and Humanities (ICCESSH 2017)*, pp. 457–460, Atlantis Press, Moscow, Russia, June 2017.

[27] Y. Zheng, Y. Chen, G. Fei, J. Dorsey, and E. Wu, “Simulation of textile stains,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 7, pp. 2471–2481, 2019.

[28] C. Luan, G. Liu, Y. Liu et al., “Structure effects of 2D materials on α-nickel hydroxide for oxygen evolution reaction,” *ACS Nano*, vol. 12, no. 4, pp. 3875–3885, 2018.

[29] M. T. Sajjad, A. Ruseckas, and I. D. W. Samuel, “Enhancing exciton diffusion length provides new opportunities for organic photovoltaics,” *Matter*, vol. 3, no. 2, pp. 341–354, 2020.

[30] S. Jeon, S. K. Sung, E. H. Jang et al., “Multilayer metal-oxide-metal nanopatterns via nanoimprint and strip-off for multispectral resonance,” *Applied Surface Science*, vol. 428, pp. 280–288, 2018.

[31] H. Arai, M. Irita, and N. Katayama, “Investigation of multilayered catalyst layers for polymer electrolyte fuel cells by electrospray deposition,” *ECS Meeting Abstracts*, vol. MA2020-02, no. 34, p. 2220, 2020.

[32] Z. Lu, “Digital Image Art Style Transfer Algorithm and Simulation Based on Deep Learning Model,” *Scientific Programming*, vol. 2022, Article ID 8409459, 9 pages, 2022.