

# On the image inpainting problem from the viewpoint of a nonlocal Cahn-Hilliard type equation

---

Brkić, Antun Lovro; Mitrović, Darko; Novak, Andrej

Source / Izvornik: **Journal of Advanced Research, 2020, 25, 67 - 76**

**Journal article, Published version**

**Rad u časopisu, Objavljena verzija rada (izdavačev PDF)**

<https://doi.org/10.1016/j.jare.2020.04.015>

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:217:559112>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

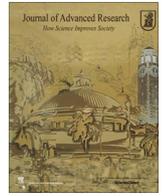
Download date / Datum preuzimanja: **2024-07-14**



Repository / Repozitorij:

[Repository of the Faculty of Science - University of Zagreb](#)





# On the image inpainting problem from the viewpoint of a nonlocal Cahn-Hilliard type equation

Antun Lovro Brkić<sup>a</sup>, Darko Mitrović<sup>b</sup>, Andrej Novak<sup>c,\*</sup>

<sup>a</sup> Institute of Physics, Bijenička cesta 46, 10000 Zagreb, Croatia

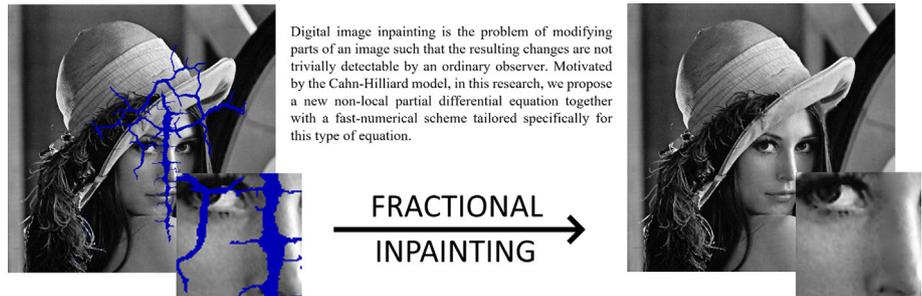
<sup>b</sup> Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

<sup>c</sup> Department of Physics, Faculty of Science, Bijenička cesta 32, University of Zagreb, Croatia

## HIGHLIGHTS

- Investigation of stationary linear fractional differential equations as inpainting tools.
- Physical motivation for introducing PDF inpainting tools and fractional generalizations.
- Development of fast numerical algorithms for the inpainting problem.
- Systematic comparison with the integer order equations.

## GRAPHICAL ABSTRACT



## ARTICLE INFO

### Article history:

Received 2 January 2020

Revised 23 April 2020

Accepted 25 April 2020

Available online 15 May 2020

### Keywords:

Fractional calculus

Image inpainting

Partial differential equations

## ABSTRACT

Motivated by the fact that the fractional Laplacean generates a wider choice of the interpolation curves than the Laplacean or bi-Laplacean, we propose a new non-local partial differential equation inspired by the Cahn-Hilliard model for recovering damaged parts of an image. We also note that our model is linear and that the computational costs are lower than those for the standard Cahn-Hilliard equation, while the inpainting results remain of high quality. We develop a numerical scheme for solving the resulting equations and provide an example of inpainting showing the potential of our method.

© 2020 The Authors. Published by Elsevier B.V. on behalf of Cairo University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## Introduction

Digital image inpainting is the problem of modifying parts of an image such that the resulting changes are not trivially detectable by an ordinary observer. It is used to recover the missing or damaged regions of an image based on the data from the known regions. It represents an ill-posed problem because the missing

or damaged regions can never be recovered correctly with absolute certainty unless the initial image is completely known.

In this paper we are concerned with the following problem. Let  $\Omega \subset \mathbb{R}^2$  be a square image domain and  $\omega \subset \Omega$  an open region with smooth boundary  $S = \partial\omega$  such that  $\text{dist}(\omega, \partial\Omega) > 0$ . Let  $f$  be the original image, known only on  $\Omega \setminus \omega$ , and let  $0 < \mu \leq \nu \leq 1$ . For a constant  $\epsilon \in \mathbb{R}$ , we aim to solve the following problem

$$(-\Delta)^\mu (H(u) - \epsilon^2 (-\Delta)^\nu u) = 0, \quad \text{on } \omega, \quad (1)$$

$$u = f(x) \quad \text{on } \Omega \setminus \omega, \quad (2)$$

where  $u : \Omega \rightarrow \mathbb{R}$  is the interpolation of the original image  $f$ . In the special case when  $\mu = \nu = 1$  and  $H(u) = \frac{a}{2}(1 - u^2)^2$  (so called

Peer review under responsibility of Cairo University.

\* Corresponding author.

E-mail addresses: [albrkic@ifs.hr](mailto:albrkic@ifs.hr) (A.L. Brkić), [darko.mitrovic@univie.ac.at](mailto:darko.mitrovic@univie.ac.at) (D. Mitrović), [andrej.novak@phy.hr](mailto:andrej.novak@phy.hr) (A. Novak).

<https://doi.org/10.1016/j.jare.2020.04.015>

2090-1232/© 2020 The Authors. Published by Elsevier B.V. on behalf of Cairo University.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

double-well potential), Eq. (1) is the famous stationary Cahn-Hilliard equation (CHE) (see the original paper [10]), a well-known macroscopic field model for the phase separation of a binary alloy at a fixed temperature. It is derived from the Helmholtz free energy

$$\mathcal{E}[u] = \int_{\Omega} \left[ H(u(x)) + \frac{1}{2} \epsilon^2 |\nabla u(x)|^2 \right] dx \quad (3)$$

where  $u$  typically denotes the concentration,  $\epsilon$  is the range of intermolecular forces,  $H(u)$  is the free energy density and the last term is a contribution to the free energy originating from the spatial fluctuations of  $u$ .

Note that it is almost a rule that nonlinear PDEs (like Perona-Malik or the CHE mentioned above; see also [35]) often capture the most interesting phenomena. This increases the computational costs and makes the numerical procedure more complicated.

In this contribution we assume  $H$  that to be quadratic, which yields a linear equation, but instead of integer order derivatives we deal with a fractional order equation. The motivation for this comes from a simple observation from fractional calculus. Namely, recall that for given boundary values linear diffusion  $\Delta u = 0$  yields only linear solutions. On the other hand,  $\Delta^\mu u = 0$  has a much wider set of solutions and due to this, it is reasonable to expect that the image inpainting using fractional equations produces images that seem more natural (see Fig. 1).

The aim of this paper is to study the application of the fractional generalization of the Cahn-Hilliard type equations (CHTE) given in (1) to the image inpainting problem and to propose a fast algorithm for obtaining its numerical solutions. Through several examples, we are going to show that fractional PDEs produce superior results over integer order PDEs.

To this end, we derive a fast algorithm based on the matrix decomposition that solves (1) (formulated as (27)) in the local as well as in the non-local case. In both cases, the idea is to use appropriate arrangements of the discrete equations obtained by the finite difference method (see [12]) so that the computed matrix of the linear system exhibits a sparse structure with block symmetry (see (35)). This structure enables us to derive the recursive relations for the computation of the decomposition that, by using simple backward and forward substitutions, yields the solution. We also carry out a comparison of this approach with the standard algorithms for numerical solutions of the sparse linear system.

We would like to emphasize that the discretized fractional order partial differential equation (PDE) under the consideration serves as a motivation for the construction of a fast and efficient inpainting algorithms rather than as a problem from a purely mathematical point of view that will be submitted to the rigorous numerical analysis.

The rest of the paper is structured as follows. In the next section, we give a short overview of the previous approaches, motivations and ideas underlying the inpainting problem. In Section “Numerical method”, we introduce the notion of the discrete Laplacean and its fractional powers with the applications to the equation under consideration. Together with that, we derive an algorithm based on matrix decompositions for both local and

nonlocal case for the purposes of fast image inpainting. In Section “Results” we present the application of the introduced ideas on several testing images, comparing it with well known linear methods. Finally in Section “Conclusions and further work”, we finish with a short discussion and ideas for the future work.

## A short overview of previous results

The literature regarding the PDEs with the applications to the image inpainting problems is extensive. The terminology of digital image inpainting first appeared in the paper of Bertalmio in [4], based on the discretization of the transport-like PDE model

$$u_t = \nabla^\perp u \cdot \nabla \Delta u \text{ on } \omega, \quad (4)$$

$$u = f(x) \text{ on } \Omega \setminus \omega, \quad (5)$$

which is, for stabilization purposes, coupled with the anisotropic diffusion

$$u_t = \zeta_\epsilon |\nabla u| \nabla \cdot \frac{\nabla u}{|\nabla u|}, \quad (6)$$

where  $\zeta_\epsilon$  is a smooth cut-off function that forces the equation to act only on  $\omega$ . Furthermore,  $\nabla^\perp u = (-u_y, u_x)$  represents the perpendicular gradient of the image and this is the term that controls the speed of the transport. In this model  $\kappa = \nabla \cdot \frac{\nabla u}{|\nabla u|}$  is the curvature along the isophotes (curves on a surface that connect points of equal brightness),  $\Delta u$  is a measure of image smoothness and  $\nabla^\perp u$  is the propagation direction, i.e., the direction of smallest spatial change. The idea was to extend the image intensity in the direction of the isophotes arriving to the subset  $\omega \subset \Omega$ , where  $\omega$  is the inpainting domain. It can be shown that the steady state equation of (4) is the equation satisfied by the steady state inviscid flow in the two dimensional incompressible Navier-Stokes equation [5]. In this concept we can identify image intensity as a stream function for which the Laplacian of the image intensity models the vorticity that results in an algorithm that continues the isophotes while matching gradient vectors at the boundary of the inpainting domain.

Given the subjective nature of the image inpainting problem it is reasonable to argue that the brain interpolates broken missing edges using elastica-type curves. More precisely, if we slightly extend the inpainting domain  $\omega$  in  $\Omega$  and denote it by  $\omega_\epsilon$ , one can extrapolate the isophotes of an image  $u$  by a collection of curves  $\{\gamma_t\}_{t \in [I_0, I_M]}$  with no mutual crossing, which coincide with the isophotes of  $u$  on  $\omega_\epsilon \setminus \omega$  and minimize the energy

$$\int_{I_0}^{I_M} \int_{\gamma_t} (\alpha + \beta |\kappa_{\gamma_t}|^p) ds dt, \quad (7)$$

where  $[I_0, I_M]$  is the intensity span and  $\kappa$  is the curvature. For some parameters  $\alpha$  and  $\beta$ , depending on the specific application, this energy penalizes a generalized Euler’s elastica energy. In [33] the authors proposed two novel inpainting models based on the seminal Mumford-Shah image model [24] and its high order correction, called Mumford-Shah-Euler image model. The second one improves

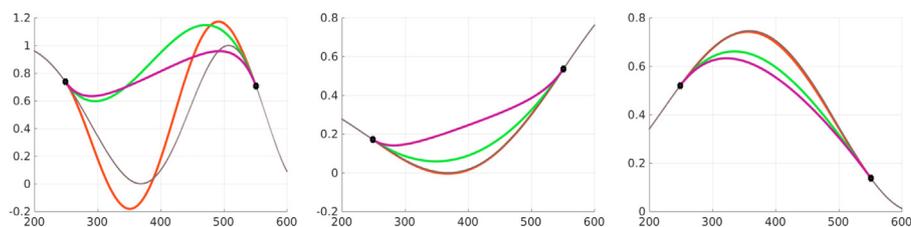


Fig. 1. Example of 1D inpainting problem on the  $\omega = [250, 550]$ . Biharmonic equation (green), integer order CHTE (magenta) and fractional order CHTE (red).

the first model by replacing the embedded straight-line curve model with Euler’s elastica, first introduced by Mumford in the context of curve modeling. This approach is not very computationally efficient, and attempts have been made to create more effective schemes, and various extensions involving augmented Lagrangians were considered (see [34,37,38]).

More recently, modified CH and Allen–Cahn equations for the inpainting of binary images have been analyzed [6,7,9,15,18]. In the integer order case, besides the standard double well-potential, researchers have investigated the nonsmooth double obstacle potential [8] (considered in this contribution) with the applications to the grayscale images, as well as the logarithmic potential [13]. They successfully demonstrated that a simpler class of models (comparing to [33]) can be modified to achieve fast inpainting of simple binary shapes, text reparation, road interpolation, and image upscaling. This was the motivation for the development of advanced numerical methods based on finite difference methods [11], finite element methods with preconditioning [8], spectral methods [9], and operator splitting [20] in order to make the practical computation faster and more efficient. For a mathematical analysis of the fourth order models we refer to [35].

Several years later the investigation of fractional models started in signal and image processing, a tool already widely used in physics [30]. Fractional derivatives with respect to space have been used in the attempts to describe more accurately the anomalous diffusion or dispersion, where a particle plume spreads at a rate inconsistent with the well known Brownian model of motion, and the plume may be asymmetric. The application of fractional calculus resulted in superior algorithms for the edge detection [21], filters for texture improvement [25], noise removal [3,39], etc. Roughly speaking, the idea is to solve the following equation

$$\nabla^\mu u (I(u, \nabla u) \nabla^\nu u) = 0, \tag{8}$$

on either the entire or a part of the image domain where  $I(u, \nabla u)$  is an appropriately chosen function, depending on the specific application. For a numerical treatment of such equations see [12,22], and for applications in image inpainting cf. [1].

In general, fractional differential equations are characterized by nonlocal and spatially heterogeneous properties in which classical models fail to provide the adequate results. Regarding image inpainting problems it has been shown that they improve the image quality as well as the peak signal to noise ratio (PSNR) [1,19,41]. For a review of the field, starting from simple harmonic inpainting to the state of the art methods in PDE based inpainting see [31,32].

*Physical reasoning in the integer case*

Even though ad hoc adjustments of the governing equations have been known to produce impressive results (for example the Perona-Malik equation [36]) it is important to keep in mind the underlying thermodynamic theory for the construction of this class of tools, at least when dealing with integer order equations. Let

$$\mathcal{E}(u) = \int_{\Omega} H_l(u, \nabla u, \nabla^2 u, \dots) dx \tag{9}$$

be the free energy functional, where  $H_l$  is the local free energy per molecule. Next, we expand  $H_l$  in a Taylor series around  $\mathbf{u}_0 = (u, 0, 0, \dots)$  and neglect higher order terms

$$\begin{aligned} H_l(u, u_{x_1}, u_{x_2}, u_{x_1 x_1}, u_{x_1 x_2}, u_{x_2 x_2}, \dots) \approx \\ H(u) + \sum_{i=1}^2 \frac{\partial H_l(\mathbf{u}_0)}{\partial u_{x_i}} u_{x_i} + \frac{1}{2} \sum_{i,j=1}^2 \frac{\partial^2 H_l(\mathbf{u}_0)}{\partial u_{x_i} \partial u_{x_j}} u_{x_i} u_{x_j} \\ + \frac{1}{2} \sum_{i,j,k=1}^2 \frac{\partial^2 H_l(\mathbf{u}_0)}{\partial u_{x_k} \partial u_{x_i x_j}} u_{x_k} u_{x_i x_j} + \frac{1}{2} \sum_{i,j,k,l=1}^2 \frac{\partial^2 H_l(\mathbf{u}_0)}{\partial u_{x_i x_j} \partial u_{x_k x_l}} u_{x_i x_j} u_{x_k x_l}. \end{aligned} \tag{10}$$

Imposing that the free energy is invariant under all rotations and reflections i.e.

$$\begin{aligned} \frac{\partial H_l(\mathbf{u}_0)}{\partial u_{x_i}} = 0, \quad \frac{\partial H_l(\mathbf{u}_0)}{\partial u_{x_i x_i}} = \varepsilon_1, \quad \frac{\partial^2 H_l(\mathbf{u}_0)}{\partial u_{x_i}^2} = \varepsilon_2, \quad i = 1, 2, \\ \frac{\partial H_l(\mathbf{u}_0)}{\partial u_{x_i x_j}} = \frac{\partial^2 H_l(\mathbf{u}_0)}{\partial u_{x_i} \partial u_{x_j}} = 0, \quad i \neq j. \end{aligned} \tag{11}$$

we get the local free energy

$$H_l(u, u_{x_1}, u_{x_2}, u_{x_1 x_1}, u_{x_2 x_2}, \dots) = H(u) + \varepsilon_1 \Delta u + \frac{\varepsilon_2}{2} |\nabla u|^2 + \dots \tag{12}$$

After integration over the domain and integration by parts we obtain the total free energy

$$\mathcal{E}(u) = \int_{\Omega} \left( H(u) + \frac{\varepsilon^2}{2} |\nabla u|^2 + \dots \right) dx, \tag{13}$$

where  $\varepsilon^2 = \varepsilon_2 - \frac{2\varepsilon_1}{\partial u}$ .

Let us consider the mixture of two miscible phases, where  $u_1$  and  $u_2$  are relative concentrations of the components such that  $u_1 = u$  and  $u_2 = 1 - u, u \in [0, 1]$ . In the general case, the corresponding flux is given by

$$J = -D(u, |\nabla u|) \nabla (\mu_2 - \mu_1), \tag{14}$$

where  $D(u, |\nabla u|)$  is the (generalized) diffusivity, and  $\mu_1, \mu_2$  are the chemical potentials of the components. By Fick’s first law, the gradient of two chemical potentials can be calculated as a variation of a corresponding free energy potential

$$\mu_2 - \mu_1 = \frac{\delta \mathcal{E}(u)}{\delta u}. \tag{15}$$

Combining (14) and (15) and assuming a general anisotropic situation, as is often the case in image processing, one obtains

$$J = -D(u, |\nabla u|) \nabla \frac{\delta \mathcal{E}(u)}{\delta u}. \tag{16}$$

Now if we assume that the mass is conserved we obtain a class of equations depending on the choice of energy functional

$$\frac{\partial}{\partial t} (\rho u) + \nabla \cdot J = 0, \tag{17}$$

$$\frac{\partial}{\partial t} u - \nabla \cdot \left( D(u, |\nabla u|) \nabla \frac{\delta \mathcal{E}[u]}{\delta u} \right) = 0. \tag{18}$$

In typical situations, while constructing PDE interpolation or a filter, one can choose either the specific diffusivity  $D(u, |\nabla u|)$  or the free energy functional  $\mathcal{E}(u)$  in an effort to process the image. In the simplest case where we neglect terms with derivatives and take  $H(u)$  to be quadratic,  $D(u, |\nabla u|) = const$  one obtains the linear diffusion equation (Gaussian filter), the very first PDE model for harmonic inpainting and image processing. Note that harmonic inpainting is a linear extension scheme, and, because of this, images obtained by employing such a technique do not produce very convincing results. In practice, one replaces the Laplace operator with the biharmonic one to define cubic inpainting by taking the total free energy in the form (13) to get

$$\frac{\partial}{\partial t} u - \nabla \cdot D \nabla (\varepsilon^2 \Delta u + au) = 0. \tag{19}$$

As for the application in image processing, we assume that  $\varepsilon \equiv const$  (or we can replace  $\varepsilon^2 \Delta u$  by a non-isotropic constant coefficient elliptic operator), and we take the stationary case, i.e. we arrive at (1) with quadratic  $H$  and  $\mu = v = 1$ .

Extension to the fractional case

Continuing in the same direction as explained in the introduction, it is natural to go beyond integer derivatives in order to increase the variety of curves which can be used during the inpainting procedure (we recall that in the harmonic case it is a linear curve and in the bi-harmonic case it is a third order polynomial; see Fig. 1 and Fig. 2).

This intention can be supported by the following arguments. The system is expected to evolve so that Helmholtz energy (3) decreases in time and approaches a minimum. For a Hilbert space  $V$ , Eq. (1) with  $\mu = \nu = 1$  can be viewed as a gradient flow

$$\frac{\partial u}{\partial t} = -D\nabla \mathcal{E}[u], \tag{20}$$

where  $D$  is some positive constant and the gradient of  $\mathcal{E}$  at a point  $u \in V$  is defined as follows

$$(\nabla \mathcal{E}[u], v)_V = \frac{d}{ds} \mathcal{E}(u + sv)|_{s=0}. \tag{21}$$

Let  $(\cdot, \cdot)_{L^2}$  denote standard  $L^2$  scalar product, now by ignoring boundary terms for the moment we can obtain that

$$\begin{aligned} \frac{d}{ds} \mathcal{E}(u + sv)|_{s=0} &= (\epsilon^2 \nabla u, \nabla v)_{L^2} + (H'(u), v)_{L^2} \\ &= (-\epsilon^2 \Delta u + H'(u), v)_{L^2}. \end{aligned} \tag{22}$$

Now we can identify two distinct situations. In the first one, we can choose  $V = L^2(\Omega)$  to obtain the gradient

$$\nabla \mathcal{E}[u] = -\epsilon^2 \Delta u + H'(u), \tag{23}$$

and the associated gradient flow is

$$\frac{\partial u}{\partial t} + (-\epsilon^2 \Delta u + H'(u)) = 0, \tag{24}$$

that gives already mentioned Allen-Cahn equation. It is well known that this equation does not preserve mass. Alternatively, one can take  $V = H^{-1}(\Omega)$  with the scalar product  $(v, w)_{H^{-1}} = ((-\Delta)^{-1/2} v, (-\Delta)^{-1/2} w)_{L^2}$ . In this case the associated gradient flow is Eq. (1) with  $\mu = \nu = 1$  that does preserves mass.

Next natural step would be to explore the gradient flow in the space  $H^{-\mu}$  where  $\mu > 0$  with the appropriate scalar product  $(v, w)_{H^{-\mu}} = ((-\Delta)^{-\mu/2} v, (-\Delta)^{-\mu/2} w)_{L^2}$ . In this case, a gradient is given by

$$\nabla \mathcal{E}[u] = (-\Delta)^\mu (-\epsilon^2 \Delta u + H'(u)). \tag{25}$$

In conclusion, one can consider a gradient flow in the Sobolev space  $H^{-\mu}$ , for  $\mu > 0$  where the choice  $\mu = 0$  one recovers Allen-Cahn equation and  $\mu = 1$  yields CHE. At this point, further generalizations could be obtained by allowing the other Laplacian in (25) to be of fractional order. For more details please see [2].

Numerical method

In order to introduce a discretization procedure, we shall first rewrite (1), (2) with

$$H(u) = \frac{a}{2} u^2 \tag{26}$$

in a more suitable way for the numerical treatment. Denote by  $\lambda_0 > 0$  a positive constant (in the applications below, we take  $\lambda_0 = 1$ ), by  $\lambda_\omega = \lambda_0 \chi_{\Omega \setminus \omega}$  for the indicator function  $\chi_{\Omega \setminus \omega}$  of the  $\Omega \setminus \omega$ . Namely, we will investigate the following equivalent variant of (1), (2)

$$\lambda_\omega (-\Delta)^\mu (au - \epsilon^2 (-\Delta)^\nu u) + (\lambda_0 - \lambda_\omega)(u - f) = 0 \text{ in } \Omega. \tag{27}$$

Fractional derivatives can be defined in several, essentially equivalent, ways (see e.g. the classical books [26,29]). However, depending on the situation, certain variants of the definition of fractional derivatives provide better operational aspects.

The fractional power of the discrete Laplace operator can be found in [12, Theorem 1.1].

**Theorem 1.** Let  $0 < \mu < 1$  and  $\mathbb{Z}_h = \{hj : j \in \mathbb{Z}\}$ . Furthermore, let  $v : \mathbb{Z}_h \rightarrow \mathbb{R}$  be such that

$$\sum_{m \in \mathbb{Z}} \frac{|v_m|}{(1 + |m|)^{1 \pm 2\mu}} < \infty. \tag{28}$$

Then

$$(-\Delta)^\mu v_j = \sum_{m \in \mathbb{Z}, m \neq j} (v_j - v_m) K_\mu(j - m), \tag{29}$$

where the discrete kernel  $K_\mu^h$  is given by

$$K_\mu^h(m) = \frac{4^\mu \Gamma(1/2 + \mu)}{\sqrt{(\pi)} |\Gamma(-\mu)|} \cdot \frac{\Gamma(|m| - \mu)}{h^{2\mu} \Gamma(|m| + 1 + \mu)}. \tag{30}$$

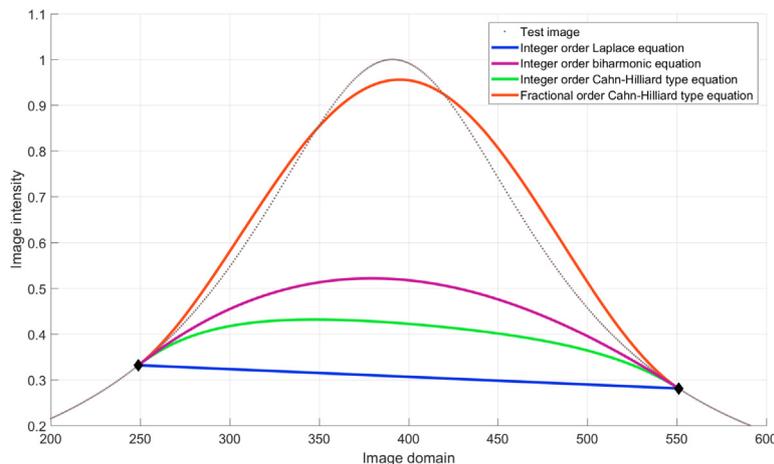


Fig. 2. Graphical representation of the image inpainting problem of the Runge function using integer order and fractional order equations with  $\mu = 0.8$  on  $\omega = [250, 550]$ .

Discretization of the integer order problem

In this section, motivated by applicability of the algorithm and methodical reasons, we want to lay out the main ideas of the numerical scheme in the integer order case, i.e.  $\mu = \nu = 1$  that will be extended to the non-integer case.

We discretize at grid points in the square domain which are at  $(x_i, y_j)$  with  $x_i = ih$  and  $y_j = jh$ , with  $h = \frac{1}{n+1}$ , where  $n$  represents the resolution of the image. Let us abbreviate  $u_{ij} = u(ih, jh)$  and  $f_{ij} = f(ih, jh)$ . By using the standard three-point discretization to approximate  $u_{xx}(x, y)|_{x=x_i, y=y_j}$  and  $u_{yy}(x, y)|_{y=y_j, x=x_i}$ , and applying it to  $\Delta u$ , one can derive the following:

$$\begin{aligned} (u_{xx}(x, y))_{xx}|_{x=x_i, y=y_j} &= u_{xxxx}(x, y)|_{x=x_i, y=y_j} \\ &\approx \frac{u_{i+2j} - 4u_{i+1j} + 6u_{ij} - 4u_{i-1j} + u_{i-2j}}{h^4}, \\ (u_{xx}(x, y))_{yy}|_{x=x_i, y=y_j} &= u_{xxyy}(x_i, y_j) \\ &\approx \frac{1}{h^4} [u_{i+1, j+1} - 2u_{i+1, j} + u_{i+1, j-1} - 2u_{ij+1} + 2u_{ij+1} \\ &\quad + u_{ij} - 2u_{ij-1} + u_{i-1, j+1} - 2u_{i-1, j} + u_{i-1, j-1}]. \end{aligned} \tag{31}$$

Using this approximation we can write the first term on the left hand side of (27) in the following form

$$\begin{aligned} -\Delta (\epsilon^2 \Delta u) - a \Delta u &= \\ &\frac{\epsilon^2}{h^4} u_{i+2j} + \left(\frac{8\epsilon^2}{h^4} - \frac{a}{h^2}\right) u_{i+1j} + \left(-\frac{20\epsilon^2}{h^4} + \frac{4a}{h^2}\right) u_{ij} \\ &+ \left(\frac{\epsilon^2}{h^4} + \frac{\epsilon^2}{h^4} - \frac{a}{h^2}\right) u_{i-1j} - \frac{\epsilon^2}{h^4} u_{i-2j} - \frac{\epsilon^2}{h^4} u_{ij+2} \\ &+ \left(\frac{8\epsilon^2}{h^4} - \frac{a}{h^2}\right) u_{ij+1} + \left(\frac{8\epsilon^2}{h^4} - \frac{a}{h^2}\right) u_{ij-1} - \frac{\epsilon^2}{h^4} u_{ij-2} \\ &- \frac{2\epsilon^2}{h^4} u_{i+1, j+1} - \frac{2\epsilon^2}{h^4} u_{i+1, j-1} - \frac{2\epsilon^2}{h^4} u_{i-1, j+1} - \frac{2\epsilon^2}{h^4} u_{i-1, j-1} + \tau_{ij}, \end{aligned} \tag{32}$$

where the truncation error  $\tau_{ij}$  is bounded by  $\mathcal{O}(h^2)$ . For  $\tilde{f}_{ij}$  defined by (40), Eq. (40) yields a set of  $n^2$  linear equations in  $n^2$  unknowns  $u_{ij}$  as follows

$$\begin{aligned} \epsilon^2 u_{i+2j} + (8\epsilon^2 - ah^2) u_{i+1j} + (-20\epsilon^2 + 4ah^2) u_{ij} \\ + (8\epsilon^2 - ah^2) u_{i-1j} - \epsilon^2 u_{i-2j} - \epsilon^2 u_{ij+2} + (8\epsilon^2 - ah^2) u_{ij+1} \\ + (8\epsilon^2 - ah^2) u_{ij-1} - \epsilon^2 u_{ij-2} - 2\epsilon^2 u_{i+1, j+1} - 2\epsilon^2 u_{i+1, j-1} - 2\epsilon^2 u_{i-1, j+1} \\ - 2\epsilon^2 u_{i-1, j-1} = h^4 \tilde{f}_{ij}, 1 \leq i, j \leq n. \end{aligned} \tag{33}$$

Let us now rewrite the  $n^2$  equations given by (33) as a single matrix equation by making the following arrangement of the  $n^2$  unknowns  $u_{ij}, 1 \leq i, j \leq n$ . We will use the linear isomorphism  $vec : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n^2}$  that reshapes an  $n \times n$  matrix into a vector of  $n^2$  elements, in the following way

$$X = vec \left( \begin{bmatrix} u_{1,1} & u_{1,2} & \dots & u_{1,n} \\ u_{2,1} & u_{2,2} & \dots & u_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n,1} & u_{n,2} & \dots & u_{n,n} \end{bmatrix} \right) \mapsto \begin{bmatrix} u_{1,1} \\ \vdots \\ u_{n,1} \\ u_{1,2} \\ \vdots \\ u_{n,2} \\ \vdots \\ u_{1,n} \\ \vdots \\ u_{n,n} \end{bmatrix}. \tag{34}$$

This approach leads to a  $n^2 \times n^2$  symmetric block pentadiagonal matrix

$$S = \begin{bmatrix} A & B & C & 0 & 0 & 0 & \dots & 0 \\ B & A & B & C & 0 & 0 & \dots & 0 \\ C & B & A & B & C & 0 & \dots & 0 \\ 0 & C & B & A & B & C & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & C & B & A & B & C \\ 0 & \dots & 0 & 0 & C & B & A & B \\ 0 & \dots & 0 & 0 & 0 & C & B & A \end{bmatrix} \tag{35}$$

abbreviated by  $S = \text{diag}(A, B, C)$ , where the matrices  $A, B$  and  $C$  are defined as follows

$$A = \text{diag}(-20\epsilon^2 + 4ah^2, 8\epsilon^2 - ah^2, \epsilon^2), \tag{36}$$

$$B = \text{diag}(8\epsilon^2 - ah^2, -2\epsilon^2), \tag{37}$$

$$C = \text{diag}(-\epsilon^2). \tag{38}$$

Thus we arrive at the problem of solving the sparse symmetric  $n^2 \times n^2$  linear system  $SX = F$ . If we consider the columns of  $S$  to be vectors in  $\mathbb{R}^{n^2}$ , we can easily conclude that they are linearly independent, so  $S$  is a regular matrix. Hence we can conclude that the linear system  $SX = F$  has a unique solution. Before turning to the question of solving it, we make a small note, relevant for the practical implementation. Namely, for implementation purposes, the matrix  $S$  can be constructed easily using the Kronecker product

$$S = I \otimes A + I_{-1} \otimes B + I_{+1} \otimes B + I_{-2} \otimes C + I_{+2} \otimes C, \tag{39}$$

where  $I$  is  $n \times n$  identity matrix,  $I_{-1} = \delta_{ij+1}, I_{+1} = \delta_{i+1j}$  and  $I_{\pm 2} = I_{\pm 1}^2$ , where  $\delta_{ij}$  is the standard Kronecker symbol.

Formulation of the linear system

Now we shift our focus to the discrete form of (27). Taking into account the finite difference equations from the previous section we want to solve

$$Zu = (\lambda_0 - \lambda_\omega)F, \tag{40}$$

where  $Z$  is defined as  $Z = \lambda_\omega S + (\lambda_0 - \lambda_\omega)I$ , where  $I$  is the identity matrix,  $F = \text{vec}(f)$  is the original image and  $\lambda_\omega$  is the characteristic function of the set  $\omega$  defined as in (27). Note that, in general,  $Z$  is not a symmetric matrix.

Solutions of the linear system

We aim to apply a suitable factorization to the symmetric matrix  $Z^T Z$  so that  $Z^T Z = LL^T$ , where  $L$  is a lower-diagonal matrix. To this end we suppose that  $L$  (and  $L^T$ ) is a lower (upper) triangular matrix of the following form

$$Z^T Z = \begin{bmatrix} A_1 & B_2 & C_3 & D_4 & E_5 & 0 & 0 & 0 & \dots & 0 \\ B_2 & A_2 & B_3 & C_4 & D_5 & E_6 & 0 & 0 & \dots & 0 \\ C_3 & B_3 & A_3 & B_4 & C_5 & D_6 & E_7 & 0 & \dots & 0 \\ D_4 & C_4 & B_4 & A_4 & B_5 & C_6 & D_7 & E_8 & \dots & 0 \\ E_5 & D_5 & C_5 & B_5 & A_5 & B_6 & C_7 & D_8 & \dots & 0 \\ 0 & E_6 & D_6 & C_6 & B_6 & A_6 & B_7 & C_8 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & 0 & E_{n-2} & D_{n-2} & C_{n-2} & B_{n-2} & A_{n-2} & B_{n-1} & C_n \\ 0 & 0 & 0 & 0 & E_{n-1} & D_{n-1} & C_{n-1} & B_{n-1} & A_{n-1} & B_n \\ 0 & 0 & 0 & 0 & 0 & E_n & D_n & C_n & B_n & A_n \end{bmatrix}, \tag{41}$$

$$L^T = \begin{bmatrix} \alpha_1^T & \beta_2^T & \gamma_3^T & \delta_4^T & \varepsilon_5^T & 0 & 0 & 0 & \dots & 0 \\ 0 & \alpha_2^T & \beta_3^T & \gamma_4^T & \delta_5^T & \varepsilon_6^T & 0 & 0 & \dots & 0 \\ 0 & 0 & \alpha_3^T & \beta_4^T & \gamma_5^T & \delta_6^T & \varepsilon_7^T & 0 & \dots & 0 \\ 0 & 0 & 0 & \alpha_4^T & \beta_5^T & \gamma_6^T & \delta_7^T & \varepsilon_8^T & \dots & 0 \\ 0 & 0 & 0 & 0 & \alpha_5^T & \beta_6^T & \gamma_7^T & \delta_8^T & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha_6^T & \beta_7^T & \gamma_8^T & \dots & 0 \\ \vdots & \ddots & \dots & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 & 0 & \alpha_{n-2}^T & \beta_{n-1}^T & \gamma_n^T \\ 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_{n-1}^T & \beta_n^T \\ 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_n^T \end{bmatrix}, \quad (42)$$

Let us note that this is a very general form and that for practical purposes the matrix  $Z^T Z$  might be even more sparse, depending of the size of the inpainting domain. We also observe that the special form of the matrix  $Z$  enables us to perform this symmetrisation in only  $\mathcal{O}(n^2)$  operations. Furthermore, keeping in mind that each  $\alpha_i, \beta_i, \gamma_i, \delta_i$  and  $\varepsilon_i$  are  $n \times n$  matrices we obtain the following recursive scheme for determining the elements of the lower-diagonal matrix  $L$  by using only the definition of the matrix  $Z^T Z$ :

$$\varepsilon_i = E_i (\alpha_{i-4}^T)^{-1}, \quad (43)$$

$$\delta_i = (D_i - \varepsilon_i \beta_{i-3}^T) (\alpha_{i-3}^T)^{-1}, \quad (44)$$

$$\gamma_i = (C_i - \delta_i \beta_{i-2}^T - \varepsilon_i \gamma_{i-2}^T) (\alpha_{i-2}^T)^{-1}, \quad (45)$$

$$\beta_i = (B_i - \gamma_i \beta_{i-1}^T - \varepsilon_i \delta_{i-1}^T - \delta_i \gamma_{i-1}^T) (\alpha_{i-1}^T)^{-1}, \quad (46)$$

$$\alpha_i \alpha_i^T = A_i - \beta_i \beta_i^T - \gamma_i \gamma_i^T - \delta_i \delta_i^T - \varepsilon_i \varepsilon_i^T, \quad i = 5, \dots, n, \quad (47)$$

where  $\alpha_1$  to  $\alpha_4, \beta_2$  to  $\beta_4, \gamma_3, \gamma_4$  and  $\delta_4$  are given by

$$\alpha_1 \alpha_1^T = A_1, \quad (48)$$

$$\beta_2 = B_1 (\alpha_1^T)^{-1}, \quad (49)$$

$$\alpha_2 \alpha_2^T = A_2 - \beta_2 \beta_2^T, \quad (50)$$

$$\gamma_3 = (C_3) (\alpha_1^T)^{-1}, \quad (51)$$

$$\beta_3 = (B_3 - \gamma_3 \beta_2^T) (\alpha_2^T)^{-1}, \quad (52)$$

$$\alpha_3 \alpha_3^T = A_3 - \beta_3 \beta_3^T - \gamma_3 \gamma_3^T, \quad (53)$$

$$\delta_4 = (D_4) (\alpha_1^T)^{-1}, \quad (54)$$

$$\gamma_4 = (C_4 - \delta_4 \beta_2^T) (\alpha_2^T)^{-1}, \quad (55)$$

$$\beta_4 = (B_4 - \gamma_4 \beta_3^T - \delta_4 \gamma_3^T) (\alpha_3^T)^{-1}, \quad (56)$$

$$\alpha_4 \alpha_4^T = A_4 - \beta_4 \beta_4^T - \gamma_4 \gamma_4^T - \delta_4 \delta_4^T. \quad (57)$$

Note that in each step one needs to compute the inverse of  $\alpha_i$  and this can be done in  $\mathcal{O}(n^3)$  operations. Now we proceed in two steps: a) solve  $LY = F$ , b) use the computed  $Y$  to solve  $L^T X = Y$  and obtain the solution  $X$ . After the computation of the lower-diagonal matrix  $L$  we have the following recursion

$$\alpha_i Y_i + \beta_i Y_{i-1} + \gamma_i Y_{i-2} + \delta_i Y_{i-3} + \varepsilon_i Y_{i-4} = F_i, \quad i = 5, \dots, n, \quad (58)$$

where  $Y_{i-1}, Y_{i-2}, Y_{i-3}$  and  $Y_{i-4}$  have already been determined, beginning with

$$\alpha_1 Y_1 = F_1, \quad (59)$$

$$\alpha_2 Y_2 + \beta_2 Y_1 = F_2, \quad (60)$$

$$\alpha_3 Y_3 + \beta_3 Y_2 + \gamma_3 Y_1 = F_3, \quad (61)$$

$$\alpha_4 Y_4 + \beta_4 Y_3 + \gamma_4 Y_2 + \delta_4 Y_1 = F_4. \quad (62)$$

Because the matrices  $\alpha_i$  have already been computed (during the decomposition step) we can perform a forward substitution

$$Y_i = \alpha_i^{-1} (F_i - \beta_i Y_{i-1} - \gamma_i Y_{i-2} - \delta_i Y_{i-3} - \varepsilon_i Y_{i-4}), \quad i = 5, \dots, n, \quad (63)$$

in order to obtain  $Y$ . The solution  $X$  is finally computed through the backward substitution given by

$$X_i = (\alpha_i^T)^{-1} (Y_i - \beta_{i+1}^T X_{i+1} - \gamma_{i+2}^T X_{i+2} - \delta_{i+3}^T X_{i+3} - \varepsilon_{i+4}^T X_{i+4}), \quad i = 1, \dots, n-4. \quad (64)$$

Here, the boundary cases are defined in the following way

$$\alpha_n^T X_n = Y_n, \quad (65)$$

$$\alpha_{n-1}^T X_{n-1} + \beta_n^T X_n = Y_{n-1}, \quad (66)$$

$$\alpha_{n-2}^T X_{n-2} + \beta_{n-1}^T X_{n-1} - \gamma_n^T X_n = Y_{n-2}, \quad (67)$$

$$\alpha_{n-3}^T X_{n-3} + \beta_{n-2}^T X_{n-2} - \gamma_{n-1}^T X_{n-1} - \delta_n^T X_n = Y_{n-3}. \quad (68)$$

Because the inverses of  $\alpha_i^T$  are known, it is easy to see that both, forward and backward substitutions can be done in  $\mathcal{O}(n^2)$  operations.

In total, this yields  $\mathcal{O}(n^4)$  operations.

#### Discretization of fractional differential equations

Now, we are ready to deal with the fractional variant of (1), (2). In our simulations we have fixed  $\mu + \nu = 2$  because numerical experiments have indicated that this could be the compromise between the quality of the inpainting results and keeping the numerical scheme relatively simple. In this way, with the appropriate selection of the parameters  $a$  and  $\varepsilon^2$ , Eq. (1) can be viewed as a fractional inpainting with a corrective local term of high order. Using Theorem 1 (and notations from there) and assuming  $a = 1$ , we have

$$\begin{aligned} & (-\Delta)^\mu u - \varepsilon^2 \Delta^2 u = \\ & (-\varepsilon^2 - K(2)) u_{i+2j} + (8\varepsilon^2 - K(1)) u_{i+1j} + (-20\varepsilon^2 + 4K(1) + 4K(2) + \\ & 4K(3) + 4K(4) + 4K(5) + 4K(6)) u_{ij} + (8\varepsilon^2 - K(1)) u_{i-1j} + \\ & (-\varepsilon^2 - K(2)) u_{i-2j} - (-\varepsilon^2 - K(2)) u_{i+2j} + (8\varepsilon^2 - K(1)) u_{i+1j} + \\ & (8\varepsilon^2 - K(1)) u_{i-1j} + (-\varepsilon^2 - K(2)) u_{i-2j} - 2\varepsilon^2 u_{i+1j+1} - 2\varepsilon^2 u_{i+1j-1} - \\ & 2\varepsilon^2 u_{i-1j+1} - 2\varepsilon^2 u_{i-1j-1} - K(3) (u_{ij+3} + u_{ij-3} + u_{i+3j} + u_{i-3j}) - \\ & K(4) (u_{i+4} + u_{i-4} + u_{i+4j} + u_{i-4j}) - K(5) (u_{i+5} + u_{i-5} + u_{i+5j} + u_{i-5j}) - \\ & K(6) (u_{i+6} + u_{i-6} + u_{i+6j} + u_{i-6j}). \end{aligned} \quad (69)$$

Here we have neglected terms containing the constant  $K(m)$ , with  $m \geq 7$  because the constants  $K(m)$  rapidly tend to zero for increasing  $m$  and because taking more terms does not seem to influence the subjective assessment of the inpainted image. Thus, it has a negligible influence in minimizing the relative  $L^2$  error (see Section "Results").

Next, we proceed similarly as in the integer case by defining.

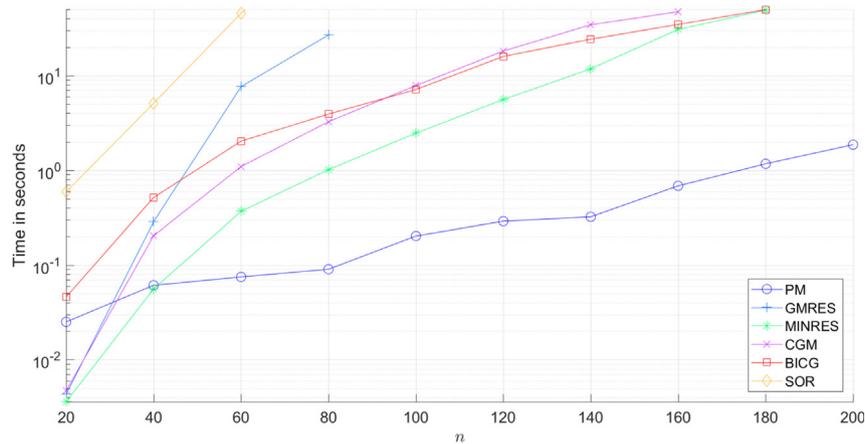
$S = \text{diag}(A, B, C, D, E, F, G)$ , where the matrices  $A, B, C, D, E, F$  and  $G$  are given by

$$A = \text{diag}(-20\varepsilon^2 + 4K(1) + 4K(2) + 4K(3) + 4K(4) + 4K(5) + 4K(6), \quad (70)$$

$$8\varepsilon^2 - K(1), -\varepsilon^2 - K(2), -K(3)),$$

**Table 1**  
Values of the parameters for the inpainting results shown on Fig. 1.

Inpainting method	$\epsilon^2$	min	max	$L^2_{rel}$
Image (a)	–	0.00	0.53	–
Integer order biharmonic eq.	1	0.06	0.53	0.223
Integer order CHTE	10	0.14	0.53	0.746
Fractional order CHTE $\mu = 0.30$	1	0.00	0.53	0.014
Image (b)	–	0.00	1.00	–
Integer order biharmonic eq.	1	0.60	1.15	0.878
Integer order CHTE	10	0.63	0.96	0.757
Fractional order CHTE $\mu = 0.80$	1	–0.18	1.17	0.320
Image (c)	–	0.14	0.75	–
Integer order biharmonic eq.	1	0.14	0.66	0.118
Integer order CHTE	100	0.14	0.63	0.163
Fractional order CHTE $\mu = 0.40$	1	0.14	0.74	0.010



**Fig. 3.** Comparison of different solvers for the system. PM denotes the approach proposed in this paper.

$$B = \text{diag}(8\epsilon^2 - K(1), -2\epsilon^2), \tag{71}$$

$$C = \text{diag}(-\epsilon^2 - K(2)), \tag{72}$$

$$D = \text{diag}(-K(3)), \tag{73}$$

$$E = \text{diag}(-K(4)), \tag{74}$$

$$F = \text{diag}(-K(5)), \tag{75}$$

$$G = \text{diag}(-K(6)). \tag{76}$$

Now we define the matrix  $Z$  as in (40) and follow the same steps as in the integer case.

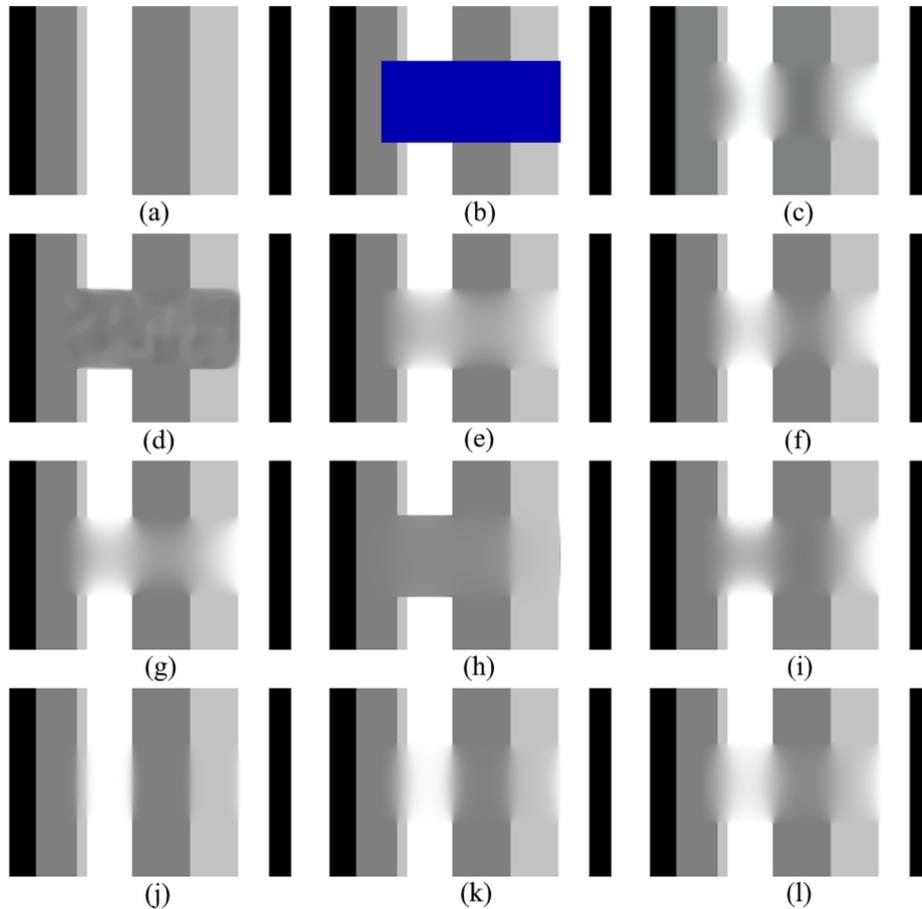
**Results**

In Fig. 2 we set  $a = 1$  and compare applications of different equations on a 1D inpainting problem on the domain  $\omega = [250, 550]$ . Let us note that the simple integer order inpainting using the Laplace equation resulted in a relative  $L^2$ -error of 0.59974, the linear integer order CHTE ( $\epsilon^2 = 10$ ) (27) produced a relative  $L^2$ -error of 0.38415, and on the other hand the (linear) fractional order CHTE ( $\epsilon^2 = 1$ ) produced a relative  $L^2$ -error of 0.05859. Besides the lower  $L^2$  relative error, fractional order equations seem to preserve the image features (also see Fig. 1) and thereby produces images that look more natural. Clearly, the fractional order CHTE delivers superior results compared to the linear integer order equations.

Furthermore, we have performed the experiment on 100 1D test images with 3 different inpainting domains and different fractional orders in (69) and compared it to the results obtained by the integer order equations. By choosing the result of the fractional order inpainting with the least  $L^2$ -relative error (with respect to the undamaged image) and comparing it to the error produced by integer order inpainting, we conclude that the fractional order inpainting approach yields on average 28.68% lower  $L^2$ -relative error. The selection of the parameters was done by exhaustive enumeration method. Selected images from this experiment are presented in Fig. 1. Further details are given in Table 1.

Moreover, for different dimensions  $n$  of the system and different numerical methods we have performed 6 independent measurements of the running times required for solving the inpainting problem. The average computational times are presented in Fig. 3, where one can compare the computational performance of the proposed approach as compared to the classical numerical methods for solving such a system. Note that the running time of the algorithm under the consideration in the case  $n^2 = 40000$  was only 1.87 s whereas other methods were not able to yield a solution within 100 s. This experiment was performed on the standard desktop computer.

In Fig. 4, the test example consists of a gray scale image that contains a wide damaged area in the shape of a rectangle in the middle of the image. For three different parameters  $\mu$  the proposed method was tested against the MATLAB inpainting function called `inpaintn`, transport equation of Bertalmio [5], Laplace and biharmonic inpainting as well as integer order CHTE and two total variation (TV and high order TV denoted by TV4) inpainting methods. The MATLAB files for Laplace, transport and total variation meth-



**Fig. 4.** Inpainted gray scale stripes image using different PDE based inpainting methods. (a) Original image, (b) Image with the inpainting domain, (c) Matlab `inpaintn` function, (d) Transport equation of Bertalmio, (e) Local Laplace inpainting, (f) Local biharmonic inpainting, (g) Integer order CHTE, (h) TV inpainting, (i) TV4 inpainting, and fractional order CHTE with (j)  $\mu = 0.7$ , (k)  $\mu = 0.8$ , (l)  $\mu = 0.9$ .

**Table 2**  
Results for the gray scale stripes inpainting using different models, presented on Fig. 4.

Inpainting method	a	$\epsilon^2$	PSNR	SSIM	$L_{rel}^2$
Image with crack	–	–	9.8941	0.78550	0.46369
Matlab <code>inpaintn</code> function	–	–	26.266	0.56987	0.064359
Transport eq.	–	–	18.284	0.80683	0.17605
Integer order Laplace eq.	0	1	24.257	0.80114	0.088172
Integer order biharmonic eq.	1	0	27.307	0.81967	0.061465
Integer order CHTE	1	10	25.742	0.81167	0.073854
TV	–	–	19.762	0.78139	0.14869
TV4	–	–	24.460	0.78756	0.085689
Fractional order CHTE $\mu = 0.7$	10	1	37.580	0.89977	0.019116
Fractional order CHTE $\mu = 0.8$	10	1	31.774	0.82172	0.037167
Fractional order CHTE $\mu = 0.9$	10	1	28.810	0.80908	0.052209

ods are available on the Web.<sup>1,2</sup> All simulations are run with standard parameters that were determined by the exhaustive enumeration approach (Fig. 4, results (d)–(i)).

For further details on values of the parameters please see Table 2.

In the Fig. 5 we have applied the same methods as in the Fig. 4, but this time to a real and more complex image – the famous Lena with a  $2.5\times$  zoom covering details of the nose and right eye region. For further details on values of the parameters please see Table 3.

Moreover, we have performed thousands of experiments with above mentioned approaches (shown in Fig. 4 and Fig. 5, results (d)–(i)), however, we do not exclude the possibility that even better results for the other approaches could be obtained by a fine tuning of the parameters. Based on the tests performed for the fractional order

CHTE, it seems that the best inpainting results are obtained for the values of  $\mu$  close to 0.7, although this is probably subjected to the features of the image under the consideration.

In addition, in Fig. 6, we have applied the proposed method on the RGB image where each color channel was treated separately. We see that the difference between the original image and the inpainted one is almost not detectable.

## Conclusions and further work

The success of the inpainting depends on the choice of curves which can be used to interpolate damaged parts of the image. If we have only linear curves or third order polynomials as in the case



**Fig. 5.** Inpainted gray scale Lena using different PDE based inpainting methods with  $2.5\times$  zoom over the nose and right eye region. (a) Original image, (b) Image with the inpainting domain in blue, (c) Matlab `inpaintn` function, (d) Transport equation of Bertalmio, (e) Local Laplace inpainting, (f) Local biharmonic inpainting, (g) Integer order CHTE, (h) TV inpainting, (i) TV4 inpainting, and fractional order CHTE with (j)  $\mu = 0.7$ , (k)  $\mu = 0.8$ , (l)  $\mu = 0.9$ .

**Table 3**  
Results for gray scale Lena image inpainting using different models, presented on Fig. 5.

Inpainting method	a	$\epsilon^2$	PSNR	SSIM	$L^2_{rel}$
Image with crack	–	–	20.612	0.92782	0.146790
Matlab <code>inpaintn</code> function	–	–	39.661	0.98699	0.008313
Transport eq.	–	–	35.339	0.97638	0.022761
Integer order Laplace eq.	0	1	38.639	0.98532	0.009822
Integer order biharmonic eq.	1	0	40.080	0.98733	0.009100
Integer order CHTE	1	10	38.982	0.98602	0.009420
TV	–	–	30.468	0.45713	0.019342
TV4	–	–	30.746	0.54271	0.010896
Fractional order CHTE $\mu = 0.7$	10	1	40.892	0.98958	0.007518
Fractional order CHTE $\mu = 0.8$	10	1	39.643	0.98802	0.008540
Fractional order CHTE $\mu = 0.9$	10	1	39.265	0.98700	0.009382

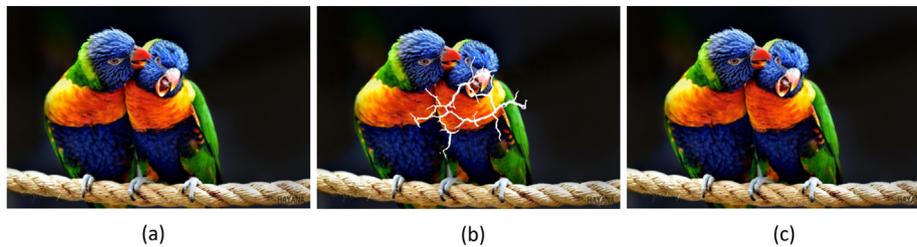


Fig. 6. Example of the image produced by the proposed equation. (a) Original image, (b) Image with inpainting domain, (c) Result of the inpainting.

of the harmonic or biharmonic inpainting approach, we cannot obtain satisfactory results. One way to overcome this limitation and still remain in the framework of analysis of harmonic and biharmonic PDEs is to add nonlinear terms (this is the case with the CHE), but such an approach decreases computational efficiency and usually requires a non-standard numerical treatment.

In the current contribution, we extended the choice of possible interpolating curves not by adding a nonlinear (correcting) terms, but by replacing integer by fractional order derivatives, staying at the same time in the linear setting. This significantly simplifies the numerical treatment of the problem and decreases computational costs. On the other hand, we find the obtained results at least equally convincing as the ones obtained using the integer order CHTE.

In future work, we shall try to extend this approach by introducing equations with nonlinear coefficients and derivatives of variable order [14] or derivatives of complex order [29]. We shall also continue in the direction of rigorously proving the convergence of the scheme and optimizing the order of the equation used for the inpainting.

### Compliance with Ethics Requirements

This article does not contain any studies with human or animal subjects.

### Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgment

This research is supported in part by COST action 15225, by the project P30233 of the Austrian Science Fund FWF, by the project M-2669 Meitner-Programme of the Austrian Science Fund, by the Croatian Science Foundation's funding of the project *Microlocal defect tools in partial differential equations* (MiTPDE) with Grant No. 2449 and by the bilateral project *Applied mathematical analysis tools for modeling of biophysical phenomena*, between Croatia and Serbia. The permanent address of D.M. is University of Montenegro, Montenegro.

### References

- [1] Abirami A, Prakash P, Thangavel K. Fractional diffusion equation-based image denoising model using CN–GL scheme. *Int J Comput Math* 2018;95:1222–39.
- [2] Ainsworth M, Mao Z. Analysis and approximation of a fractional Cahn–Hilliard equation. *SIAM J Numer Anal* 2017;55(4):1689–718.
- [3] Bai J, Feng X-C. Fractional-order anisotropic diffusion for image denoising. *IEEE Trans Image Process* 2007;16:2492–502.
- [4] Bertalmio M, Sapiro G, Caselles V, Ballester C. Image inpainting. In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co.; 2000.
- [5] Bertalmio M, Bertozzi A, Sapiro G. Navier–stokes, fluid dynamics, and image and video inpainting. In: *Proceedings of the 2001 IEEE computer society conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001. vol. 1*. IEEE, 2001.
- [6] Bertozzi A, Esedoglu S, Gillette A. Inpainting of binary images using the Cahn–Hilliard equation. *IEEE Trans Image Process* 2007;16:285–91.
- [7] Bertozzi A, Esedoglu S, Gillette A. Analysis of a two-scale Cahn–Hilliard model for binary image inpainting. *Multiscale Model Simul* 2007;6:913–36.
- [8] Bosch J, Kay D, Stoll M, Wathen AJ. Fast solvers for Cahn–Hilliard inpainting. *SIAM J Imaging Sci* 2014;7:67–97.
- [9] Bosch J, Stoll M. A fractional inpainting model based on the vector-valued Cahn–Hilliard equation. *SIAM J Imaging Sci* 2015;8(4):2352–82.
- [10] Cahn JW, Hilliard JE. Free energy of a nonuniform system. I. Interfacial free energy. *J Chem Phys* 1958;28:258–67.
- [11] Cheng M, Warren JA. An efficient algorithm for solving the phase field crystal model. *J Comput Phys* 2008;227:6241–8.
- [12] Ciaurri Ó, Roncal L, Stinga PR, Torrea JL, Varona JL. Nonlocal discrete diffusion equations and the fractional discrete Laplacian, regularity and applications. *Adv Math* 2018;330:688–738.
- [13] Copetti MIM, Elliott CM. Numerical analysis of the Cahn–Hilliard equation with a logarithmic free energy. *Numer Math* 1992;63(1):39–65.
- [14] Diening L, Harjulehto P, Hästö P, Ruzicka M. *Lebesgue and Sobolev spaces with variable exponents*. Springer; 2011.
- [15] Esedoglu S, Tsai Y-HR. Threshold dynamics for the piecewise constant Mumford–Shah functional. *J Comput Phys* 2006;211:367–84.
- [16] Kim J, Lee S, Choi Y, Lee S-M, Jeong D. Basic principles and practical applications of the Cahn–Hilliard equation. *Math Probl Eng* 2016;2016:11.
- [17] Larnier S, Mecca R. Fractional-order diffusion for image reconstruction. In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE; 2012.
- [18] Li Y, Jeong D, Choi J, Lee S, Kim J. Fast local image inpainting based on the Allen–Cahn model. *Dig Signal Process* 2015;37:65–74.
- [19] Mathieu B, Melchior P, Oustaloup A, Ceyral Ch. Fractional differentiation for edge detection. *Signal Process* 2003;83:2421–32.
- [20] Meerschaert MM, Scheffler H-P, Tadjeran C. Finite difference methods for two-dimensional fractional dispersion equation. *J Comp Phys* 2006;211:249–61.
- [21] Mumford D, Shah J. Optimal approximations by piecewise smooth functions and associated variational problems. *Commun Pure Appl Math* 1989;42:577–685.
- [22] Pu Y-F, Zhou J-L, Yuan X. Fractional differential mask: a fractional differential-based approach for multiscale texture enhancement. *IEEE Trans Image Process* 2010;19:491–511.
- [23] Podlubny I. *Fractional differential equations: an introduction to fractional derivatives*. San Diego - New York - London: Academic Press; 1999.
- [24] Samko SG, Kilbas AA, Marichev OL. *Fractional integrals and derivatives – theory and applications*. Amsterdam: Gordon and Breach Science Publishers; 1993.
- [25] Schneider WR, Wyss W. Fractional diffusion and wave equations. *J Math Phys* 1989;30:134–44.
- [26] Schönlieb C. *Partial differential equation methods for image inpainting, cambridge monographs on applied and computational mathematics*. Cambridge: Cambridge University Press; 2015. <https://doi.org/10.1017/CBO9780511734304>.
- [27] Schönlieb CB. *Partial differential equation methods for image inpainting, vol. 29*. Cambridge University Press; 2015.
- [28] Shen J, Kang SH, Chan TF. Euler's elastica and curvature-based inpainting. *SIAM J Appl Math* 2003;63:564–92.
- [29] Tai X-C, Hahn J, Chung GJ. A fast algorithm for Euler's elastica model using augmented Lagrangian method. *SIAM J Imaging Sci* 2011;4:313–44.
- [30] Theljani A, Theljani A, Belhachmi Z, Kallel M, Moakher M. A multiscale fourth-order model for the image inpainting and low-dimensional sets recovery. *Math Meth Appl Sci* 2017;40:3637–50.
- [31] Weickert J. *Anisotropic diffusion in image processing*. ECMI, B.G. Teubner Stuttgart; 1998.
- [32] Zhu W, Tai Xue-Cheng, Chan T. Augmented Lagrangian method for a mean curvature based image denoising model. *Inverse Probl. Imaging* 2013;7:1409–32.
- [33] Zhang J, Chen R, Deng C, Wang S. Fast linearized augmented lagrangian method for Euler's elastica model. *Numer Math: Theory Meth Appl* 2017;10:98–115.
- [34] Zhang J, Chen K. A total fractional-order variation model for image restoration with nonhomogeneous boundary conditions and its numerical solution. *SIAM J Imaging Sci* 2015;8:2487–518.
- [35] Zhou M et al. Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans Image Process* 2012;21:130–44.