MATHEMATICAL MODELING AND COMPUTING, Vol. 9, No. 2, pp. 351–364 (2022)

Time-fractional diffusion equation for signal and image smoothing

Ben-Loghfyry A., Hakim A.

LAMAI, University of Cadi Ayyad, Marrakesh, Morocco

(Received 6 October 2020; Revised 6 May 2022; Accepted 7 May 2022)

In this paper, we utilize a time-fractional diffusion equation for image denoising and signal smoothing. A discretization of our model is provided. Numerical results show some remarkable results with a great performance, visually and quantitatively, compared to some well known competitive models.

Keywords: *image denoising, fractional derivative, time-fractional order derivative, tensor diffusion.*

2010 MSC: 35K55, 35K57, 94A08 **DOI:** 10.23939/mmc2022.02.351

1. Introduction

Image processing is a mathematical process for improving the image quality. This field has many classes like image denoising [1–4], image super-resolution [5,6], image recognition [7,8], image registration [9, 10], image restoration and enhancement [11,12], etc. In literature, there exist many techniques that can be applied in image denoising, but in our paper, we will concentrate on using partial differential equations (PDEs) [13,14].

The PDEs are power tools when it comes to denoising, it has been proven that they are handy in deleting noises and also in preserving important detail of the observed image. They were first utilize with the so-called heat equation [15], by using the Laplacian as a diffusion operator, but the result image is useless because of the appearance of blurriness caused by the isotropic diffusion (Laplacian operator). To remedy this problem, Perona and Malik proposed a anisotropic diffusion equation [16], by the name Perona–Malik equation. They used the gradient of the image to formulate the diffusion process and to control it. Let's consider u an observed image, t the time, and $c(\cdot)$ the diffusion function, the Perona–Malik diffusion can be presented as follows

$$\frac{\partial u}{\partial t} = \nabla \cdot (c(|\nabla u|)\nabla u). \tag{1}$$

The function c is chosen to enhance the homogeneous region in a fast way while keeping the details of the original image and preserving features as much as possible. Perona–Malik equation has shown its capability against noise, and its preservation of features. Despite its effectiveness, Perona–Malik diffusion has weaknesses regarding the appearance of blocky effect, because the diffusion function cannot detect the corners and edges with an adequate precision, so that the diffusion process can be over smoothly. In [17], the author managed to propose an improvement of the Perona–Malik model, by changing the argument of the diffusion function c with $|\nabla(G_\sigma * u)|$, where G_σ is a Gaussian convolution kernel, it has the following form $G_\sigma(x) = \frac{1}{2\pi\sigma^2}\exp\big(-\frac{|x|^2}{2\sigma^2}\big)$. This choice has led them to explore a new reliable edge detector.

Another successful attempt was introduced by Weickert [18]. His idea is to generalize the modified Perona–Malik equation, instead of using directions with the function $c(\cdot)$, he replaces it with orientations by applying a diffusion matrix D. His model has the following form:

$$\frac{\partial u}{\partial t} = \nabla \cdot (D\,(J_\rho(\nabla(G_\sigma * u)))\,\nabla u) \tag{2}$$

with proper conditions. More details about the terms D and J_ρ will be mentioned next. Moreover, this approach shows some magnificent results compared to the other well-known models. But unfortunately, it destroys details near edges and strong corners.

After defining the general frame, we can talk about the tool used in this paper, which is the fractional derivative [19, 20]. This later is a powerful tool modeling general nonlinear phenomenons, and now it is considered as a needful gear in image processing [21]. The time-fractional derivative is a new and a powerful tool, which can be found and applied in many domains (see [22–24] for more info).

In our paper, we are going to apply the time-fractional order derivative in the sense of Caputo [25,26] on the problem (2) instead of the first derivative. A good reason why we made this change is that a fractional-order derivative over time of an image u can be looked as a weighted sum of the successive terms u_{t_0}, u_{t_1}, ..., u_{t_N}. In other words, it uses explicitly the old versions of the observed image until it converges to the desired image. The model based on the time-fractional derivative reduces the noise in an image because it improves the smoothing process. One may wonder, what diffusion process will we be using, that can give great results with time-fractional derivative? The answer of that question is by using a diffusion based on orientations instead of directions, like in the Perona–Malik [16] or the improved Perona–Malik [17] models.

The paper is organized as follows, in section (2), we will explicitly introduce our model. A discretization of our equation is provided using a finite difference method. Section (3) will show the results of our model, visually and quantitatively, compared to some other competitive models. Some discussions will be provided. Finally, we will sum up with a conclusion in section (4).

2. The proposed time-fractional diffusion filtering

In this section, we introduce our model. For $x \in \Omega \subset \mathbb{R}^2$, $t \in (0,T)$ and $0 < \alpha < 1$, the diffusion filtering can be described as follows

$$\begin{cases} \dfrac{\partial^\alpha u}{\partial t^\alpha}(t,x) = \nabla \cdot \left(D\left(J_\rho(\nabla u_\sigma)\right)\nabla u\right) + \lambda\big(u - u_0\big) & \text{in} \quad (0,T)\times\Omega, \\ u(0,x) = u_0(x) & \text{on} \quad \Omega, \\ \langle D(J_\rho(\nabla u_\sigma)\nabla u, n\rangle = 0 & \text{on} \quad (0,T)\times\partial\Omega. \end{cases} \tag{3}$$

Where u_0 is the initial (observed) noisy image, λ is positive constant.

The term $\frac{\partial^\alpha u}{\partial t^\alpha}(t,x)$ is the time-fractional derivative in the sense of Caputo [25], it has the following form:

$$\frac{\partial^\alpha u}{\partial t^\alpha}(t,x) = \frac{1}{\Gamma(1-\alpha)}\int_0^t \frac{\partial u}{\partial \tau}(\tau,x)\frac{1}{(t-\tau)^\alpha}d\tau, \quad 0<\alpha<1.$$

D is an anisotropic diffusion based tensor which depends on its structure J_ρ. This latter is defined by

$$J_\rho(\nabla u_\gamma) = G_\rho * (\nabla u_\gamma \otimes \nabla u_\gamma) = G_\rho * (\nabla(G_\gamma * u)\otimes \nabla(G_\gamma * u)^t),$$

G_ρ and G_σ are two Gaussian convolution kernels, they have the following form $G_\tau = \frac{1}{2\pi\tau^2}\exp\big(-\frac{|x|^2}{2\tau^2}\big)$.

The diffusion tensor D has been calculated using the eigenvalues and the eigenvectors of its structure tensor J_ρ. Its purpose is to preserve the initial image details, like corners and edge, while deleting the noise. The function D is defined as follow:

$$D := f_+(\lambda_+,\lambda_-)\theta_+\theta_+^t + f_-(\lambda_+,\lambda_-)\theta_-\theta_-^t, \tag{4}$$

where $\lambda_{+/-}$ (resp. $\theta_{+/-}$) are the the eigenvalues (resp. the eigenvectors) of the tensor structure J_ρ. The eigenvalues are calculated as:

$$\lambda_{+/-} = \frac{1}{2}\left(\operatorname{trace}(J_\rho) \pm \sqrt{\operatorname{trace}^2(J_\rho) - 4\det(J_\rho)}\right), \tag{5}$$

one could give a reasonable cause of this choice using the geometric characteristics of the restored image. In other words, if $\lambda_{+/-} \approx 0$, the smoothing in homogeneous regions is isotropic, but if $\lambda_+ \gg \lambda_- \approx 0$

(or $\lambda_- \gg \lambda_+ \approx 0$), the smoothing process in this case is anisotropic and directed along the edges. However, if $\lambda_+ \gg \lambda_- \gg 0$, the smoothing goes off into corners.

For the functions $f_+(\lambda_+, \lambda_-)$ and $f_-(\lambda_+, \lambda_-)$, they are chosen to fulfill the following constraints:

— isotropic: in the homogeneous regions, the diffusion has to be isotropic;
— anisotropic: beside sharp edges and corners, the diffusion in this case has to be anisotropic.

While this process in a generalization of all the diffusion equation applied on image processing, one must wonder, the functions $f_{+/-}$ associated to the Perona–Malik equation, which are defined as:

$$f_{+/-}(\lambda_+, \lambda_-) = \exp\left(-\frac{(\lambda_+ + \lambda_-)^2}{k}\right),$$

where k is a positive constant, chosen numerically. The functions $f_{+/-}$ associated with the Perona–Malik model [16] or the Weickert model [27] destroy the details of the restored image, especially corners and singularities. To remedy this, we propose a new choice of the functions $f_{+/-}$, so that the orientation of the diffusion works in a perfect way, while preserving and conserving features, corners and characteristics of the observed image. In other words, the eigenvalues λ_+ and λ_- are needed when the diffusion is near a corner or a contour, in this case λ_+ is high so the diffusion will be carried by the direction θ_-.

A considerable choice of those functions are the following:

$$\begin{cases} f_+(\lambda_+, \lambda_-) = \exp\left(-\frac{\lambda_+}{k_1}\right), \\ f_-(\lambda_+, \lambda_-) = \exp\left(-\frac{\lambda_-}{k_2}\right) + 0.001 \exp\left(-\frac{\lambda_+}{k_1}\right), \end{cases} \tag{6}$$

where k_1 and k_2 (respectively) are two thresholds that control the diffusion along the directions θ_+ and θ_- (respectively).

2.1. Discretization of the proposed model

For simplicity reasons, we consider a one-dimensional function u, and for $0 < \alpha < 1$, the Caputo's fractional derivative is approached by:

$$\frac{\partial^\alpha u_i^k}{\partial t^\alpha} \cong \sigma_{\alpha,\tau} \sum_{l=1}^{k} \omega_l^{(\alpha)} \left(u_i^{k-l+1} - u_i^{k-l}\right) = \sigma_{\alpha,\tau} \left[u_i^k - \sum_{l=1}^{k-1} \left(\omega_l^{(\alpha)} - \omega_{l+1}^{(\alpha)}\right) u_i^{k-l} - \omega_k^{(\alpha)} u_i^0\right],$$

where

$$\sigma_{\alpha,\tau} = \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}, \quad \omega_l^{(\alpha)} = l^{1-\alpha} - (l-1)^{1-\alpha}, \quad 1 = \omega_1^{(\alpha)} > \omega_2^{(\alpha)} > \cdots > \omega_k^{(\alpha)}.$$

In order to introduce a discretization of the equation 3, we define a spatial partition of the image domain Ω. Let u be an image with the size of $N \times M$ in Ω. We denote $u_{i,j}$ the value of u at the pixel (i, j), for all $i = 1, \ldots, N$ and $j = 1, \ldots, M$.

Now, we present a discretization of the equation. In order to do that, we need to control the diffusion. In other words, the term $\operatorname{div}(D\nabla u)$ must be discretized in a way that our discrete model can be stabilized. In our case, we consider a method presented by Weickert in [18] which use a convolution on the discrete image, called non-negative stencil, in order to stabilize its scheme.

We rewrite the fractional diffusion matrix as follows

$$D = \begin{pmatrix} a & b \\ b & c \end{pmatrix},$$

practically, a, b and c can be calculated

$$\begin{aligned} a &= \lambda_1 \cos^2\theta + \lambda_2 \sin^2\theta, \\ b &= (\lambda_1 - \lambda_2) \sin\theta \cos\theta, \\ c &= \lambda_1 \sin^2\theta + \lambda_2 \cos^2\theta. \end{aligned}$$

Where θ is

$$\tan(2\theta) = \frac{2J_{12}}{J_{11} - J_{22}},$$

where again, J_{11}, J_{12} and J_{22} are the components of the structure tensor J_ρ. It has the following form:

$$J_\rho = \begin{pmatrix} J_{11} & J_{12} \\ J_{12} & J_{22} \end{pmatrix}.$$

Now we can start calculating the term $\mathrm{div}(D\nabla u)$,

$$\begin{aligned} \mathrm{div}(D\nabla u) &= \mathrm{div}\left[\begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} \partial_x u \\ \partial_y u \end{pmatrix} \right] \\ &= \partial_x(a\partial_x u) + \partial_x(b\partial_y u) + \partial_y(b\partial_x u) + \partial_y(c\partial_y u) \\ &= \partial_x a\partial_x u + a\partial_{xx} u + \partial_x b\partial_y u + b\partial_{xy} u \\ &\quad + \partial_y b\partial_x u + b\partial_{yx} u + \partial_y c\partial_y u + c\partial_{yy} u. \end{aligned}$$

We denote $(\partial_x u)_{i,j}$ (resp. $(\partial_y u)_{i,j}$) the discretization of the operators $\partial_x u$ (resp. $\partial_x u$). They can be calculated as follows:

$$(\partial_x u)_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} & \text{if } \ i < N, \\ 0 & \text{if } \ i = N, \end{cases}$$

$$(\partial_y u)_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} & \text{if } \ j < M, \\ 0 & \text{if } \ j = M. \end{cases}$$

For the second derivatives:

$$(\partial_{xx} u)_{i,j} = \begin{cases} u_{i+1,j} - 2u_{i,j} + u_{i-1,j} & \text{if } \ 1 < i < N, \\ u_{i+1,j} - u_{i,j} & \text{if } \ i = 1, \\ u_{i-1,j} - u_{i,j} & \text{if } \ i = N, \end{cases}$$

$$(\partial_{yy} u)_{i,j} = \begin{cases} u_{i,j+1} - 2u_{i,j} + u_{i,j-1} & \text{if } \ 1 < j < M, \\ u_{i,j+1} - u_{i,j} & \text{if } \ j = 1, \\ u_{i,j-1} - u_{i,j} & \text{if } \ j = N, \end{cases}$$

$$(\partial_{xy} u)_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} + u_{i-1,j+1} + u_{i-1,j} & \text{if } \ 1 < i < N, \ 1 < j < M, \\ 0 & \text{if } \ i = 1, \\ 0 & \text{if } \ i = N, \end{cases}$$

$$(\partial_{xy} u)_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} + u_{i+1,j-1} + u_{i,j-1} & \text{if } \ 1 < i < N, \ 1 < j < M, \\ 0 & \text{if } \ j = 1, \\ 0 & \text{if } \ j = M. \end{cases}$$

Using those derivatives, we rewrite the term $\mathrm{div}(D\nabla u)$ at the inner point (i, j) by:

$$\begin{aligned} [\mathrm{div}(D\nabla u)]_{i,j} &= (\partial_x a)_{i,j}(\partial_x u)_{i,j} + a_{i,j}(\partial_{xx} u)_{i,j} + (\partial_x b)_{i,j}(\partial_y u)_{i,j} + b_{i,j}(\partial_{xy} u)_{i,j} \\ &\quad + (\partial_y b)_{i,j}(\partial_x u)_{i,j} + b_{i,j}(\partial_{yx} u)_{i,j} + (\partial_y c)_{i,j}(\partial_y u)_{i,j} + c_{i,j}(\partial_{yy} u)_{i,j}. \quad (7) \end{aligned}$$

The discretization of (7) can be seen as a convolution product on u at a point (i, j) using an array of size 3×3 called the Stencil. However, one of the conditions imposed by Weickert on the matrix associated with the diffusion term is the positivity of its components. It is required for the stability of the scheme. Moreover, (7) does not guarantee this condition. To deal with this problem, Weickert proposed in [18] a new non-negative discretization summarized in Table 1.

Table 1. Non-negative discretization stencil.

$\frac{\lvert b_{i-1,j+1}\rvert - b_{i-1,j+1}}{4}$ $+\frac{\lvert b_{i,j}\rvert - b_{i,j}}{4}$	$\frac{c_{i,j+1}+c_{i,j}}{2}$ $-\frac{\lvert b_{i,j+1}\rvert + \lvert b_{i,j}\rvert}{2}$	$\frac{\lvert b_{i+1,j+1}\rvert + b_{i+1,j+1}}{4}$ $+\frac{\lvert b_{i,j}\rvert + b_{i,j}}{4}$
$\frac{a_{i-1,j}+a_{i,j}}{2}$ $-\frac{\lvert b_{i-1,j}\rvert + \lvert b_{i,j}\rvert}{2}$	$-\frac{a_{i-1,j}+2a_{i,j}+a_{i+1,j}}{2}$ $-\frac{\lvert b_{i-1,j+1}\rvert - b_{i-1,j+1}+\lvert b_{i+1,j+1}\rvert + b_{i+1,j+1}}{4}$ $-\frac{\lvert b_{i-1,j-1}\rvert + b_{i-1,j-1}+\lvert b_{i+1,j-1}\rvert - b_{i+1,j-1}}{4}$ $-\frac{\lvert b_{i-1,j}\rvert + \lvert b_{i+1,j}\rvert + \lvert b_{i,j-1}\rvert + \lvert b_{i,j+1}\rvert + 2\lvert b_{i,j}\rvert}{2}$ $-\frac{c_{i,j-1}+2c_{i,j}+c_{i,j+1}}{2}$	$\frac{a_{i+1,j}+a_{i,j}}{2}$ $-\frac{\lvert b_{i+1,j}\rvert + \lvert b_{i,j}\rvert}{2}$
$\frac{\lvert b_{i-1,j-1}\rvert + b_{i-1,j-1}}{4}$ $+\frac{\lvert b_{i,j}\rvert + b_{i,j}}{4}$	$\frac{c_{i,j-1}+c_{i,j}}{2}$ $-\frac{\lvert b_{i,j-1}\rvert + \lvert c_{i,j}\rvert}{2}$	$\frac{\lvert b_{i+1,j-1}\rvert - b_{i+1,j-1}}{4}$ $+\frac{\lvert b_{i,j}\rvert - b_{i,j}}{4}$

Let's consider $A(u)$ the matrix after discretization, that contains the components of the operator $\operatorname{div}(D\nabla u)$. We also consider the matrix S of size 3×3 containing the components of Table (1), we note:

$$S = \begin{pmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{pmatrix}.$$

For a fixed point (i, j), the term $[A(u)]_{i,j}$ can be calculated as follows:

$$\begin{aligned}[A(u)]_{i,j} = {} & u_{i-1,j-1}d_{11} + u_{i-1,j}d_{12} + u_{i-1,j+1}d_{13} \\ & + u_{i,j-1}d_{21} + u_{i,j}d_{22} + u_{i,j+1}d_{23} \\ & + u_{i+1,j-1}d_{31} + u_{i+1,j}d_{32} + u_{i+1,j+1}d_{33}.\end{aligned}$$

We completed the discretization of (7). The final following discretization of our PDE is:

$$\sigma_{\alpha,\tau}\left[u_i^k - \sum_{l=1}^{k-1}\left(\omega_l^{(\alpha)} - \omega_{l+1}^{(\alpha)}\right)u_{i,j}^{k-l} - \omega_k^{(\alpha)}u_{i,j}^0\right] = \left[A(u^{k-1})\right]_{i,j} + \lambda\left(u_{i,j}^{k-1} - (u_0)_{i,j}\right)$$

for $k \geqslant 1$, $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant M$.

This last implies that:

$$u_{i,j}^k = \sum_{l=1}^{k-1}\left(\omega_l^{(\alpha)} - \omega_{l+1}^{(\alpha)}\right)u_{i,j}^{k-l} + \omega_k^{(\alpha)}u_{i,j}^0 + \frac{1}{\sigma_{\alpha,\tau}}\left[\left[A(u^{k-1})\right]_{i,j} + \lambda\left(u_{i,j}^{k-1} - (u_0)_{i,j}\right)\right] \tag{8}$$

for $k \geqslant 2$, $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant M$, where

$$u_{i,j}^0 = (u_0)_{i,j}, \quad u_{i,j}^1 = \omega_1^{(\alpha)}u_{i,j}^0 + \frac{1}{\sigma_{\alpha,\tau}}\left(\left[A(u^0)\right]_{i,j} + \lambda\left(u_{i,j}^0 - (u_0)_{i,j}\right)\right).$$

3. Results and discussion

This section will provide some important results, to prove the efficiency of our time-fractional diffusion model, compared to some competitive well-known denoising methods.

In our case, we will compare the obtained results from our model, with the improved Perona–Malik [17], the Weickert filter [18] and the fractional-order Total Variation [28] i.e. TV^β for $1 < \beta < 2$. Which is a better performed version of the well-known Total Variation model [29]. For the sake of comparison. In this work, all the simulations were done by Matlab 2016, on a computer with a processor at 3 GHz and 8 Gb of RAM. To test the performance of our approach compared to the others, we used the peak signal-to-noise ratio (PSNR) and also the structural similarity index (SSIM) in order to measure the quality of the result images. All the parameters in all the models were set manually in terms of PSNR and SSIM values.

3.1. Enhancement in case of noise

In order to study the effectiveness of the proposed method, we will perform some experiments. First experiment will be 3 images corrupted by a Gaussian noise with zero mean and standard deviation $\sigma = 30$. Figure 1 shows denoising results for the 3 images. We can see clearly the difference between the restored images as the noise is reduced. We take $\lambda = \frac{1}{\sigma}$; Maxiter = 1000 and $\alpha = 0.6$. For the choices of the diffusivity function $f_{+/-}$ In Table 2, we present a quantitative comparison between the obtained images with different values of the noise level σ. The PSNR and SSIM values are set and show a big improvement compared to the other models.

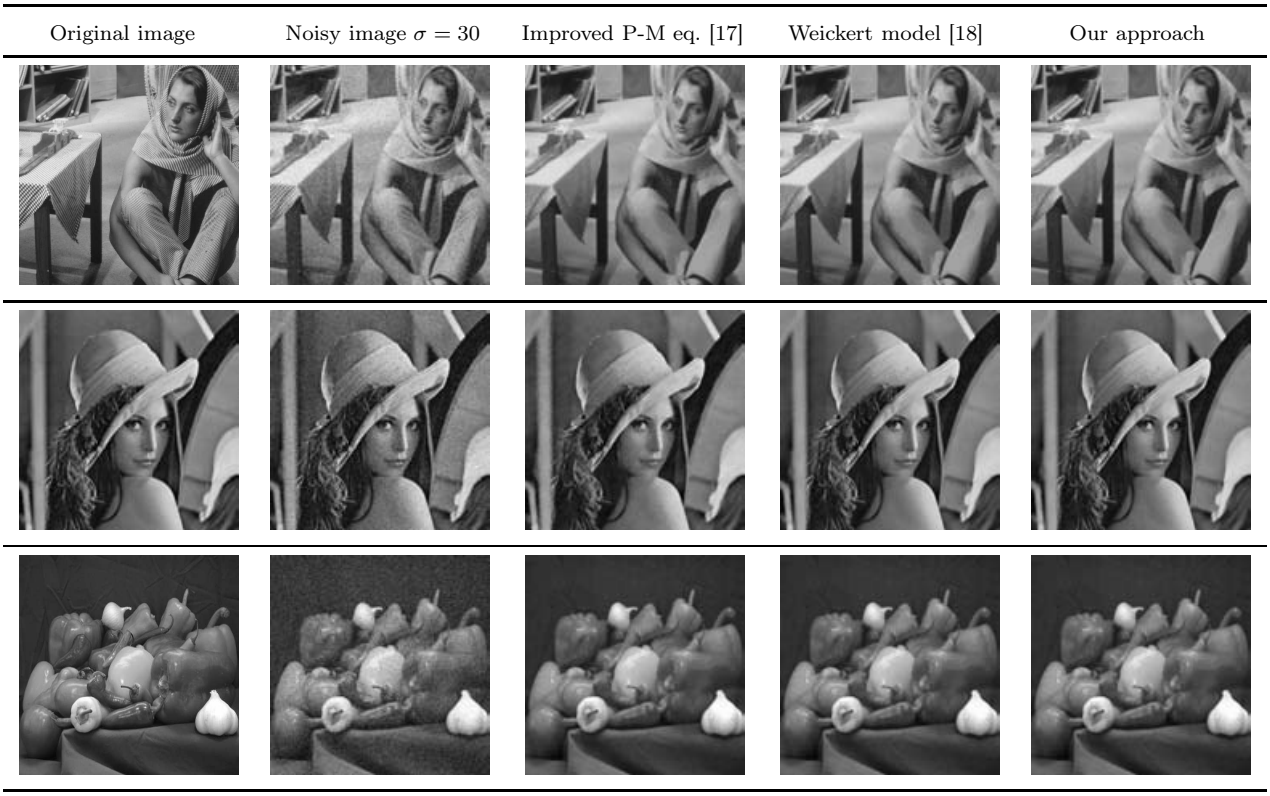
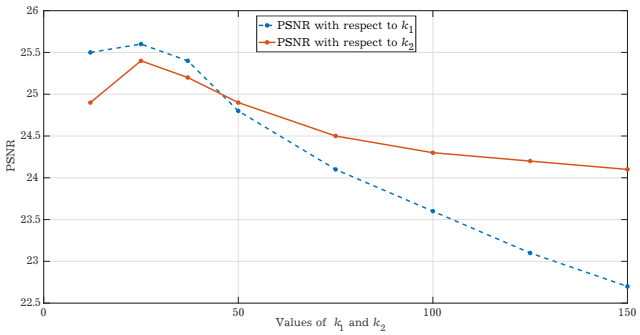


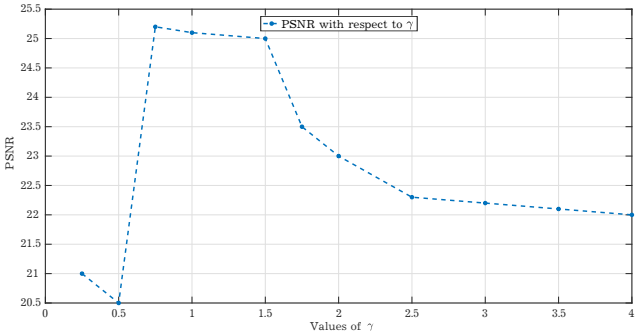
Fig. 1. Denoising results applied on three noisy test images, with a noise level $\sigma = 30$.

Table 2. Comparison of different models with 3 different images using the PSNR and SSIM values.

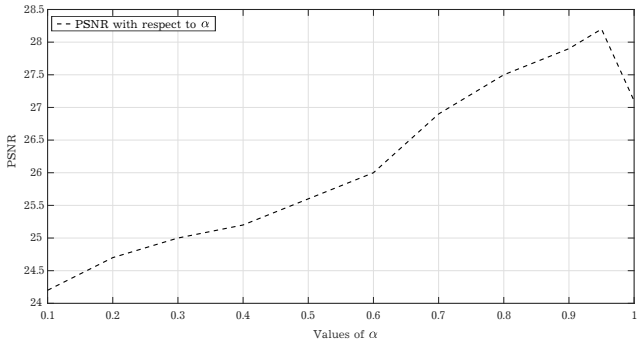
Image		Our approach	TV^{β} [28]	Improved P-M [17]	Weikert filter [18]
Barbara	$\sigma = 10$	PSNR=**31.7012** SSIM=**0.898**	PSNR=31.2850 SSIM= 0.888	PSNR= 31.2427 SSIM= 0.890	PSNR= 31.3077 SSIM=0.896
	$\sigma = 20$	PSNR=**28.2458** SSIM=**0.839**	PSNR= 27.7976 SSIM=0.820	PSNR= 26.6231 SSIM=0.819	PSNR= 27.5048 SSIM=0.831
	$\sigma = 30$	PSNR=**25.0945** SSIM=**0.786**	PSNR= 24.9479 SSIM= 0.778	PSNR= 24.4496 SSIM=0.773	PSNR=24.7579 SSIM = 0.782
Lena	$\sigma = 10$	PSNR=**34.5859** SSIM=**0.925**	PSNR= 34.4747 SSIM= 0.902	PSNR=33.6393 SSIM=0.917	PSNR=34.0937 SSIM=0.851
	$\sigma = 20$	PSNR= **31.4599** SSIM=**0.855**	PSNR=31.2153 SSIM=0.828	PSNR=29.7665 SSIM=0.831	PSNR= 30.2254 SSIM=0.844
	$\sigma = 30$	PSNR=**29.9105** SSIM=**0.807**	PSNR=29.3855 SSIM= 0.766	PSNR= 27.4437 SSIM=0.725	PSNR=28.7537 SSIM=0.786
Peppers	$\sigma = 10$	PSNR=**34.1985** SSIM=**0.982**	PSNR= 34.0619 SSIM=0.989	PSNR=33.6967 SSIM= 0.966	PSNR= 33.9521 SSIM= 0.978
	$\sigma = 20$	PSNR=**31.3677** SSIM=**0.941**	PSNR=31.0285 SSIM= 0.938	PSNR=30.0275 SSIM= 0.924	PSNR= 31.0017 SSIM= 0.926
	$\sigma = 30$	PSNR= **30.0596** SSIM=0.884	PSNR=29.0046 SSIM=**0.888**	PSNR=27.4982 SSIM=0.865	PSNR= 28.7789 SSIM=0.880



a PSNR (k_1) and PSNR (k_2).



b PSNR (γ).



c PSNR (α) and SSIM (α).

Fig. 2. Sensitivity test of the four parameters for the *Man* test with $\sigma = 35$.

3.2. Sensitivity test of the parameter efficacy

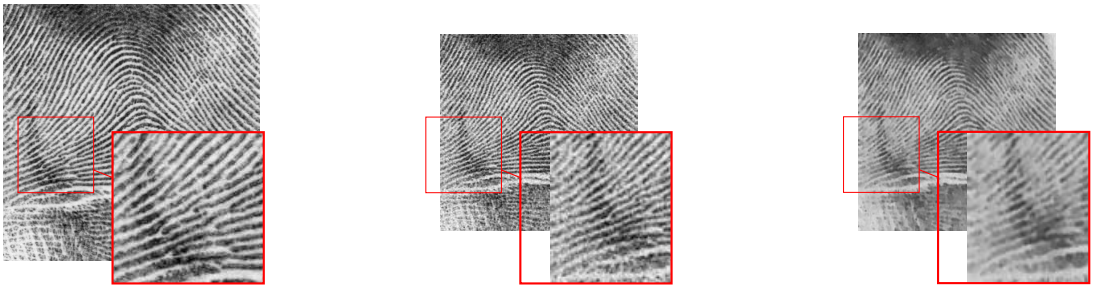
In general, the choice of parameters depends on the image, its nature, the noise level and discontinuity of lines.

We start with the parameter selection in (6), the two threshold k_1 and k_2 are chosen in order to show the best values of PSNR as shown in Fig. 2*a*, for an optimized value of $k_{1/2}$, we fix $k_{1/2} = 25$. We use the same technique to pick the parameter γ (see Fig. 2*b*). For ρ, Weickert has demonstrated in [27] that in general $\rho = 3\gamma$.

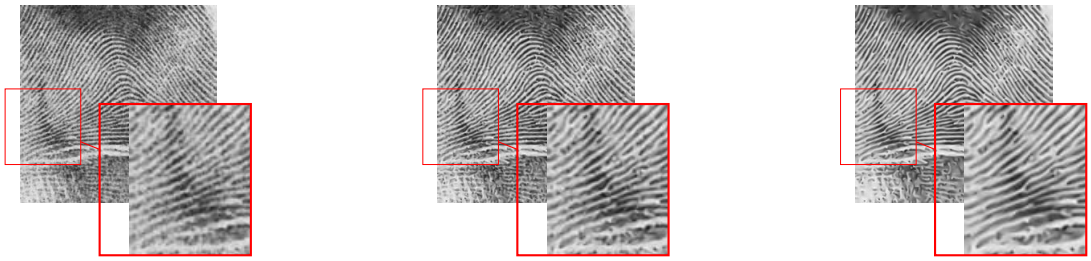
The last important parameter α (the time-fractional order) mostly depends on the nature of the image. Moreover, as is shown in Fig. 2*c*, the values of α are obtained by the best value of the PNSR.

3.3. Robustness in the case of noisy images

Almost every model in the literature can reduce small noises, but a finite of those being robust can face big noises and preserve as much as possible at the same time. To test the robustness of our proposed approach against high level of noise, we take more test images using different Gaussian additive noise variances ($\sigma = 40$ and $\sigma = 50$). Figures 3 to 8 show the denoised images of different models. We see clearly that for the proposed method, the noise is reduced and important details are conserved. These results demonstrate that our model is more robust against noise compared to the others. Visually, some differences are visible in image locations with smoothly varying intensities, such as Fingerprint's lines and in Barbara's face. While, even if the competitive model are feature preserving, especially the TV^{α} regularization (that proves to be more useful than the Total Variation, which is known by it success in avoiding staircasing effect), the noise is not completely removed and the texture is damaged. However, the proposed time-fractional diffusion model does not suffer from the presence of the staircasing effect. The results show that the proposed model outperforms the others visually and quantitatively in terms of both PSNR and SSIM values.

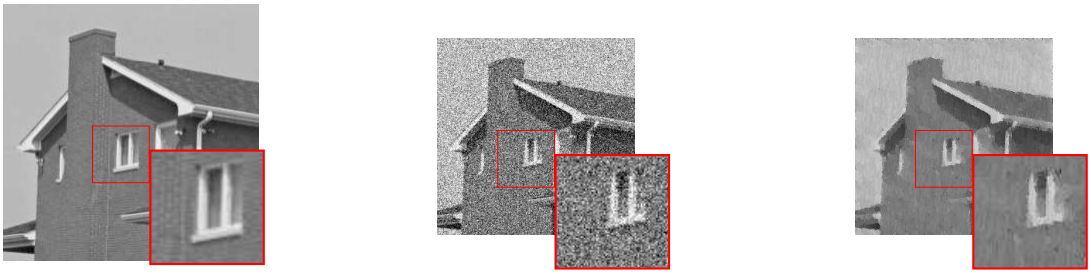


a Original image. ***b*** Noisy image $\sigma = 50$. ***c*** TV^{β} model [28].

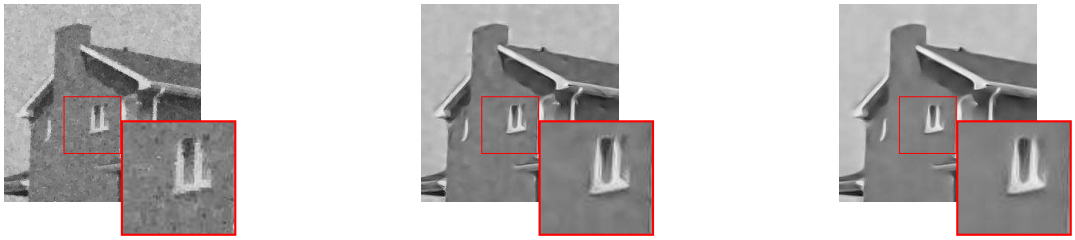


d Improved PM [17]. ***e*** Weickert model [18]. ***f*** Our approach.

Fig. 3. Comparison of our approach with other models using *Fingerprint* image where the noise variance is $\sigma = 50$.

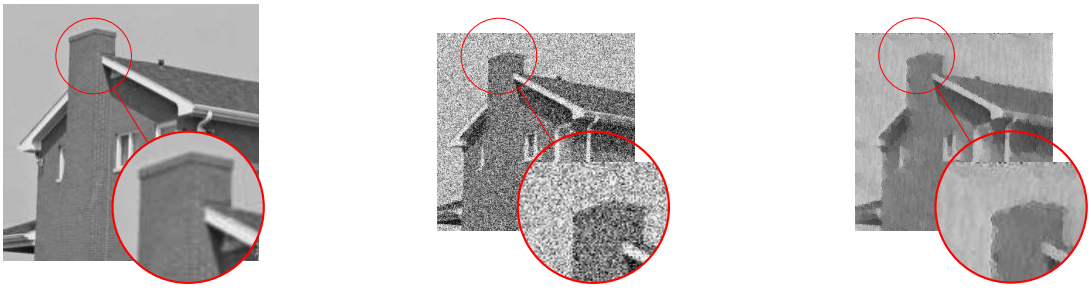


a Original image. ***b*** Noisy image σ=40. ***c*** TV^{β} model [28].

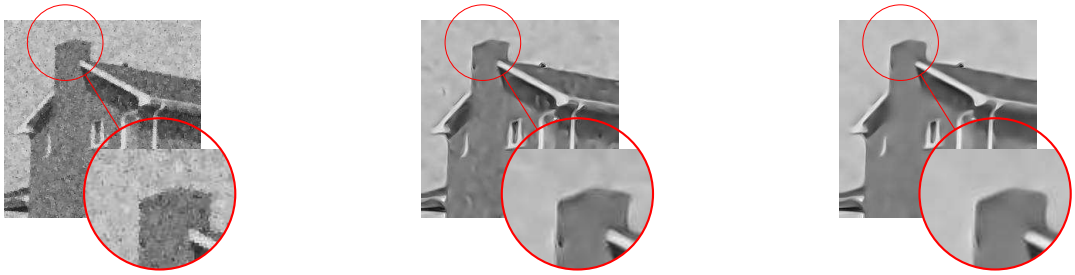


d Improved PM [17]. ***e*** Weickert model [18]. ***f*** Our approach.

Fig. 4. Comparison of our approach with other models using *House* image where the noise variance is $\sigma = 40$.



a Original image. ***b*** Noisy image $\sigma = 50$. ***c*** TV^{β} model [28].



d Improved PM [17]. ***e*** Weickert model [18]. ***f*** Our approach.

Fig. 5. Comparison of our approach with other models using *House* image where the noise variance is $\sigma = 50$.

a Original image. ***b*** Noisy image $\sigma = 40$. ***c*** TV^{β} model [28].

d Improved PM [17]. ***e*** Weickert model [18]. ***f*** Our approach.

Fig. 6. Comparison of our approach with other models using *Man* image where the noise variance is $\sigma = 40$.

a Original image.

b Noisy image $\sigma = 50$.

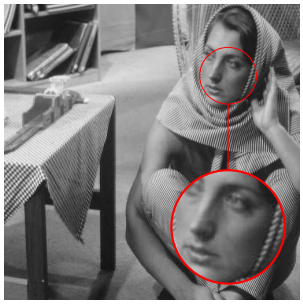
c TV^{β} model [28].

d Improved PM [17].

e Weickert model [18].

f Our approach

Fig. 7. Comparison of our approach with other models using *Man* image where the noise variance is $\sigma = 50$.



a Original image.

b Noisy image $\sigma = 40$.

c TV^{β} model [28].

d Improved PM [17].

e Weickert model [18].

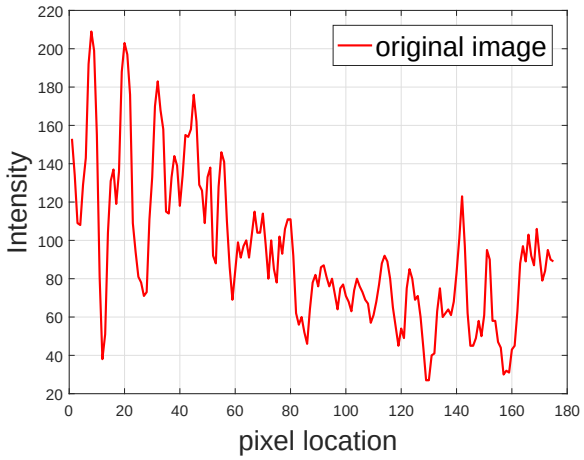
f Our approach.

Fig. 8. Comparison of our approach with other models using *Barbara* image where the noise variance is $\sigma = 40$.

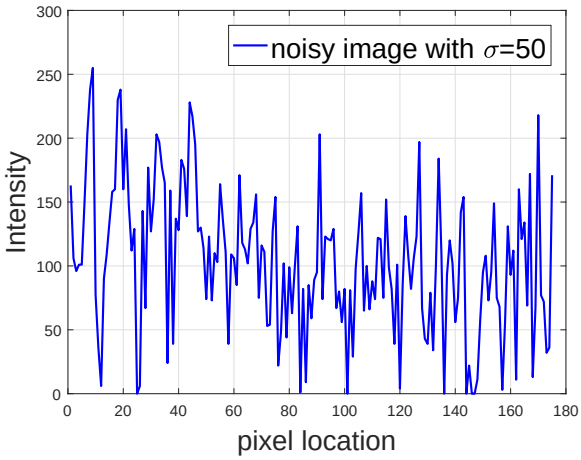
3.4. Smoothing signals

In this subsection, we will present some results in the smoothing process of signals to extract useful data from them. Figires 9 and 10 show the recovery of data from destroyed signals shaped in smooth data. From these figures, it is clear that the recovery of peaks and details from noisy signals is successfully guaranteed using our approach, contrarily to the competitive models. Hight frequencies are preserved thanks to the time-fractional derivative and its memory effect potential.

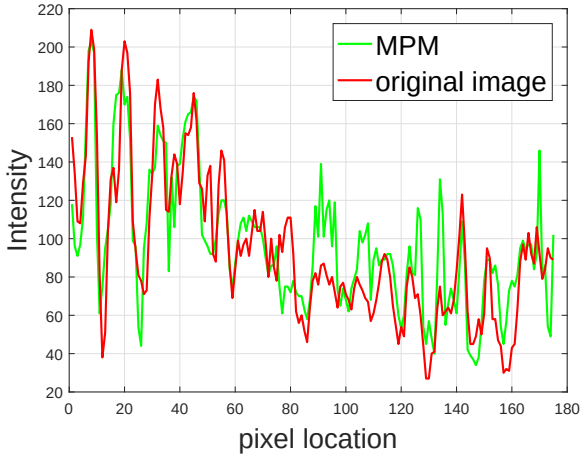
Additionally, our approach fixes the drawbacks of the competitive models, such as Staircasing effect. It can be seen in some hight peaks that were not treated by either of the competitive models. However, this drawback is reduced in our case with the help of the diffusion process and the fractional derivative.



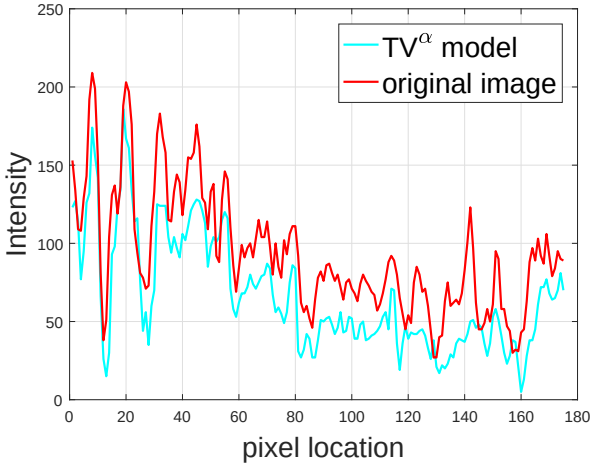
a The original signal.



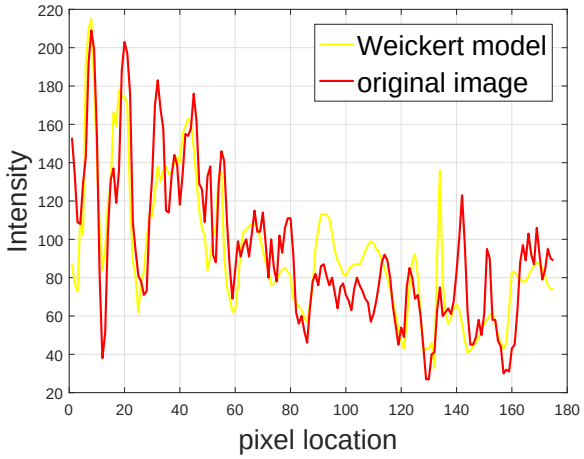
b Noisy signal from an image with $\sigma = 50$.



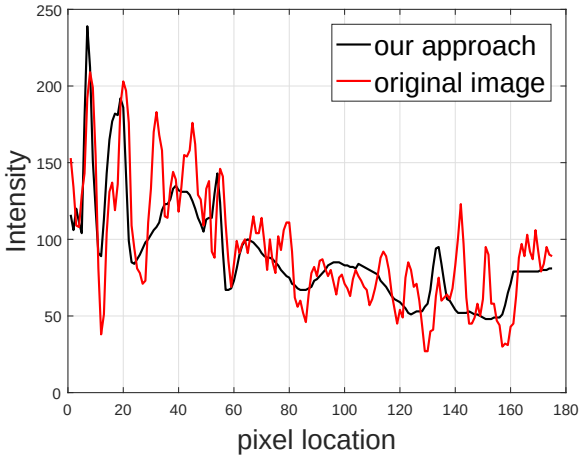
c Original and recovered image (signal) by MPM.



d Original and recovered image (signal) by TV^{α}.

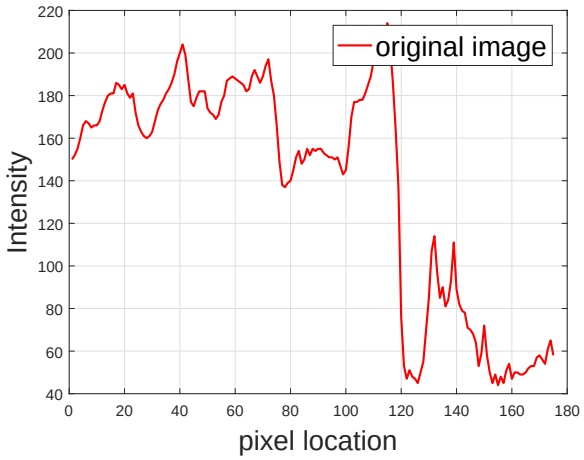


e Original and recovered image (signal) by Weickert model.

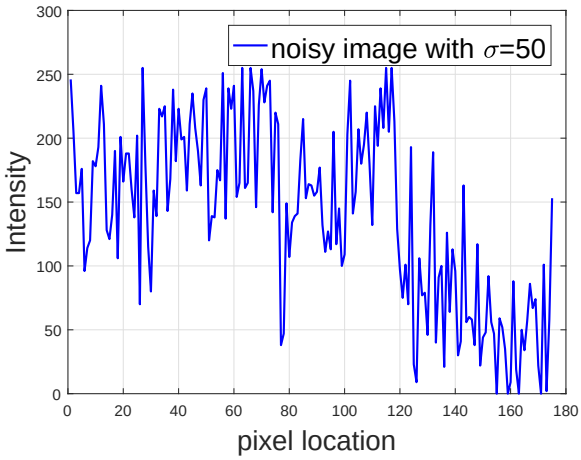


f Original and recovered image (signal) by our approach.

Fig. 9. Signal of cropped part of *Fingerprint* image with different smoothing methods.



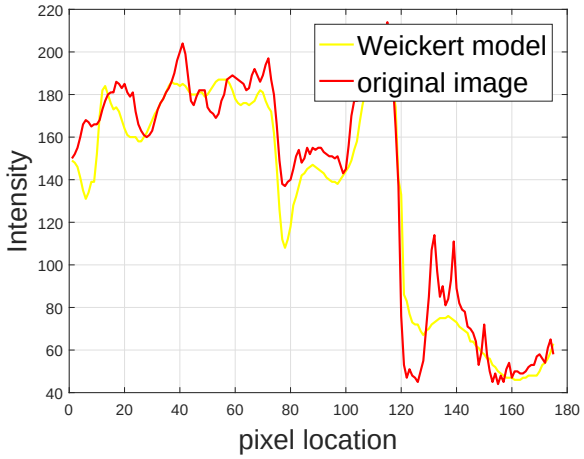
a The original signal.



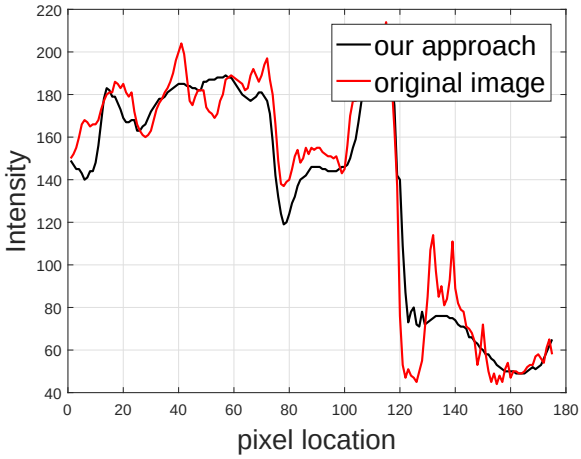
b Noisy signal from an image with $\sigma = 50$.

c Original and recovered image (signal) by MPM.

d Original and recovered image (signal) by TV^{α}.



e Original and recovered image (signal) by Weickert model.



f Original and recovered image (signal) by our approach.

Fig. 10. Signal of cropped part of *Man* image with different smoothing methods.

4. Conclusion

In this work, we manage to integrate a powerful tool into our approach with a strong diffusion process based on tensor diffusion regularization, with a considerable choice of the functions that controls the isotropic behavior near flat regions and nonlinear filter beside sharp edges, corners and contours. To legalize our model, Simulated visual results show that the efficacy of our new approach and also unfold the robustness with respect to noise reduction compared to the competitive models. However, for quantitative results, we have shown that the model yields to better PSNR and SSIM values in every comparison. In future work, a double fractional-order derivative on time and the diffusion term will be included in one model. This choice has a powerful background in the literature.

[1] Ben-loghfyry A., Hakim A. Robust time-fractional diffusion filtering for noise removal. Mathematical Methods in the Applied Sciences. 1–17 (2022).

[2] El Alaoui El Fels A., Ben-loghfyry A., El Ghorfi M. Performance of denoising algorithms in the improvement of lithological discrimination. Modeling Earth Systems and Environment. 1–8 (2022).

[3] Hakim A., Ben-Loghfyry A. A total variable-order variation model for image denoising. AIMS Mathematics. **4** (5), 1320–1335 (2019).

[4] Pan H., Wen Y. W., Zhu H. M. A regularization parameter selection model for total variation based image noise removal. Applied Mathematical Modelling. **68**, 353–367 (2019).

[5] Laghrib A., Ben-Loghfyry A., Hadri A., Hakim A. A nonconvex fractional order variational model for multi-frame image super-resolution. Signal Processing: Image Communication. **67**, 1–11 (2018).

[6] Hakim M., Ghazdali A., Laghrib A. A multi-frame super-resolution based on new variational data fidelity term. Applied Mathematical Modelling. **87**, 446–467 (2020).

[7] Baloochian H., Ghaffary H. R., Balochian S. Enhancing fingerprint image recognition algorithm using fractional derivative filters. Open Computer Science. **7** (1), 9–16 (2017).

[8] Ferrah I., Chaou A. K., Maadjoudj D., Teguar M. Novel colour image encoding system combined with ANN for discharges pattern recognition on polluted insulator model. IET Science, Measurement & Technology. **14**, 718–725 (2020).

[9] Nasreddine K., Benzinou A., Fablet R. Signal and image registration: Application to decrypt marine biological archives. Traitement du Signal. **26** (4), 255–268 (2009).

[10] Frohn-Schauf C., Henn S., Witsch K. Multigrid based total variation image registration. Computing and Visualization in Science. **11** (2), 101–113 (2008).

[11] Alaa H., Alaa N. E., Aqel F., Lefraich H. A new Lattice Boltzmann method for a Gray–Scott based model applied to image restoration and contrast enhancement. Mathematical Modeling and Computing. **9** (2), 187–202 (2022).

[12] Alaa K., Atouni M., Zirhem M. Image restoration and contrast enhancement based on a nonlinear reaction-diffusion mathematical model and divide & conquer technique. Mathematical Modeling and Computing. **8** (3), 549–559 (2021).

[13] Guichard F., Moisan L., Morel J. M. A review of PDE models in image processing and image analysis. Journal de Physique IV. **12** (1), 137–154 (2002).

[14] Chan T. F., Shen J., Vese L. Variational PDE models in image processing. Notices of the American Mathematical Society. **50** (1), 14–26 (2003).

[15] Aubert G., Kornprobst P. Mathematical problems in image processing: partial differential equations and the calculus of variations. Vol. 147. Springer Science & Business Media (2006).

[16] Perona P., Malik J. Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on pattern analysis and machine intelligence. **12** (7), 629–639 (1990).

[17] Catté F., Lions P. L., Morel J. M., Coll T. Image selective smoothing and edge detection by nonlinear diffusion. SIAM Journal on Numerical analysis. **29** (1), 182–193 (1992).

[18] Weickert J. Applications of nonlinear diffusion in image processing and computer vision. Acta Math. Univ. Comenianae. **70** (1), 33–50 (2001).

[19] Sabatier J., Agrawal O. P., Machado J. T. Advances in fractional calculus. Vol. 4. Springer (2007).

[20] Machado J. T., Kiryakova V. The chronicles of fractional calculus. Fractional Calculus and Applied Analysis. **20** (2), 307–336 (2017).

[21] Yang Q., Chen D., Zhao T., Chen Y. Fractional calculus in image processing: a review. Fractional Calculus and Applied Analysis. **19** (5), 1222–1249 (2016).

[22] Uchaikin V. V. On time-fractional representation of an open system response. Fractional Calculus and Applied Analysis. **19** (5), 1306 (2016).

[23] Chang A., Sun H. Time-space fractional derivative models for CO_2 transport in heterogeneous media. Fractional Calculus and Applied Analysis. **21** (1), 151–173 (2018).

[24] Zhao X., Sun Z. Z. Time-fractional derivatives. Numerical Methods. **3**, 23–48 (2019).

[25] Oliveira D. S., de Oliveira E. C. On a Caputo-type fractional derivative. Advances in Pure and Applied Mathematics. **10** (2), 81–91 (2019).

[26] Li C., Qian D., Chen Y. On Riemann-Liouville and Caputo derivatives. Discrete Dynamics in Nature and Society. **2011**, 562494 (2011).

[27] Weickert J. Anisotropic diffusion in image processing. Vol. 1. Teubner Stuttgart (1998).

[28] Zhang J., Chen K. A total fractional-order variation model for image restoration with nonhomogeneous boundary conditions and its numerical solution. SIAM Journal on Imaging Sciences. **8** (4), 2487–2518 (2015).

[29] Rudin L. I., Osher S., Fatemi E. Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena. **60** (1–4), 259–268 (1992).

Рівняння дифузії з дробовою часовою похідною для згладжування сигналу та зображення

Бен-Логфірі А., Хакім А.

LAMAI, Університет Каді Айяд, Марракеш, Марокко

У цій статті використовується рівняння дифузії з дробовою часовою похідною для знешумлення зображення та згладжування сигналу. Подано дискретизацію запропонованої моделі. Числові результати показують деякі чудові результати з високою продуктивністю, як візуально, так і кількісно, порівняно з деякими добре відомими конкурентними моделями.

Ключові слова: *знешумлення зображення, дробова похідна, похідна дробового порядку за часом, тензор дифузії.*