Bol. Soc. Paran. Mat.
©SPM – E-ISSN-2175-1188
SPM: www.spm.uem.br/bspm

(3s.) **v. 2025 (43) 1** : 1–10.
ISSN-0037-8712
doi:10.5269/bspm.78430

New Gradient EDGE Detection Method Based on Caputo k-Fractional Derivative*

Ahfouda Belhadi†, Bachir Dehda, Abdelaziz Azeb Ahmed, Fares Yazid and Fatima Siham Djeradi

ABSTRACT: In this paper, we propose a new gradient edge detection method based on the Caputo k-fractional derivative. In this method, we construct a new 33 fractional order mask to obtain the gradient components. The main property of this technique is that it has a double memory effect in detecting edges. The experimental results for some natural and medical test images show that our proposed edge detection method outperforms both Caputo and Caputo-Fabrizio fractional methods in terms of both visual perception and peak signal to noise ratio.

Key Words: Caputo fractional derivative, Caputo-Fabrizio fractional derivative, Caputo k-fractional derivative, traditional detectors, edge detection, peak signal to noise ratio (PSNR).

Contents

1. Introduction

Edge detection is an important task in image processing and computer vision problems. Its goal is to determine an object via its boundaries using curves. Mathematically, edge detection means finding the discontinuous points of an image. In this context, many mathematical methods have been applied to detect edges such as gradient and Laplacian methods [1]. The Sobel, Roberts and Prewitt detectors are one of the first classical techniques [2,3,4] that compute the discrete first order derivative using convolution with kernels or masks. Since these masks are constructed using integer-order derivatives, these detectors are very sensitive to noise and present false edges. To avoid these defects, many researchers have applied the fractional derivative [5] as a tool to freely generate different masks by their fractional parameters. For instance, G. Asumu et al [6] have used the Caputo and Caputo-Fabrizio fractional derivative for detecting edges. However, the mask that corresponds to these operators has only one parameter to process, then it is less adaptive when we want more detail for the image. For this reason, in this paper we propose a new more adaptive gradient method based on Caputo k-fractional derivative [7] that is characterized by two parameters α and k (α for derivative order and k for fractional derivative operators). Because this operator has a dual memory effect property, our method can overcome both Caputo and Caputo-Fabrizio fractional derivative gradient methods. The obtained results prove the efficiency and robustness of our method.

* This research work is supported by the General Direction of Scientific Research and Technological Development (DGRSDT), Algeria.
† Corresponding author.
2010 *Mathematics Subject Classification:* 26A33, 94A08.
Submitted August 15, 2025. Published December 19, 2025

This paper is organized as follows: In Sect. 2, we remember the traditional detectors such as Sobel, Roberts and Prewitt operators. In Sect. 3, we introduce main fractional derivatives operators like as Caputo and Caputo k-fractional derivatives. In Sect. 4, we present our proposed edge detection method that is characterized by its algorithm. In Sect. 5, we give experimental results comparing our edge detection method with other methods for both natural and medical test images.

Finally, we close this paper with a conclusion and main references.

2. Traditional gradient methods for edge detection

For an image $f(x, y)$, its gradient is the vector $G = \begin{pmatrix} G_x \\ G_y \end{pmatrix}$, such as G_x and G_y are horizontal and vertical approximations of the first derivatives respectively.

The first step in the edge detection process is to convolve the image with a kernel $h(x, y)$ as the form:

$$(f * h)(x, y) = \sum_{i,j} f(x - i, y - j)h(i, j). \tag{2.1}$$

This formula leads to calculate G_x and G_y according to the considered kernel $h(x, y)$. For example: In the **Sobel technique**, we use the following expressions:

$$G_x = f * \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, \quad G_y = f * \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}. \tag{2.2}$$

In the **Roberts technique**, G_x and G_y are calculated by:

$$G_x = f * \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad G_y = f * \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{2.3}$$

And, in the **Prewitt technique**, we have the following formula:

$$G_x = f * \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}, \quad G_y = f * \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}. \tag{2.4}$$

The second step for getting edges is to compute the gradient magnitude [3,8,9,10] as follows:

$$\|G(x, y)\|_2 = \sqrt{G_x^2 + G_y^2}. \tag{2.5}$$

The third step is to apply the thresholding method to gradient magnitude by the following relation:

$$edge = \begin{cases} 0, & \text{if } \|G(x, y)\|_2 < Threshold, \\ 1, & \text{if } \|G(x, y)\|_2 \geq Threshold. \end{cases} \tag{2.6}$$

3. Caputo k-fractional derivative operator

In this section, we recall some basic definitions and properties of fractional derivatives that are useful for our study.

Definition 3.1 *[7] Let $0 < \alpha < 1$ and $u \in H^1(a, b)$, $b > a$. The Caputo fractional derivative of order α of u is defined by:*

$${}^C D_a^\alpha u(t) = \frac{1}{\Gamma(1 - \alpha)} \int_a^t \frac{u'(\tau)}{(t - \tau)^\alpha} d\tau. \tag{3.1}$$

Definition 3.2 *[7] Let $\alpha > 0$, $\alpha \notin \{1, 2, 3, \dots\}$, and $k \geq 1$ (k is a real number), $n = [\alpha] + 1$. The Caputo k-fractional derivative of order α of u is given by:*

$${}^C D_a^{\alpha,k} u(t) = \frac{1}{k\Gamma_k\left(n - \frac{\alpha}{k}\right)} \int_a^t \frac{u^{(n)}(\tau)}{(t - \tau)^{\frac{\alpha}{k} - n + 1}} d\tau, \tag{3.2}$$

where

$$\Gamma_k(x) = k^{\frac{x}{k}-1}\Gamma\left(\frac{x}{k}\right). \tag{3.3}$$

In a special case for $0<\alpha<1$, we have the following definition.

Definition 3.3 *Let $0<\alpha<1,\ k\geq 1,$ and $u\in H^1(a,b),\ b>a$. The Caputo k-fractional derivative of order α is given by:*

$$^{C}D_a^{\alpha,k}u(t) = \frac{1}{k\Gamma_k\left(1-\frac{\alpha}{k}\right)}\int_a^t \frac{u'(\tau)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau. \tag{3.4}$$

Properties 1 *The Caputo k-fractional derivative of order α verifies:*

1. *For $n-1<\alpha<n$, we have*
$$\lim_{\substack{>\\ k\longrightarrow 1}} {}^{C}D_a^{\alpha,k}u(t) = {}^{C}D_a^{\alpha}u(t). \tag{3.5}$$

2. *For $n-1<\alpha<n$, we have*
$$\lim_{\substack{<\\ \alpha\longrightarrow n}} {}^{C}D_a^{\alpha,1}u(t) = u^{(n)}(t). \tag{3.6}$$

4. Proposed gradient edge detection method

In this section, we aim to provide approximate formulas for G_x and G_y and to describe the algorithm corresponding to our proposed method.

4.1. Gradient approximation using Caputo k-fractional derivative

We consider the Caputo k-fractional derivative that is given by (3.4) and we divide the interval $[a,t]$ into equal distances as follows:

$$t_\ell = a+\ell\Delta t, \quad \Delta t = \frac{t-a}{N},\ \ell=0,\dots,N.$$

To simplify calculus, we take $a=0$. The formula (3.4) can be expressed as:

$$^{C}D_a^{\alpha,k}u(t) = \frac{1}{k\Gamma_k\left(1-\frac{\alpha}{k}\right)}\sum_{\ell=0}^{\ell=N-1}\int_{t_\ell}^{t_{\ell+1}} \frac{u'(\tau)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau. \tag{4.1}$$

In each sub-interval $[t_\ell, t_{\ell+1}]$ we interpolate u' at points $t_\ell,\ t_{\ell+1}$ as

$$u'(\tau) \approx -\frac{(\tau-t_{\ell+1})}{\Delta t}u'(t_\ell) + \frac{(\tau-t_\ell)}{\Delta t}u'(t_{\ell+1}),$$

then

$$\int_{t_\ell}^{t_{\ell+1}} \frac{u'(\tau)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau \approx -\frac{u'(t_\ell)}{\Delta t}\int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_{\ell+1})}{(t-\tau)^{\frac{\alpha}{k}}}d\tau + \frac{u'(t_{\ell+1})}{\Delta t}\int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_\ell)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau,$$

by approximating $u'(t_\ell),\ u'(t_{\ell+1})$ with forward, backward finite differences schema respectively, we obtain

$$\begin{aligned}
\int_{t_\ell}^{t_{\ell+1}} \frac{u'(\tau)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau \approx& \frac{(u_{\ell+1}-u_\ell)}{(\Delta t)^2}\left[\int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_\ell)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau - \int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_{\ell+1})}{(t-\tau)^{\frac{\alpha}{k}}}d\tau\right]\\
\approx& \frac{(u_{\ell+1}-u_\ell)}{(\Delta t)^2}\left[\int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_\ell)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau - \int_{t_\ell}^{t_{\ell+1}}\frac{(\tau-t_\ell-\Delta t)}{(t-\tau)^{\frac{\alpha}{k}}}d\tau\right]\\
\approx& \frac{(u_{\ell+1}-u_\ell)}{\Delta t}\int_{t_\ell}^{t_{\ell+1}}\frac{d\tau}{(t-\tau)^{\frac{\alpha}{k}}}\\
\approx& k\frac{(u_{\ell+1}-u_\ell)}{(\alpha-k)\Delta t}\left[(t-t_\ell-\Delta t)^{1-\frac{\alpha}{k}} - (t-t_\ell)^{1-\frac{\alpha}{k}}\right].
\end{aligned}$$

Therefore,

$$
{}^{C}D_{0}^{\alpha,k}u(t) \approx \frac{1}{\Delta t(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} \sum_{\ell=0}^{N-1} \left[(t-(\ell+1)\Delta t)^{1-\frac{\alpha}{k}} - (t-\ell\Delta t)^{1-\frac{\alpha}{k}} \right] (u_{\ell+1}-u_{\ell}). \tag{4.2}
$$

Now, if we put

$$
c_{\ell} = \frac{1}{\Delta t(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} \sum_{\ell=0}^{N-1} \left[(t-(\ell+1)\Delta t)^{1-\frac{\alpha}{k}} - (t-\ell\Delta t)^{1-\frac{\alpha}{k}} \right], \ \ell = 0, \dots, N-1,
$$

then,

$$
{}^{C}D_{0}^{\alpha,k}u(t) \approx -c_0u_0+(c_0-c_1)u_1+(c_1-c_2)u_2+\cdots+(c_{N-3}-c_{N-2})u_{N-2}+(c_{N-2}-c_{N-1})u_{N-1}+c_{N-1}u_N. \tag{4.3}
$$

For numerical images $\Delta t = 1$, i.e., $(N = t)$, and in practice we can suffice only with the last two terms of (4.3). This leads to (4.3) becoming

$$
{}^{C}D_{a}^{\alpha,k}u(t) \approx (c_{t-2}-c_{t-1})u(t-1)+c_{t-1}u(t). \tag{4.4}
$$

Consequently, at the point (x, y) the gradient components of an image $u(x, y)$ take the following form:

$$
\begin{aligned}
G_x &= w_0u(x,y)+w_1u(x-1,y),\\
G_y &= w_0u(x,y)+w_1u(x,y-1),
\end{aligned} \tag{4.5}
$$

such as,

$$
\begin{aligned}
w_0 &= -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)},\\
w_1 &= -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}\left[2-2^{1-\frac{\alpha}{k}}\right].
\end{aligned}
$$

One of the masks that we can construct is as follows:

$$
h_x = \begin{pmatrix}
\frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}\\
\frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}\\
\frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}
\end{pmatrix},
$$

$$
h_y = \begin{pmatrix}
\frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[2-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}\\
\frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & \frac{\left[1-2^{1-\frac{\alpha}{k}}\right]}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}\\
-\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)} & -\frac{1}{(\alpha-k)\Gamma_k\left(1-\frac{\alpha}{k}\right)}
\end{pmatrix}. \tag{4.6}
$$

4.2. Proposed edge detection algorithm

Here, we describe the algorithm steps and its flowchart.
Step 1. Upload the noisy image.
Step 2. Calculate the gradient components at each pixel (x, y) by the convolution using (4.6).
Step 3. Compute the gradient magnitude at each pixel (x, y) using (2.5).
Step 4. Select a threshold value and apply the global thresholding technique.
Step 5. Get the resulting edge image.

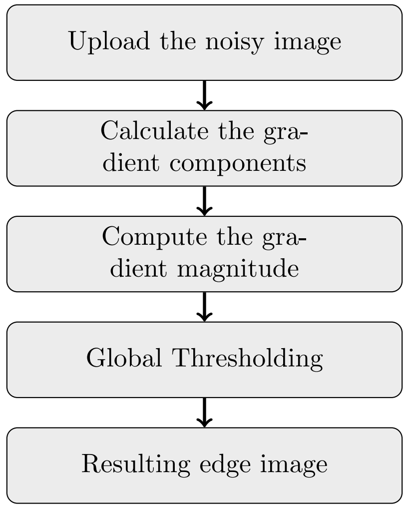

Figure 1: Flowchart corresponding to our proposed method algorithm

5. Experimental results

To show the effectiveness and good performance of our proposed edge detection method, we compare it to some recent methods that are implemented in [6]. To verify that, we perform the test on a number of natural and medical grayscale images with a size of 512512 pixels. These images are corrupted by white gaussian noise with $\sigma = 1$. For the threshold T, we take some quantiles as its value. The algorithm is implemented in Matlab, and the simulation results are presented in the table 1

Images	Implemented method in Caputo sense	Implemented method in Caputo-Fabrizio sense	Proposed method	Parameters		
				α	k	T
Cameraman	5.5468	5.4506	5.5880	0.90	1.01	139.0036
	5.4136	5.3909	5.4354	0.69	1.02	97.0987
	5.3763	5.3729	5.3955	0.99	1.03	86.9632
Boat	5.6332	5.3653	5.7588	0.90	1.01	136.9786
	5.3712	5.3475	5.4046	0.96	1.02	115.7935
	5.3142	5.3134	5.3438	0.99	1.03	90.5973
Humanbrain	8.3416	7.8182	8.4970	0.90	1.01	105.8569
	7.4598	7.3719	7.5720	0.69	1.02	80.1996
	7.9632	7.9486	8.0204	0.99	1.03	104.7245

Table 1: Comparison of PSNR (dB) results between the methods implemented in [6] and our proposed method

This table presents PSNR values for three images at different values of the parameters. The PSNR is used as a criterion to measure the quality between the original and the detected edge image, such as a greater PSNR value gives better image quality. Obviously, the simulation results show that at each parameter, the PSNR value corresponding to our proposed method is the highest compared to all other methods for each image. Representatively, in the Fig 2, Fig 3 and Fig 4 our detector outperforms others. Consequently, with our pro-posed detection method, the edges indicate objects better than other methods.

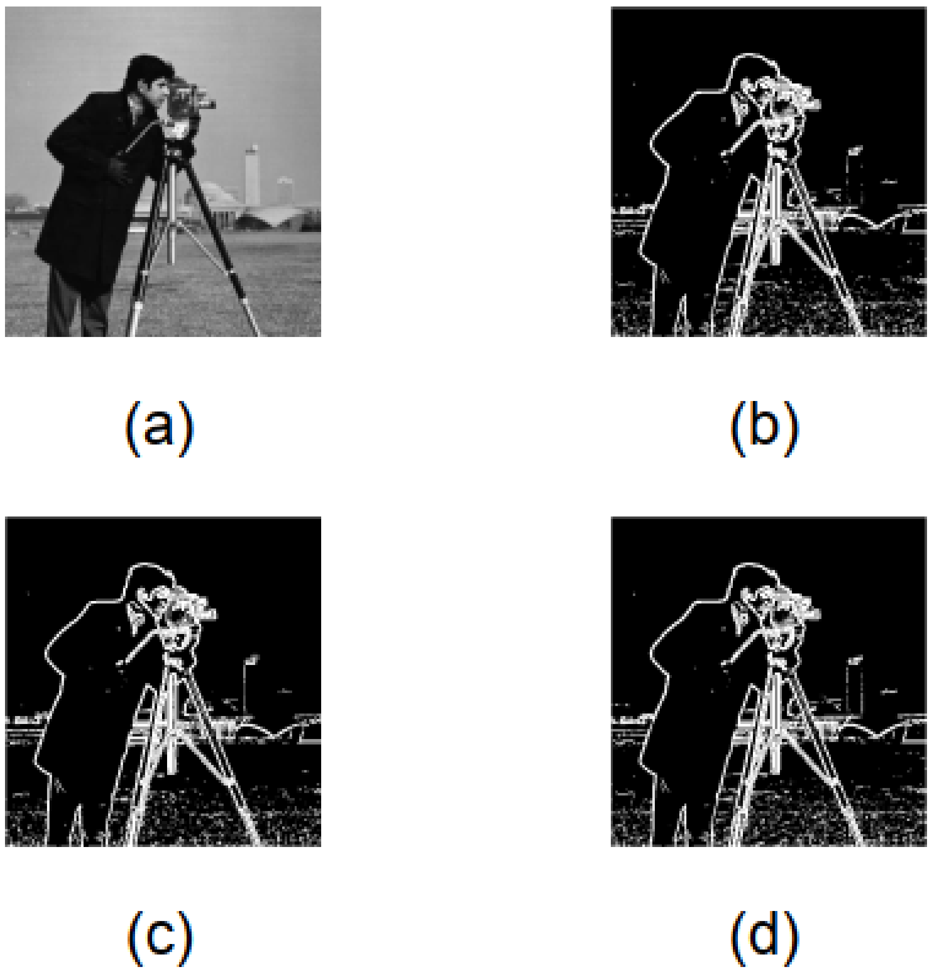

Figure 2: Original and edge detected images, (a) original, (b) with Caputo sense, (c) with Caputo-Fabrizio sense, (d) Proposed method for $\alpha = 0.98,\ k = 1.01,\ 73.54$

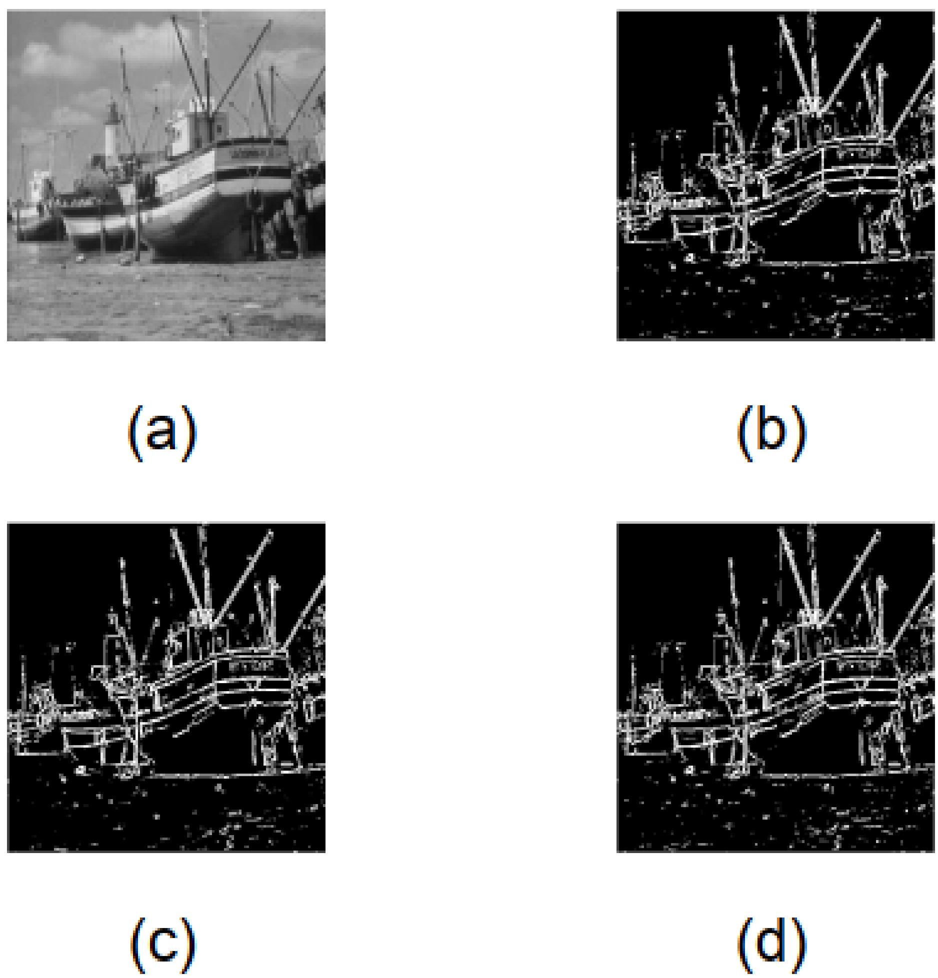

Figure 3: Original and edge detected images, (a) original, (b) with Caputo sense, (c) with Caputo-Fabrizio sense, (d) Proposed method for $\alpha = 0.98,\ k = 1.01,\ 89.30$

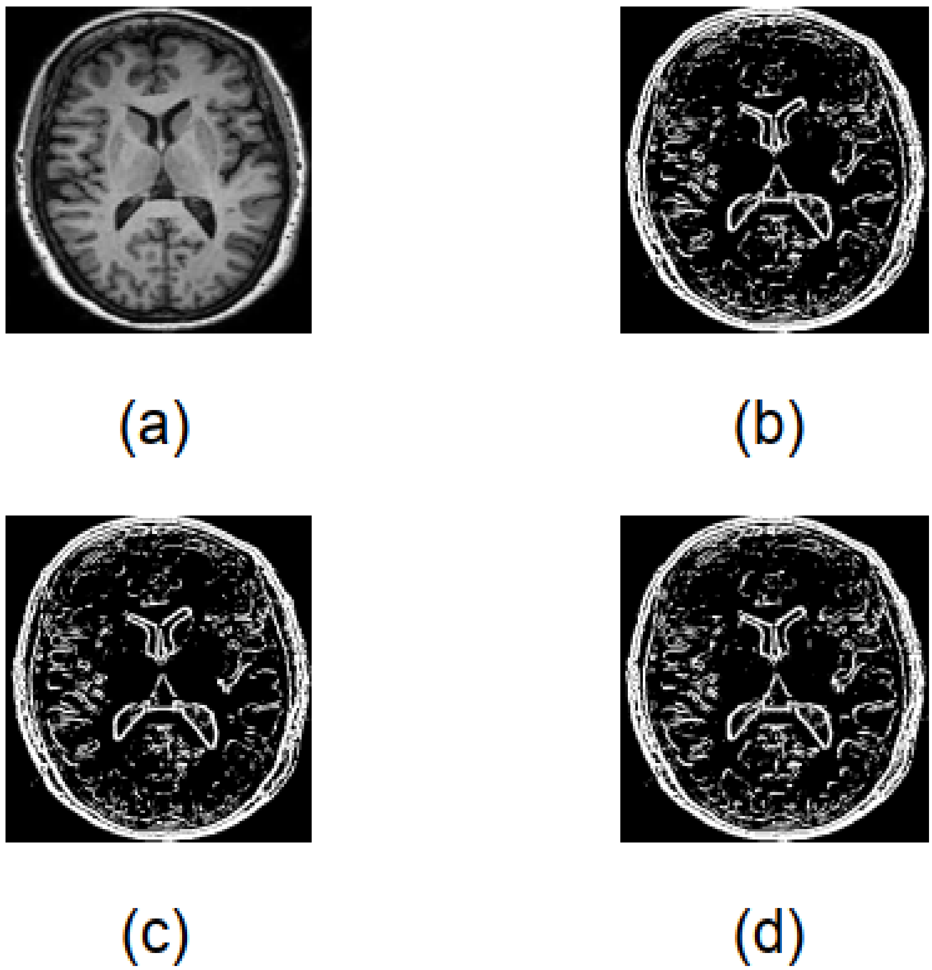

Figure 4: Original and edge detected images, (a) original, (b) with Caputo sense, (c) with Caputo-Fabrizio sense, (d) Proposed method for $\alpha = 0.98,\ k = 1.01,\ 89.30$

6. Conclusion

In this paper, a new edge detection method based on Caputo k-fractional derivative has been developed. Its characteristic of using two parameters makes it more adaptive. To examine the accuracy of this method, we com-pared it with two recent methods applied to three images. The experimental results showed that our proposed method provides better edges map. In perspective, we will apply the fractional derivative in the Laplacian methods to detect edges.

Acknowledgments

The authors are highly grateful to the anonymous referee for his/her valuable comments and suggestions for the improvement of the paper.

References

1. H. Spontón, J. Cardelino, *A Review of Classic Edge Detectors*, Image Processing On Line 5, 90–123, (2015).
2. J. Kaur, S. Agrawal, R. Vig, *A Comparative Analysis of Thresholding and Edge Detection Segmentation Techniques*, International journal of computer applications 39(15), 29–34, (2012).
3. S. Bhardwaj, A. Mittal, *A Survey on Various Edge Detector Techniques*, Procedia Technology 4, 220–226, (2012).
4. I.E. Igbinosa, *Comparison of edge detection technique in image processing techniques*, International journal of Information technology and Electrical Engineering 2(1), 25–29, (2013).
5. S. Qureshi, N.A. Rangaig, D. Baleanu, *New Numerical Aspects of Caputo-Fabrizio Fractional Derivative Operator*,Mathematics 7(4), 374, (2019).
6. G.A.M.B. Nchama, L.D.L. Alfonso, A.P. Cosme, Augusto Pedroso, *Natural images edge detection using Prewitt fractional differential algorithm via Caputo and Caputo-Fabrizio definitions*, Global Journal of Pure and Applied Mathematics 16(6), 789–809, (2020).
7. G. Farid, N. Latif, M. Anwar, A. Imran, M. Ozair, M. Nawaz, *On applications of Caputo k-fractional derivatives*, Advances in Difference Equations 2019(1), 439, (2019).
8. P.P. Acharjya, R. Das, D. Ghoshal, *A Study on Image Edge Detection Using the Gradients*, International Journal of Scientific and Research Publications 2(12), 1–5, (2012).
9. M. Abo-Zahhad, R. Ragab Gharieb, S.M. Ahmed, A.A. Donkol, *Edge Detection with a Preprocessing Approach*, Journal of Signal and Infor-mation Processing 5, 123–134, (2014).
10. M. Juneja, P. Singh Sandhu, *Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain*, International Journal of Computer Theory and Engineering 1(5), 614, (2005).

Ahfouda Belhadi,
Department of Mathematics,
Laboratory of Operators Theory and Partial Differential Equations, Basics and Applications,
University of Echahid Hamma Lakhdar El oued,
Algeria.
E-mail address: belhadi67@gmail.com

and

Bachir Dehda,
Department of Mathematics,
Laboratory of Operators Theory and Partial Differential Equations, Basics and Applications,
University of Echahid Hamma Lakhdar El oued,
Algeria.
E-mail address: dehda-bachir@univ-eloued.dz

and

Abdelaziz Azeb Ahmed,
Department of Mathematics,
Laboratory of Operators Theory and Partial Differential Equations, Basics and Applications,

University of Echahid Hamma Lakhdar El oued,
Algeria.
E-mail address: aziz-azebahmed@univ-eloued.dz

and

Fares Yazid,
Department of Mathematics,
Laboratory of Pure and Applied Mathematics, Amar Telidji University, Laghouat 03000
Algeria.
E-mail address: f.yazid@lagh-univ.dz

and

Fatima Siham Djeradi,
Department of Mathematics,
Laboratory of Pure and Applied Mathematics, Amar Telidji University, Laghouat 03000
Algeria.
E-mail address: fs.djeradi@lagh-univ.dz