Research Article

Field Coupling-Based Image Filter for Sand Painting Stylization

Tao Wu ^{1,2}, Xia Chen,¹ and Liqiong Lu³

¹School of Information Engineering, Lingnan Normal University, Zhanjiang 524048, China

²Guangdong Engineering and Technological Development Center for E-Learning, Zhanjiang 524048, China

³School of Printing and Packaging, Wuhan University, Wuhan 430079, China

Correspondence should be addressed to Tao Wu; wutao@whu.edu.cn

Received 17 May 2017; Revised 13 November 2017; Accepted 23 November 2017; Published 4 March 2018

Academic Editor: Federica Caselli

Copyright © 2018 Tao Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Sand painting is a very new and popular performance art technique, and it has important meaning for the artistic creation. Computer-aided sand painting is now in the ascendant, whose fully automated mechanism is still an attractive challenge. Image data field, inspired from short-range nuclear force in physics, is introduced into sand painting stylization. With preserving edges and features, a fully automatic image filter pipeline based on image field coupling is proposed, which consists of three main steps, including image data field generating, multiscale field coupling, and additional details optimizing. The visual comparisons indicate that the proposed pipeline and image filter are feasible and effective, and sand painting-like artistic images can be produced by using multiscale field coupling. The qualities of various rendering images are also investigated from the perspective of feature similarity measures. The quantitative and qualitative results suggest that the field coupling-based filter is appropriate for sand painting stylization, and the time cost is satisfied for practice applications. Generally, the proposed method provides a reference for automatic creation of computer-aided sand painting art.

1. Introduction

Nonphotorealistic rendering (NPR for short) is an area of computer graphics and image processing that focuses on enabling a wide variety of expressive styles for digital art, which is inspired by artistic styles such as painting, drawing, technical illustration, and animated cartoons. Over the last decade, there has been continuous, even growing, interest in using various approaches to solve this problem. The key techniques focus on how to create a desired artistic style other than realism from two-dimensional content of traditional, photorealism digital image, referred to image-based artistic rendering (IBAR). A recent survey over IBAR methods is provided in [1]. Nonetheless, designing new artistic styles remains a challenging research topic, since there are so many types of art in the real world, such as watercolor, oil, pen-and-ink, pencil drawing, and pointillism [2], as well as paper-cut, embroidery, and pyrography from China [3].

On the other hand, as a new form of art, sand animation or sand painting, can tell stories by creating animated images with sand, which was pioneered by Caroline Leaf and more

recently by Ferenc Cakó. The process begins by applying sand to a lighted surface, after which images are rendered on the surface by drawing lines and figures with bare hands. The essential step of sand painting is to create a series of sand images from raw materials (generally photograph or videos) and ensure the originality of performing arts. Sand painting has increasingly attracted audiences and artists because of its innovative and expressive graphic style. But sand painting performance spaces are very difficult to set up and maintain, which prevents many novices from getting started. In addition, it is also quiet difficult to acquire sand control skills using the basic production equipment, and people who are nonnative to this technique are generally unable to use sand painting.

To solve this problem in computer graphics and image processing, computer-based sand art technique must be developed to create sand painting images from photograph images, that is, a new digital artistic form inspired by sand animation. In fact, sand painting has also surfaced in NPR in recent years [4–10]. Unfortunately, the existing methods mainly simulate sand art using mobile devices in

FIGURE 1: Sand image examples by Photoshop and real performance: (a) image by Photoshop from Baidu Tieba, (b) image by Photoshop from Baidu Jingyan, and (c) and (d) two images by real performance.

an interactive manner but consider none of full-automation, except for three explorative works [6, 9, 10].

As sand painting is gaining popularity; an imperative and urgent demand for automatically creating this style-like effect is also rising. The sand painting lovers try their best to achieve this goal. As shown in Figures 1(a) and 1(b), there are two result images generated by Internet users using Adobe Photoshop, which are publicly available in Baidu Tieba (<https://tieba.baidu.com/>) and downloaded from Baidu Jingyan (<https://jingyan.baidu.com/>) under Creative Commons licenses. However, these results are really not as good as those by real performance (see Figures 1(c) and 1(d)). Thus, IBAR for sand painting is still an open issue, and sustained effort is required for it. In this paper, we confine our technique to converting an image into a sand painting-like effect, which makes the users seem to view a live performance of a sand art on a screen.

In these contexts, we present an IBAR method for generating artistic image with a sand painting style, and our intentions or research questions are the following four aspects: (1) How different types of fields and field coupling have an influence on the various features of an input image? We generated various image data field using different image features, and then field coupling is implemented in order to create a sand painting filter. (2) How could the sand painting-like effects be automatically produced with a field coupling-based algorithm? We proposed a field coupling-based pipeline, which used an original image as input and produced sand painting-like effects. In addition, multiscale field coupling is introduced with different fields of multiresolution for

improving the artistic quality. (3) What effects would the sand painting images have? The visual comparisons and the aesthetic qualities of various rendering images were validated, as well as time performance analysis. (4) What is a suitable parameter configuration for image-guided sand painting filter using field coupling? We investigated the effects of different parameters for the algorithm performance and provided the default parameters.

The rest of the paper is organized as follows. In Section 2, we provide a brief introduction to sand art and NPR. Section 3 presents an introductory explanation of data field, as well as image data field. In Section 4, we detail the implementation of the proposed method, and the algorithm analysis is as well presented, such as parameter setup and computational complexity. Section 5 shows the experimental results and discussion. Finally, in Section 6, we make the conclusions and suggest areas for future research.

2. Related Works

2.1. Sand Art. In this sections, we briefly summarize the works in technologically enhanced static and performance art creation systems, especially some related systems and algorithms for sand motion simulation.

From the perspective of computer-aided information processing, sand art can be divided into three groups: sand drawing based on medium or environmental simulation, sand animation based on artistic style simulation, and automatic sand painting creation and performance. In turn, manual interaction is involved less and less, and automation

degree becomes higher and higher. Because of the novelty, we have only found several sand art systems worth noting.

First, there are some systems of sand materials on mobile devices. Mark Hancock and colleagues' sand tray therapy system allows storytelling on a sand background [11], but users manipulate figurines instead of sand. Masahiro Ura and colleagues developed a tool for painting with simulated sand [12], but it reduces input to discrete points. iSand is an iPhone application for sand art that shares this limitation (<https://itunes.apple.com/cn/app/isand-sand-particle-painting/id346966446?mt=8>), and its sand granules are much larger than those used in traditional sand animation. Sand-Draw is a similar tool by Kalrom Systems LTD (<http://itunes.apple.com/cn/app/sand-draw/id589049492?mt=8>). These works achieve various human-computer interfaces, which simulate sand medium or environment with people oriented drawing. We can classify them into the first level of sand art, that is, sand drawing based on medium or environmental simulation, which requires users with excellent operation ability and high painting skill.

Second, there are some applications of sand interactions on mobile devices. Kazi and his students developed a new multitouch digital artistic medium named SandCanvas that simplifies the creation of sand animations [4]. SandArt is a similar tool by PSoft LTD (<http://psftmobile.net/en/sandart.html>). Lin and Fuh also program a software application named UncleSand, which can be used on the mobile tablets [5]. These works simulate hand gestures and sand interactions using mobile devices and achieve dynamic visual effects on sand art, which focuses on expressive forms and aesthetic styles. We can classify them into the second level of sand art, that is, sand animation based on artistic style simulation, which requires users with high painting skill and excellent gesture technique.

The above two categories of research works have not touched the automatic generation of sand image and animation. Finally, there are some techniques on sand art generations on mobile devices. Urbano provided an artificial sand painting that has drawn inspiration from *Temnothorax albipennis* ant societies [13]. The abstract images correspond to visualizations of the paths made by a group of virtual ants on a bidimensional space. These ants start from unorganized placement of virtual sand grains and then rearrange and create some interesting patterns composed of scattered pointillistic and imperfect circles. Urbano's results are not very satisfying from the point of visual aesthetics, but its contribution is autonomous evolution by computers without human intervention. Fan et al. investigate how to manage the image by computer to create sand painting style using matrix low-rank decomposition [9]. Song and Yoon make an effort to generate sand animation by a series of sand images, and one can produce sand image and animation without prior knowledge or skills [6]. Moreover, we have investigated and summarized computational aesthetics analysis on sand painting styles of various artists, both inside and outside China [14]. It is noted that, in the following section, we cannot conduct an experimental comparison in the absence of applications of these above two works [6, 9], but we still investigated the performance of the proposed method by

the comparisons with our previous work [10] and a GIMP plug-in, as well as a quantitative analysis using various measures.

2.2. IBAR. IBAR refers to a group of NPR techniques in this paper, which transform 2D input images into artistically stylized renderings. Jan Eric Kyprianidis and collaborators present a taxonomy of the 2D IBAR algorithms developed over the past two decades [1], including stroke-based rendering, region-based techniques, example-based rendering, image processing, and filtering.

Our technique is in the range of image processing and filtering. Many image processing filters have been explored for IBAR. Bilateral filter and Difference of Gaussians were first introduced into the stylization of cartoon renderings based on images and videos [15], but bilateral filter would remove salient visual features with no abstraction. Kuwahara filter performs comparatively well on high-contrast images from the point of view of edge-preserving [16], but it is unstable for homogenous regions. Several variations, such as generalized and anisotropic Kuwahara filter [17, 18], were proposed to overcome the limitations of the original Kuwahara filter.

In our previous work [10], bilateral filter [19] (BF), Kuwahara Filter [17] (KF), median filter [20], and anisotropic diffusion [21], along with FDoG [22], were employed into a fully automatic pipeline for sand art painting, which consists of four steps, including image filtering, color and texture mapping, primary contour enhancing, and details optimizing.

Except for the above one, there is nothing about IBAR-based methods for creating sand painting images. Comparatively, image processing technique is still an interesting alternative for the other three groups because of parallelization and GPU implementations, although the rendering results may not be so perfect from an artistic point of view. This is probably because these filters are often concerned with the restoration and recovery of photorealistic imagery. By contrast, IBAR generally aims for simplification. Thus, IBAR-based methods for fully automated sand painting is worth being considered.

Throughout the current researches, the following tendencies are reflected. (1) Novelty: computer-aided sand painting art is novel and interesting in IBAR. This tendency corresponds to the sixth one of seven research questions in [1], and more and more new forms of art should be considered into account. (2) Partiality: the major concern of existing techniques for sand painting stylization partially focuses on the human-computer interaction and seriously depends on mobile devices and users' drawing skills. This tendency meets the fourth one of seven research questions in [1], and we should develop automated tools that support creativity and the artist. (3) Scarcity: in view of the rising of this form of artistic stylization, there have been no systematic studies, since computer-aided sand painting art is only in its infancy. According to the future directions and challenges [1], we try to make the effort to achieve automated artistic stylization and provide a fully automatic image filter pipeline for sand art painting.

3. Image Data Field for Sand Painting Filter

3.1. Data Field. Data field is proposed by Li and Du [23] in recent years and has been of particular interest to researchers. Its main idea is originated from physical field. In nuclear field, nuclear force binds protons and neutrons together to form the nucleus of an atom. Data field provides an analogy with the mechanism of the nuclear field theory. Given a data space, data field describes the complex correlation among data objects, where there are some effects and interactions in an unknown way, and it expresses the power of an item in the universe of discourse by a potential function as the physical field does.

Data field is introduced into various applications [24–26] and also has been successfully used in image processing [27]. There are two general categories of data field, static and dynamic ones, and we use the former in the paper. The static data field corresponds to the stable active field in theoretical physics. Inspired by preliminary work [28], we introduce the static data field and propose a sand art generation method for image-based painting.

Given a data object p in data space Ω , let $\varphi_p(q)$ be the potential at any position $q \in \Omega$ produced by p ; then $\varphi_p(q)$ can be computed by any of the following equations:

$$\varphi_p(q) = M_p^q \times \exp\left(-\left(\frac{\|p - q\|}{\sigma}\right)^k\right) \quad (1)$$

$$\varphi_p(q) = G \times \frac{M_p^q}{\left(1 + (\|p - q\|/\sigma)^k\right)} \quad (2)$$

$$\varphi_p(q) = \frac{M_p^q}{\left(4\pi\epsilon_0 \times \left(1 + (\|p - q\|/\sigma)^k\right)\right)}, \quad (3)$$

where $\|p - q\|$ is the distance between p and q , the strength of interaction $M_p^q \geq 0$ can be regarded as mass or charge of data objects, a natural number k is the distance index, and $\sigma \in (0, +\infty)$ is the influential factor that indicates the range of interaction. Additionally, the distance is usually measured by Euclidean, Manhattan, or Chebyshev metric.

In general, there is more than one object in data space. To obtain the precise potential value of any position under these circumstances, all interactions from data objects should be regarded. Given a data set $D = \{p_1, p_2, \dots, p_n\}$, because of overlap, the potential of any position q in the data space is the sum of all data radiation,

$$\varphi(q) = \sum_{i=1}^n \varphi_{p_i}(q), \quad (4)$$

where $\varphi_{p_i}(q)$ is calculated by one of (1)–(3).

Equations (1)–(3) are three common choices of the above potential function. Equation (1) imitates nuclear field with Gaussian potential, while (2) and (3) imitate gravitational field and electrostatic field, respectively, where G and ϵ_0 are the constants depending on the law of gravitation and the Coulomb law. Mathematically considered, therefore, the latter two seem essentially the same. Nuclear force field

using (1) is a short-range interaction in the physical world, and, conversely, gravitational field using (2) is a long-range interaction. Field intensity of the former attenuates rapidly with the increment of interaction distance. Just for the same influential factor, the potential value of nuclear force field tends to attenuate more rapidly than that of gravitational field. Obviously, this feature of short-range interaction is more suited to describing the grayscale change relationships in the image neighbourhood, since two pixels in an image should have a relatively lower relevance if the distance between them is too long. However, we need to state that there are several alternative formulae for $\varphi_p(q)$, such as electromagnetic field, temperature field, or nuclear field with exponential potential.

In the following, we use the nuclear field with Gaussian potential as (1), and we fix $k = 2$ and choose the Chebyshev distance for $\|p - q\|$. Furthermore, how to determine the mass M_p^q is also an open problem, and various applications use various definitions of the mass. Three kinds of masses, inertial mass, active gravitational mass, and passive gravitational mass, are defined in theoretical physics. For example, the mass used in [27] refers to the active gravitational mass of p acting on q , while a passive gravitational mass can be also a useful measure by an object in a known gravitational field. Thus, the active gravitational mass that acts on pixel p by pixel q is significant for image data field. We define various image data fields with various forms of the mass according to the requirement of sand art, and then field coupling is introduced into producing a sand painting.

3.2. Image Data Field. Image data field is a natural extension of data field in image processing. Each image pixel is as a data particle with mass and has interactions with neighbouring pixels. The potential sum at any pixel is calculated by obeying the law of short-range nuclear force field.

Suppose $\Omega = \{p = (x_p, y_p) \mid x_p \in [1, h], y_p \in [1, w]\}$ is a finite space consisting of two-dimensional pixels, $g : \Omega \rightarrow [0, L - 1]$ is a mapping, and $f(p)$ denotes the grayscale value of the pixel p , then an input image is a pair $I = (\Omega, g)$, where h, w , and L are the height, width, and gray level of the image, respectively.

According to data field, each pixel $p \in \Omega$ is a particle with mass, and the grayscale change interactions (attraction or repulsion) between each other form an image data field on Ω . Assuming two pixels $p, q \in \Omega$, let $\varphi_p(q)$ be the potential at any pixel p produced by q , and then it can be computed by

$$\begin{aligned} \varphi_p(q) &= M_p^q \\ &\times \exp\left(-\left(\frac{\max\{|x_p - x_q|, |y_p - y_q|\}}{\sigma_d}\right)^2\right), \end{aligned} \quad (5)$$

where M_p^q related to $g(p), g(q)$ is the strength of interaction and can be considered as the mass of data object in image data field. σ_d denotes the influential factor related to distance.

FIGURE 2: The pipeline of our technique for creating sand painting-like effect.

Given a two-dimensional space Ω , each pixel acts on each other by some interaction forces and forms an image data field, the potential of any pixel q can be defined as follows:

$$\varphi(q) = \sum_{p \in \Omega} \varphi_p(q), \quad (6)$$

where $p \in \Omega$ denotes the neighbourhood of q in the image.

Generally, there are too many pixels p in a given image space Ω . The above potential simulates the short-range nuclear force's field corresponding to the Gaussian potential field, which satisfies *three-sigma rule*. In the image data field, the influential range of a data object is the geometrical neighbourhood with the distance shorter than $3\sigma_d/\sqrt{2}$; that is to say, data objects beyond a certain distance are almost not influenced by a specified data object, and the potential values contributed become zero. Especially for the image, too wide range of neighbourhood is time-consuming and unnecessary. To reduce the time cost, the potential of any pixel q can be simplified as follows:

$$\varphi(q) = \sum_{q \in \xi(p)} M_p^q \times \exp\left(-\left(\frac{\max\{|x_p - x_q|, |y_p - y_q|\}}{\sigma_d}\right)^2\right), \quad (7)$$

where $\xi(q) = \{p \mid p \in \Omega, \max\{|x_p - x_q|, |y_p - y_q|\} \leq 3\sigma_d/\sqrt{2}\}$ denotes the neighbouring pixels affected by the central pixel q .

4. Proposed Method

4.1. Overall Procedure. In the real sand paintings, artists achieve the effect of accentuation by controlling not only

the level of details but also the appearance of strokes. The salient region is painted in fine details with clear and sharp strokes, while the periphery area is drawn with rough and flat strokes and the outer region is simply omitted. To achieve a similar effect, the proposed method uses image data field to obtain the saliency map, to define the relevance of each area. To locally adapt the density and appearance of strokes to the relevance, we first build a field pyramid from the input image and then calculate a cumulative weighted average from field coupling using various resolution, such as the salient, gradient, noise, and texture field. In addition, a real sand painting is usually with scabrous granular spot or texture. To simulate this feature, we introduce two additional steps of mathematical morphology and fragmentation.

We present the components of our technique, and a block diagram is given in Figure 2, where two purple dashed blocks indicate optional steps.

Given an input image, the proposed method automatically converts it into nonphotorealistic rendering of a sand painting-like stylization by the following steps:

- (1) Generate a grayscale image from the input image, and obtain the gradient map from the input image.
- (2) Build image data field using the absolute mass, and obtain the gradient field from the gradient map.
- (3) Apply image transformation using image data field with the relative mass, and generate a saliency field from the input image.
- (4) Apply various influence factor to generate a field pyramid, and build a multiscale field coupling by weighted averaging pixels from different fields of the multiresolution field pyramid according to the pixel's similarity of potential value in sand painting field.
 - (i) Combining salient field and gradient field, generate a joint edge field using field coupling. In

addition, the layer flag and the importance map are also prepared for the sand animation.

- (ii) Generate the noise field by applying random dithering to the transformed image in Step (3).
 - (iii) Generate the blurring field using field coupling on gradient field and noise field.
 - (iv) Generate the sand stroke field using field coupling on edge field and blurring field.
 - (v) Generate the sand painting field by composite of sand stroke field and sand sample texture.
- (5) Apply color mapping pixel by pixel, as well as mathematical morphology and fragmentation, to produce the sand stylization.
 - (6) Obtain the output image, that is, sand painting-like effect, and salience layer and gradient importance are prepared for the extension of sand animation.

4.2. Image Transformation Using Image Data Field. According to [28], image data field has good performance on outliers. Image transformation can be the combination of the grayscale value and the potential-weighted sum of image data field, which is determined by the interaction intensity and distance. Taking the grayscale weight into account, image data field can be generated for an input image $I = (\Omega, g)$, and the relative mass of each object is formulated as (8), and the absolute mass as (9).

$$M_p^q = Mr_p^q = \exp\left(-\left(\frac{g(p) - g(q)}{\sigma_m}\right)^2\right), \quad (8)$$

$$M_p^q = Ma_p^q = \exp\left(-\frac{g(p)}{\sigma_m}\right)^2, \quad (9)$$

where σ_m denotes influential factor related to interaction mass, and $d_p^q = \exp(-(\max(|x_p - x_q|, |y_p - y_q|)/\sigma_d)^2)$ in (7) is the distance of interaction, as well as the spatial weight.

To further use this property, an image transformation can be produced. For an image $I = (\Omega, g)$, the improved intensity $g'(q)$ of any pixel q is calculated by

$$g'(q) = \frac{\sum_{p \in \xi(q)} \varphi_p(q) g(p)}{\varphi(q)}. \quad (10)$$

Given the central pixel q , (10) is the weighted sum of grayscale value $g(p)$ and potential interaction φ_p of each neighbouring pixel $p \in \xi(q)$. Therefore, the transformed result g' is still bounded in the interval $[0, L - 1]$.

As can be seen from (10), the three following properties of the transformed image are obtained (1) The outliers lie in the neighbourhood of a central pixel q ; $g'(q)$ does not relate primarily to the outliers since the dominant weights are owned by homogeneous pixels, which have the similar gray level with q . (2) While the central pixel q is outlier, $g'(q)$ would be smoothed by the other homogeneous pixels, since although they have litter weights, the quantity of these pixels occupies the absolute superiority. (3) For a homogeneous

neighbourhood, the potential value of the central pixel is very small, even zero. However, for a neighbourhood consisting of several classes of homogeneous pixels, the potential value of the central pixel is generally high. The lower the potential value of a central pixel, the more likely that the central pixel is in the interior of a homogeneous region. In the homogeneous regions, the force field is uniform, pixels attract each other, and they group into a cluster. Meanwhile, in the transition regions, the force field is nonuniform, pixels are repulsed by homogeneous pixels, and then they are separated from the homogeneous regions. It can be a measurement of gray level changes, which is useful to extract transition regions.

Given the input image $I = (\Omega, g)$, we build image data field with the relative mass in (8) and then obtain the salient field using (7). For each central pixel q , the potential value in saliency field is denoted by $\varphi_{\text{salient}}(q)$, which is calculated by

$$\varphi_{\text{salient}}(q) = \sum_{p \in \xi(q)} \exp\left(-\frac{(g(p) - g(q))^2}{\sigma_m}\right) \cdot \exp\left(-\left(\frac{\max\{|x_p - x_q|, |y_p - y_q|\}}{\sigma_d}\right)^2\right). \quad (11)$$

Applying image transformation, the transformed image $I' = (\Omega, g')$ is determined using (10). In other words, for an image $I = (\Omega, g)$, the improved intensity $g'(q)$ of any pixel q is calculated by

$$g'(q) = \frac{\sum_{p \in \xi(q)} \varphi_{\text{salient}_p}(q) g(p)}{\varphi_{\text{salient}}(q)}, \quad (12)$$

where $\varphi_{\text{salient}_p}(q)$ denotes the potential value at the position q in salient field produced by p .

It should be noted that most of grayscale-level methods handle the *uint8* images, and it would be perfect if the transformed results can be converted into this type by using functions *round()* and *uint8()* in MATLAB.

To depict the features of transition region adequately, the saliency field would be synthesized as a new descriptor in the following section. Therefore, the factor is first normalized via the following way to avoid one factor being neglected due to large differences between each other.

$$\begin{aligned} & \varphi_{\text{salient}}(q) \\ &= (L - 1) \frac{\varphi_{\text{salient}}(q) - \min_{p \in \Omega} \varphi_{\text{salient}}(p)}{\max_{p \in \Omega} \varphi_{\text{salient}}(p) - \min_{p \in \Omega} \varphi_{\text{salient}}(p)}. \end{aligned} \quad (13)$$

Taking the logo image of National Natural Science Foundation of China as an example (see Figure 3(a)), we apply the image transformation using image data field and demonstrate the transformed grayscale image in Figure 3(b) and the salient field in Figures 3(c) and 3(d). One can observe that the salient field highlights the visual saliency; the potential value is determined primarily by how different a given location is

FIGURE 3: The sample image and its salient fields: (a) the original image, (b) the transformed grayscale image, (c) the vector formulation of salient field, and (d) the equipotential line of salient field.

from its surroundings in grayscale level. In Figure 3(c), the salient field is overlapped by the original image and labelled as the direction of data field. In Figure 3(d), the equipotential line of salient field is shown as electric field in physics.

4.3. Coupling on Gradient Field and Salient Field. In this subsection, we generated the edge field using the coupling on gradient field and salient field. We first calculate the gradient map $F = (\Omega, f)$ using the *gradient()* function in MATLAB, which returns the x and y components of the two-dimensional numerical gradient; f_x corresponds to $\partial F / \partial x$, the differences in x (horizontal) direction, while f_y corresponds to $\partial F / \partial y$, the differences in the y (vertical) direction. Then, the intensity of pixel q in the gradient map is determined by $f(q) = \sqrt{f_x(q)^2 + f_y(q)^2}$.

Given an input image $I = (\Omega, g)$, we obtain the corresponding gradient map $F = (\Omega, f)$ and then build the gradient field according to (7) using the absolute mass in (9). For each central pixel q , the potential value in gradient field is denoted by $\varphi_{\text{gradient}}(q)$, which is calculated using

$$\varphi_{\text{gradient}}(q) = \sum_{p \in \xi(q)} \exp\left(\frac{-f(p)}{\sigma_m}\right)^2 \cdot \exp\left(-\left(\frac{\max\{|x_p - x_q|, |y_p - y_q|\}}{\sigma_d}\right)^2\right). \quad (14)$$

For the same purpose as salient field, we also normalize the potential values in gradient field:

$$\begin{aligned} \varphi_{\text{gradient}}(q) \\ = (L - 1) \frac{\varphi_{\text{gradient}}(q) - \min_{p \in \Omega} \varphi_{\text{gradient}}(p)}{\max_{p \in \Omega} \varphi_{\text{gradient}}(p) - \min_{p \in \Omega} \varphi_{\text{gradient}}(p)}. \end{aligned} \quad (15)$$

Once the salient field and the gradient field are presented, one can generate a joint edge field using field coupling, and the potential value $\varphi_{\text{edge}}(q)$ at each central pixel q in edge field is described as (16). In addition, the layer flag and the importance map are also prepared for the sand animation.

$$\begin{aligned} \varphi_{\text{edge}}(q) = & (\omega_1 \varphi_{\text{salient}}(q) + (1 - \omega_1) \varphi_{\text{gradient}}(q)) \\ & \cdot \tau_{\text{salient}}(q) \tau_{\text{gradient}}(q), \end{aligned} \quad (16)$$

where $\omega_1 \in [0, 1]$ is a weight balancing contribution of normalized salient field and gradient field.

In an extreme case, when $\omega_1 = 1$, the new field approximately degenerates to salient field, and, conversely, when $\omega_1 = 0$, it is equivalent to gradient field. Therefore, ω_1 should be between 0 and 1, and $\omega_1 \in (0, 0.5]$ is more perfect since an accentuated gradient is necessary in the real sand art. We fix $\omega_1 = 0.2$ as the default value of the proposed algorithm. The two switch functions $\tau_{\text{salient}}(q)$ and $\tau_{\text{gradient}}(q)$ are the control factors related to the density and the width

FIGURE 4: The gradient field and its coupling result with salient field: (a) the gradient map, (b) the gradient field, and (c) the edge field.

of edges, and they are calculated by the following equations, respectively:

$$\tau_{\text{salient}}(q) = \begin{cases} 1 & \text{if } \varphi_{\text{salient}}(q) > \overline{\varphi_{\text{salient}}}(q) - T_{\text{salient}}, \\ 0 & \text{otherwise,} \end{cases} \quad (17)$$

$$\tau_{\text{gradient}}(q) = \begin{cases} 1 & \text{if } \varphi_{\text{gradient}}(q) > T_{\text{gradient}}, \\ 0 & \text{otherwise,} \end{cases} \quad (18)$$

where $\overline{\varphi_{\text{salient}}}(q)$ is the average of potential values from the pixel q 's neighbourhood in salient field.

The constant T_{salient} is a salient threshold to control the width of edges, and we fix the default value $T_{\text{salient}} = 5$ using try and error method. The variable parameter T_{gradient} is a gradient threshold to control the density of edges, and we introduce the Rosin method [29] to obtain this threshold automatically.

We also take the logo image as an example and build the image data field from the gradient map shown in Figure 4(a), and the gradient field is listed in Figure 4(b). The edge field is generated using field coupling, as shown in Figure 4(c).

4.4. Coupling on Gradient Field and Noise Field. In this subsection, we generated the blurring field using the coupling on gradient field and noise field. The noise field is generated by applying random dithering to the transformed image

according to (12). The value $\varphi_{\text{noise}}(q)$ at the pixel q is calculated in the following way:

$$\varphi_{\text{noise}}(q) = \begin{cases} g(q) & \text{if } r_n > \frac{\eta g(q)}{255}, \\ 255 & \text{otherwise,} \end{cases} \quad (19)$$

where $r_n \in [0, 1]$ controls the frequency of noises and $\eta \in (0, 1]$ controls the intensity of noises.

In the following experiments, we fix $\eta = 0.5$ as a default setting. Then the blurring field is determined using field coupling on gradient field and noise field, and the value at each central pixel q in blurring field is denoted as $\varphi_{\text{blur}}(q)$, which is formulated as follows:

$$\varphi_{\text{blur}}(q) = \frac{\sum_{p \in \xi(q)} \varphi_{\text{noise}_p}(q) \varphi_{\text{gradient}_p}(q)}{\varphi_{\text{gradient}}(q)}. \quad (20)$$

For the logo image, we generated the noise field using the above method, as shown in Figure 5(a). Applying the field coupling, we obtain the blurring field as shown in Figure 5(b).

4.5. Coupling on Edge Field and Blurring Field. In this subsection, we generated the sand stroke field using the coupling on edge field and blurring field.

FIGURE 5: The noise field and its coupling result with gradient field: (a) the noise field and (b) the blurring field.

FIGURE 6: The results with the constant influence factor: (a) the sand stroke field, (b) the sand painting field, and (c) the sand painting image.

The sand stroke field resulted in rough and wide strokes at the area with a low potential value, which can be determined by

$$\varphi_{\text{stroke}}(q) = \omega_2 \varphi_{\text{edge}}(q) + (1 - \omega_2) \varphi_{\text{blur}}(q), \quad (21)$$

where $\omega_2 \in [0, 1]$ is a weight balancing contribution of edge field and blurring field.

This parameter has a similar effect as ω_1 , and the only difference between them is the value range. When $\omega_2 = 1$,

the new field approximatively degenerates to edge field, and, conversely, when $\omega_2 = 0$, it is equivalent to blurring field. To find a compromise between them, we fix the arithmetic average value of edge field and blurring field as the default value of ω_2 ; that is, $\omega_2 = 0.5$. For the logo image, we obtain the sand stroke field as shown in Figure 6(a).

4.6. Coupling on Texture Field and Sand Stroke Field. In this subsection, we generated the sand painting field using the

coupling on texture field and sand stroke field, and then a temporary image can be generated.

Given a sample image with sand texture, the texture field is obtained by the same steps of salient field, including image data field, image transformation, and potential normalization. The only difference between them is the used mass, and the texture field uses the absolute mass corresponding to (9). In other words, the potential value in texture field can be defined by modifying (13). In sand painting field, the value $\varphi_{\text{sand}}(q)$ at the pixel q is calculated by

$$\varphi_{\text{sand}}(q) = \omega_3 \varphi_{\text{texture}}(q) + (1 - \omega_3) \varphi_{\text{stroke}}(q), \quad (22)$$

where $\omega_3 \in [0, 1]$ is a weight balancing contribution of texture field and stroke field.

This parameter also has a similar effect as ω_1 , even the bounds of values. With $\omega_3 = 1$, the sand painting field approximately degenerates to texture field, and, conversely, when $\omega_3 = 0$, it is equivalent to sand stroke field. Certainly, $\omega_3 \in (0, 0.5]$ is more perfect since sand stroke field is more important than the sample texture field, and our purpose is the original image-guided art stylization. Therefore, we fix $\omega_3 = 0.1$ as the default value because of the aesthetic purpose.

For the logo image, we obtain the sand painting field as shown in Figure 6(b). After coupling on texture field and sand stroke field is done, the result field has more abundant details, color information, and texture features. In comparison, the sand stroke field in Figure 6(a) is relatively poor and monotonous in background pixels.

Although the data field has been normalized into $[0, L-1]$, it is only with a grayscale channel and appears significantly different than the real sand art. To achieve the visual effect of sand art, we apply the color mapping and convert a data field to a result image. The color mapping is calculated by the following way:

$$\begin{aligned} \mathbf{R} &= 239 \left(\frac{\varphi_{\text{sand}}}{255} \right)^{0.4}, \\ \mathbf{G} &= 221 \left(\frac{\varphi_{\text{sand}}}{255} \right)^{0.5}, \\ \mathbf{B} &= 111 \left(\frac{\varphi_{\text{sand}}}{255} \right)^{0.6}, \end{aligned} \quad (23)$$

where $\mathbf{R}$, $\mathbf{G}$, $\mathbf{B}$ denote the color component of red, green, and blue channels, respectively.

For the logo image, we obtain the sand painting image as shown in Figure 6(c). The result approximately simulated the style of sand painting. For a large object, the edge and contour pixels are well described. To get more detail, we provide a local enlarged part, attached on the lower-left corner using a rectangular area with red borders. One can find that, the difference between the background and foreground pixels in Figure 6(c) is not very clear, even for a very simple image with salient object. Thus, further improvement is necessary.

4.7. Multiscale Field Coupling. This field coupling filter remains the bottleneck of our methodology; therefore we also developed a multiscale approach based on a field pyramid

built in a fine-to-coarse manner, controlled by various influence factors. In other words, one can apply various influence factors to generate a field pyramid and build a multiscale field coupling by weighted averaging pixels from different fields of the multiresolution field pyramid according to the pixel's the similarity of potential value in sand painting field. In the first scale, we use the original image as the input and conduct the field coupling (denoted by $Fc(\cdot)$ in (24)) using the minimal influence factor δ_d^1 ; the result Res_1 is the preliminary grounding. From the second scale to the last scale, the field coupling is also conducted using various influence factors, and only the difference between results of two successive scales is considered as the details modification. The process can be formalized as

$$\text{Res}_i = \begin{cases} Fc(I, \delta_d^1), & i = 1, \\ Fc(I, \delta_d^i) - \text{Res}_{i-1}, & i > 1, \end{cases} \quad (24)$$

where Res_i is the transition state of each round, I is the original image, and $Fc(\cdot)$ represents those steps of various field coupling using the influence factor δ_d^i , as mentioned previously.

Once multiscale field coupling has been completed, the weighted sum of each staged result Res_i is then the final tone before postprocessing; that is, the tone is calculated by

$$\text{Res} = \sum_{i=1}^n \xi_i \text{Res}_i. \quad (25)$$

In (25), n is the number of scales and ξ_i is an open weight of each state. There are several methods to determine this weight, and a useful strategy is that the weight should be reduced with the increase of i and be as close as possible to each other but have enough discrimination. We proposed a possible scheme as

$$\xi_i = s_{\max} \left(1 - \frac{\delta_d^i}{\delta_d^n + 1} \right)^\kappa, \quad (26)$$

where κ is the attenuation index and $s_{\max}$ is distinguishing coefficient, corresponding to the two conditions of the above strategy. In the following experiments, we fix $\kappa = 0.5$ and $s_{\max} = 2$ as a default setting. The filtered result with multiscale field coupling is listed in Figure 7(d). The transition fields are also attached in Figures 7(a)–7(c). Comparing Figure 7(d) with Figure 6(c), the high contrast between object and background is more intense, and the pixels on edges and contours are clearer. However, there is an obvious artificial drawing trace, and the sand texture is also absent.

4.8. Additional Postprocessing. To achieve a closer visual effect of real sand painting, we introduce two additional components as the postprocessing. Excepting color mapping pixel by pixel, the final steps involve adding a few more components to the rendered image to improve its sand stylization, including mathematical morphology and fragmentation.

In the step of mathematical morphology, only imdilate [30] is used. As a result, the regions with high intensity

FIGURE 7: The results with multiscale field coupling: (a)–(c) the difference image from 3 to 11, (d) the sand painting image with multiscale field coupling, and (e) the sand painting image with postprocessing.

are dilated, especially the edge pixels. We use the function *imdilate()* provided by MATLAB, which performs grayscale dilation with a rolling ball structuring element. In the fragmentation step, the values of central pixels are substituted by values from random neighbourhood pixels. Correspondingly, there appears a scabrous granular texture, which also exists in the real sand painting. In general, the visual effect of simulating the sand painting is acceptable. In the following section, we would provide more results and analyse them using various measures on feature similarity.

After additional postprocessing being finished, the final result is shown in Figure 7(e). The sand painting-like effect is then simulated by the proposed method. According to our knowledge, the edges or the salient objects in a real sand painting are not very stiff, which only pursue a limited and incomplete realistic aesthetics, or more spiritual expression. The rendering image should look fine until one sees it at close quarters, when the sand grains are clearly visible. The enlarged view attached on Figure 7 demonstrates this point. Comparing Figure 7(d) and Figure 7(e), the line in Figure 7(d) is too perfectly straight and lacks the fluidity of real sand. The overall result is without an artistic sense of beauty. Seemingly fantastic objects and their outlines exist in Figure 7(e), which have different shapes after a rough look, but boundaries are really invisible with a detailed view. We believe that the postprocessing can improve the rendering quality for most of the input images.

5. Experimental Results

5.1. Experimental Setup. In this section, several experiments and comparisons are presented to examine the performance

of the proposed technique. We have conducted our experiments on a PC platform with an Intel 2.5 GHz Core i7-6500U CPU and 8 GB RAM, running on a Windows 10 operating system.

Before demonstrating and evaluating our rendering results, let us first recapitulate all the related parameters required from the users, so that these results are reproducible. Currently there are totally ten parameters, including σ_d in (7), σ_m in (8) and (9), ω_1 in (16), T_{salient} in (17), T_{gradient} in (18), η in (19), ω_2 in (21), ω_3 in (22), and κ and s_{max} in (26). Three of them, that is, σ_d , σ_m , and T_{gradient} , could be set automatically. The optimized parameter σ_d is obtained by the Shannon entropy-based method, and σ_m can be determined according to the grayscale changes related to σ_d [28]. As mentioned previously, T_{gradient} is calculated by the Rosin method [29]. An optimal T_{salient} can be experimentally obtained from 3 to 7 by a trial-and-error method. As previously explained, the default value of T_{salient} is set to be 5 for all the images in Figure 8 according to our experiments. Out of the remaining six, at least three of them, that is, η , κ , and s_{max} , could be set as constants across all the results; we fix $\eta = 0.5$, $\kappa = 0.5$, and $s_{\text{max}} = 2$. Thus, there remain at most three ones as the stylization control parameters to be provided by users, including ω_1 , ω_2 , and ω_3 . With a default configuration of $\omega_1 = 0.2$, $\omega_2 = 0.5$, and $\omega_3 = 0.1$, we will further discuss these three parameters by experiments.

The techniques based on bilateral filter and Kuwahara filter produced acceptable results for sand painting stylization [10]. Thus, we compare the proposed method with these two methods. In addition, a texture transfer-based method, named Resynthesizer, is also involved in comparisons. Resynthesizer is a GIMP plug-in for texture synthesis by Paul

FIGURE 8: The visual results by various methods: (a) the original images, and the result images by (b) the BF method, (c) the KF method, (d) the GIMP method, and (e) our method.

Harrison, which re-rendered an image with the style of a different image. Given a sample of a texture and an original image, Resynthesizer can create a new image with the transferred texture style, and it can be publicly downloaded from <http://registry.gimp.org/node/25219>. For three

compared methods, the parameters (if any) are used as a default setting.

5.2. Visual Results. We first examine how the proposed technique can benefit the fully automated sand painting. As

TABLE 1: Quantitative evaluation results.

Measures	BF	KF	GIMP	Ours
PSNR	0.2810	0.3331	0.4297	0.3051
MSSIM	0.6530	0.6383	0.6010	0.6279
FSIM	0.4813	0.3552	0.6628	0.7429
VSI	0.5558	0.4012	0.7836	0.5616
GSM	0.5415	0.4323	0.5739	0.7003
GMSD	0.3831	0.5659	0.5046	0.2661

shown in Figure 8(a), eight input images of dramatically different scenes (close-ups, portrait, landscape, and buildings) are chosen for the demonstration. Note that, to make the paper more compact, we have chosen to shrink all resulting images. However, this should not damage the image quality as each image is associated with a high resolution. Therefore any reader who is interested in seeing more details from an image can always enlarge the corresponding portion.

To show that our rendering results (see Figure 8(e)) are indeed better than those by others, we compare our results with the ones generated by BF-based method (BF for short), KF-based method (KF for short), and Resynthesizer in GIMP (GIMP for short), as shown in Figures 8(b)–8(d) from left to right, respectively. Five original images from top to bottom are downloaded from Baidu Gallery (<http://image.baidu.com/>), and the result images are listed in the top five rows of Figure 8.

The first two images are relatively easy to process, in which there are a salient object with a flat background. For the flower image in the first row, all the methods obtain moderate results, but only the proposed method yields an appropriate result, and the others cannot describe the smallest petal, as labelled by a red rectangle. For the building image, our method and the BF method produce similar results, the rendered background is reasonably indistinct, and the building is highlighted. Our method processes the cloud more delicately than the BF method, while the KF method and the GIMP method obtain an almost completely failed result, since the building has been lost in the background. The next two images are a little more complex because of the difficulty of extracting the salient objects. For these two images, the proposed method obtains better visual effect, which can be justified by comparing the filtering results perceptually. The details are well preserved by the proposed method, for example, the bell on the top of tower and the hollow decoration on the door. The fifth image is the Temple of Heaven with uneven light, and the key point is how to reduce the disturbance of illumination change. Our method produces the good result for the image, especially for the stereo feeling.

Example outputs using test images of the recent NPR benchmark [31] are shown in the remaining three rows of Figure 8. Generally, the BF and KF methods perform not very well for these three images. In rim lighting image, there is a portrait with a clean background and strong rim lighting. The KF method and the GIMP method suffer failures on near-uniform background. The BF method generates a not-bad result, and our result is also acceptable. Both of the methods

express the man's facial appearance. In the next image, there is a cat with the blurry but varied background, and it is noticeable that the good stylization result is difficult to be obtained without a priori information because of the complex background. The GIMP method and the proposed method provide the reasonable results. The cat's whiskers are thin but definite features, and our method well captures this point. The last one is a portrait of a Mac user, with generally light tones. The man's features are partially occluded by the Mac. Some important edges are blurred owing to the shallow depth of field; for example, the ear is not very clear, but it is a crucial feature. Thus, this image is slightly complicating stylization. We could see that the results by the compared methods are not very good, and our result is basically reasonable. At least the ear is painted.

In summary, the experimental results of these images indicate that the proposed method is effective in yielding the approximately ideal results of sand painting stylization.

5.3. Quantitative Results. To measure the robustness of the considered methods, we studied quantitative scores of the rendering outputs with various images. Table 1 shows the average response with respect to eight images by using six measures on feature similarity, including the standard pixel-wise Peak Signal to Noise Ratio (PSNR), Mean Structural SIMilarity measure (MSSIM) [32], Feature SIMilarity (FSIM) [33], Visual Saliency based Index (VSI) [34], Gradient SIMilarity based Metric (GSM) [35], and Gradient Magnitude SIMilarity Deviation (GMSD) [36].

These metrics aim to use computational models to measure the image quality consistently with subjective evaluations, with a full reference image, which is the original input image in this paper, and the distorted image refers to the stylized result. As shown in Table 1, the proposed method outperforms other methods in respect of preserving edges and features, since it demonstrates the highest FSIM and GSM scores, as well as the lowest GMSD value. FSIM is based on the fact that human visual system understands an image mainly according to its low-level features, while GSM and GMSD are based on gradient similarity. In other words, our results achieve a high similarity with the input images from the view of visual features and gradient magnitude. Various metrics in Table 1 measure the feature similarity from different points. Our method yielded not so high scores by the measures of PSNR and MSSIM. The reason is maybe that the proposed method modifies and improves the original images according to the style of sand painting, but not only a simple image filter. Specifically, most of the background pixels or the

TABLE 2: Points received by measures based on the rank.

Measures	BF	KF	GIMP	Ours
PSNR	4	2	1	3
MSSIM	1	2	4	3
FSIM	3	4	2	1
VSI	3	4	1	2
GSM	3	4	2	1
GMSD	2	4	3	1
Total	16	20	13	11
Average	2.7	3.3	2.2	1.8

FIGURE 9: The rendering results using various parameters: (a) $\omega_1 = 0.01, 0.5$, (b) $\omega_2 = 0.1, 0.9$, and (c) $\omega_3 = 0.01, 0.5$.

nonsalient objects are replaced by the sand texture. In fact, the filter we described maintains only the relevant parts of the contours, while reducing the presence of irrelevant edges which are usually due to noise. To sum up, the quantitative comparison demonstrates that our method is comparatively robust.

In order to “compress or reconstruct” the data from Table 1, we decided to replace the actual average scores with points. In each row of Table 1, we can replace the average score with the rank of that score in the row. We can assign points by the rank from the number of elements, that is, resulting in a score between 1 and 4. 1 point ranks first, and 4 points rank last. If we apply this simple rule, we then obtain Table 2, as well as the total and the average points. However, Table 2 should not be regarded as a strict competition in aesthetic quality of the related methods, but more as an indication of the cross-correlation between a feature similarity measure and others. Some other measures can be employed to assess the image quality of sand painting stylization. If one method totally scores low in this table, then it suggests that the results are relatively dominant from the perspective of these groups of feature similarity measures. As a result, our method gains 1.8 points, and the corresponding results appear to have a higher similarity with the original images in edges, gradient, and visual saliency. At least, the results reported by these

metrics suggest that our method is effective to some extent, and it has shown some advantages in preserving edges and features.

In general, we still state that the aesthetic quality of the stylization methods is rather subjective and cannot be easily evaluated via metrics. Nevertheless, our method is effective at generating sand painting-looking images, both visual effects and quantitative results.

5.4. Parameter Selection. The visual effect of the proposed artistic processing technique is primarily controlled by ten parameters, which can be divided into 3 groups, three of them are obtained automatically, four of them are used as a default configuration, and the remaining can be fixed by users to control the style. In this subsection, we discuss these stylization parameters, that is, ω_1 , ω_2 , and ω_3 . The three parameters are used during the process of multiscale field coupling.

In the phase of edge field generation, $\omega_1 \in (0, 0.5]$ controls the weight balancing contributions of normalized salient field and gradient field. We generate two result images using $\omega_1 = 0.01, 0.5$, as shown in Figure 9(a). For visual comparison, we also list the visual result of the default parameter $\omega_1 = 0.2$. Half of each result is matched up and the lower triangular part is from the result using $\omega_1 = 0.2$, while the other

TABLE 3: Average elapsed time in seconds.

Images number	h	w	Average time costs (s)			
			BF	KF	GIMP	Ours
(1)	1024	653	13.85	10.04	90.04	30.65
(2)	1280	860	7.87	8.72	145.03	38.87
(3)	1024	680	14.59	13.74	93.15	30.47
(4)	1024	669	8.58	8.93	86.52	30.40
(5)	1024	745	14.02	12.49	129.62	33.15
(6)	1024	1024	15.93	13.17	233.73	45.97
(7)	1024	852	9.94	13.06	122.46	42.91
(8)	1024	1024	15.33	12.91	236.42	51.18

part is the ones using $\omega_1 = 0.01, 0.5$; see two subfigures in Figure 9(a). With a smaller value of ω_1 , the edge field approximatively degenerates to salient field, and vice versa. That is to say, if users think an input image's saliency is more important than its gradient, a smaller ω_1 can be fixed so that the saliency can be highlighted. As can be seen from Figure 9(a), pixels from both edge and background are also involved into the visual effect. With a smaller ω_1 , the edge field is as a fine and crease edge, and the background is rough and dark. Otherwise, the edge field is considered a coarse and silhouette edge, since the high gradient magnitude pixels (usually on the edge) are fragmented with no salient pixels being processed. Furthermore, various sand painting fields in the subsequent process should produce different results by the color mapping, and then the background would be flat and bright.

In the phase of sand stroke field generation, $\omega_2 \in [0, 1]$ controls the weight balancing contributions of edge field and blurring field. As shown in Figure 9(b), we also generate two result images using $\omega_2 = 0.1, 0.9$. The sand stroke field is equal to edge field with a higher value of ω_2 , and otherwise as a blurring field. To control the visual effect of an edge, ω_2 are defined accordingly. The difference between results using a pair of values of ω_2 is the same as that for ω_1 in a sense. These two parameters seem to be grouped together, but the accurate relation between them has not been discovered because of a complex and successive process. However, this parameter ω_2 is still user-adjustable.

In the phase of sand painting field generation, $\omega_3 \in (0, 0.5]$ controls the weight balancing contributions of texture field and sand stroke field. As shown in Figure 9(c), we generate two result images using $\omega_3 = 0.01, 0.5$. With a smaller value of ω_3 , the sand painting field is approximatively equivalent to sand stroke field, and vice versa. As a result, the visual effect using a smaller ω_3 is a flat background without any texture. At the other extreme, the visual effect of the edge and saliency object are neutralized by the texture.

Overall, an appropriate parameter configuration is obviously required to generate a better visual effect, and one should fix these three parameters placidly and moderately, although our method provides the freedom and flexibility of adjusting them.

5.5. Computational Complexity. The computational complexity is a key problem in practical engineering applications, and

it is desired that result images with the style of sand painting can be generated in real time. Except for the parameter selection, the main time cost of the proposed framework lies in image data field generating and multiscale field coupling. The former consumes the time of order $O((2\text{round}(3\sigma_d/\sqrt{2})+1)^2hw)$, where the affiliated term is generally much less than hw . The latter repeatedly calculates the weighted sum in a pixel-by-pixel manner, which scans each pixel once taking time of about $O(hw)$ in a single run. Additionally, the field coupling is implemented several times, no more than 5 in our experience. Of course, the presented method for parameters selection costs time no more than $O(hw)$. Therefore, the time complexity of the proposed algorithm is approximatively linear in the size of the original image, that is, hw . Thus, the computational complexity analysis indicates that our fully automatic pipeline is efficient.

The running times are listed in Table 3. For the reference, we also list the average elapsed time of related methods. With the increase of the size of the input image, the generation time of all methods has increased. From the time cost's point of view, the BF and KF methods are the fastest, but noneffective, and the visual results are a little poor, which can be seen from the above sections. The GIMP method provides an acceptable result, but it is too time-consuming to be applied. For a medium size image, one would have to take some time to wait for the run of the GIMP method. In a certain sense, the longer the processing time, the better the visual effect. By comparison, our method slightly solved such a problem. For a true color image with an approximate size of 1024×768 , the average running time of our unoptimized MATLAB implementation is nearly 30 seconds on the PC, except the step of parameter selection of T_{salient} through the try and error method. From this perspective, the proposed method generally demonstrates an acceptable time performance. In addition, some image processing methods in GPU would be used to accelerate the rendering procedure [37, 38]. As future extensions, optimizations like parallelization through GPUs and multicore CPUs to compute multiscale field coupling can still be made; thus a significant speed-up can be expected.

5.6. The User Study. Here we conduct a user study to further validate the fact that our approach is useful for sand painting stylization and it is a better alternative to the GIMP method.

FIGURE 10: The scores from user survey.

We start by inviting 129 users to participate in this investigation, who volunteered from our university, and there are 48 sand painting novices, 26 sand painting enthusiasts, 50 undergraduates in digital media, and 5 professional teachers of artistic design. All of them declared that they have seen the real performance of sand painting and are with various related experiential knowledge. Then, the users are asked to generate rendering images by various methods considering the original images in Figure 8 as the input. Later, they score each result image according to a sliding scale of measure value, using a Likert scale in the range [0 10], and 10 means a perfect result. The evaluation covers five measures. (1) Similarity: we asked the users to score the similarity preserving of each result, and 10 means the most similar features. (2) Texture: each user measures how much the result appears as a sand-like texture, and 10 means the most beauty of texture feeling. (3) Color: the users are asked to evaluate the color themes and the harmony of color pattern, and 10 means the best visual effect. (4) Complexity: the users choose a score to measure the complexity of painting style, and 10 means the most possible to be used in real applications of sand painting stylization. (5) Time: each participant is asked to score the satisfaction of running time or response time, and 10 means the most satisfied.

Figure 10 shows a summary of the survey results. Our method obtains the highest score on similarity preserving, close to the BF method. From this perspective, the KF method and the GIMP method perform a not very good performance. One can also observe that our method yields better features on texture, color, and complexity. Though our method fails to take the least time to complete a stylization task, it is still acceptable by users since the average score is above 6. This user study shows that our method produces the results with overall highest quality and relatively less time consumption, which demonstrates that our method has better usability than the others, and it is well suited for sand painting stylization with none a priori knowledge. In general, we believe that our approach can be a good candidate for novice users, without prior knowledge or skills, to convert their photos into sand painting style.

5.7. Discussions. In this subsection, we provide a brief discussion on the proposal. Relating to research question (1) in Section 1, we generated various image data fields using different image features of the NSFC logo image and investigated the influence of the various features on an image data field, and then field coupling is implemented in order to create a sand painting filter. Relating to research question (2), we proposed a field coupling-based pipeline with a fully automatic mechanism for sand painting stylization (see Figure 2), which takes an original image as input and produces sand painting-like effects. In addition, multiscale field coupling is introduced with different fields of multiresolution for improving the artistic quality. An example of sand painting stylization is shown in Figure 7(e). Relating to research question (3), we conducted three groups of experiments using eight images and provided the visual comparisons (see Figure 8) and the aesthetic quality assessment (see Tables 1 and 2), as well as time performance analysis (see Table 3). Relating to research question (4), we investigated the effects of different parameters for the algorithm performance and provided the default parameters, in which three of them are the stylization control parameters. Generally, the field coupling-based method performs a good performance on the computer-based simulation of sand painting image, but the simulation of sand animation is still open and remains a challenging task.

6. Conclusions

In this paper, we focus on the sand art and proposed an IBAR method to simulate sand painting style image from photographs: first, the image features are obtained using image data field generation, and then a painterly filter based on field coupling is applied to the original image. We hope that this paper spurs others to consider more automatic painterly sand rendering, as opposed to the more interactive examples of the hardware medium that have been popular in NPR of sand painting [4, 5]. The experiment results verify the efficiency and feasibility of the proposed method, and our process is effective at generating sand painted-looking images.

Our methodology meets the following two essential rules for artistic stylization algorithms: (1) it is not supervised, and one does not have to provide any parameter value and (2) it is simple and innovative with respect to relevant literature.

Our method has some limitations, and there are a couple of issues that will be considered in the future research. (1) Sand painting artist also includes various styles, other than the one that has been simulated by the proposed method. Introducing other novel forms of image data field into the proposed method handling sand painting filter is currently under investigation and will be reported later. (2) Sand painting is far more than the single-frame image with sand style. The journey of artistic generation may be just as or more interesting than the result image. Thus, with the GPUs and multicore CPUs improvements of time performance, the extension of the proposed method to produce an animated sequence or video of multiframe images is well worth further studying.

Disclosure

Many images were used from Baidu Gallery under Baidu's policy (<https://www.baidu.com/duty/right.html>).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (no. 61402399), by the Natural Science Foundation of Guangdong, China (no. 2017A030307030), and by Foundation for Distinguished Young Teachers in Higher Education of Guangdong, China (no. YQ2014117).

References

- [1] J. E. Kyprianidis, J. Collomosse, T. Wang, and T. Isenberg, "State of the 'Art': a taxonomy of artistic stylization techniques for images and video," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 5, pp. 866–885, 2013.
- [2] D. Chi, "A natural image pointillism with controlled ellipse dots," *Advances in Multimedia*, vol. 2014, Article ID 567846, 17 pages, 2014.
- [3] 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, pp. 14559–14579, 2017.
- [4] R. H. Kazi, K.-C. Chua, S. Zhao, R. C. Davis, and K.-L. Low, "SandCanvas: A multi-touch art medium inspired by sand animation," in *Proceedings of the 29th Annual CHI Conference on Human Factors in Computing Systems, CHI 2011*, pp. 1283–1292, Canada, May 2011.
- [5] C.-F. Lin and C.-S. Fuh, "Uncle sand: A sand drawing application in ipad," in *Proceeding of Computer Vision, Graphics, and Image Processing Conference*, Nantou, Taiwan, 2012.
- [6] G. Song and K.-H. Yoon, "Sand image replicating sand animation process," in *Proceedings of the 19th Korea-Japan Joint Workshop on Frontiers of Computer Vision, FCV 2013*, pp. 74–77, Republic of Korea, February 2013.
- [7] M. Yang, X. He, C. Hu, T. Wang, and G. Yang, "Algorithm for interactive simulation of sand painting," *Journal of Computer-Aided Design and Computer Graphics*, vol. 28, no. 7, pp. 1084–1093, 2016.
- [8] X. Xiaochen, K. Liquan, H. Xie, and Y. Xiaowen, "Sand painting gesture recognition based on multi-touch," *Computer Engineering and Applications*, vol. 53, no. 1, pp. 244–248, 2017.
- [9] H. Fan, Z. Chen, and J. Li, "Image sand style painting algorithm," *Applied Mathematics & Information Sciences*, vol. 8, no. 2, pp. 765–771, 2014.
- [10] T. Wu, R. Hou, and L. Zhang, "Sand painting stylization using image filter," in *Proceedings of The 15th National Conference on Image and Graphics, NCIG '16*, 2016.
- [11] M. Hancock, T. Ten Cate, S. Carpendale, and T. Isenberg, "Supporting sandtray therapy on an interactive tabletop," in *Proceedings of the 28th Annual CHI Conference on Human Factors in Computing Systems, CHI 2010*, pp. 2133–2142, New York, NY, USA, April 2010.
- [12] M. Ura, M. Yamada, M. Endo, S. Miyazaki, and T. Yasuda, "A paint tool for image generation of sand animation style," *Human Interface*, vol. 11, no. 21, pp. 7–12, 2009.
- [13] P. Urbano, "The T. albipennis Sand Painting Artists," in *International Conference on Applications of Evolutionary Computation*, pp. 414–423, 2011.
- [14] T. Wu, J. Yang, and G. Ran, "Computational aesthetics analysis on sand painting style," *Journal of Frontiers of Computer Science and Technology*, vol. 10, no. 7, pp. 1021–1034, 2016.
- [15] H. Winnemöller, S. C. Olsen, and B. Gooch, "Real-time video abstraction," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 1221–1226, 2006.
- [16] M. Kuwahara, K. Hachimura, S. Eiho, and M. Kinoshita, "Processing of ri-angiocardigraphic images," *igital Processing of Biomedical Images*, pp. 187–202, 1976.
- [17] G. Papari, N. Petkov, and P. Campisi, "Artistic edge and corner enhancing smoothing," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2449–2462, 2007.
- [18] J. E. Kyprianidis, H. Kang, and J. Döllner, "Image and video abstraction by anisotropic Kuwahara filtering," *Computer Graphics Forum*, vol. 28, no. 7, pp. 1955–1963, 2009.
- [19] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 6th International Conference on Computer Vision (ICCV '98)*, pp. 839–846, Bombay, India, January 1998.
- [20] E. Arias-Castro and D. L. Donoho, "Does median filtering truly preserve edges better than linear filtering?" *The Annals of Statistics*, vol. 37, no. 3, pp. 1172–1206, 2009.
- [21] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [22] H. Kang, S. Lee, and C. K. Chui, "Coherent line drawing," in *Proceedings of the 5th International Symposium on Non-Photorealistic Animation and Rendering (NPAR '07)*, pp. 43–50, August 2007.
- [23] D. Li and Y. Du, *Artificial Intelligence with Uncertainty*, Chapman & Hall, Boca Raton, Fla, USA, 2007.
- [24] S. Wang, Wenyan Gan, D. Li, and D. Li, "Data field for hierarchical clustering," *International Journal of Data Warehousing and Mining*, vol. 7, no. 4, pp. 43–63, 2011.
- [25] S. Wang, J. Fan, M. Fang, and H. Yuan, "HGCUDF: Hierarchical grid clustering using data field," *Journal of Electronics*, vol. 23, no. 1, pp. 37–42, 2014.
- [26] S. Wang and Y. Chen, "HASTA: A hierarchical-grid clustering algorithm with data field," *International Journal of Data Warehousing and Mining*, vol. 10, no. 2, pp. 39–54, 2014.
- [27] J. Zhao and M. Jia, "Segmentation algorithm for small targets based on improved data field and fuzzy c-means clustering," *Optik - International Journal for Light and Electron Optics*, vol. 126, no. 23, pp. 4330–4336, 2015.
- [28] T. Wu, "Image data field-based framework for image thresholding," *Optics & Laser Technology*, vol. 62, pp. 1–11, 2014.
- [29] P. L. Rosin, "Unimodal thresholding," *Pattern Recognition*, vol. 34, no. 11, pp. 2083–2096, 2001.
- [30] R. E. W. Gonzalez and S. L. C. R. Eddins, *Digital Image Processing Using MATLAB*, Gatesmark Publishing, 2009.
- [31] D. Mould and P. L. Rosin, "A benchmark image set for evaluating stylization," in *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling*

- and Non-Photorealistic Animation and Rendering, *Expresive 16*, Eurographics Association, pp. 11–20, Aire-la-Ville, Switzerland, 2016.
- [32] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
 - [33] L. Zhang, L. Zhang, X. Mou, and D. Zhang, “FSIM: a feature similarity index for image quality assessment,” *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.
 - [34] L. Zhang, Y. Shen, and H. Li, “VSI: a visual saliency-induced index for perceptual image quality assessment,” *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4270–4281, 2014.
 - [35] A. Liu, W. Lin, and M. Narwaria, “Image quality assessment based on gradient similarity,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1500–1512, 2012.
 - [36] W. Xue, L. Zhang, X. Mou, and A. . Bovik, “Gradient magnitude similarity deviation: a highly efficient perceptual image quality index,” *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 684–695, 2014.
 - [37] M. Abdellah, A. Eldeib, and A. Sharawi, “High performance GPU-Based Fourier volume rendering,” *International Journal of Biomedical Imaging*, vol. 2015, Article ID 590727, 13 pages, 2015.
 - [38] J. Kyprianidis, H. Kang, and J. Döllner, “Anisotropic kuwahara filtering on the gpu,” in *GPU Pro - Advanced Rendering Techniques*, W. Engel, Ed., pp. 247–264, 2010.

Submit your manuscripts at
www.hindawi.com