Finite Difference Method for Perona-Malik Model with Fractional Derivative and Its Application in Image Processing

Achraf Sayah, Noureddine Moussaid and Omar Gouanouane

March 15, 2022

Finite difference method for Perona-Malik model with fractional derivative and its application in image processing

Achraf SAYAH
Laboratory of mathematics and applications
Hassan II University, FST
Mohammedia Morocco
achraf.sayah@yahoo.com

Noureddine MOUSSAID
Laboratory of mathematics and applications
Hassan II University, FST
Mohammedia Morocco
noureddine.moussaid@fstm.ac.ma

Omar GOUASNOUANE
Laboratory of mathematics and applications
Hassan II University, FST
Mohammedia Morocco
omar.gouasnouane@fstm.ac.ma

***Abstract*—In this work, we consider the Perona-Malik (PM) model with fractional derivatives and its application for image processing. This model is obtained from the standard PM equation by replacing the ordinary derivative with a fractional derivative. the numerical resolution of this model is based on the finite difference method, we analyse efficient numerical methods for the fractional model, and we give practical experiments with natural images which are showing that the fractional approach is more efficient than the ordinary integer one. the proposed model has good performance in visual quality and high signal to noise ratio (SNR)/ Peak signal to noise (PSNR)**

***Keywords*- Perona-Malik; image processing; fractional derivative; Finite difference**

I. Introduction

In recent years, the use of fractional partial differential models has become increasingly popular and has also been widely applied various applications in different fields of research [20], including electromagnetism [12], stochastics, fractals, diffusion processes [17], complex networks, image processing ...etc. different models using fractional partial differential equations have been proposed, and there has been significant interest in developing numerical schemes for their solution [3], [5] such as finite element/difference methods or other methods.

In this context several authors have used models with fractional derivatives in image processing, the more studied model was proposed by Perona and Malik, researchers have proposed models based on (PM) by adding or modifying one or more parameters , in [4] the authors propose to modify the classical PM model by introducing the Caputo-Fabrizio fractional gradient inside the diffusivity function, in [18] the authors reinterpret the Perona–Malik model in the language of Gaussian scale mixtures and derive some extensions of the model, in [19] they developed a new noise removal model by combining the modified isotropic diffusion model and the modified Perona-Malik (PM) model, in [14], [15] new diffusion coefficients are proposed for Image denoising

In this work we study a numerical approach to the Perona-Malik (PM) model with Caputo's time-fractional derivatives and its application for image processing

A. Perona-Malik model with time-fractional derivative

Let $\Omega \subset \mathbb{R}^2$ denote a bounded rectangular domain of $\mathbb{R}^2$ We consider the following Perona-Malik problem :

$$\begin{cases} {}^{c}D_t^{\alpha} u(\mathbf{x},t) = div\big(G(\|\nabla u(\mathbf{x},t)\|)\nabla u(\mathbf{x},t)\big) & \text{in } \Omega \times I, \\ \dfrac{\partial u}{\partial n} = 0 & \text{on } \partial\Omega \times I, \\ u(\mathbf{x}, t=0) = u_0(\mathbf{x}) & \text{on } \Omega. \end{cases} \tag{1}$$

where :

- The operator ${}^{c}D_t^{\alpha}$ is the time fractional derivative of order α in Caputo sense with $0 < \alpha < 1$ and
$$ {}^{c}D_t^{\alpha} u(\mathbf{x},t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial u(\mathbf{x},t)}{\partial s} \frac{ds}{(t-s)^{\alpha}} $$
- $u(\mathbf{x},t)$ represents the smoothed image intensity function in the position $\mathbf{x} = (x,y) \in \Omega$ in time t
- $u_0(\mathbf{x})$ is the original image intensity function (noisy image)
- ∇ (respectively. $div(.)$) denotes the gradient (respectively divergence) operator
- $\dfrac{\partial u}{\partial n}$ is the normal derivative to the boundary
- $I = [0,T]$ where $T > 0$
- Γ designates the Gamma function
- G the diffusion coefficient

Pietro Perona and Jitendra Malik proposed in 1990 two functions as diffusion coefficient:

$$ G(\|\nabla u\|) = e^{-\left(\frac{\|\nabla u\|}{\lambda}\right)^2} $$

and

$$ G(\|\nabla u\|) = \frac{1}{1 + \left(\dfrac{\|\nabla u\|}{\lambda}\right)^2} $$

where :

λ is a positive constant controls the sensitivity to edges, is usually fixed manually(methodologies to estimate the contrast parameter λ is given in [13] and [16]) or as a function of the noise in the image.
Other expressions for the diffusion coefficient are given in [14] and [15]

II. Numerical Schemes

A. Discretization in time: a finite difference scheme

Let $0 = t_0 < t_1 < t_2 < ... < t_n = T$ an uniform discretization of $[0,T]$ of step Δt where $\Delta t = T/n$ ($t_k = k\Delta t$, $k = 0,1,...,n$)
We discretize the derivative operator in Caputo's sense by a finite difference approach by:
for all $0 \le k \le n-1$:

$$\begin{aligned} {}^cD_t^\alpha u(\mathbf{x},t_{k+1}) &= \frac{1}{\Gamma(1-\alpha)}\sum_{j=0}^{k}\int_{t_j}^{t_{j+1}}\frac{\partial u(\mathbf{x},s)}{\partial s}\frac{ds}{(t_{k+1}-s)^\alpha} \\ &= \frac{1}{\Gamma(1-\alpha)}\sum_{j=0}^{k}\frac{u(\mathbf{x},t_{j+1})-u(\mathbf{x},t_j)}{\Delta t}\int_{t_j}^{t_{j+1}}\frac{ds}{(t_{k+1}-s)^\alpha}+\overline{R}_{k+1} \end{aligned} \tag{2}$$

where $\overline{R}_{k+1}$ is the truncation error satisfying :

$$\overline{R}_{k+1} \le C_u \Delta t^{2-\alpha}$$

for a positive constant c_u depend only on u (see [5])
by computing $\displaystyle\int_{t_j}^{t_{j+1}}\frac{ds}{(t_{k+1}-s)^\alpha}$, we write (2) as:

$$\begin{aligned} {}^cD_t^\alpha u(\mathbf{x},t_{k+1}) &= \frac{1}{(\Delta t)^\alpha\Gamma(2-\alpha)}\sum_{j=0}^{k}((k-j+1)^{1-\alpha}-(k-j)^{1-\alpha}) \\ &\quad (u(\mathbf{x},t_{j+1})-u(\mathbf{x},t_j))+\overline{R_{k+1}} \end{aligned} \tag{3}$$

or by a change of index as :

$$\begin{aligned} {}^cD_t^\alpha u(\mathbf{x},t_{k+1}) &= \frac{1}{(\Delta t)^\alpha\Gamma(2-\alpha)}\sum_{j=0}^{k}((j+1)^{1-\alpha}-(j)^{1-\alpha}) \\ &\quad (u(\mathbf{x},t_{k-j+1})-u(\mathbf{x},t_{k-j}))+\overline{R_{k+1}} \end{aligned} \tag{4}$$

let us take $c_j = (1+j)^{1-\alpha} - j^{1-\alpha}$ for all $j = 0,1...,k$
and introduce the parameter $\alpha_0 = \Gamma(2-\alpha)\Delta t^\alpha$
we can easily verify that :

$$\sum_{j=0}^{k}(c_j - c_{j+1}) + c_{k+1} = (1-c_1) + \sum_{j=1}^{k-1}(c_j - c_{j+1}) + c_k$$

And note that $c_0 = 1$
we can write 4 as

$$\begin{aligned} {}^cD_t^\alpha u(\mathbf{x},t_{k+1}) &= \frac{1}{\alpha_0}\big(u(\mathbf{x},t_{k+1})-(1-c_1)u(\mathbf{x},t_k) \\ &\quad -\sum_{j=1}^{k-1}B_j u(\mathbf{x},t_{k-j}) - c_k u(\mathbf{x},t_0)\big)+\overline{R}_{k+1} \end{aligned} \tag{5}$$

where $B_j = c_j - c_{j+1}$
Let be $u_k(\mathbf{x}) \approx u(\mathbf{x},t_k)$ the approximation of $u(\mathbf{x},t_k)$
we write finally :

$${}^cD_t^\alpha u^{k+1} \simeq \frac{1}{\alpha_0}\big(u^{k+1}-(1-c_1)u^k-\sum_{j=1}^{k-1}B_j u^{k-j}-c_k u^0\big) \tag{6}$$

B. Discretization in space

Let be Δx and Δy be the space steps such that $\Delta x = \Delta y = 1$
Let us take

$$A(u(\mathbf{x},t)) := div\big(G(\|\nabla u(\mathbf{x},t)\|)\nabla u(\mathbf{x},t)\big) \tag{7}$$

we can write

$$A(u(\mathbf{x},t)) = \frac{\partial}{\partial x}\left(G(\|\nabla u\|)\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(G(\|\nabla u\|)\frac{\partial u}{\partial y}\right) \tag{8}$$

we pose :

$$\phi(\mathbf{x},t) = G(\|\nabla u(\mathbf{x},t)\|)\frac{\partial u(\mathbf{x},t)}{\partial x}$$

and

$$\psi(\mathbf{x},t) = G(\|\nabla u(\mathbf{x},t)\|)\frac{\partial u(\mathbf{x},t)}{\partial y}$$

we've using a centered finite difference

$$\frac{\partial\phi(x_i,y_j,t_k)}{\partial x} = \frac{\phi(x_{i+1/2},y_j,t_k)-\phi(x_{i-1/2},y_j,t_k)}{\Delta x}+O(\Delta x) \tag{9}$$

$$\simeq G(|\nabla u^k_{i+1/2,j}|)(u^k_{i+1,j}-u^k_{i,j}) - G(|\nabla u^k_{i-1/2,j}|)(u^k_{i-1,j}-u^k_{i,j}) \tag{10}$$

where $u^k_{i,j}$ is the approximation of $u(x_i,y_j,t_k)$
and

$$G(|\nabla u^n_{i+1/2,j}|) = G(|u^n_{i+1,j}-u^n_{i,j}|)$$

and

$$G(|\nabla u^n_{i-1/2,j}|) = G(|u^n_{i,j}-u^n_{i-1,j}|)$$

the same for

$$\frac{\partial\phi(x_i,y_j,t_k)}{\partial y} = \frac{\phi(x_i,y_{j+1/2},t_k)-\phi(x_{i-1/2},y_{j-1/2},t_k)}{\Delta y}+O(\Delta y) \tag{11}$$

$$\simeq G(|\nabla u^k_{i,j+1/2}|)(u^k_{i,j+1}-u^k_{i,j}) - G(|\nabla u^k_{i,j-1/2}|)(u^k_{i,j-1}-u^k_{i,j}) \tag{12}$$

we write finally

$$A(u(x_i,y_j,t_n)) \simeq A(u^n_{i,j}) \tag{13}$$

$$\simeq G_N.\nabla_N u^n_{i,j} + G_S.\nabla_S u^n_{i,j} + G_E.\nabla_E u^n_{i,j} + G_W.\nabla_W u^n_{i,j} \tag{14}$$

where:

$$\nabla_N u^n_{i,j} = u^n_{i-1,j} - u^n_{i,j}$$
$$\nabla_S u^n_{i,j} = u^n_{i+1,j} - u^n_{i,j}$$
$$\nabla_E u^n_{i,j} = u(^n_{i,j+1} - u^n_{i,j}$$
$$\nabla_W u^n_{i,j} = u^n_{i,j-1} - u^n_{i,j}$$

and

$$G_m = G\Bigg(|\nabla_m u(x_i, y_j, t)|\Bigg) \qquad\qquad m = N, S, E, W$$

C. Full Discretization

We will take an upper step in time for the fractional derivative in time
then we obtain an equivalence to our problem:

$$u^{k+1} = (1-c_1)u^k + \sum_{j=1}^{k-1} B_j u^{k-j} + c_k u^0 + \alpha_0 A(u^k) \quad (15)$$

and for $k = 0$

$$u^1 = \alpha_0 A(u^0) + u^0 \quad (16)$$

without forgetting the boundary condition and the initial condition

III. Experimental Results:

In this section, we give some experimental results which are obtained with application of the proposed model on images which have been corrupted by Gaussian noise, It is usually common in images acquired from cameras and telescopes, and it alters all pixels in the image.
Our proposed model is compared to the standard model of Perona-Malik, or/and classical heat equation.
We take $\Delta t = 0.1$ and $\Delta x = \Delta y = 1$
The number of iterations used is fixed in 17, and the performance of the models has been assessed by using PSNR(peak signal-to-noise ratio) and SNR(signal-to-noise ratio) which are defined by:

$$SNR = 10\log\left(\frac{\sum_{n,m} u^2}{\sum_{n,m}(\widehat{u}-u)^2}\right) \quad (17)$$

and

$$PSNR = 10\log\left(\frac{255^2}{\sum_{n,m} MSE}\right) \quad (18)$$

where u the original image and $\widehat{u}$ the restored image
and MSE is the Mean squared error given by

$$MSE = \frac{1}{MN}\sum_{n,m}(\widehat{u}-u)^2)$$

with M and N signifiant the width and height of the image
And for reasons of simplification we'll take $\lambda = 6$
Here we give a comparison between the proposed model and other models

A. First experience:

eyes original image in the left, noisy image with Gaussian noise with mean = 0 and variance $\sigma = 0.01$ in the middle, and noisy image with Gaussian noise with mean = 0 and $\sigma = 0,02$ in the right

result after 17 iterations for $\sigma = 0.01$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.63$

result after 17 iterations for $\sigma = 0.02$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.63$

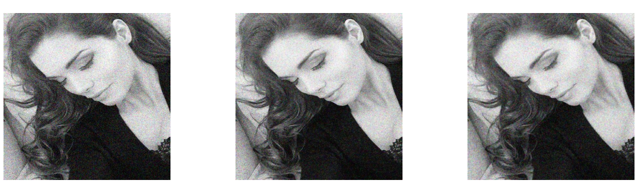
model	SNR	$PSNR$
heat equation	17.9148	23.1598
PM model	18.3240	23.5775
Proposed model with $\alpha = 0.63$	20.9576	26.2677
Proposed model with $\alpha = 0.81$	21.0049	26.8741

Table I: Performance of the proposed model for the first test with $\sigma = 0.01$

B. Second experience:

original image in the left, noisy image with Gaussian noise with mean = 0 and $\sigma = 0.01$ in the middle, and noisy image with Gaussian noise with mean = 0 and $\sigma = 0,02$ in the right

result after 17 iterations for $\sigma = 0.01$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.7$



result after 17 iterations for $\sigma = 0.02$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.7$

model	SNR	$PSNR$
heat equation	18.2883	23.6139
PM model	19.4911	24.8358
Proposed model with $\alpha = 0.7$	22.1066	27.4772

Table II: Performance of the proposed model for the second test with $\sigma = 0.01$

C. 3rd experience:

Lena ,original image in the left, noisy image with Gaussian noise with mean = 0 and $\sigma = 0.01$ in the middle, and noisy image with Gaussian noise with mean = 0 and $\sigma = 0,02$ in the right

result after 17 iterations for $\sigma = 0.01$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.7$

result after 17 iterations for $\sigma = 0.02$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.7$

model	SNR	$PSNR$
heat equation	14.7579	20.3170
PM model	15.7359	21.3160
Proposed model with $\alpha = 0.8$	19.9973	24.9802
Proposed model with $\alpha = 0.9$	19.0326	24.6910

Table III: Performance of the proposed model for the 3rd test with $\sigma = 0.02$

D. 4th experience:

Boat ,original image in the left, noisy image with Gaussian noise with mean = 0 and $\sigma = 0.01$ in the middle, and noisy image with Gaussian noise with mean = 0 and $\sigma = 0,02$ in the right

result after 17 iterations for $\sigma = 0.01$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.83$

result after 17 iterations for $\sigma = 0.02$, heat equation in the left, PM model in the middle and proposed method in the right with $\alpha = 0.83$

model	SNR	$PSNR$
heat equation	17.1683	23.2715
PM model	17.9907	24.1097
Proposed model with $\alpha = 0.83$	21.3823	27.4801
Proposed model with $\alpha = 0.9$	20.8720	27.0342

Table IV: Performance of the proposed model for the 4th test with $\sigma = 0.01$

model	SNR	$PSNR$
heat equation	14.4018	20.4189
PM model	15.2763	21.3265
Proposed model with $\alpha = 0.83$	18.6263	25.4392
Proposed model with $\alpha = 0.9$	18.3691	24.4915

Table V: Performance of the proposed model for the 4th test with $\sigma = 0.02$

IV. Conclusion

In this paper we have given a numerical scheme for the Perona-Malik model with fractional time derivative, we have applied the proposed model to images to which we have added Gaussian noise, we have compared the model with the heat equation and/or the classical Perona-Malik model, we have given the results and we have calculated the SNR and the PSNR it can be seen that the PSNR/SNR values of the proposed modelare higher than those of other models

Acknowledgment

All numerical results are obtained with Matlab $2016a$

References

[1] 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, 1990

[2] Yu-Li You and M. Kaveh *Fourth-Order Partial Differential Equations for Noise Removal*. IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 10, OCTOBER 2000

[3] M.R. Sidi Ammi , I. Jamiai, and D.F.M. Torres *A finite element approximation for a class of Caputo time-fractional diffusion equations*. Computers & Mathematics with Applications Vol. 78, Issue 5, 1 September 2019, Pages 1334-1344

[4] G. Asumu, M. Nchama, A.L Mecias, and M.R. Ricard *Perona-Malik Model with Diffusion Coefficient Depending on Fractional Gradient via Caputo-Fabrizio Derivative*. Abstract and Applied Analysis Volume 2020, Article ID 7624829, 15 pages

[5] Y. Lin and C. Xu , *Finite difference/spectral approximations for the time-fractional diffusion equation*. Journal of Computational Physics 225 (2007) 1533–1552

[6] M. Dehghan, M. Safarpoor, *Application of the dual reciprocity boundary integral equation approach to solve fourth-order time-fractional partial differential equations*. Int. J. Comput. Math. 95 (10) (2018) 2066–2081.

[7] B.A. Jacobs, C. Harley, *Application of nonlinear time-fractional partial differential equations to image processing via hybrid Laplace transform method*. J. Math. 2018 (2018) 8924547, 9 pp.

[8] M. Sarboland, *Numerical solution of time fractional partial differential equations using multiquadric quasi-interpolation scheme*. Eur. J. Comput. Mech. 27 (2) (2018) 89–108.

[9] G.J. Fix, J.P. Roop, *Least squares finite element solution of a fractional order two-point boundary value problem*. Comput. Math. Appl. 48 (2004) 1017–1033.

[10] O.P. Agrawal, *Solution for a fractional diffusion-wave equation defined in a bounded domain*. J. Nonlinear Dynam. 29 (2002) 145–155.

[11] S. Hu, Z. Liao, and W. Chen, *Sinogram Restoration for Low-Dosed X-Ray Computed Tomography Using Fractional-Order Perona-Malik Diffusion*. Mathematical Problems in Engineering Volume 2012, Article ID 391050, 13 pages

[12] N. Engheta, *On the role of fractional calculus in electromagnetic theory*. Department of Electrical & Systems Engineering,University of Pennsylvania. August 1997

[13] J. Losada and J. Nieto, *Properties of a new fractional derivative without singular kernel," Progress in Fractional Differentiation and Applications*. vol. 1, pp. 87–92, 2015.

[14] J. Yuan and J. Wang, *Perona-Malik model with a new diffusion coefficient for image denoising," International Journal of Image and Graphics*. vol. 16, no. 2, article 1650011, 2016.

[15] V. Kamalaveni, R.A. Rajalakshmi, and K.A. Narayanankutty, *Image denoising using variations of Perona-Malik model with differents edges stopping functions*. Procedia Computer Science, vol. 58, pp. 673–682, 2015.

[16] M.B. Fernández, M.G. Hidalgo, and A.L.Mecías, *New estimation method of the contrast parameter for the Perona–Malik diffusion equation*. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, vol. 4, pp. 238–252, 2014.

[17] I.Bazhlekova and E.Bazhlekovab, *Fractional derivative modeling of bioreaction-diffusion processes*. AIP Conference Proceedings 2333, 060006 (2021);

[18] L.M. Mescheder and D.A. Lorenz, *An Extended Perona–Malik Model Based on Probabilistic Models*. Journal of Mathematical Imaging and Vision volume 60, pages 128–144 (2018)

[19] N. Wang, Y. Shang, Y. Chen, M. Yang, Q. Zhang, Y. Liu and Z. Gui *A Hybrid Model for Image Denoising Combining Modified Isotropic Diffusion Model and Modified Perona-Malik Model*. in IEEE Access, vol. 6, pp. 33568-33582, 2018

[20] S.S. Ray, A. Atangana, S.C.O. Noutchie, M. Kurulay, N. Bildik , and A. Kilicman *Fractional Calculus and Its Applications in Applied Mathematics and Other Sciences*. Mathematical Problems in Engineering Volume 2014, Article ID 849395, 2 pages

[21] M.R.S. Ammi and I. Jamiai *Finite difference and Legendre spectral method for a time-fractional diffusion-convection equation for image restoration*. Discrete and Continuous Dynamical Systems - Series S 11(1) p 103-117