MATHEMATICAL MODELING AND COMPUTING, Vol. 9, No. 3, pp. 536–546 (2022)

A nonlinear fractional partial differential equation for image inpainting

Gouasnouane O.[1,2], Moussaid N.[1], Boujena S.[2], Kabli K.[2]

[1] *University Hassan II of Casablanca, FST Mohammedia, Laboratory of Mathematics, Computer Science and Applications (LMCSA), PO Box 146, Mohammedia, Morocco*

[2] *University Hassan II of Casablanca, Ain-Chock Sciences Faculty, Laboratory of Modelisation, Analysis, Control and Statistics (MACS), Km 8 Route d'El Jadida, B.P 5366 Maarif, Casablanca, Morocco*

(Received 5 January 2022; Accepted 19 April 2022)

Image inpainting is an important research area in image processing. Its main purpose is to supplement missing or damaged domains of images using information from surrounding areas. This step can be performed by using nonlinear diffusive filters requiring a resolution of partial differential evolution equations. In this paper, we propose a filter defined by a partial differential nonlinear evolution equation with spatial fractional derivatives. Due to this, we were able to improve the performance obtained by known inpainting models based on partial differential equations and extend certain existing results in image processing.
The discretization of the fractional partial differential equation of the proposed model is carried out using the shifted Grünwald–Letnikov formula, which allows us to build stable numerical schemes.
The comparative analysis shows that the proposed model produces an improved image quality better or comparable to that obtained by various other efficient models known from the literature.

Keywords: *image processing, image inpainting, fractional calculus, fractional order partial differential equation, nonlinear diffusion, fractional derivative.*

2010 MSC: 35K55, 65M06, 68U10, 94A08, 65K05 **DOI:** 10.23939/mmc2022.03.536

1. Introduction

Inpainting technique was first appeared in the framework of digital restoration in the work of Bertalmio et al. [1]. It consists of filling in the missing areas or modifying the damaged ones in a non-detectable way by an observer not familiar with the original images. Digital inpainting serves a wide range of applications, such as text removal, restoring old photos, and creating special effects such as object disappearance from a scene.

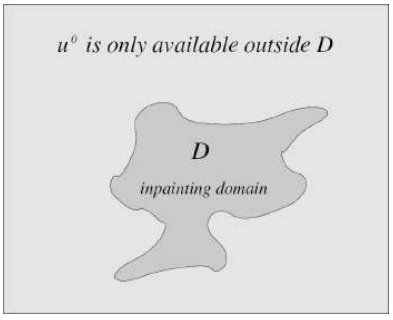

Fig. 1. Inpainting domain.

To scalize, let u_0 be a given image defined on an image domain Ω, the problem is to reconstruct the original image u in the damaged domain $D \subset \Omega$, called the inpainting domain (Figure 1).

Several techniques and methods of inpainting have been proposed in recent decades [2–5]. The first, and widely used approach is due to Bertalmio, Sapiro, Caselles, and Ballester [1], their model is based on observations about the work of museum artists, who restorate old paintings. Their approach follows the principle of prolongating the image intensity in the direction of the level lines, sets of image points in image with constant grayvalue, arriving at the hole which is the incomplete part of the image. This results in solving a discrete approximation of the PDE:

$$\frac{\partial u}{\partial t} = \nabla^{\perp} u \cdot \nabla \Delta u, \tag{1}$$

which is implemented to fetch the direction of the level lines in which the intensity of the image needs to be extended in order to fill in the gaps. Equation (1) is a transport equation for the image smoothness modeled by Δu along the level lines of the image. Here $\nabla^{\perp} u$ is the perpendicular gradient of the image function u, i.e., it is equal to $(-\frac{\partial u}{\partial y}, \frac{\partial u}{\partial x})$. Their algorithm produces very impressive results, however, usually requires several minutes for the inpainting of relatively small areas. Note that Bertalmio et al.'s model is a third-order nonlinear partial differential equation.

Inspired by the work of Bertalmio et al. Chan and Shen proposed in [6] a new approach for the inpainting technique whose objective is to create a model motivated by existing denoising/segmentation methods and mathematically easier to understand and analyze. Their approach is based on the best known image processing model, the total variation (TV) model, it results in the action of anisotropic diffusion inside the inpainting domain, which preserves the contours while diffusing homogeneous regions and small oscillations such as noise. Specifically, the corresponding steepest descent equation reads

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + \lambda_e(u_0 - u), \tag{2}$$

and is valid on the entire domain Ω. Where λ_e is the extended Lagrange multiplier $\lambda_e = \lambda(1 - \chi_D)$, χ_D is the characteristic function of the inpainting domain D, and λ is given by the TV denoising scheme [6]. The TV inpainting is effective for small area that require anisotropic diffusion based on the contrast of the isophotes, in this model the conductivity coefficient defined by $g(|\nabla u|) = \frac{1}{|\nabla u|}$ and depends on the strength of an isophote.

The Curvature diffusion model (CDD) [7] was then developed and allowed to extend that of TV by introducing an anisotropic diffusion conductivity coefficient which depends on the curvature of level lines. This new approach is a third order diffusion PDE, unlike for the TV model where it is second order one. This EDP is the following

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\frac{c(\kappa)}{|\nabla u|}\nabla u\right) + \lambda(u_0 - u), \tag{3}$$

where c is a continuous function, which penalizes large curvatures, and encourages diffusion when the curvature is small. Typical example for the conductivity coefficient c which, in fact, have been used in [7], is: $c(s) = s^p$, $s > 0$, $p \geqslant 1$. The curvature κ is given by: $\kappa = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)$.

In recent years, fractional calculus began to shift from fundamental mathematics formulations to applications in various domains including biology, physics and mechanics amount to replace the classical derivatization in an evolution PDE with a fractional order derivative. In particular in the image processing field [8–14] the nonlocal properties of fractional differential models appear to give better results than traditional differential models. Many researchers have applied fractional calculus different steps of image processing particularly in inpainting. In [15, 16] Zhang et al. have proposed a fractional order model to improve the inpainting's performances. For this model the fractional differential is

$$\begin{aligned}\frac{\partial u}{\partial t} &= \overline{(-1)^{\alpha}}\operatorname{div}^{\alpha}\left(\frac{\nabla^{\alpha} u}{(|\nabla^{\alpha} u|)^{2-p}}\right) + \lambda_D(u - u_0)\\ &= \overline{(-1)^{\alpha}}\operatorname{curv}^{\alpha} u, \quad p \in [1, 2],\end{aligned} \tag{4}$$

where ∇^{α} designate a fractional order derivative and $\operatorname{curv}^{\alpha}$ is a fractional curvature given by

$$\operatorname{curv}^{\alpha} = D^{\alpha}_{x^-}\left(\frac{D^{\alpha}_{x^+} u}{(D^{\alpha}_{x^+} u + D^{\alpha}_{y^+} u + \varepsilon)^{\frac{2-p}{2}}}\right) + D^{\alpha}_{y^-}\left(\frac{D^{\alpha}_{y^+} u}{(D^{\alpha}_{x^+} u + D^{\alpha}_{y^+} u + \varepsilon)^{\frac{2-p}{2}}}\right). \tag{5}$$

The fractional order is developed based on Grünwald–Letnikov (GL) definition.

Bosch and Stoll develope a modified Cahn–Hilliard vector inpainting model for grayscale images that consists of PDEs with fractional derivatives in space [17]. This model is the generalized version of Cahn–Hilliard's binary image inpainting method for grayscale images. It is achieved by replacing whole order differentiation with fractional order differentiation. To better perform the inpainting results, we propose in this article, a new version of the Perona–Malik model [18] for which we work with fractional order derivatives with respect to space. In this new model the p-shifted Grünwald–Letnikov formula is applied to implement the numerical scheme instead of the discrete Fourier transform (DFT) which imposes the period boundary condition on the proposed equations. Our algorithm is easy to implement compared to other algorithms that use discrete Fourier transform in which an $m \times m$ input image is folded over the lines $x = m - 1$ and $y = m - 1$ to produce an image of $2m \times 2m$ to obtain, at the same time, symmetrical and continuous edges of the original image but the size of the extended image requires additional memory and affects the cost and precision of the calculation.

2. Fractional derivatives

The best known definitions of the fractional derivative are Grünwald–Letnikov, Riemann–Liouville, and Caputo, the first two are the most used in the field of image processing. The definition of Grünwald–Letnikov is obtained from the definition of the integral power derivative, while the definitions of Riemann–Liouville and Caputo are taken from the Cauchy formula of integral order.

The left and right Grünwald–Letnikov derivatives of order $\alpha > 0$ for a given function $f(x)$, $x \in [a, b]$ are defined by

$$ {}_{GL}\mathrm{D}^{\alpha}_{a,x} f(x) = \lim_{\substack{h \to 0 \\ Nh = x-a}} \left(\frac{1}{h}\right)^{\alpha} \sum_{i=0}^{N} (-1)^{\alpha} \binom{\alpha}{i} f(x - ih), \tag{6} $$

and

$$ {}_{GL}\mathrm{D}^{\alpha}_{x,b} f(x) = \lim_{\substack{h \to 0 \\ Nh = b-x}} \left(\frac{1}{h}\right)^{\alpha} \sum_{i=0}^{N} (-1)^{\alpha} \binom{\alpha}{i} f(x + ih), \tag{7} $$

respectively.

The left and right Riemann–Liouville derivatives with order $\alpha > 0$ of the given function $f(x)$, $x \in [a, b]$ are defined as

$$ {}_{RL}\mathrm{D}^{\alpha}_{a,x} f(x) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_a^x \frac{1}{(x-s)^{\alpha+1-n}} f(s)\, ds, $$

and

$$ {}_{RL}\mathrm{D}^{\alpha}_{x,b} f(x) = \frac{(-1)^n}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_x^b \frac{1}{(s-x)^{\alpha+1-n}} f(s)\, ds, $$

respectively, where Γ is the Euler's gamma function and n is a positive integer satisfying $n-1 \leqslant \alpha \leqslant n$.

If $f(x)$ is suitably smooth, i.e. $f \in \mathcal{C}^m[a, b]$ the Grünwald–Letnikov and Riemann–Liouville derivatives of $f(x)$ are equivalent but in general the two definitions are not equivalent [19].

3. Inpainting and nonlinear fractional order model

Our approach consists to extend the classical Perona–Malik model by introducing the fractional order derivatives with respect to the spatial variables as follows,

$$ \begin{cases} \dfrac{\partial u}{\partial t} - \lambda \operatorname{div}^{\alpha}(\mu(|\nabla^{\alpha} u|)\nabla^{\alpha} u) = 0, \quad \text{in } Q, \\ u(x, 0) = u_0(x), \quad \forall x \in \Omega, \\ \dfrac{\partial u}{\partial \nu}(x, t) = 0, \quad \forall x \in \partial\Omega,\ \forall t \in [0, T], \end{cases} \tag{8} $$

where u_0 is a given (distorted) image occupying a bounded domain Ω in $\mathbb{R}^d$ (with $d \leqslant 3$ in most applications) for which boundary is $\partial\Omega$. Q is defined by $Q = \Omega \times [0,T]$, for some given $T > 0$, and ν is the external vector normal to the domain boundary. $\lambda = \chi_D$, the operators $\mathrm{div}^\alpha = \sum_{i=1}^{d} \frac{\partial^\alpha}{\partial x_i^\alpha}$ and $\nabla^\alpha = \left(\frac{\partial^\alpha}{\partial x_1^\alpha}, \frac{\partial^\alpha}{\partial x_2^\alpha}, \ldots, \frac{\partial^\alpha}{\partial x_d^\alpha}\right)$.

For numerical approximation of spatial fractional order derivatives we can use the formula derived from the Grünmald–Letnikov definition given by

$$\left({}_{GL}\mathrm{D}^\alpha_{0,x} f(x)\right)_{x=x_k} \approx \left(\frac{1}{h}\right)^\alpha \sum_{i=0}^{k} \omega_i^{(\alpha)} f(x_k - ih),$$

where $\omega_i^{(\alpha)} = (-1)^i \binom{\alpha}{i}$ is the polynomial coefficients of $(1-z)^\alpha$, and can be calculate by the following recurrence formula

$$\begin{cases} \omega_0^{(\alpha)} = 1, \\ \forall i \in \{1, 2, \ldots, N\}, \quad \omega_i^{(\alpha)} = \left(1 - \frac{\alpha+1}{i}\right) \omega_{i-1}^{(\alpha)}. \end{cases} \tag{9}$$

The above approximation is a convergent approximation of order 1 for any $\alpha > 0$ [19]. Nevertheless the standard Grünwald–Letnikov formula can lead to unstable numerical schemes in the resolution of fractional differential equations for $1 < \alpha < 2$ [19]. To avoid this inconvenient the p-shifted Grünwald–Letnikov formula is useful for constructing stable numerical schemes.

The right shifted Grünwald–Letnikov formula is defined by

$$\left({}_{GL}\mathrm{D}^\alpha_{a,x} f(x)\right)_{x=x_k} \approx \left(\frac{1}{h}\right)^\alpha \sum_{i=0}^{k+p} \omega_i^{(\alpha)} f(x - ih + p). \tag{10}$$

This shifted Grünwald–Letnikov formula gives a first order accuracy; the best performance comes from minimizing $|p - \alpha/2|$ [20, 21]. If $1 < \alpha \leqslant 2$, the optimal choice is $p = 1$. The case of $\alpha = 2$ reduces to the second order central difference method for the second order classical derivative.

To solve numerically the proposed model (8) we treat at first the fractional order gradient.

Let u be an image with $(n+1) \times (m+1)$ pixels. According to (10), the discrete fractional-order derivatives at the point (i, j) with the order α along the horizontal and the vertical direction, are respectively

$${}_{GL}\mathrm{D}^\alpha_x u(i,j) = \left(\frac{1}{h}\right)^\alpha \sum_{k=0}^{k+p} \omega_i^{(\alpha)} u(k,j), \quad \forall j \in \{0, 1, 2, \ldots, m\}. \tag{11}$$

And

$${}_{GL}\mathrm{D}^\alpha_y u(i,j) = \left(\frac{1}{h}\right)^\alpha \sum_{k=0}^{k+p} \omega_i^{(\alpha)} u(i,k), \quad \forall i \in \{0, 1, 2, \ldots, n\}. \tag{12}$$

Using the above formulas and by adopting the matrix form, the discretization of the fractional order gradient vector with α order is given, respectively for $j = 0, 1, 2, \ldots, m$ and for $i = 0, 1, 2, ..., n$ by

$$\begin{pmatrix} \frac{\partial^\alpha}{\partial x^\alpha} u_{0,j} \\ \frac{\partial^\alpha}{\partial x^\alpha} u_{1,j} \\ \frac{\partial^\alpha}{\partial x^\alpha} u_{2,j} \\ \vdots \\ \frac{\partial^\alpha}{\partial x^\alpha} u_{n-2,j} \\ \frac{\partial^\alpha}{\partial x^\alpha} u_{n-1,j} \\ \frac{\partial^\alpha}{\partial x^\alpha} u_{n,j} \end{pmatrix} = \left(\frac{1}{h}\right)^\alpha B_n^{(\alpha)} \begin{pmatrix} u_{0,j} \\ u_{1,j} \\ u_{2,j} \\ u_{3,j} \\ \vdots \\ u_{n-1,j} \\ u_{n,j} \end{pmatrix} \tag{13}$$

$$\begin{pmatrix} \frac{\partial^\alpha}{\partial y^\alpha}u_{i,0} \\ \frac{\partial^\alpha}{\partial y^\alpha}u_{i,1} \\ \frac{\partial^\alpha}{\partial y^\alpha}u_{i,2} \\ \vdots \\ \frac{\partial^\alpha}{\partial y^\alpha}u_{i,m-2} \\ \frac{\partial^\alpha}{\partial y^\alpha}u_{i,m-1} \\ \frac{\partial^\alpha}{\partial y^\alpha}u_{i,m} \end{pmatrix} = \left(\frac{1}{h}\right)^\alpha B_m^\alpha \begin{pmatrix} u_{i,0} \\ u_{i,1} \\ u_{i,2} \\ u_{i,3} \\ \vdots \\ u_{i,m-1} \\ u_{i,m} \end{pmatrix}, \tag{14}$$

where $u_{i,j} = u(i,j)$ for $(i,j) \in \{0,1,2,\ldots,n\} \times \{0,1,2,\ldots,m\}$,

$$B_n^{(\alpha)} = \begin{pmatrix} \omega_1^\alpha & \omega_0^\alpha & 0 & 0 & 0 & \cdots & 0 \\ \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 & 0 & \cdots & 0 \\ \omega_3^\alpha & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \omega_{n-2}^\alpha & \omega_{n-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 \\ \omega_{n-1}^\alpha & \omega_{n-2}^\alpha & \omega_{n-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha \\ \omega_n^\alpha & \omega_{n-1}^\alpha & \omega_{n-2}^\alpha & \omega_{n-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha \end{pmatrix}$$

and

$$B_m^{(\alpha)} = \begin{pmatrix} \omega_1^\alpha & \omega_0^\alpha & 0 & 0 & 0 & \cdots & 0 \\ \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 & 0 & \cdots & 0 \\ \omega_3^\alpha & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \omega_{m-2}^\alpha & \omega_{m-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha & 0 \\ \omega_{m-1}^\alpha & \omega_{m-2}^\alpha & \omega_{m-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha & \omega_0^\alpha \\ \omega_m^\alpha & \omega_{m-1}^\alpha & \omega_{m-2}^\alpha & \omega_{m-3}^\alpha & \cdots & \omega_2^\alpha & \omega_1^\alpha \end{pmatrix}.$$

We discretize the scaling interval $[0,T]$. Choosing $N \in \mathbb{N}$ we obtain the length of uniform discrete scale step $\Delta t = \frac{T}{N}$. The nonlinear terms of the equation are treated from the previous scale step while the linear terms are considered on the current scale level.

For every $k = 1,\ldots,N$, we look for a function u^k, a solution of the equation

$$\frac{u^k - u^{k-1}}{\Delta t} - \lambda \,\mathrm{div}^\alpha \left(\mu(|\nabla^\alpha u^{k-1}|)\nabla^\alpha u^{k-1}\right) = 0. \tag{15}$$

4. Numerical experiments and interpretation

In this section, we show typical digital examples and applications of the model to retouching an old image, removing text from an image, restoring damaged regions in an image as well as to the special effect which consists of removing an object from a scene.

The performance of the proposed model will be compared to several well know inpainting models [6, 7, 16].

The discrete scaling step is selected to be $\Delta t = 1.\mathrm{E}-2$ for the four models. We set the nonlinear diffusion coefficient $\mu(s) = 1/\sqrt{1+s^2}$. The values of the fractional order are taken with the range $\alpha \in]1,2]$.

All testing problems were implemented using Matlab 2018a on Intel(R) Core(TM) i5 at 1.8GHz, 6 GB memory, system type 64-bit and Windows 10.

Four performance metrics are considered here to evaluate the performance of restored image; peak signal to noise ratio (PSNR), signal to noise ratio (SNR), mean-squared error (MSE) and structural similarity index measure (SSIM).

The mean-squared error (MSE) between two images I_1 and I_2 is:

$$\mathrm{MSE} = \frac{1}{n \times m} \sum_{n,m} \left(I_1(n,m) - I_2(n,m)\right)^2, \tag{16}$$

n and m are the number of rows and columns in the input images, respectively,

$$\mathrm{SNR} = 10 \times \log_{10} \frac{\sum_{n,m} I_2^2}{\sum_{n,m} \left(I_1 - I_2\right)^2}, \tag{17}$$

I_1 restored image and I_2 original image,

$$\mathrm{PSNR} = 10 \times \log_{10} \frac{R^2}{\mathrm{MSE}}, \tag{18}$$

R is the maximum fluctuation in the input image data type. For example, if the input image has a double-precision floating-point data type, then R is 1. If it has an 8-bit unsigned integer data type, R is 255, etc.

The Structural Similarity Index Measure (SSIM) is used for measuring the similarity between two images,

$$\mathrm{SSIM} = \frac{2\mu_x\mu_y + C1}{\mu_x^2 + \mu_y^2 + C1} \times \frac{2\sigma_{xy} + C2}{\sigma_x^2 + \sigma_y^2 + C2}, \tag{19}$$

where μ_x, μ_y, σ_x, σ_y, and σ_{xy} are the means, standard deviations, and cross-covariance for images x, y. $C1$ and $C2$ denote constants used to maintain stability.

Figures 2–6 show the comparison results of inpainting images between the proposed model and the previously mentioned baseline methods.

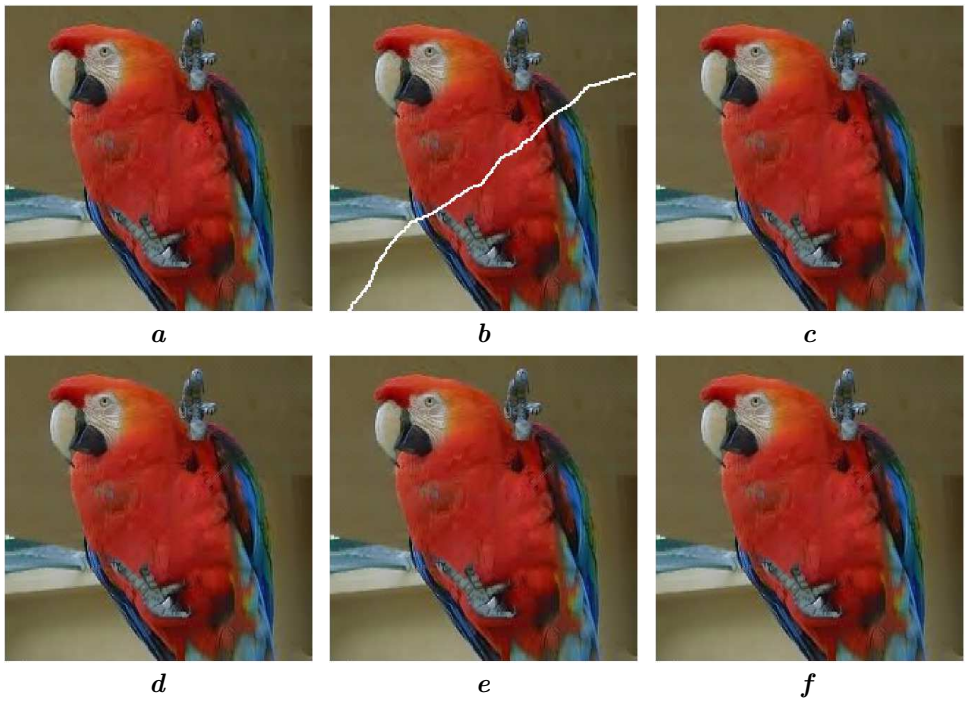

Fig. 2. Comparison of inpainting models for damaged image of parrot. (***a***) Clean image (650×420). (***b***) Damaged image. (***c***) Processed by the proposed model with $\alpha = 1.8$, PSNR = 32.22, SNR = 22.43, MSE = 62.25, SSIM = 0.942. (***d***) processed by Chan and Shen model, PSNR = 29.51, SNR = 19.85, MSE = 86.31, SSIM = 0.868. (***e***) Processed by CDD model, PSNR = 30.62, SNR = 20.04, MSE = 77.52, SSIM = 0.901. (***f***e) Processed by Zhang model, PSNR = 30.62, SNR = 20.38, MSE = 74.28, SSIM = 0.903.

Figure 2 shows an parrot image with the region to be inpaint. Notice that the inpainting region after applying the proposed model is filled effectively and contours are well reconstructed and recovered. The results were compared with other popular methods in the literature, the measures of proposed inpainting technique achieve the highest values.

In Tables 1–3 we show numerical results obtained with various values of fractional order α. We can observe than the PSNR and SNR reaches a maximum at $\alpha = 1.8$.

Table 1. The effect of the parameter α on PSNR, SNR, MSE and SSIM for parrot image.

α	PSNR	SNR	MSE	SSIM
1.2	27.93	19.87	102.85	0.813
1.4	29.76	21.03	97.42	0.878
1.6	30.51	21.75	81.13	0.913
1.7	31.93	22.02	68.57	0.931
1.8	32.22	22.43	62.25	0.942
1.9	31.85	21.84	71.14	0.914
2	28.73	19.71	101.51	0.807

Table 2. The effect of the parameter α on PSNR, SNR, MSE and SSIM for canyon image.

α	PSNR	SNR	MSE	SSIM
1.2	28.93	26.98	39.48	0.897
1.4	30.82	28.12	31.57	0.97
1.6	32.34	29.83	29.9	0.981
1.7	33.95	31.01	27.75	0.987
1.8	34.86	31.97	21.21	0.99
1.9	32.74	30.82	28.07	0.985
2	30.13	28.63	33.01	0.961

Table 3. The effect of the parameter α on PSNR, SNR, MSE and SSIM for Sherlock image.

α	PSNR	SNR	MSE	SSIM
1.2	40.85	38.8	13.02	0.97
1.4	43.1	40.75	10.48	0.981
1.6	44.76	42.04	8.91	0.987
1.7	46.68	44.13	6.77	0.998
1.8	47.56	44.67	5.13	0.9984
1.9	47.02	44.11	6.38	0.997
2	45.62	41.95	9.57	0.989

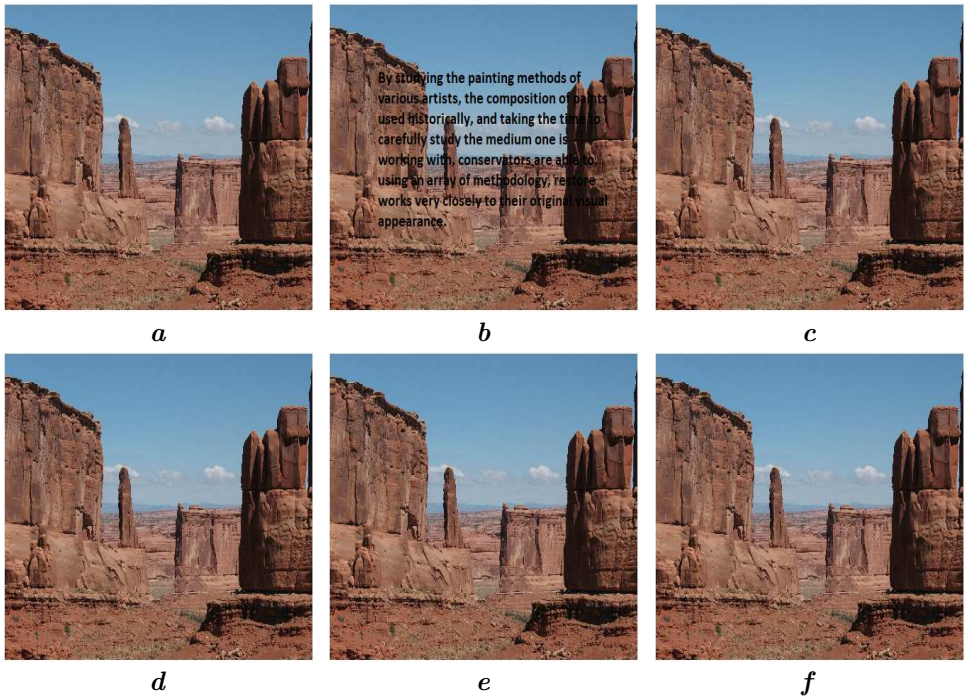

Fig. 3. Results for text removing corresponding to the canyon image. (***a***) Clean image (1536×2048). (***b***) Image with overlaid text, PSNR = 19.7, SNR = 16.81, MSE = 696.17, SSIM = 0.847. (***c***) Processed by the proposed model with $\alpha = 1.8$, PSNR = 34.86, SNR = 31.97, MSE = 21.21, SSIM = 0.99. (***d***) By Chan and Shen model, PSNR = 33.41, SNR = 30.73, MSE = 24.58, SSIM = 0.97. (***e***) Processed by CDD model model, PSNR = 33.53, SNR = 30.82, MSE = 22.72, SSIM = 0.972. (***f***) Processed by Zhang model, PSNR = 33.72, SNR = 30.9, MSE = 22.89, SSIM = 0.98.

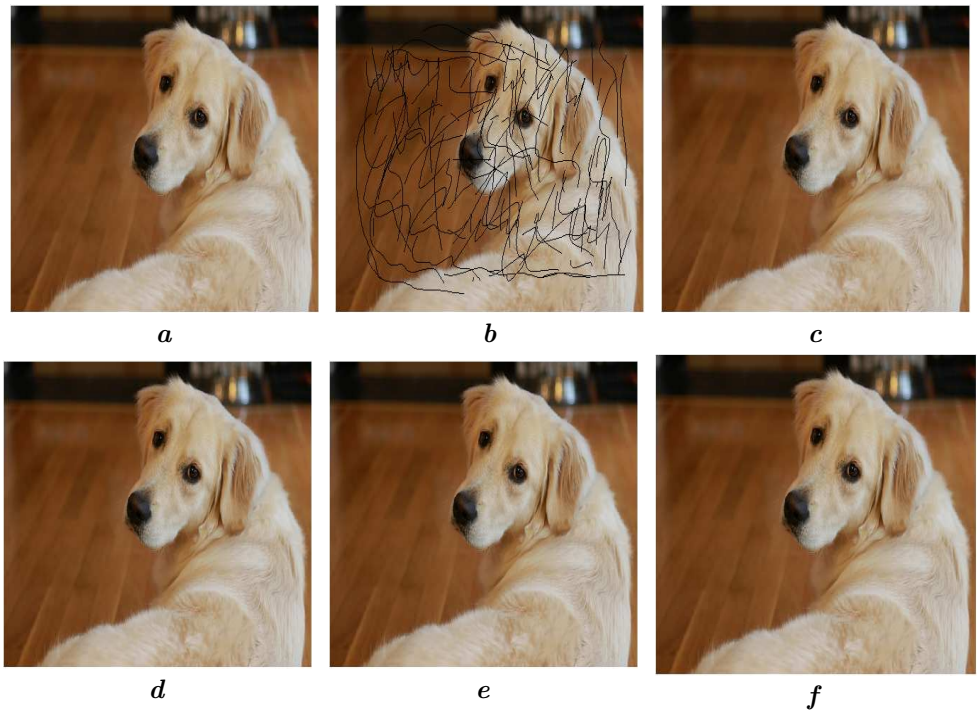

Fig. 4. Repairing results corresponding to the Sherlock image. (***a***) Clean image (560×420). (***b***) Image with hand text, PSNR = 19.96, SNR = 17.07, MSE = 656.19, SSIM = 0.867. (***c***) Processed by the proposed model with $\alpha = 1.8$, PSNR = 47.56, SNR = 44.67, MSE = 5.13, SSIM = 0.9984. (***d***) By Chan and Shen model, PSNR = 44.37, SNR = 41.92, MSE = 10.47, SSIM = 0.989. (***e***) Processed by CDD model model, PSNR = 45.53, SNR = 42.68, MSE = 9.48, SSIM = 0.992. (***f***) Processed by Zhang model, PSNR = 46.02, SNR = 43.16, MSE = 7.09, SSIM = 0.997.

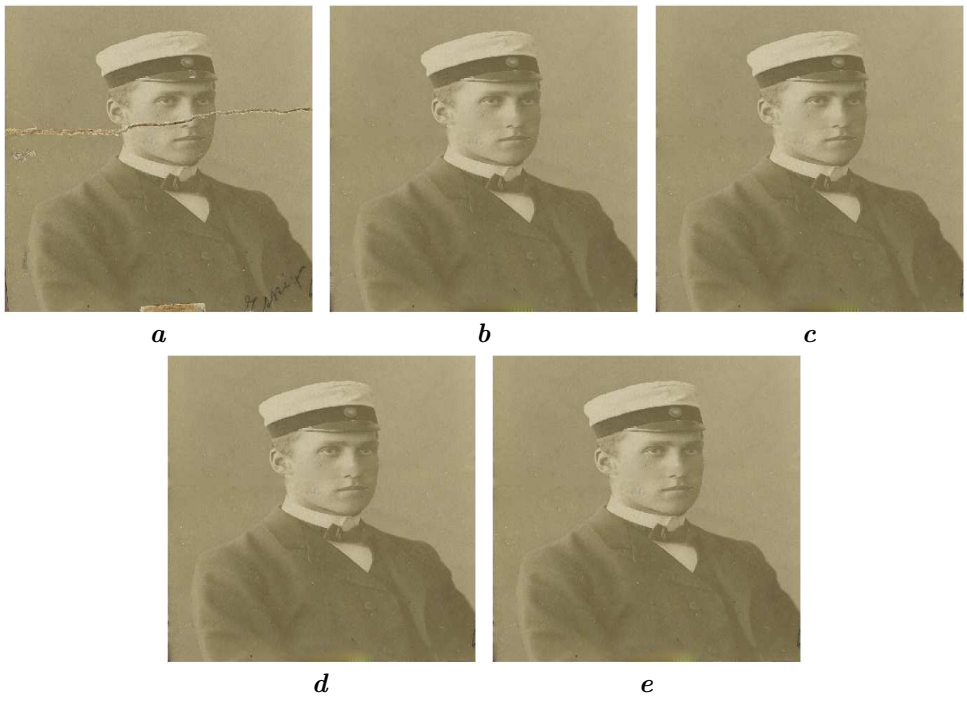

Fig. 5. Restoration of an old image. (***a***) Old image (691×928). (***b***) Processed by the proposed model with $\alpha = 1.8$. (***c***) Processed by Chan and Shen model. (***d***) Processed by CDD model model. (***e***) Processed by Zhang model.

The Proposed technique is used to remove text and handwritten from an image as shown in Figures 3 and 4. It provides an effective restoration of the degraded image, completing successfully the missing zones. It also preserves the image details. The fractional order α is considered for optimal reconstruction result $\alpha = 1.8$. Our model has a good performance in visual quality, a lower mean square error (MSE), a higher value of peak signal-to-noise (PSNR) as well as for the rate of the signal-to-noise (SNR), and a better measure of the structural similarity index measure (SSIM) compared to those obtained by the previously mentioned baseline methods.

Figure 5 shows an old image and its reconstruction. In this case the qualitative assessment of the completed images is easily measured by the human perception because we do not have the reference image with which we can quantitatively compare the inpainted results.

Finally, Figure 6 shows an object (microphone) disappearance from a scene, also in this case we do not have the reference image. For this reason, comparative results are placed side-by-side for our readers' judgement.

Fig. 6. Object disappearance from a scene. (***a***) Original image (218×290). (***b***) Processed by the proposed model with $\alpha = 1.8$. (***c***) Processed by Chan and Shen model. (***d***) Processed by CDD model model. (***e***) Processed by Zhang model.

5. Conclusion

In this work, we propose a fractional order nonlinear model for image inpainting in which the integer order spatial derivative in the classical Perona–Malik model is replaced by a fractional order spatial derivative. This new version has shown an improvement in the inpainting capacity compared to the inpainting models already existing in the literature. Indeed, the interpretation of the numerical simulations allows us to note that our model presents a good performance in visual quality, a lower mean square error (MSE), higher values of peak signal-to-noise (PSNR) and of signal-to-noise ratio (SNR), as well as a better measure of the structural similarity index measure (SSIM) compared to results obtained by other models considered.

[1] Bertalmio M., Sapiro G., Caselles V., Ballester C. Image Inpainting. Proceedings of the 27th annual conference on Computer graphics and interactive techniques – SIGGRAPH '00. 417–424 (2000).

[2] Boujena S., El Guarmah E., Gouasnouane O., Pousin J. An Improved Nonlinear Model for Image Restoration. Pure and Applied Functional. **2** (4), 599–623 (2017).

[3] Boujena S., El Guarmah E., Gouasnouane O., Pousin J. On a derived non linear model in image restoration. Proceedings of 2013 International Conference on Industrial Engineering and Systems Management (IESM). 1–3 (2013).

[4] Boujena S., Bellaj K., Gouasnouane O., El Guarmah E. An improved nonlinear model for image inpainting. Applied Mathematical Sciences. **9** (124), 6189–6205 (2015).

[5] Boujena S., Bellaj K., Gouasnouane O., El Guarmah E. One approach for image denoising based on finite element method and domain decomposition technique. International Journal of Applied Physics and Mathematics. **7** (2), 141–147 (2017).

[6] Chan T. F., Shen J. Mathematical models for local nontexture inpaintings. SIAM Journal on Applied Mathematics. **62** (3), 1019–1043 (2002).

[7] Chan T. F., Shen J. Nontexture inpainting by curvature driven diffusions (CDD). Journal of Visual Communication and Image Representation. **12** (4), 436–449 (2001).

[8] Gouasnouane O., Moussaid N., Boujena S. A Nonlinear Fractional Partial Differential Equation for Image Denoising. Third International Conference on Transportation and Smart Technologies (TST). 59–64 (2021).

[9] Sayah A., Moussaid N., Gouasnouane O. Finite difference method for Perona–Malik model with fractional derivative and its application in image processing. 2021 Third International Conference on Transportation and Smart Technologies (TST). 101–106 (2021).

[10] Li H., Yu Z., Zhou J. Fractional differential and variational method for image fusion and super-resolution. Neurocomputing. **171**, 138–148 (2016).

[11] Chen D., Chen Y., Xue D. 1-D and 2-D digital fractional-order Savitzky–Golay differentiator. Signal, Image and Video Processing. **6** (3), 503–511 (2012).

[12] Cuesta E., Codes J. Image processing by means of a linear integro-differential equation. 3rd IASTED Int. Conf. Visualization, Imaging and Image Processing. **1**, 438–442 (2003).

[13] Didasa S., Burgeth B., Imiya A., Weickert J. Regularity and scalespace properties of fractional high order linear filtering. Scale Spaces and PDE Methods in Computer Vision. Scale-Space 2005. Lecture Notes in Computer Science. **3459**, 13–25 (2005).

[14] Ben-Loghfyry A., Hakim A. Time-fractional diffusion equation for signal and image smoothing. Mathematical Modeling and Computing. **9** (2), 351–364 (2022).

[15] Zhang Y., Pu Y. F., Zhou J. Two new nonlinear PDE image inpainting models. Computer Science for Environmental Engineering and EcoInformatics. CSEEE 2011. Communications in Computer and Information Science. **158**, 341–347 (2011).

[16] Zhang Y., Pu Y. F., Hu J., Zhou J. L. A class of fractional-order variational image inpainting models. Applied Mathematics & Information Sciences. **6** (2), 299–306 (2012).

[17] Bosch J., Stoll M. A fractional inpainting model based on the vector-valued cahn-hilliard equation. SIAM Journal on Imaging Sciences. **8** (4), 2352–2382 (2015).

[18] Perona P., Malik J. Scale-espace and edge detection using anisotropic diffusion. IEEE Transaction on Pattern Analysis and Machine Intelligence. **12** (7), 429–439 (1990).

[19] Changpin L., Fanhai Z. Numerical Methods for Fractional Calculus. CRC Press (2015).

[20] Meerschaert M., Tadjeran C. Finite difference approximations for fractional advection-dispersion flow equations. Journal of Computational and Applied Mathematics. **172** (1), 65–77 (2004).

[21] Oldham K., Spanier J. The Fractional Calculus. Academic Press, New York (1974).

Нелінійне рівняння в частинних похідних для зафарбовування зображень

Гуаснуан О.[1,2], Муссаід Н.[1], Бужена С.[2], Каблі К.[2]

[1] *Університет Хасана II Касабланки, факультет наук і технологій Мохаммедіа, Лабораторія математики, інформатики та додатків (LMCSA), Мохаммедіа, Марокко*
[2] *Університет Хасана II Касабланки, факультет наук Айн-Чок, Лабораторія моделювання, аналізу, контролю та статистики (MACS), Касабланка, Марокко*

Зафарбовування зображень є важливим напрямком досліджень в обробці зображень. Його основна мета — доповнити відсутні або пошкоджені області зображень, використовуючи інформацію з навколишніх областей. Цей крок може бути виконаний, використовуючи нелінійні дифузійні фільтри, які вимагають розв'язування диференціальних еволюційних рівнянь у частинних похідних. У цій роботі пропонується фільтр, який визначається нелінійним еволюційним рівнянням у частинних похідних із дробовими просторовими похідними. Завдяки цьому вдалося покращити продуктивність відомих моделей зафарбовування, які базуються на диференціальних рівняннях у частинних похідних, і розширити деякі існуючі результати в обробці зображень. Дискретизація дробового диференціального рівняння у частинних похідних пропонованої моделі здійснюється за допомогою зміщеної формули Грюнвальда–Летнікова, що дозволяє будувати стійкі чисельні схеми. Порівняльний аналіз показує, що запропонована модель забезпечує покращену якість зображення, якіснішу або співмірну з якістю, отриманою на основі інших ефективних моделей, відомих в літературі.

Ключові слова: *обробка зображень, зафарбовування зображень, дробове числення, диференціальне рівняння в частинних похідних дробового порядку, нелінійна дифузія, дробова похідна.*