Science Journal of Applied Mathematics and Statistics
2024, Vol. 12, No. 1, pp. 13-19
https://doi.org/10.11648/j.sjams.20241201.12

Research Article

Optimization of the Non-Linear Diffussion Equations

Rukia Nasimiyu Fwamba[1, *], Isaac Chepkwony[1], Wekulo Saidi Fwamba[2]

[1]Department of Mathematics & Actuarial Science, Kenyatta University, Nairobi, Kenya

[2]Department of Disaster Preparedness and Engineering Management, Masinde Muliro University of Science and Technology, Kakamega, Kenya

Abstract

Partial Differential Equations are used in smoothening of images. Under partial differential equations an image is termed as a function; f(x, y), XÎR^2. The pixel flux is referred to as an edge stopping function since it ensures that diffusion occurs within the image region but zero at the boundaries; $u_x(0, y, t) = u_x(p, y, t) = u_y(x, 0, t) = u_y(x, q, t)$. Nonlinear PDEs tend to adjust the quality of the image, thus giving images desirable outlooks. In the digital world there is need for images to be smoothened for broadcast purposes, medical display of internal organs i.e MRI (Magnetic Resonance Imaging), study of the galaxy, CCTV (Closed Circuit Television) among others. This model inputs optimization in the smoothening of images. The solutions of the diffusion equations were obtained using iterative algorithms i.e. Alternating Direction Implicit (ADI) method, Two-point Explicit Group Successive Over-Relaxation (2-EGSOR) and a successive implementation of these two approaches. These schemes were executed in MATLAB (Matrix Laboratory) subject to an initial condition of a noisy images characterized by pepper noise, Gaussian noise, Brownian noise, Poisson noise etc. As the algorithms were implemented in MATLAB, the smoothing effect reduced at places with possibilities of being boundaries, the parameters C_v (pixel flux), C_f (coefficient of the forcing term), b (the threshold parameter) alongside time t were estimated through optimization. Parameter b maintained the highest value, while C_v exhibited the lowest value implying that diffusion of pixels within the various images i.e. CCTV, MRI & Galaxy was limited to enhance smoothening. On the other hand the threshold parameter (b) took an escalated value across the images translating to a high level of the force responsible for smoothening.

Keywords

Partial Differential Equations, Image Processing, Holes, Non-Linear Diffusion Equations, Pixel Flux, Threshold Parameter, Coefficient of the Forcing Term

1. Introduction

Partial differential equations (PDE's) [22] describe fundamental laws of physics. Examples: Hydrodynamics - Navier Stokes equations. Electrodynamics - Maxwell's equations. Thermodynamics - Diffusion equation. Quantum Mechanics - Schr ̈odinger equation. Typically, these equations determine the evolution of a scalar or vector field that depends on space and time [31]. Steady state problems in two or three spatial dimensions are also described by PDE's.

PDEs got some basic properties including;

a) Linearity: Express the PDE in terms of a differential

*Corresponding author: rukiafwamba2015@gmail.com (Rukia Nasimiyu Fwamba)

Received: 27 February 2024; **Accepted:** 12 March 2024; **Published:** 2 April 2024

operator,

$$\xi_u = f \tag{1}$$

f , is a function of the independent variable.

$$\xi = \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \ldots$$

Linearity means that ξ is a linear operator: For any function u and v

$\xi(u+v) = \xi u + \xi v$ and $\xi(cu) = c\xi(u)$, c is a constant

b) Homogeneity, a PDE is homogeneous if it can be expressed as $\xi_u = 0$. The PDE is inhomogeneous if it has a so called source term: $\xi_u = f$, where f is some function of the independent variables (e.g. Poisson's equation in electrostatics). Note that one can add solutions of the homogeneous PDE to the inhomogeneous solution. If a PDE is both linear and homogeneous then it obeys the principle of superposition: If f and g are two solutions of a linear, homogeneous PDE, then $c_1 f + c_2 g$ is also a solution, where c_1, c_2 are some constants.

In order to have a well defined problem we not only need the partial differential equation that governs the physics, but also a set of boundary conditions (BC) and initial conditions (IC) to specify the problem. e.g. Heat equation for an insulated bar with its two ends immersed in heat baths [21] Typical boundary conditions:

Dirichlet boundary conditions: The value of the function is specified at the boundary. e.g. temperature of heat bath.

von Neumann boundary conditions: The normal gradient of the function is specified at the boundary. e.g. the electric field from a given potential at the boundary.

Cauchy boundary conditions: Both Dirichlet and von von Neumann boundary conditions are specified.

There are usually two different problems that arise in physics:

Initial value problem: This deals with time evolution of u. At a certain time t_0, the value of u (and possibly its derivative) is identified [29].

Boundary value problem: Steady state, time independent problems. The value of u and/or its derivative is determined along a boundary that encloses the region of interest. e.g. electrostatic problems where the potential at the boundaries is specified. In some cases can also have open boundaries.

The applications of partial differential equations (PDEs) to computer vision and image processing started as early as the 1960s [11]. However, this technique did not take root until the introduction of the concept of scale space by Koenderink [19] and Witkin [30] in the 1980s. Perona and Malik's work on anisotropic diffusion [26] further drew great interest from researchers towards PDE-based methods. Nowadays, PDEs have been successfully used to model image processing and computer vision [10, 8]. There are three methods used to design PDEs. For the first method, PDEs are written down directly, based on some mathematical understandings on the properties of the PDEs (e.g., anisotropic diffusion [26], shock filter [23] and curve evolution based equations [28]. The second methods basically define an energy functional first, which collects the wish list of the desired properties of the output image, and then derives the evolution equations by computing the Euler-Lagrange equation of the energy functional [15]). The third kind of methods are axiomatic approaches, which first prescribe the properties, i.e., axioms, that the PDEs should hold and then deduce the form of PDEs from these axioms (e.g., Alvarez et al.'s axiomatic formulation of the scale-space theory [1]). All of these methods require both good insight to what properties to hold and high mathematical skills, in order to acquire the desired output.

Consider a diffusion equation coupled to chemical reactions modeled by a nonlinear term $f(u)$;

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u + f(u) \tag{2}$$

This is a physical process composed of two individual processes: u is the concentration of a substance that is locally generated by a chemical reaction $f(u)$, while u is spreading in space because of diffusion [25]. There are obviously two time scales: one for the chemical reaction and one for diffusion. Typically, fast chemical reactions require much finer time stepping than slower diffusion processes [5]. It could therefore be advantageous to split the two physical effects in separate models [12] and use different numerical methods for the two. A natural spitting in the present case is;

$$\frac{\partial u^*}{\partial t} = \alpha \nabla^2 u^* \tag{3}$$

$$\frac{\partial u^{**}}{\partial t} = f(u^{**}) \tag{4}$$

Non-linear Diffusion equations can be applied to initially given images. Let the image to be processed be, $\Phi^0(x,y), \Phi^0 : \Omega \to \mathbb{R}, \Omega \subset \mathbb{R}^n$ is a domain in space. Ω is a surface and $n = 2$. Introducing time t this image deforms in a partial differential evolution equation (3) according to;

$$\frac{\partial \Phi}{\partial t} = F[\Phi(x,y,t)] \tag{5}$$

where:

$\Phi(x,y,t) : \Omega \times [0,T] \to \mathbb{R}$ is the evolving image [20]

$F : \mathbb{R} \to \mathbb{R}$ is an operator that characterizes the given algo-

rithm. The solution of this differential equation gives the processed image at scale t [27].

Perona et al [26] transformed the diffusion equation from linear to non- linear restoring borderlines. Currently the following nonlinear PDE is used,

$$w_t - \nabla.(f(|\nabla G_\sigma * w|)\nabla_w) = 0 \tag{6}$$

$w(t, y)$ is an unknown function defined in $R_T = [0, T] \times \Omega$

This equation is accompanied by Neumann boundary conditions and initial conditions;

$$\frac{\partial w}{\partial v} = 0, \text{ on } \; I \times \partial\Omega \quad w(0, y) = w_0(y) \;\; \text{ in } \;\; \Omega$$

w, the unit normal vector to the boundary of Ω. Assuming that

$\Omega \subset \mathbb{R}^d$, a bounded rectangular domain

$J = [0, T]$, a scaling interval,

$g : \mathbb{R}_0^+ \to \mathbb{R}^+$, a non-increasing function,

$g\sqrt{s}$ is smooth,

$g(0) = 1, g(s) \to 0, s \to \infty$,

$G_\sigma \in C^\infty(\mathbb{R}^d)$, a smoothing kernel (e.g. the Gauss function),

$$\begin{aligned} &\int_{\mathbb{R}^d} G_\sigma(y)dy = 1, \int_{\mathbb{R}^d} |\nabla G_\sigma| dy \le C_\sigma, \\ &G_\sigma(y) \to \delta_y, \sigma \to 0, \end{aligned} \tag{7}$$

δy is the Dirac measure at the point, $y, w_0 \in L_\infty(\Omega)$

This research modeled an equation of nonlinear diffusion for image processing, optimized the nonlinear heat equation. The objectives of this research was to; optimize the nonlinear heat equation, establish viable values for C_v, C_f, b that enhance the image smoothening process.

2. Materials and Methods

Digital images are given on discrete (regular) grids [13]. This lends itself for discretizing the diffusion equation to obtain numerical schemes [24] that can be solved using MATLAB. The considered numerical schemes are stable and efficient.

As the numerical algorithms are executed in MATLAB, Optimization is done to establish the most appropriate values of C_v, C_f, b, t. These values are responsible for smoothening of images that are initially noisy [14].

A model below was used in this research;

$$\frac{\partial u}{\partial t} = c_v\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + c_f u(1-u)(u-b) \tag{8}$$

2.1. Numerical Approaches

There is a wide body of literature on the numerical solution of PDE's [7]. Many algorithms are available including:

Finite difference method which is the most common approach. Based on approximating derivatives by finite differences. e.g. $\frac{\partial u(x,t)}{\partial t} \approx \frac{u_{j+1}^n - u_j^n}{\Delta x}$

where $u_j^n = u(j\Delta x, n\Delta t)$.

Finite Element method: Basic idea is to divide volume of selected region into discrete pieces (polygons) and approximate the solution by simple functions on these pieces. Complex geometries whose boundaries are curved are easily implemented too. It is therefore common among engineers to model solids and structures.

Spectral methods: Approximate the solution and the boundary condition with a Fourier series and substitute into the PDE. Use the inverse Fast Fourier transform to compute the Fourier coefficients of the solution.

Monte Carlo: Use stochastic methods to find solutions of a PDE.

The fact that linear PDEs fall into three canonical forms [17] is not of importance from computational point of view. What is relevant whether we are dealing with;

Time evolution - Initial value problem: Wave equation, diffusion equations. Compute the time evolution for a given initial condition subject to boundary conditions.

Static solution - Boundary value problem: Laplace equation, Poisson equation: Find solutions that satisfy the boundary conditions around the region of interest.

Because of the way the boundary conditions need to be enforced, the numerical methods will be different for these two types of problems. In boundary value problems one cannot "integrate in from the boundary" [18] in the same sense one can "integrate forward in time" for initial value problems. For static problems the goal of the numerical method is to converge on the correct solution everywhere at once.

To apply numerical methods of solution to a domain it must be divided into equal spaced points. An example of a rectangular 2D domain can be an image or a photograph.

2.1.1. Alternating Direction Implicit (ADI) Method

The numerical discretization in time was conducted using the Alternate Direction Implicit method The merit of using the ADI method is good stability even when there is alarge convection to diffusion ratio (i.e. large Peclet number, P e > 2), which is typically the case for pultrusion.

Alternating direction implicit (ADI) methods modify Crank-Nicolson to maintain convergence but introduce the

tridiagonal matrices which are more efficient [4].

The ADI methods are based on a factored form. The factors are used to split each full timestep into two partial timesteps [9]. The equation for ADI below was used to solve the non-linear diffusion equation in MATLAB;

$$\begin{aligned}\eta u_{i+1,j}^{n+\frac{1}{2}} + (1+2\eta)u_{i,j}^{n+\frac{1}{2}} - \eta u_{i-1,j}^{n+\frac{1}{2}} &= \varphi u_{i,j+1}^{n} + (1-2\varphi)u_{i,j}^{n} + \varphi u_{i,j-1}^{n} + C_f u_{i,j}^{n}(1-u_{i,j}^{n})(u_{i,j}^{n}-b)\\ -\varphi u_{i,j+1}^{n+1} + (1+2\varphi)u_{i,j}^{n+1} - \varphi u_{i,j-1}^{n+1} &= \eta u_{i+1,j}^{n+\frac{1}{2}} + (1-2\eta)u_{i,j}^{n+\frac{1}{2}} + \eta u_{i+1,j}^{n+\frac{1}{2}} + C_f u_{i,j}^{n}(1-u_{i,j}^{n})(u_{i,j}^{n}-b)\end{aligned} \tag{9}$$

2.1.2. Two-Point Explicit Group Successive Over-Relaxation (2-EGSOR)

2-EGSOR iterative method can be displayed in a grid as below:

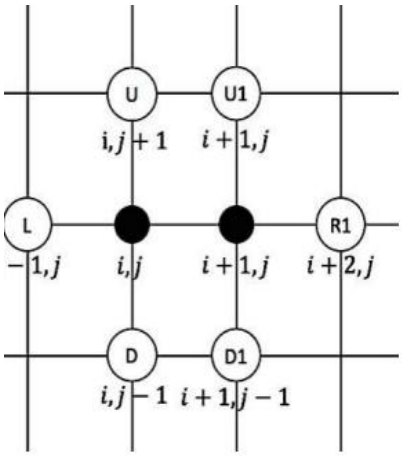

Figure 1. *Grid points in 2-EGSOR.*

The following equation was used in MATLAB to smoothen sampled noisy images of CCTV, MRI & Galaxy nature

$$\begin{pmatrix}\alpha & C_v\\ C_v & \alpha_1\end{pmatrix}\begin{pmatrix}u_{i,j}^{n+1}\\ u_{i+1,j}^{n+1}\end{pmatrix} = \begin{pmatrix}u_{i,j}^{n} + C_v(u_{i,j+1}^{n} + u_{i,j-1}^{n+1} + u_{i-1,j}^{n+1})\\ u_{i+1,j}^{n} + C_v(u_{i+1,j+1}^{n} + u_{i+1,j-1}^{n+1} + u_{i+2,j}^{n})\end{pmatrix} + C_f u_{i,j}^{n}(1-u_{i,j}^{n})(u_{i,j}^{n}-b) \tag{10}$$

2.2. Optimization

Let $\Omega \subset \mathbb{R}^2$ denote a subset of the plane and $I(.,t):\Omega \to \mathbb{R}$ be a family of gray scale images, and then anisotropic diffusion is defined as;

$$\frac{\partial I}{\partial t} = div\left(c(x,y,t)\nabla I\right) = \nabla_c.\nabla I + c(x,y,t)\Delta I \tag{11}$$

Δ, Laplacian

∇, gradient

$div(.....)$, divergence operator

$c(x,y,t)$, Diffusion coefficient controls the rate of diffusion [16] and is usually chosen as a function of the image gradient so as to preserve edges in the image [3].

Perona et al [26] pioneered the idea of anisotropic diffusion and proposed two functions for the diffusion coefficient:

$$c\left(\|\nabla I\|\right) = e^{-\left(\|\nabla I\|/K\right)^2} \tag{12}$$

And

$$c\left(\|\nabla I\|\right) = \frac{1}{1+\left(\|\nabla I\|/K\right)^2} \tag{13}$$

K Controls the sensitivity to edges and is chosen experimentally or as a function of the noise in the image.

Let M denote the manifold of smooth images, then the diffusion equations presented above can be interpreted as the gradient descent equations for the minimization of the energy [6] function $E: M \to \mathbb{R}$ defined by,

$$E(I)=\frac{1}{2}\int_{\Omega} g\left(\|\nabla I(x)\|^2\right)dx \tag{14}$$

Where $g:\mathbb{R}\to\mathbb{R}$ is a real-valued function which is intimately related to the diffusion coefficient. Then for any compactly supported infinitely differentiable test function h,

$$\left.\frac{d}{dx}\right|_{t=0} E[I+th]=\left.\frac{d}{dx}\right|_{t=0}\frac{1}{2}\int_{\Omega} g\left(\|\nabla(I+th)(x)\|^2\right)dx \tag{15}$$

$$=\int_{\Omega} g'\left(\|\nabla I(x)\|^2\right)\nabla I.\nabla h dx$$

$$=-\int_{\Omega} div\left(g'(\|\nabla I(x)\|^2\right)\nabla I)h dx$$

The last line of [15] follows from multidimensional integration by parts [2]. Letting ∇E_i denote the gradient of E with respect to $L^2(\Omega,\mathbb{R})$ the inner product evaluated at I, this gives;

$$\nabla E_I=-div(g'\left(\|\nabla I(x)\|^2\right)\nabla I \tag{16}$$

Thus, the gradient descent equations on the function E are given by;

$$\frac{\partial I}{\partial t}=-\nabla E_I=div\left(g'\left(\|\nabla I(x)\|^2\right)\right)\nabla I) \tag{17}$$

Thus by letting $c=g'$ the anisotropic diffusion equations are obtained.

Optimization was done for (8) using the objective function;

$$T(c_v,c_f)=b\sqrt{c_v^2+c_f} \tag{18}$$

c_v, pixel flux

c_f, Coefficient of theForcing term

b, the threshold parameter

The minimized values of C_v, C_f, b were applied in the numerical schemes of ADI and 2-EGSOR in MATLAB generating smoothened images.

3. Results

Parameter estimation gives the approximate values of time and the constants C_v, C_f and b in (8).

3.1. Approximated Parameters Under EGSOR

Table 1. *Parameter estimation for MRI image under EGSOR.*

t	C_v	C_f	b
0	0.000125	0.005	4
0.111111	0.00000006968	0.006957	4.626973
0.222222	0.00000001597	0.007049	4.624980
0.333333	0.0000003090288	0.006933	4.624997
0.444444	0.0000002684966	0.006931	4.624979
0.555555	0.000000213459	0.007012	4.624996
0.666666	0.00000000845238	0.006977	4.624980
0.777777	0.00000022759	0.006979	4.626915
0.888888	0.00000000882199	0.06988	4.624981
1	0.0000000413959	0.07018	4.624981

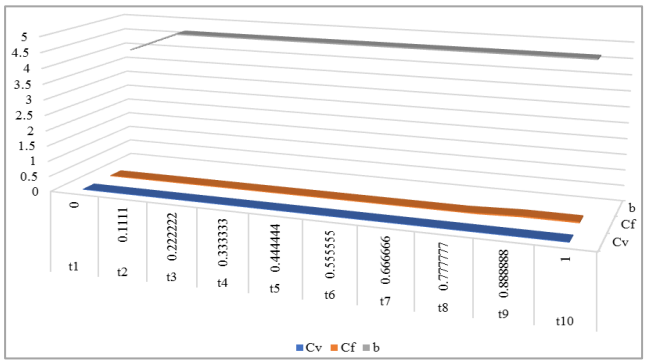

Figure 2. *Time against C_v, C_f and b for MRI image under EGSOR.*

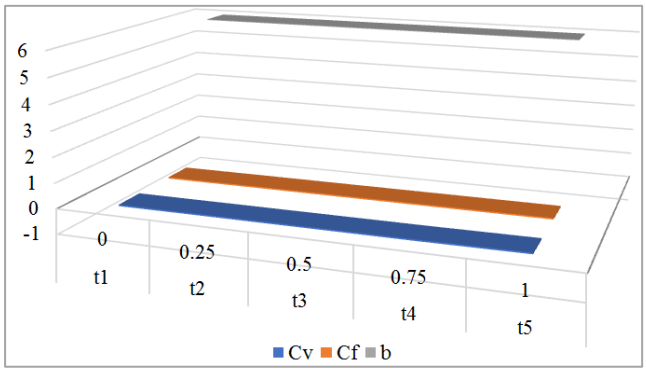

Figure 3. *Time against Cv, Cf and b for CCTV image under EGSOR.*

Table 2. *Parameter estimation for CCTV image under EGSOR.*

t	C_v	C_f	b
0	0.000008	0.005	6
0.25	0.000000068621	0.011784	5.999747
0.5	0.000000256962	0.011789	5.999747
0.75	0.0000002491727	0.011791	5.999743
1	0.000000157290	0.011793	5.999752

3.2. Approximated Parameters Under ADI

Table 3. *Parameter estimation for MRI image under ADI.*

t	C_v	C_f	b
0	0.000125	0.005	4
0.111111	0.000003	0.012811	4.000023
0.222222	0.00000045924	0.012840	4.002498
0.333333	0.00000002024	0.012792	4.000023
0.444444	0.0000000731162	0.012818	4.002586
0.555555	0.0000001149168	0.012883	4.000027
0.666666	0.0000020747	0.012809	4.002546
0.777777	0.0000002403366	0.012870	4.002569
0.888888	0.001742	0.012647	4.000024
1	0.000745	0.012642	4.011794

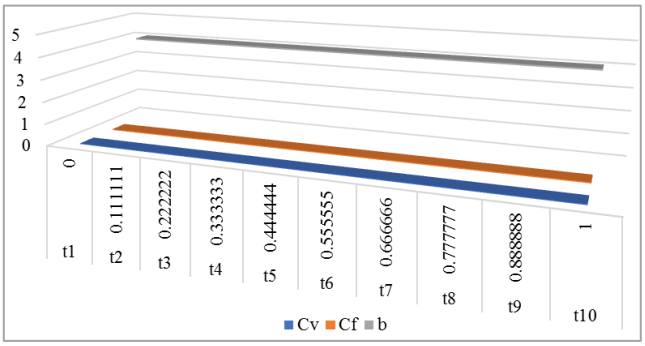

Figure 4. *Time against C_v, C_f and b for MRI image under ADI.*

Table 4. *Parameter estimation for CCTV image under ADI.*

t	C_v	C_f	b
0	-0.000000068622	0.011784	5.999747
0.25	0.00196019959	0.01607929220	5.999766
0.5	0.00196019959	0.01607929220	5.999766
0.75	0.00196019959102	0.01607929220	5.999766
1	0.0196019959102702	0.01607929220	5.999766

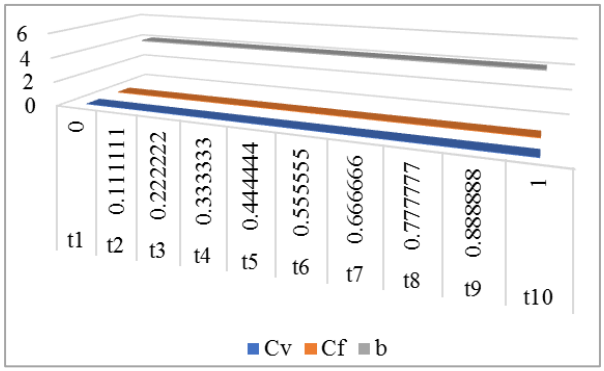

Figure 5. *Time against C_v, C_f, b for CCTV image under ADI.*

4. Discussion

From the graphs the parameter b which is the threshold parameter maintains the highest value as opposed to C_v which is the pixel flux, it registers the lowest value implying pixel diffusion is limited in the smoothening process. On the other hand the forcing term coefficient C_f is of moderate value, inputing minimum force into the image in the course of smoothening hence limiting blurring.

5. Conclusion

The paper sort to optimize parameters b, C_v, C_f. Parameter b works best when optimized to the highest value as compared to C_v, C_f. Smoothening of images can be done well when parameters are minimized. I recommend use of other computer programs other than MATLAB in future optimization.

Abbreviations

MRI: Magnetic Resonance Imaging
CCTV: Closed Circuit Television
ADI: Alternating Direction Implicit
2-EGSOR: Two-point Explicit Group Successive Over-Relaxation
MATLAB: Matrix Laboratory
PDEs: Partial differential equations

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Alvarez, L., Guichard, F., Lions, P. L., Morel, J. M. (1993): Axioms and fundamental equations of image processing. Arch. for Rational Mechanics 123(3), 199–257.

[2] Benson, D. A., Wheatcraft, S. W., and Meerschaert, M. M. (2000). Application of a fractional advection-dispersion equation. *Water Resources Research, 36(6): 1403-1412*.

[3] Bueno-Orovio, A., Kay D., and Burrage, K. (2012). Fourier spectral methods for fractional in-space reaction-diffusion equations. *Journal of Computational Physics*.

[4] Caselles, V., and Sbert, C. (1996) What is the best causal scale space for three-dimensional images? *SIAM Journal applied mathematics*, 56(4): 1119–1246.

[5] Caputo, M. (1967). Linear models of dissipation whose q is almost frequency Independent Geophysical. *Journal of the Royal Astronomical Society*, 13(5): 529-539.

[6] Cattaneo, C. (1948). Sulla conduzione del calore. Atti Semin. Mat. Fis. Della Universita di Modena, 3: 3.

[7] Chen, Y., Yu, W., and Pock, T. (2015). Learning optimized reaction diffusion processes for effective image restoration. *Proceedings of IEEE conference on computer vision and pattern recognition (Boston: IEEE)* pp 5261-69.

[8] Caselles, V., Morel, J. M., Sapiro, G., A. Tannenbaum (1998): Special issue on partial differential equations and geometry-driven diffusion in image processing and analysis. IEEE Trans. Image Processing 7(3).

[9] Compte, A., and Metzler, R. (1997). The generalized Cattaneo equation for the description of anomalous transport processes. *Journal of Physics A: Mathematical and General, 30: 7277-7289*.

[10] Ebihara, M., Mahara, H., Osa, A., and Miike, H. (2003) Segmentation and edge detection of noisy image and low contrast image based on a reaction-diffusion model The Journal of the Institute of Image Electronics Engineers of Japan, 32: 378–385.

[11] Einstein, A. (1905). "die von der molekularkinetischen Theorie der Warme geforderte Bewegung von in ruhenden Flussigkeiten suspendierten Teilchen "Annalen der Physik (in German). 322(8): 549-560.

[12] Gabor, D (1965) Information theory in electron microscopy. Laboratory Investigation 14, 801–807.

[13] Gerbrands, J. J., Schavemaker, J. G. M., Reinders, M. J. T., and Backer, E. (2000). Image Sharpening by morphological filtering, Pattern Recognition, 33: 997–1012.

[14] Gilboa, G., Sochen, N., Zeevi, Y. (2004): Image enhancement and denoising by complex diffusion processes. IEEE Trans. Pattern Analysis and Machine Intelligence 26(8), 1020–1036.

[15] Grégoire, N., and Ana, I. T. M. (2020). "On the maximization problem for solutions of reaction–diffusion equations with respect to their initial data". In: Mathematical Modelling of Natural Phenomena 15, p. 71. https://doi.org/10.1051/mmnp/2020030

[16] Haar Romeny, B. M. (1994). Geometry-Driven Diffusion in Computer Vision. Kluwer Academic Publishers.

[17] Jacobs, B. A., and Harley, C. (2013). A comparison of two hybrid methods for solving linear time-fractional partial differential equations on a two-dimensional domain. Sub-mitted to Abstract and Applied Analysis Special Issue: New Trends on Fractional and Functional Differential Equations.

[18] Lee, J. S. (1980). Digital image enhancement and noise filtering by use of local statistics. IEEE Pat. Anal. Mach. Intell., 2: 165–168.

[19] Jumarie, G. (2005). On the solution of the stochastic differential equation of exponential growth driven by fractional Brownian motion. Applied Mathematics Letters, 18: 817-826.

[20] Koenderink, J. (1984) The structure of images. Biological Cybernetics 50, 363–370.

[21] Malladi, R., Sethian, J. A., and Vemuri, B. C. (1995). "Shape modeling with front propagation," IEEE Trans. Pattern Anal. Machine Intel, vol. 17, pp. 158–175, Feb.

[22] Mumford, D., and Shah, J. (1989). "Optimal approximations by piecewise smooth functions and variation problems," Commun. Pure Appl. Math., vol. 42.

[23] Oldham, K. B., and Spanier, J. (1974). The Fractional Calculus. Academic Press, Inc.

[24] Osher, S., and Rudin, L. (1990) Feature oriented image enhancement using shock filters. SIAM J. Numerical Analysis, 27: 919–940.

[25] Ozkan, M. K., Sezan, M. I., and Tekalp, A. M. (1993) Adaptive motion compensated filtering of noisy image sequences. IEEE Trans. Circuits and Systems for Video Technology, 3: 277–290.

[26] Peaceman, D. W., and Rachford Jr. H. H. (1955). The numerical solution of parabolic and elliptic differential equations. *Journal of the Society for Industrial and Applied Mathematics*, 3(1): 28–41.

[27] Perona, P., and Malik, J. (1990). "Scale-space and Edge Detection Using Anisotropic Diffusion," IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(7), pp. 629-639.

[28] Rasmussen, F. S., Sonne M. R., Larsen, M. J., Spangeberg, Lilleheden L. T. and Hattel, J. H.(2018). A characterization study relating cross-sectional distribution of fiber volume fraction and permeability. Proc. 22nd Int. Conf. on Compos. Mater. (Melbourn, Australia).

[29] Sapiro, G. (2001) Geometric Partial Differential Equations and Image Analysis. Cambridge University Press.

[30] J. H. (2017): 2D Numerical Modelling of thge Resin Injection Pultrusion Process Including Experimental Resin Kinetics and Temperature Validation. Proceeding: ICCM21.

[31] Witkin, A. (1983) Scale-space filtering. In: Proc. Int. Joint Conf. Artificial Intelligence.