Digital inpainting with SPH method

Fernando Alves Mazzini
IMPA
Rio de Janeiro, Brazil
falvesmazzini@gmail.com

Fabiano Petronetto do Carmo
Departamento de Matemática – UFES
Vitória, Brazil
fabiano.carmo@ufes.br

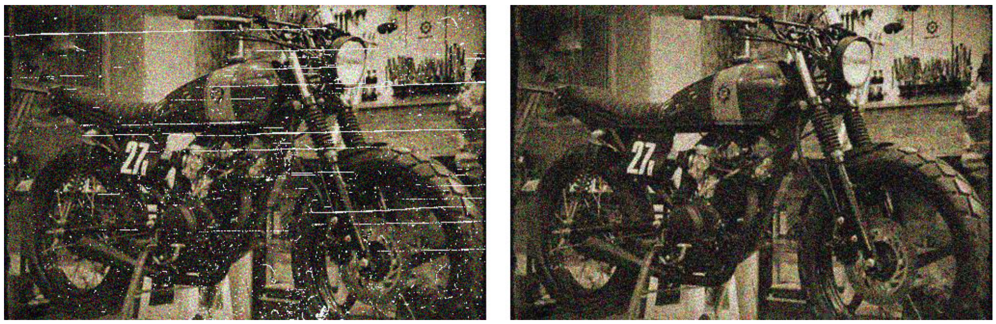


Fig. 1. Digital inpainting with our algorithm: corrupted image (left), restored image (right).

***Abstract*— Digital inpainting consists in reconstructing damaged parts of an image in order to restore its original aspect. In this paper, we propose the use of Smoothed Particle Hydrodynamics (SPH) method into the digital inpainting context. The formulation of the SPH method and its flexibility towards treatment on inpainting domains with complex geometry determine a simple algorithm. Moreover, we compute the particle approximation of the SPH method using the gather approach which ensures computational efficiency. We illustrate the effectiveness of our inpainting algorithm in a set of restored images where not only damaged images were recovered, but also parts of images obstructed by objects. The results achieved in affordable computational time are close to the ones obtained with traditional inpainting algorithms.**

***Keywords*- Digital inpainting; SPH method; gather approach.**

I. INTRODUCTION

Inpainting consists in restoring degraded parts of an image in order to restore its natural visual aspect. Digital inpainting seeks to translate artistic restoration concepts to a mathematical language, reproducing its results in damaged digital images. Once the user determines which image regions must be fixed, the inpainting algorithm uses the information from the surrounding area to fill the selected regions and returns an image, in which is expected for an observer who is not familiar with the original one not to be able to detect alterations [1]. The main digital inpainting algorithms, which can include the seminal work of Bertalmio et al [1] and some extensions [2], [3] can be described from the temporal evolution of the image based on a Partial Differential Equation. Numerical solutions are used in the traditional finite difference method due to direct formulation from the discrete model of an image. An alternative technique, producing good results, was proposed in [4], which repeatedly applies the convolution of the image with a Gaussian filter.

There are many topics of interest related to the digital inpainting. Among them, we would like to mention the image denoising. In this field, a interesting approach is presented in [5]. The Smoothed Particle Image Reconstruction (SPIR) algorithm makes use of the versatility granted by the SPH method to rebuild the image. SPH determines the value of a property over a particle by interpolating the known values on particles in its surroundings with no connectivity needed among the particles in the discretization.

Contributions: This paper presents an algorithm based on SPH method for digital inpainting, named as Gather - Smoothed Particle Image Reconstruction (G-SPIR). The proposed approach uses a radial interpolation provided by SPH method to fill damaged pixels in the target image. In contrast with previous denoising SPH method which uses scatter approach, our approach combines the definition of the area element geometrically with the gather approach in a simple and efficient algorithm able to reconstruct lost or deteriorated large parts of images.

A. Related work

Digital inpainting was initially proposed by Bertalmio et al [1], and their pioneer work is considered a reference for many other studies in the area. Being Ω the region to be inpainted and $\partial\Omega$ its boundary, most digital inpainting algorithms roughly consists in progressively prolong to Ω the isophotes from $\partial\Omega$ in a way that the continuity of the gray-level is preserved. Simultaneously, the regions defined by the prolongation of these isophotes are filled with color corresponding the information of the surrounding area. Thus, Ω is iteratively shrunk transporting the information of its frontier smoothly to its interior (Fig. 2).

Oliveira et al [4] have proposed a simple algorithm that reproduces results close to the ones obtained in [1] with a considerable reduction in execution time. The simpler version of this algorithm tends to introduce an undesirable blur where

(a) Damaged image (b) Intermediate step (c) Final result

Fig. 2. Progressive nature of a digital inpainting algorithm.

it crosses high contrast borders. This drawback can be corrected by applying diffusion barriers that limit the diffusion process in the Ω interior.

Di Blasi et al [5] have introduced a new approach for the reconstruction of noised images using the SPH method. Their algorithm, called SPIR (Smoothed Particle Image Reconstruction), uses the scatter approach in the discretization of SPH's integral representation. While it is effective in removing noise, the adopted approach is not appropriate for the problem of the digital inpainting.

II. Technical background

A. Smoothed Particle Hydrodynamics

The SPH method determines an approximation for a function and its derivatives using a local average. More precisely starting from a particle set, points that represent the simulation object and have properties inherent to the problem, each function and their derivatives in a determined point are approximated by weighted average of contributions given by particles near this point. The basic formulation of SPH can be divided into two steps: integral representation and approximation by particles. These steps will be briefly described in the following.

The integral representation of a function f with a domain $\Omega \subset \mathbb{R}^2$ is defined by the convolution of f with a smooth function $W_h : \mathbb{R}^2 \to \mathbb{R}$ called kernel. We use as kernel a radial basis function $W_h = W(R)$, where $R = \|\mathbf{x}\|/h$ with $W(r) = 0$ when $r > \kappa$. The parameter $\kappa \in \mathbb{R}$ is associated with kernel W. Hence, the integral representation is given by

$$f^h(\mathbf{u}) = \int_\Omega f(\mathbf{x}) W_h(\mathbf{u} - \mathbf{x}) d\Omega. \quad (1)$$

In SPH method, the kernel W_h is generally choosen as a differentiable function, with support domain and unitary integral.

The particle approximation of SPH method consists in replace the integral representation by summing the contributions of a point collection (or particles) arbitrarily distributed on problem domain. The compactness condition ensures that only a finite number of particles have to be considered in approximation. The effective domain of kernel W_h in a particle $\mathbf{u} \in \Omega$ is given by $V(\mathbf{u}) = \{\mathbf{x} \in \mathbb{R}^2;\ \|\mathbf{x} - \mathbf{u}\| \leq \kappa h\}$ and is called support domain of $\mathbf{u}$. Hence, the $f(\mathbf{u})$ approximation is made by computing the average of $f(\mathbf{x})$ values, obtained by the contribution of the neighboring particles $\mathbf{x}$ from the particle $\mathbf{u}$:

$$f^h(\mathbf{u}) = \sum_{\mathbf{x} \in V(\mathbf{u})} f(\mathbf{x}) W_h(\mathbf{u} - \mathbf{x}) V_x, \quad (2)$$

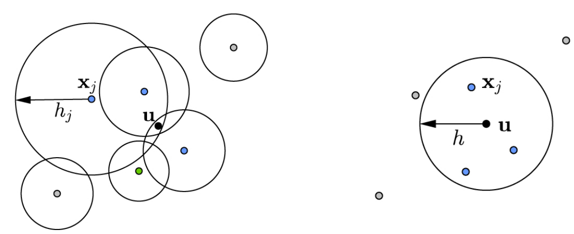
where the infinitesimal discrete area $d\Omega$ is replaced by the area element V_x. The success in the approximation depends directly on the smoothing length h or, more precisely, on the number of nearest neighboring particles (NNP). A small value of h may result in an insufficient number of neighboring particles in the support domain. On the other hand, to achieve a particular number of particles it is necessary a large value of h, there will be local smoothing properties. Hence, determining the NNP must occur before computing the SPH method. We will see that there are two interpretations to the search of neighboring particles: the gather approach and the scatter approach. Both differ in the way of inferring whether a given particle is a neighboring particle. Choosing the most suitable approach depends on the kind of problem to be solved.

B. Determining the nearest neighboring particles

In the scatter approach, the smoothing length h_j refers to the support domain radius of the particle $\mathbf{x}_j$, and it is considered one of the NNP if the particle $\mathbf{u}$ belongs to its support domain (Fig. 3 (a)). By fixing a k number of NNP, the values of h_j must be increased until $\mathbf{u}$ belongs to k support domains.

This approach is is numerically efficient in noise treatment of images, once the pixels to be restored are surrounded by color data pixels, as proposed by Di Blasi et al. [5] in their Smoothed Particle Image Reconstruction (SPIR) method. However, it becomes ineffective in the restoration of a great number of gathered pixels, frequent event in the image restoration field, which is the purpose of digital inpainting.

In the gather approach, the smoothing length h refers to the support domain radius of particle $\mathbf{u}$ and its NNP correspond to the $\mathbf{x}_j$ particles contained in its support (Fig. 3 (b)). This, by fixing a positive integer k, the value h must be increased gradually until it reaches a k number of NNPs contained in support domain of $\mathbf{u}$ (k nearest neighbors).



(a) Scatter approach (b) Gather approach

Fig. 3. Note that the third nearest particle (green) was not classified as NNP (blue) in scatter approach (a). In gather approach (b) the NNP (blue) are the k nearest particles.

III. Our approach

Our proposed method consists in applying the SPH method in the digital inpainting context, using a gather approach in the particle approximation step. This choice led to results close to the ones obtained with the known textbook methods.

In this work, we define the same area element in all particles, which represents the geometric property of the infinitesimal element. This definition avoids the use of an unbalanced smoothing kernel function even when a large image region is restored, allowing the use of our method in traditional image inpainting cases.

Given a degraded image, the user supplies a mask that specifies the regions (Ω) to be restored and a number $k \in \mathbb{N}$ of NNP for each damaged pixel in Ω. For each damaged pixel, the algorithm searches for the k nearest neighbors using the gather approach (through the kNN algorithm), and the SPH assigns color to the pixel based on the nearest neighbors color interpolation. The whole process is schematized in Fig. 4.

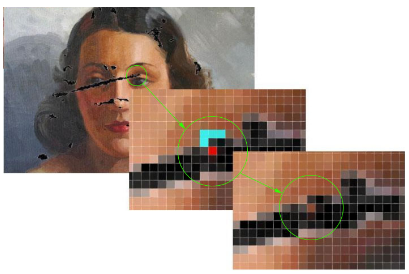


Fig. 4. The inpainting steps. From top-left to bottom-right: damaged image; search for neighbors (blue); assigning a color to the retouched pixel (red).

A. Implementation

In the algorithm presented in this paper, the image to be restored

$$I : [1, a] \times [1, b] \subset \mathbb{N}^2 \to [0, 255] \subset \mathbb{N}, \tag{3}$$

will be discretized in a set of particles $\{p_1, p_2, \cdots, p_n\}$ with $n = a \cdot b$ where each particle p_k represents the pixel coordinate $(i_k, j_k) \in [1, a] \times [1, b]$ and the color $I_k = I(i_k, j_k)$ stored in this pixel.

Besides, the inpainting domain Ω is set from an image

$$M : [1, a] \times [1, b] \subset \mathbb{N}^2 \to [0, 255] \subset \mathbb{N}, \tag{4}$$

for which is assigned a zero value only for inpainting pixels. That is,

$$p_k \in \Omega \iff M(i_k, j_k) = 0.$$

A particle $p_k \in \Omega$ is restored from the equation

$$I_k = \sum_{j \in N_k} I_j W_{kj} V_j, \tag{5}$$

where the sum runs through all neighbor particles of p_k, denoted as N_k, $W_{kj} = W_h(r_{kj})$ with r_{kj} given by the distance between particles p_k and p_j, and V_j is the area element of the particle p_j.

The k NNP spotting should be done before performing any computation with the SPH method. Given a particle p_i, since the kNN algorithm has determinate the N_i, set of k nearest neighbors outside Ω, the smoothing length h_i is given by

$$h_i = \alpha \max\{d_{ij};\ j \in N_i\},$$

where d_{ij} is the euclidean distance between particles p_i and p_j, and α is a scaling parameter of the support radius wich avoids that contribution of distant neighbors particles of particle p_i be almost zero. The discrete area element V_j of particle p_j is given by $V_j = 1/n$.

The choice of k to kNN algorithm is subjective. The tests showed that increasing the value of k incurs the increase of blurring in inpainted regions, which may be desirable or not, depending on each case.

The simple formulation of the proposed technique is translated into pseudo-code below that describes the G-SPIR algorithm.

Data: I: damaged image
Ω: mask
k: kNN parameter
Result: R: inpainted image

%Compute NNP for each damaged pixel p_i
for *each pixel* p_i *to reconstruct* **do**
| Compute N_i with $N_i \cap \Omega = \emptyset$
end
%Start restored image: copy outside Ω pixels
$R = I$
%Evaluate pixel color value for each damaged pixel p_i
for $p_i \in \Omega$ **do**
| Compute $R(p_i)$ with equation (5)
end

Algorithm 1: G-SPIR algorithm

IV. Results and Comparisons

Next, we present some results obtained by applying the G-SPIR algorithm. The tests were sorted into three categories, often mentioned in digital inpainting textbooks: text-over-image removal, restoration of degraded images, and object-over-image removal. The algorithm and its variations (color images treatment) were implemented using MATLAB® and all tests were run using Intel® Core™ i3 2.13 GHz processor with 4 GB RAM. In our results, we use the parameters $k = 4$ and $\alpha = 1.5$ in G-SPIR algorithm.

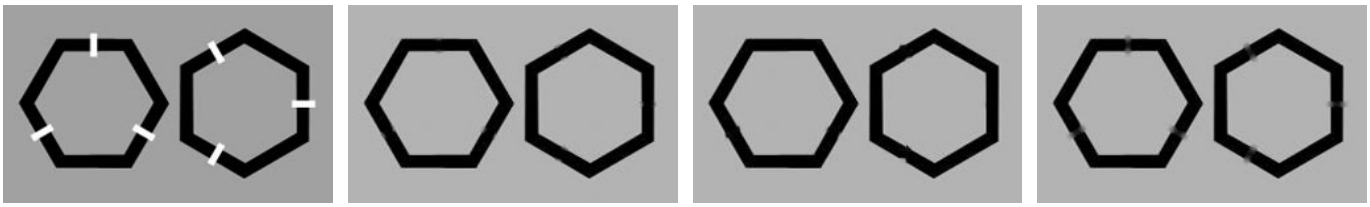
Fig. 1 and Fig. 5 show the restorations of degraded images. The inpainting domain Ω in Fig. 1 has 5716 pixels and the inpainting time was 3.35 seconds. For comparison purposes, Fig. 5 also shows the results obtained from the BSCB algorithm, from Bertalmio et al [1] and Fast algorithm, from Oliveira et al [4]. Fig. 6 shows the results obtained (G-SPIR, BSCB and Fast) from the object-over-image removal in a synthetic image, with the intention of evaluating the efficiency of the algorithm in high-contrast edges connection. In Fig. 7, (a) and (b), we see an object-removal example in a real scene, while (c) and (d) show a text removal example. All results are accompanied by the number of pixels in inpainting domain ($\#\Omega$) and its inpainting times shown in figures subtitles.

V. Concluding remarks

In this work we present a digital inpainting algorithm with a simple formulation that combines numerical efficiency of data

(a) Original image ($\#\Omega = 5126$) (b) G-SPIR (1.53 s) (c) BSCB (450.53 s) (d) Fast (7.73 s)

Fig. 5. Restoration of degraded images.



(a) Original image ($\#\Omega = 1153$) (b) G-SPIR (4.59 s) (c) BSCB (2396.91 s) (d) Fast (9.74 s)

Fig. 6. Reconnection of high-contrast edges. Note the slight differences in reconnecting the edges on each result. G-SPIR presents a good result, but with a slight defect in the connections. BSCB could not deal properly with the propagation direction of edges, while Fast produces a blurring in inpainted region (without the use of diffusion barriers).

(a) Original image ($\#\Omega = 6158$) (b) G-SPIR (3.38 s) (c) Original image ($\#\Omega = 6626$). (d) G-SPIR (1.53 s)

Fig. 7. Object-over-image removal:(a) and (b); Text-over-image removal: (c) and (d).

Fig. 8. Limitation: texture is not reproduced.

in the surrounding area with the versatility of the SPH method to assign colors in damaged pixels. The results obtained with the G-SPIR algorithm are close, in their majority, to the ones obtained with representative inpainting (BSCB and Fast). In general, G-SPIR inpainting time was slightly faster than the Fast algorithm. However, like other inpainting algorithm previously mentioned, it was not able to transport texture to the inpainting domain when filling a large area (Fig. 8).

For future studies we intend to analyze the viability of other SPH operators to enhance results, the influence of the k value (NNP amount) on the quality of inpainting, and improve the method with the use of diffusion barriers [4] to avoid blurring in regions where inpainting domain crosses high-contrast edges.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 417–424.

[2] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–355.

[3] W. Au and R. Takei, "Image inpainting with the navier-stokes equations," *Final Report, APMA*, vol. 930, 2001.

[4] M. M. Oliveira, B. Bowen, R. McKenna, and Y.-S. Chang, "Fast digital image inpainting," in *Appeared in the Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain*, 2001, pp. 106–107.

[5] G. Di Blasi, E. Francomano, A. Tortorici, and E. Toscano, "A smoothed particle image reconstruction method," *Calcolo: a quarterly on numerical analysis and theory of computation*, vol. 48, no. 1, pp. 61–74, 2011.