

Article

Fractional-Order Colour Image Processing

Manuel Henriques ¹, Duarte Valério ^{1,*}, Paulo Gordo ² and Rui Melicio ^{1,3}

¹ IDMEC, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal; manuel.henriques@tecnico.ulisboa.pt (M.H.); ruimelicio@gmail.com (R.M.)

² CENTRA, Faculdade de Ciências, Universidade de Lisboa, Campo Grande, 1749-016 Lisboa, Portugal; prgordo@fc.ul.pt

³ ICT, Universidade de Évora, Rua Romão Ramalho, 59, 7000-671 Évora, Portugal

* Correspondence: duarte.valerio@tecnico.ulisboa.pt; Tel.: +351-218417351

Abstract: Many image processing algorithms make use of derivatives. In such cases, fractional derivatives allow an extra degree of freedom, which can be used to obtain better results in applications such as edge detection. Published literature concentrates on grey-scale images; in this paper, algorithms of six fractional detectors for colour images are implemented, and their performance is illustrated. The algorithms are: Canny, Sobel, Roberts, Laplacian of Gaussian, CRONE, and fractional derivative.

Keywords: Fractional Derivatives; Image Processing; Colour Images, Satellite Images; Very-High-Resolution Satellite; Satellite Imagery Processing

MSC: 26A33; 68U10; 94A08



Citation: Henriques, M.; Valério, D.; Gordo, P.; Melicio, R. Fractional Order Colour Image Processing. *Mathematics* **2021**, *9*, 457. <https://dx.doi.org/10.3390/math9050457>

Academic Editor: Ivo Petras

Received: 17 January 2021

Accepted: 20 February 2021

Published: 24 February 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Conventional edge detectors are based on integer derivatives. With the introduction of fractional calculus, some new non-integer edge detectors were created [1] and the existent integer ones adapted [2–4].

The aforementioned detectors were developed for grey-scale images. Since the input is often a coloured image, two options are available. One is to convert the image into grey-scale and then apply the detector. This may compromise the performance of edge detection. Thus, the other approach is to create colour-based edge-detection detectors [5,6].

This paper presents, in sequence for six different detection algorithms, three variants: conventional grey-scale integer detectors, grey-scale fractional detectors, and colour-based fractional detectors. The former two can be found in the literature; to the best of our knowledge, colour-based fractional detectors presented in this paper are the first ever used, with the exception of Canny, already given for integer orders, in [5]. While fractional derivatives are linear operators [7], the algorithms for image processing include other steps and operations, resulting in a non-linear treatment of the input data.

This paper is organised in the following manner. Section 2 explains the theoretical formulations for the edge detectors. Section 3 presents an application of the formulated methods to satellite images and the results which serve as an illustration of their performance. In Section 4 discussion of the results is made, and conclusions are drawn.

2. Materials and Methods

2.1. Grünwald-Letnikov Definition

The first-order derivative of a function is given by

$$D^1 f(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}. \quad (1)$$

Iterating, it is possible to arrive to the n th derivative of a function. A general expression can be deduced by induction:

$$D^n f(x) = \lim_{h \rightarrow 0} h^{-n} \sum_{m=0}^n (-1)^m \binom{n}{m} f(x - mh) \tag{2}$$

Combinations of n things taken m at a time are given by

$$\binom{n}{m} = \frac{n!}{m!(n - m)!} \tag{3}$$

The Grünwald–Letnikov (GL) derivative is a generalization of the derivative in (2). The idea behind it is that h should approach 0 as n approaches infinity. However, before doing so, binomial coefficients must cope with real numbers to extend this expression. For that, the Euler gamma function is used, instead of the factorials in (3):

$$\binom{a}{b} = \begin{cases} \frac{\Gamma(a + 1)}{\Gamma(b + 1)\Gamma(a - b + 1)}, & \text{if } a, b, a - b \notin \mathbb{Z}^- \\ \frac{(-1)^b \Gamma(b - a)}{\Gamma(b + 1)\Gamma(-a)}, & \text{if } a \in \mathbb{Z}^- \wedge b \in \mathbb{Z}_0^+ \\ 0, & \text{if } [(b \in \mathbb{Z}^- \vee b - a \in \mathbb{N}) \wedge a \notin \mathbb{Z}^-] \vee (a, b \in \mathbb{Z}^- \wedge |a| > |b|) \end{cases} \tag{4}$$

Combining (2) with (4) provides the main justification for the following extension of the integer-order derivative to any real-order α , which was proposed independently by Grünwald [8] and Letnikov [9]:

$${}^{\text{GL}}D^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{k=0}^\infty (-1)^k \binom{\alpha}{k} f(t - kh), \quad t \in \mathbb{R}. \tag{5}$$

For practical reasons, a truncation of the expression above was introduced, corresponding to an initial value c . This version of the derivative starting in c can be applied to functions that are not defined in the interval from $-\infty$ to c :

$${}_c D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{k=0}^N (-1)^k \binom{\alpha}{k} f(t - kh), \quad N = \left\lfloor \frac{t - c}{h} \right\rfloor, \quad t > c. \tag{6}$$

Expression (6) is the formulation that was applied to the different integer edge detectors in order to adapt them to fractional orders. The corresponding code is available from a public repository [10].

2.2. Derivative Filters

Derivative filters measure the rate of change in pixel value of a digital image. When filters of this kind are used, the result allows for enhancement of contrast, detection of boundaries and edges, and measurement of feature orientation.

Convolution of the specimen image with derivative filters is known as a derivative filtering operation. In most of the times, there is a filter to each direction; thus, convolution is performed twice. Using two dimensions, a gradient can be measured from the combination of the convolutions in x and y :

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x'} & \frac{\partial f}{\partial y} \end{bmatrix}. \tag{7}$$

The gradient points in the direction of the largest intensity increase. The magnitude (8) and orientation (9) of this gradient are frequently used to combine the two convolutions in the processing of images with edges with different orientations:

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \tag{8}$$

$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right). \tag{9}$$

2.3. Canny Edge Detector

Original grey-scale: The Canny edge detector is a very popular edge detection algorithm. It was developed in 1986 by John F. Canny [11]. The algorithm is composed of the following steps: noise reduction, gradient calculation, non-maximum suppression, and hysteresis thresholding.

The first step of the Canny algorithm is noise reduction. Since image processing is always vulnerable to noise, it is important to remove or reduce it before processing. This is done by the convolution of the image $f(x, y)$ with a Gaussian filter, defined as

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \tag{10}$$

Then, a simple 2D, first-derivative operator (which, in the case of the algorithm used in this work, is the derivative of the Gaussian function used to smooth the image) is applied to the image already smoothed $G(x, y, \sigma) * f(x, y)$. This highlights the zones of the image where first spatial derivatives are significant:

$$\nabla g(x, y) = \nabla(G(x, y, \sigma) * f(x, y)) = (\nabla G(x, y, \sigma)) * f(x, y). \tag{11}$$

Finally, the gradients in each direction E_x and E_y are given by

$$E_x = \frac{\partial G(x, y, \sigma)}{\partial x} * f(x, y) \quad E_y = \frac{\partial G(x, y, \sigma)}{\partial y} * f(x, y). \tag{12}$$

This step of the process is called gradient calculation.

After computing gradient magnitude and orientation with (8) and (9), a full scan is performed in order to remove any unwanted pixels which may not constitute edges (non-maximum suppression).

The final step of the algorithm is hysteresis thresholding. In this phase, the algorithm decides which edges are suitable for the output image. Each edge has an intensity proportional to the magnitude of the gradient. For this, two threshold values are defined, the minimum and maximum values. All gradients higher than the maximum threshold are considered "sure-edges". In contrast, the gradients that are lower than the minimum threshold are considered "non-edges". For the gradients that are between the two values, two instances may occur [12]:

- If the pixels in question are connected to "sure-edge" pixels, they are considered to be part of the edges;
- Otherwise, these pixels are also discarded and considered non-edges.

Fractional grey-scale: When adapting the grey-scale fractional Canny, the only change was that, instead of calculating the first-order gradient of the Gaussian kernel, the GL derivative was applied to the Gaussian function with the desired order α . This means that for each point of the Gaussian mask, its fractional derivative is obtained:

$${}_c D_t^\alpha f_G(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{k=0}^N (-1)^k \binom{\alpha}{k} f_G(t - kh), \quad N = \left\lfloor \frac{t - c}{h} \right\rfloor, \quad t > c, \tag{13}$$

where $\begin{pmatrix} \alpha \\ k \end{pmatrix}$ is given by (4).

After computing the fractional derivative, the algorithm follows the same steps of the conventional Canny, including non-maximum suppression and hysteresis thresholding.

Colour: In 1987, Kanade introduced an extension of the Canny operator [5] for colour edge detection. The operator is based on the same steps as the conventional Canny, but the computations are now vector-based. This means that the algorithm determines the first partial derivatives of the smoothed image in both x and y directions.

A three-component colour image assumes, for each of its points in the plane, a value which is a vector in the colour space. In the RGB space, which is a three-dimensional (3D) space to represent colour by a mixture of red (R), green (G), and blue (B), this corresponds to a function $C = (R, G, B)$. It is possible now to define the Jacobian matrix, which is the matrix that contains the first partial derivatives for each component of the colour vector:

$$J = \begin{pmatrix} R_x & R_y \\ G_x & G_y \\ B_x & B_y \end{pmatrix} = (C_x, C_y). \tag{14}$$

Indexes x and y are used to represent partial derivatives:

$$R_x = \frac{\partial R}{\partial x} \quad \text{and} \quad R_y = \frac{\partial R}{\partial y}. \tag{15}$$

The direction along which the largest variation in the colour image can be found is the direction of the eigenvector of $J^T J$ that corresponds to the largest eigenvalue:

$$J^T J = \begin{pmatrix} J_x & J_{xy} \\ J_{yx} & J_y \end{pmatrix} \tag{16}$$

$$\begin{aligned} J_x &= R_x^2 + G_x^2 + B_x^2 \\ J_y &= R_y^2 + G_y^2 + B_y^2 \end{aligned} \tag{17}$$

$$J_{xy} = J_{yx} = R_x R_y + G_x G_y + B_x B_y.$$

In order to calculate the magnitude, one has to compute $\det(J^T J - \lambda I) = 0$, which yields

$$\lambda = \frac{J_y + J_x \pm \sqrt{(J_y + J_x)^2 - 4(J_x J_y - J_{xy}^2)}}{2}. \tag{18}$$

The orientation θ of a colour edge is determined in an image by

$$\tan(2\theta) = \frac{2 \cdot C_x \cdot C_y}{\|C_x\|^2 - \|C_y\|^2}. \tag{19}$$

After the magnitude is determined for each edge, non-maximum suppression is used. This eliminates broad edges, thanks to a threshold value.

According to the literature [13], even though colour edges and intensity edges are identical in over 90% of the cases, the former describes object geometry in the scene better than the latter.

2.4. Sobel Edge Detector

Original grey-scale: The Sobel operator measures the spatial gradient of an image. In this way, regions where there are sudden increases of pixel intensity are highlighted. Such regions correspond to edges.

The operator consists of two masks, one for each direction of x and y (\mathbf{G}_x and \mathbf{G}_y , respectively). Note that the mask, or kernel, for \mathbf{G}_y is nothing more than that for \mathbf{G}_x rotated 90 degrees [14]:

$$\begin{array}{|c|c|c|} \hline -1 & 0 & +1 \\ \hline -2 & 0 & +2 \\ \hline -1 & 0 & +1 \\ \hline \end{array}
 \quad
 \begin{array}{|c|c|c|} \hline +1 & +2 & +1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}
 \tag{20}$$

Those edges that are vertical and horizontal in relation to the pixel grid cause a maximal response of these kernels. They can be applied to the input image separately. The resulting measurements of the gradient component in each direction (\mathbf{G}_x and \mathbf{G}_y) are thereafter combined, so as to find at each point both the magnitude of the gradient and its orientation, using (8) and (9), respectively.

The resulting image gradient components can be expressed as

$$\mathbf{G}_x = -f(x-1, y-1) - 2f(x-1, y) - f(x-1, y+1) + f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1) \tag{21}$$

$$\mathbf{G}_y = -f(x-1, y-1) - 2f(x, y-1) - f(x+1, y-1) + f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1). \tag{22}$$

Fractional grey-scale: Following the same line of thought, Yaacoub [2] presented a fractional Sobel operator.

Applying the GL definition (6) to \mathbf{G}_x (21), the fractional α -order derivative of G_x yields

$$\begin{aligned}
 D^\alpha \mathbf{G}_x = & \frac{\alpha(-\alpha+1)(-\alpha+2)}{12} \cdot [f(x-4, y-1) + 2f(x-4, y) + f(x-4, y+1)] \\
 & + \frac{\alpha(-\alpha+1)}{4} \cdot [f(x-3, y-1) + 2f(x-3, y) + f(x-3, y+1)] \\
 & + \left[\frac{\alpha}{2} - \frac{\alpha(-\alpha+1)(-\alpha+2)}{12} \right] \cdot [f(x-2, y-1) + 2f(x-2, y) + f(x-2, y+1)] \\
 & + \left[-\frac{1}{2} - \frac{\alpha(-\alpha+1)}{4} \right] \cdot [f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1)] \\
 & - \frac{\alpha}{2} [f(x, y-1) + 2f(x, y) + f(x, y+1)] \\
 & + \frac{1}{2} [f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)]. \tag{23}
 \end{aligned}$$

This gradient is obtained convolving the image $f(x, y)$ with a filter mask:

$\frac{\alpha(-\alpha+1)(-\alpha+2)}{12}$	$\frac{\alpha(-\alpha+1)(-\alpha+2)}{6}$	$\frac{\alpha(-\alpha+1)(-\alpha+2)}{12}$
$\frac{\alpha(-\alpha+1)}{4}$	$\frac{\alpha(-\alpha+1)}{2}$	$\frac{\alpha(-\alpha+1)}{4}$
$\frac{\alpha}{2} - \frac{\alpha(-\alpha+1)(-\alpha+2)}{12}$	$\alpha - \frac{\alpha(-\alpha+1)(-\alpha+2)}{6}$	$\frac{\alpha}{2} - \frac{\alpha(-\alpha+1)(-\alpha+2)}{12}$
$-\frac{1}{2} - \frac{\alpha(-\alpha+1)}{4}$	$-1 - \frac{\alpha(-\alpha+1)}{2}$	$-\frac{1}{2} - \frac{\alpha(-\alpha+1)}{4}$
$-\frac{\alpha}{2}$	$-\alpha$	$\frac{\alpha}{2}$
$\frac{1}{2}$	1	$\frac{1}{2}$

(24)

Since the mask has an even number of rows, the origin is not centered. In the mask above, the origin is considered to be located on the fifth row, in the second column, shown in bold.

Similar reasoning can be applied to the y -direction. That is why, in this case, the mask in y is not the mask in x transposed.

According to the authors, this edge detector, compared with the conventional Sobel edge detector, resulted in thinner edges and reduced the number pixels of false edges.

Colour: A novel, colour-based, fractional Sobel was introduced by applying the same colour-based formulation of Section 2.3, only this time with the mask in (24). A Jacobian

was constructed, and the largest eigenvalue of $J^T J$ computed; this allows for discovery of the direction in the image, along which the largest variation in the chromatic image function occurs.

2.5. Roberts Edge Detector

Original grey-scale: The Roberts Cross operator [15] is a simpler, quick way to find the gradient of an image. It also finds zones with great variations in pixel intensity that correspond to edges.

The Roberts operator consists of a pair of 2×2 masks, again, one for each direction. Here, the mask to compute the gradient in one direction is the other mask rotated by 90° [14]:

$$\begin{matrix} +1 & 0 \\ 0 & -1 \end{matrix} \quad \begin{matrix} 0 & +1 \\ -1 & 0 \end{matrix} \tag{25}$$

The combination of the two gradients in order to find the magnitude and orientation is performed once more using the above-mentioned expressions.

Fractional grey-scale: The authors of [3] presented the application of the GL derivative to the integer Roberts edge detector and arrived to a kernel for a fractional-order operator. It is known that the Roberts expression for the gradients stands:

$$g(x, y) = \{ [f(x, y + 1) - f(x + 1, y)]^2 + [f(x + 1, y + 1) - f(x, y)]^2 \}^{\frac{1}{2}}. \tag{26}$$

Combining (6) with (4), the authors arrived at expressions for the gradient's components:

$$\begin{aligned} \frac{\partial^\alpha f(x, y)}{\partial x^\alpha} &\approx f(x, y) + (-\alpha)f(x - 1, y) + \frac{(-\alpha)(-\alpha + 1)}{2}f(x - 2, y) \\ &+ \dots + \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)}f(x - n, y) \end{aligned} \tag{27}$$

$$\begin{aligned} \frac{\partial^\alpha f(x, y)}{\partial y^\alpha} &\approx f(x, y) + (-\alpha)f(x, y - 1) + \frac{(-\alpha)(-\alpha + 1)}{2}f(x, y - 2) \\ &+ \dots + \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)}f(x, y - n). \end{aligned} \tag{28}$$

Referring to (27) and (28), the 3×3 fractional differential mask can be constructed in the eight central symmetric directions, viz. positive and negative x and y coordinates, and left and right downward and upward diagonals. The sum of the eight directional masks yields

$$\begin{matrix} \frac{\alpha^2 - \alpha + 2}{2} & \frac{\alpha^2 - \alpha + 2}{2} & \frac{\alpha^2 - \alpha + 2}{2} \\ \frac{\alpha^2 - \alpha + 2}{2} & -8\alpha & \frac{\alpha^2 - \alpha + 2}{2} \\ \frac{\alpha^2 - \alpha + 2}{2} & \frac{\alpha^2 - \alpha + 2}{2} & \frac{\alpha^2 - \alpha + 2}{2} \end{matrix} \tag{29}$$

Combining the fractional mask with the Roberts operator defined by (26), the authors arrived at a solution for edge detection in which the texture of the image is enhanced and small edges are also detected. The mathematical formulation for this combination is

$$D^\alpha [g(x, y)] = \frac{\partial^\alpha g(x, y)}{\partial x^\alpha} + \frac{\partial^\alpha g(x, y)}{\partial y^\alpha} \tag{30}$$

$$\begin{aligned} \frac{\partial^2 g(x, y)}{\partial x^2} &\approx g(x, y) + (-\alpha)g(x - 1, y) + \frac{(-\alpha)(-\alpha + 1)}{2}g(x - 2, y) \\ &\quad + \dots + \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)}g(x - n, y) = \\ &= \left\{ [f(x, y + 1) - f(x + 1, y)]^2 + [f(x + 1, y + 1) - f(x, y)]^2 \right\}^{\frac{1}{2}} + \\ &\quad (-\alpha) \left\{ [f(x - 1, y + 1) - f(x, y)]^2 + [f(x, y + 1) - f(x - 1, y)]^2 \right\}^{\frac{1}{2}} + \\ &\quad \frac{(-\alpha)(-\alpha + 1)}{2} \left\{ [f(x - 2, y + 1) - f(x - 1, y)]^2 + \right. \\ &\quad \left. + [f(x - 1, y + 1) - f(x - 2, y)]^2 \right\}^{\frac{1}{2}} + \dots + \\ &\quad \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)} \left\{ [f(x - n, y + 1) - f(x - n + 1, y)]^2 + \right. \\ &\quad \left. + [f(x - n + 1, y + 1) - f(x - n, y)]^2 \right\} \end{aligned} \tag{31}$$

$$\begin{aligned} \frac{\partial^\alpha g(x, y)}{\partial y^\alpha} &\approx g(x, y) + (-\alpha)g(x, y - 1) + \frac{(-\alpha)(-\alpha + 1)}{2}g(x, y - 2) \\ &\quad + \dots + \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)}g(x, y - n) \\ &= \left\{ [f(x, y + 1) - f(x + 1, y)]^2 + [f(x + 1, y + 1) - f(x, y)]^2 \right\}^{\frac{1}{2}} + \\ &\quad (-\alpha) \left\{ [f(x, y) - f(x + 1, y - 1)]^2 + [f(x + 1, y) - f(x, y - 1)]^2 \right\}^{\frac{1}{2}} + \\ &\quad \frac{(-\alpha)(-\alpha + 1)}{2} \left\{ [f(x, y - 1) - f(x + 1, y - 2)]^2 + \right. \\ &\quad \left. + [f(x + 1, y - 1) - f(x, y - 2)]^2 \right\}^{\frac{1}{2}} + \dots + \\ &\quad \frac{(-1)^{n-1}\Gamma(\alpha + 1)}{(n - 1)!\Gamma(\alpha - n + 2)} \left\{ [f(x, y - n + 1) - f(x + 1, y - n)]^2 + \right. \\ &\quad \left. + [f(x + 1, y - n + 1) - f(x, y - n)]^2 \right\}^{\frac{1}{2}}, \end{aligned} \tag{32}$$

where $f(x, y)$ is the input image, and $g(x, y)$ is the output image using an integer Roberts operator.

From the experimental results in [3], and comparing this fractional algorithm with the original Roberts algorithm, it was concluded that edge detection was enhanced, while the thinner edges of the original algorithm are preserved.

Colour: The reasoning used to implement the colour-based Sobel operator was also used here. The fractional Roberts requires convolutions with two masks—first with the integer masks, and then with the fractional-derivatives operator. The colour-based vector convolution and Jacobian computations were applied only to the first one with the integer Roberts. Then, the output of this first integer colour-based edge detection serves as input to the fractional derivative operation.

2.6. Laplacian of Gaussian Detector

Original grey-scale: The Laplacian is a measure of the second spatial derivative of an image. It allows for the identification of zones where intensity changes fast. It is thus often used for edge detection. The Laplacian of a 2D image is given by

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}. \tag{33}$$

In the discrete domain, the simplest approximation of the continuous Laplacian is the numerical first-derivative of the numerical first-derivative, yielding

$$\frac{\partial^2 f}{\partial x^2} = f(i, j + 1) - 2f(i, j) + f(i, j - 1) \tag{34}$$

$$\frac{\partial^2 f}{\partial y^2} = f(i + 1, j) - 2f(i, j) + f(i - 1, j). \tag{35}$$

Substituting (34) and (35) in (33), the first kernel of (36) is obtained. The second, a non-separable eight-neighbor Laplacian defined by the gain-normalized impulse response array, was suggested by Prewitt. The third mask in (36) is a separable eight-neighbor version of the Laplacian [16].

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

(36)

To tackle sensitivity of second-order derivatives to noise, the image is smoothed with a Gaussian filter, and only then is the Laplacian filter reducing high-frequency noise applied. The smoothing filter can also be convolved first with the Laplacian kernel, and only then is the result convolved with the input image.

The Laplacian of Gaussian (LoG) operator of a 2D image is illustrated in Figure 1 and defined by [14]

$$: \text{LoG}(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}}. \tag{37}$$

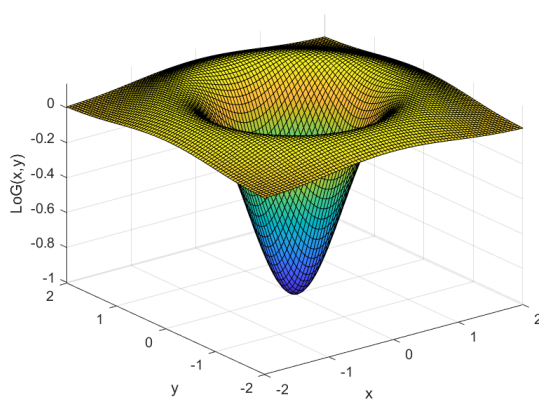


Figure 1. 2D Laplacian of Gaussian operator (37) with $\sigma = \frac{3}{4}$.

Fractional grey-scale: In 2014, the authors of [4] presented a fractional adaptation for the first operator in (36) (using the symmetric mask).

In a discrete function (f), the operator corresponds to approximation

$$G(f) = -f(x - 1, y) - f(x, y - 1) + 4f(x, y) - f(x, y + 1) - f(x + 1, y). \tag{38}$$

Decomposing and noting that for this case, $h = 1$,

$$G(f) = \frac{\partial f(x, y)}{\partial x} + \frac{\partial f(x, y)}{\partial y} - \frac{\partial f(x, y + 1)}{\partial y} - \frac{\partial f(x + 1, y)}{\partial x} \tag{39}$$

$$= -\frac{\partial^2 f(x + 1, y)}{\partial x^2} - \frac{\partial^2 f(x, y + 1)}{\partial y^2}.$$

By generalizing the order from integer to fractional, a fractional-order differential form of the Laplacian operator can be obtained:

$$G^\alpha(f) = -\frac{\partial^\alpha f(x + 1, y)}{\partial x^\alpha} - \frac{\partial^\alpha f(x, y + 1)}{\partial y^\alpha}. \tag{40}$$

Using the GL definition for the fractional-order derivative as it was used for the other operators, one may arrive at

$$G^\alpha(f) = -\sum_{k=0}^{K-1} (-1)^k C_k^\alpha f(x + 1 - k, y) - \sum_{k=0}^{K-1} (-1)^k C_k^\alpha f(x, y + 1 - k) \tag{41}$$

$$= -\left[f(x + 1, y) - \alpha f(x, y) + \frac{\alpha^2 - \alpha}{2} f(x - 1, y) + \dots + (-1)^{K-1} C_{K-1}^\alpha f(x + 2 - K, y) \right]$$

$$- \left[f(x, y + 1) - \alpha f(x, y) + \frac{\alpha^2 - \alpha}{2} f(x, y - 1) + \dots + (-1)^{K-1} C_{K-1}^\alpha f(x, y + 2 - K) \right].$$

With the definition above, the mask that performs the calculation of the fractional Laplacian may be built:

0	...	0	$(-1)^K C_{K-1}^\alpha$	0
⋮	⋮	⋮	⋮	⋮
0	...	0	$(\alpha - \alpha^2)/2$	0
$(-1)^K C_{K-1}^\alpha$...	$(\alpha - \alpha^2)/2$	2α	-1
0	...	0	-1	0

(42)

Experiments with (42) show that, the larger the order of differentiation is, the better the image feature is preserved, but the more noise that appears too.

Colour: A novel, colour-based, fractional LoG operator was implemented and tested in this study. The conventional algorithm finds edges searching for zero-crossings. This means that previous formulations cannot be adapted. According to [17], a pixel of a colour image is considered as part of an edge if zero-crossings are found in any of the colour channels. Thus, the fractional grey-scale operator formulated in [4] can be applied to each colour channel of a colour image. Then, a search for zero-crossings in the convolution outputs may be performed. If a zero-crossing is found in any of the channels, the corresponding pixel is flagged as part of an edge. The output of this algorithm is a binary image with all edges found.

2.7. CRONE

Original fractional grey-scale: In 2002, Benoît Mathieu wanted to prove that an edge detector based on fractional differentiation could improve edge detection and detection selectivity in the case of parabolic luminance transition.

The first derivative of a function $f(x)$, calculated with increasing x , can be defined by

$$\underline{D}_\rightarrow^n f(x) = \frac{f(x) - f(x - h)}{h} \tag{43}$$

with decreasing x ,

$$\underline{D}^n f(x) = \frac{f(x) - f(x+h)}{h} \tag{44}$$

with h being infinitesimally small.

A shift operator q is consequently introduced, defined by

$$qf(x) = f(x+h) \tag{45}$$

$$q^{-1}f(x) = f(x-h). \tag{46}$$

Using the shift operator on the directional derivative yields

$$\underline{D}f(x) = \frac{1-q^{-1}}{h}f(x) \tag{47}$$

$$\underline{D}f(x) = \frac{1-q}{h}f(x). \tag{48}$$

From the expressions above, it is clear that

$$\underline{D} = \frac{1-q^{-1}}{h} \tag{49}$$

$$\underline{D} = \frac{1-q}{h}. \tag{50}$$

Generalizing to an order n , \underline{D}^n and \underline{D}^n can be defined as

$$\underline{D}^n = \left(\frac{1-q^{-1}}{h}\right)^n \tag{51}$$

$$\underline{D}^n = \left(\frac{1-q}{h}\right)^n. \tag{52}$$

As explained before, the bidirectional detector can be constructed by a composition of the two unidirectional operators using the following expression:

$$\underline{D}^n = \underline{D}^n - \underline{D}^n = \frac{1}{h^n} \left[(1-q^{-1})^n - (1-q)^n \right]. \tag{53}$$

Expanding $(1-q^{-1})^n$ and $(1-q)^n$ using Newton's binomial formula, the expression above can be rewritten:

$$\underline{D}^n = \frac{1}{h^n} \sum_{k=0}^{\infty} (-1)^k \frac{n(n-1)\cdots(n-k+1)}{k!} \times (q^{-k} - q^k). \tag{54}$$

Applying the operator to a function, such as the transition studied before,

$$\underline{D}^n f(x) = \frac{1}{h^n} \sum_{k=0}^{\infty} a_k [f(x-kh) - f(x+kh)] \tag{55}$$

where

$$\begin{aligned} a_k &= (-1)^k \binom{n}{k} \\ &= (-1)^k \frac{n(n-1)\cdots(n-k+1)}{k!}. \end{aligned} \tag{56}$$

In order to detect edges on images, the formulated detector must be designed in two dimensions. Two independent vector operators for x and y , each of them a truncated CRONE detector, given respectively by

$$[+a_m \quad \cdots \quad +a_1 \quad 0 \quad -a_1 \quad \cdots \quad a_m] \tag{57}$$

$$\begin{bmatrix} +a_m \\ \vdots \\ +a_1 \\ 0 \\ -a_1 \\ \vdots \\ -a_m \end{bmatrix} \tag{58}$$

are used for this purpose. The detector was experimented in artificial and real images, and performance compared with Prewitt operators. In all cases, the CRONE detector showed better immunity to noise.

Colour: In this paper, a novel, colour-based, fractional CRONE was implemented, following the same steps as in the colour Canny already formulated. The colour channels are convolved with the masks for each direction constructing a Jacobian. Then, the maximum eigenvalue of $J^T J$ is computed in order to find the pixels where the variation in chromatic image is higher than a designated threshold.

2.8. Fractional Derivative Operator

Original fractional grey-scale: In the fractional Roberts, the fractional mask is used in addition to the conventional integer mask. In this study, the Fractional Derivatives mask in (29) was implemented individually as a fractional edge detector.

Colour: A new colour-based version for this detector was also implemented. In this case, there is only one mask to detect edges in eight different directions. Thus, the Jacobian is reduced to a vector (one dimension):

$$J = \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{59}$$

The magnitude of the gradients was computed with the 3D vector formula:

$$\|\nabla f\| = \sqrt{R^2 + G^2 + B^2}. \tag{60}$$

3. Results

3.1. Implementation

All edge detectors were implemented in MATLAB to perform fractional detection. A low-nebulosity image from ESA’s Sentinel-2 satellite [18] is used below to illustrate performance. The image retrieved from the website was analysed, and a ground truth was manually taken using GIMP. Both the selected photograph and its corresponding ground truth are given in Figure 2. Additional results for a greater number of images from the

same source are available in Excel files from our public repository [10]. Performance was checked with the usual metrics

$$J(A, B) = \frac{TP}{FP + TP + FN} \quad (61)$$

$$DSC = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (62)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (63)$$

$$\text{Specificity} = \frac{TN}{FP + TN} \quad (64)$$

of which the first two are, respectively, the Jaccard coefficient and the Dice similarity coefficient. In (61)–(64), true-positives, false-positives, true-negatives, and false-negatives are defined as usual (see Table 1).

Table 1. Performance instances.

		Processed Image	
		0	1
Ground Truth	0	TN	FP
	1	FN	TP

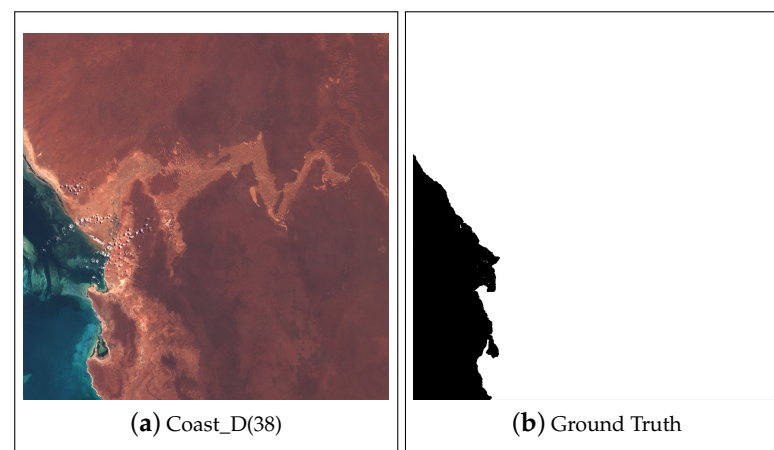


Figure 2. Photograph used to illustrate performance and its ground truth.

All the algorithms in Chapter 2 were tested with parameters varying within wide ranges ($\alpha = -3 : 0.1 : 3$, threshold = $0.1 : 0.2 : 0.9$, $\sigma = 0.2 : 0.2 : 2$, $k = 1 : 1 : 5$). The best performances of the different colour detectors are presented as an example in Figure 3; for additional examples, see the repository [10].

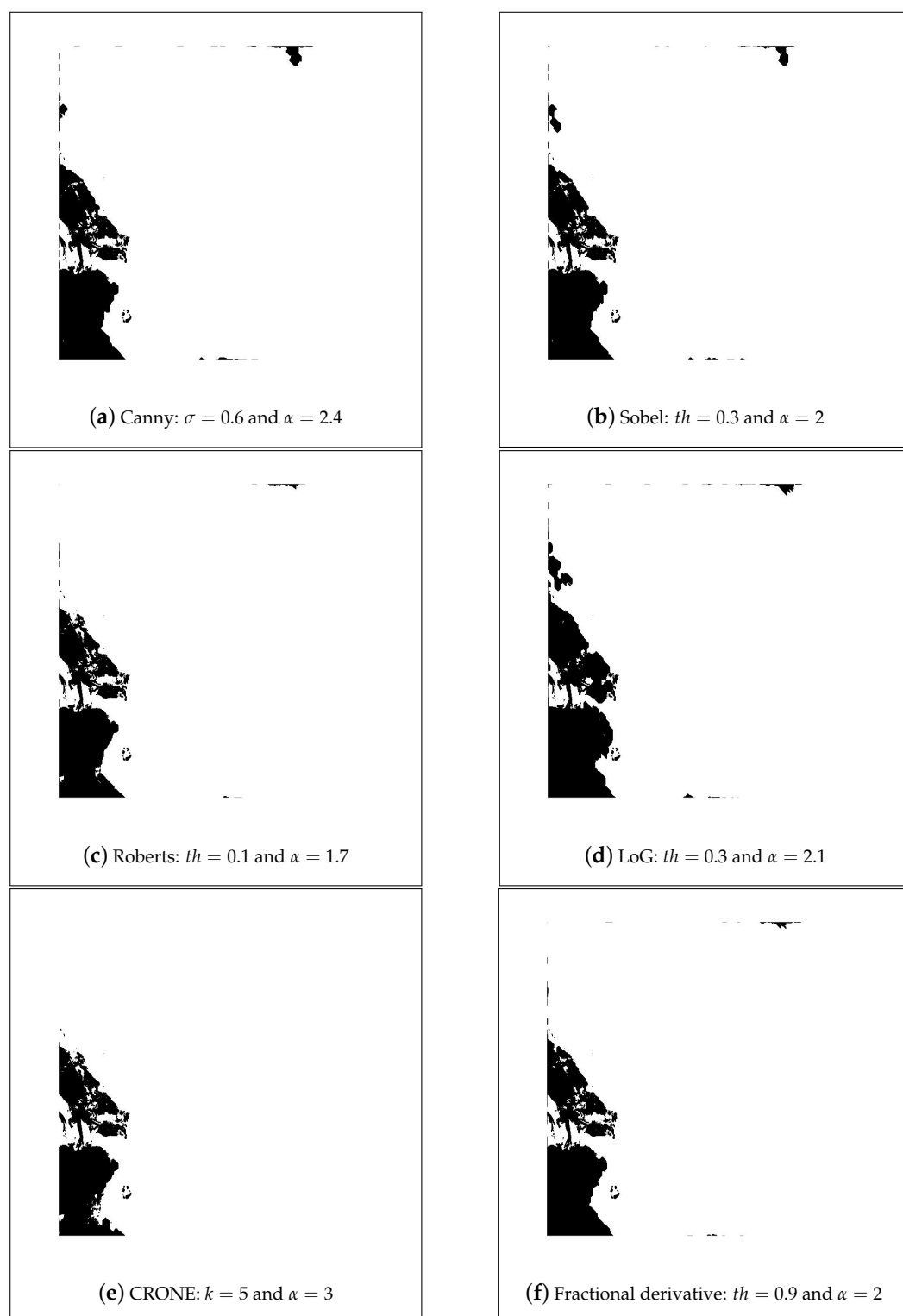


Figure 3. Best results for Figure 2 processed using the different algorithms for colour images; th is the threshold.

4. Discussion

Fractional-order adaptations of grey-scale edge detection algorithms are known to usually have better performance than the original method [19]. This happens in the example shown, where the results for Canny serve as an illustration: the integer algorithm achieves a Jaccard coefficient of 51.80%, the fractional method increases this result to 73.56%, and the colour version was able to almost entirely close the smooth inner-land gradients, achieving

a Jaccard coefficient of 96.32%. Even when the increase in performance is low in percentage, since we are dealing with images that are composed of more than 120 million pixels, a 1% increase corresponds to more than one million correctly identified pixels. On the other hand, it is true that the colour-based detector is also heavier computationally, since it usually requires more than one convolution (at least one for each colour channel).

To summarize, in this study, seven novel fractional edge-detection methods were introduced (viz. one grey-scale and six colour-based versions). Future work includes a statistical assessment of their relative performance, in different types of images and applications, to statistically measure how their use improves performance.

Author Contributions: M.H.: methodology, software, investigation, data curation, writing—original draft preparation, visualization; D.V.: conceptualization, methodology, validation, investigation, writing—review and editing, supervision; P.G.: validation, investigation, writing—review and editing; R.M.: conceptualization, methodology, validation, investigation, data curation, writing—review and editing, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by FCT, through IDMEC, under LAETA, project UIDB/50022/2020; by FCT under the ICT (Institute of Earth Sciences) project UIDB/04683/2020; and by FCT, through the CENTRA project UIDB/00099/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: See [10].

Acknowledgments: The authors would like to acknowledge ESA, Copernicus for kindly providing access to the database for this research.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

		Appears in
CRONE	Commande Robuste d'Order Non-Entier (Non-integer order robust control, in French)	Section 2.7
DSC	Dice similarity coefficient	Equation (62)
ESA	European Space Agency	Section 3.1
FN	False negative	Table 1
FP	False positive	Table 1
GIMP	GNU Image Manipulation Program	Section 3.1
GL	Grünwald-Letnikoff	Section 2.1
LoG	Laplacian of Gaussian	Section 2.6
TN	True negative	Table 1
TP	True positive	Table 1

References

1. Mathieu, B.; Melchior, P.; Oustaloup, A.; Ceyral, C. Fractional differentiation for edge detection. *Signal Process.* **2003**, *83*, 2421–2432. doi:10.1016/S0165-1684(03)00194-4.
2. Yaacoub, C.; Zeid Daou, R.A. Fractional Order Sobel Edge Detector. In Proceedings of the 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), Istanbul, Turkey, 6–9 November 2019; pp. 1–5. doi:10.1109/IPTA.2019.8936101.
3. Chen, X.; Fei, X. Improving edge-detection algorithm based on fractional differential approach. In Proceedings of the 2012 International Conference on Image, Vision and Computing, Shanghai, China, 25–26 August, 2012; Volume 50. doi:10.7763/IPCST.2012.V50.48.

4. Tian, D.; Wu, J.; Yang, Y. A Fractional-Order Laplacian Operator for Image Edge Detection. *Appl. Mech. Mater.* **2014**, *536-537*, 55–58. doi:10.4028/www.scientific.net/AMM.536-537.55.
5. Kanade, T. Image Understanding Research at Carnegie Mellon. In *Proceedings of a Workshop on Image Understanding Workshop*; Morgan Kaufmann Publishers Inc.: Los Angeles, CA, USA, 1987; pp. 32–48.
6. Cumani, A. Edge detection in multispectral images. *CVGIP Graph. Model. Image Process.* **1991**, *53*, 40–51. doi:10.1016/1049-9652(91)90018-F.
7. Valério, D.; da Costa, J.S. Introduction to single-input, single-output fractional control. *IET Control Theory Appl.* **2011**, *5*, 1033–1057. doi:10.1049/iet-cta.2010.0332.
8. Grünwald, A.K. Über “begrenzte” Derivationen und deren Anwendung. *Z. Für Math. Und Phys.* **1867**, *12*, 441–480.
9. Letnikov, A. Theory of Differentiation with an Arbitrary Index (Russian). *Moscow Matem. Sbornik* **1872**, *6*, 413–445.
10. Henriques, M. Results Repository. Available online: https://drive.google.com/drive/folders/1GMeKvc3oqNWfzd4h-GyRwHT9yFJ_UDLP?usp=sharing (accessed on 15 January 2021).
11. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, 679–698. doi:10.1109/TPAMI.1986.4767851.
12. CVonline. Edges: The Canny Edge Detector. Available online: http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MARBLE/low/edges/canny.htm. (accessed on 31 July 2020).
13. Koschan, A.; Abidi, M. Detection and classification of edges in color images. *IEEE Signal Process. Mag.* **2005**, *22*, 64–73. doi:10.1109/MSP.2005.1407716.
14. Fisher, R.; Perkins, S.; Walker, A.; Wolfart, E. Image Processing Learning Resources. 2004. Available online: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/> (accessed on 15 January 2021).
15. Roberts, L. Machine Perception of 3-D Solids. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1963.
16. Pratt, W.K. Second-Order Derivative Edge Detection. In *Digital Image Processing*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2007. doi:10.1002/0470097434.
17. Zhu, S.Y.; Plataniotis, K.N.; Venetsanopoulos, A.N. Comprehensive analysis of edge detection in color image processing. *Opt. Eng.* **1999**, *38*, 612–625. doi:10.1117/1.602105.
18. ESA. Copernicus Open Access Hub. 2020. Available online: <https://scihub.copernicus.eu/> (accessed on 31 March 2020).
19. Bento, T.; Valério, D.; Teodoro, P.; Martins, J. Fractional order image processing of medical images. *J. Appl. Nonlinear Dyn.* **2017**, *6*, 181–191.