Research Article

Mohammad F. Al-Jamal*, Ahmad Baniabedalruhman, and Abedel-Karrem Alomari

Data smoothing with applications to edge detection

https://doi.org/10.1515/math-2022-0036
received November 11, 2021; accepted April 9, 2022

Abstract: The aim of this paper is to present a new stable method for smoothing and differentiating noisy data defined on a bounded domain $\Omega \subset \mathbb{R}^N$ with $N \geq 1$. The proposed method stems from the smoothing properties of the classical diffusion equation; the smoothed data are obtained by solving a diffusion equation with the noisy data imposed as the initial condition. We analyze the stability and convergence of the proposed method and we give optimal convergence rates. One of the main advantages of this method lies in multivariable problems, where some of the other approaches are not easily generalized. Moreover, this approach does not require strong smoothness assumptions on the underlying data, which makes it appealing for detecting data corners or edges. Numerical examples demonstrate the feasibility and robustness of the method even with the presence of a large amounts of noise.

Keywords: data smoothing, numerical differentiation, noisy data, diffusion equation, regularization, edge detection

MSC 2020: 65F22, 47A52, 35K20, 65D19

1 Introduction

It is very common in many scientific applications, such as biological imaging, image processing and computer vision, inverse source and identification problems, to have to differentiate functions specified by data. In many cases, the data is experimental and so substantial noise or imprecision may be present. Since differentiation is an ill-posed problem [1], naive numerical differentiation techniques will amplify the noise resulting in inaccurate and mostly useless results. A concrete example is given by the function

$$f^{\delta}(x) = f(x) + \delta \sin(\omega x), \quad x \in (0, 1),$$

where f^{δ} is the noisy version of the true data f, and δ is a small number representing the noise level. The computed derivative is then

$$\frac{\mathrm{d}f^{\delta}}{\mathrm{d}x}(x) = \frac{\mathrm{d}f}{\mathrm{d}x}(x) + \omega\delta \cos(\omega x), \quad x \in (0, 1).$$

Evidently, the maximum error in the data is δ, which is small, while the maximum error in the computed derivative can be arbitrarily large, depending on the value of the frequency ω.

The last example showcases the *instability* associated with the differentiation process: small errors in the data might induce large errors in the computed derivative. To overcome the instability issue of

*** Corresponding author: Mohammad F. Al-Jamal,** Department of Mathematics, Yarmouk University, Irbid, 21163, Jordan, e-mail: mfaljamal@yu.edu.jo
Ahmad Baniabedalruhman: Department of Mathematics, Yarmouk University, Irbid, 21163, Jordan, e-mail: ahmad_a@yu.edu.jo
Abedel-Karrem Alomari: Department of Mathematics, Yarmouk University, Irbid, 21163, Jordan, e-mail: abdalkareem@yu.edu.jo

the differentiation operator, several regularization techniques have been developed. In [2], a perturbation based method is used for single variable functions. In [3], a mollification method is given. Smoothing splines is adopted in [4], and results for single variable functions are given. Other methods such as the Lanczos' generalized derivative [5], and the differentiation by integration using Jacobi polynomials [6], are known for single variable functions only. For a summary of other popular methods, see [7]. Despite its importance in real life applications, only few methods dealing with higher dimensional cases have reported [8–11]. A side from limited applicability due to dimension restriction, the aforementioned approaches are relatively slow to suit real life applications, particularly with big data.

In this paper, the smoothing property of the classical diffusion equation is utilized to reconstruct the gradient $\nabla f(x)$ of a multivariable function $f(x)$ from given noise-corrupted data $f^{\delta} \in L^2(\Omega)$ satisfying

$$\|f^{\delta} - f\|_{L^2(\Omega)} \leq \delta,$$

where $\Omega \subset \mathbb{R}^N$ is a bounded domain. For the time being, we shall assume that $f\,|_{\partial\Omega} = 0$; see the Appendix for the general case. More precisely, the proposed method amounts to take the function $\nabla_x u$ as stable approximation to ∇f where u solves the diffusion equation:

$$\partial_t u(x,t) = \Delta_x u(x,t), \quad x \in \Omega, \; t > 0, \tag{1}$$

subject to the initial and boundary conditions:

$$u(x,0) = f^{\delta}(x), \quad x \in \Omega, \quad u(x,t) = 0, \quad x \in \partial\Omega, \; t > 0, \tag{2}$$

where Δ is the Laplacian operator. The time t can be considered as a design parameter, and its rule is to control the smoothness of the desired approximation; in the context of regularization theory, t represents the *regularization parameter*. The precise formulation of the suggested method as well as the motivation behind it will be given in more detail in the sequel.

Under reasonable assumptions on true data f, we prove stability results and we propose optimal convergence rate for the proposed method. Particularly, we show that the method gives an optimal rate of convergence of order $O(\delta^{2/3})$. Several examples and applications in computer vision demonstrating the validity and efficiency of the proposed approach are also provided.

The rest of this paper is composed of five sections. Section 2 recalls some preliminary and auxiliary results. In Sections 3, we state the precise formulation of the proposed method and give the motivation behind it, and then we prove the main convergence results in Section 4. We provide several numerical experiments and applications in Section 5. More detailed discussion on the general case is provided in the Appendix.

2 Preliminaries

First, we recall some definitions and results from the theory of Sobolev spaces. Readers may refer to [12,13] for more detailed discussions. Throughout the rest of the paper, we assume that Ω is a bounded domain in $\mathbb{R}^N$, $N \geq 1$, with sufficiently smooth boundary $\partial\Omega$.

We use the notation $L^2(\Omega)$ to denote the usual Lebesgue space of square-integrable functions, which is a Hilbert space with the inner product

$$(u,v)_{L^2(\Omega)} = \int_{\Omega} uv, \quad \forall u, v \in L^2(\Omega).$$

The induced norm will be denoted by $\|\cdot\|_{L^2(\Omega)}$.

Let $\alpha = (\alpha_1, \ldots, \alpha_N)$ be a multi-index, and set $|\alpha| = \alpha_1 + \cdots + \alpha_N$. For a positive integer m, the Sobolev space $H^m(\Omega)$ is defined by

$$H^m(\Omega) = \{u \in L^2(\Omega) : \partial^{\alpha} u \in L^2(\Omega) \quad \forall |\alpha| \leq m\},$$

where the partial derivatives are taken in the distributional sense. It is a Hilbert space under the inner product

$$(u, v)_{H^m(\Omega)} = \sum_{|\alpha|\leq m} (\partial^\alpha u, \partial^\alpha v)_{L^2(\Omega)},$$

and a Banach space under the the corresponding induced norm. In particular, the norm on $H^1(\Omega)$ is given by

$$\|u\|^2_{H^1(\Omega)} = \|u\|^2_{L^2(\Omega)} + \|\nabla u\|^2_{L^2(\Omega)}.$$

The Hilbert space $H^1_0(\Omega)$ is defined as the closure of $C^\infty_0(\Omega)$ in $H^1(\Omega)$, where $C^\infty_0(\Omega)$ is the set of all functions that are infinitely differentiable on Ω and compactly supported in Ω. We have the characterization

$$H^1_0(\Omega) = \{u \in H^1(\Omega) \quad u|_{\partial\Omega} = 0\},$$

where the boundary value is understood in the trace sense.

By $H^{1/2}(\partial\Omega)$, we mean the space of all restrictions on $\partial\Omega$ from functions in $H^1(\Omega)$. It is a Hilbert space with the norm

$$\|u\|_{H^{1/2}(\partial\Omega)} = \inf_{v\in H^1(\Omega)} \Big\{\|v\|_{H^1(\Omega)} : v|_{\partial\Omega} = u\Big\}.$$

From the Sturm-Liouville theory [14], the eigenvalues of the eigenvalue problem

$$-\Delta\varphi = \lambda\varphi \text{ in } \Omega, \quad \varphi = 0 \text{ on } \partial\Omega \tag{3}$$

form a nondecreasing sequence of positive numbers $\{\lambda_n\}$ tending to infinity, and the corresponding (normalized) eigenfunctions $\{\varphi_n\} \subset H^1_0(\Omega) \cap H^2(\Omega)$ form an orthonormal basis for the space $L^2(\Omega)$. Moreover,

$$\|\nabla u\|^2_{L^2(\Omega)} = \sum_{n=1}^{\infty} (u, \varphi_n)^2_{L^2(\Omega)} \lambda_n, \quad \forall u \in H^1_0(\Omega). \tag{4}$$

Associated with the eigenvalue problem (3), the Hilbert space X_β is defined as follows:

$$X_\beta = \left\{ u \in L^2(\Omega) : \|u\|^2_{X_\beta} = \sum_{n=1}^{\infty} (u, \varphi_n)^2_{L^2(\Omega)} \lambda_n^{2\beta} < \infty \right\}.$$

The parameter β characterizes the regularity properties of the space X_β. We have $X_0 = L^2(\Omega)$, $X_{1/2} = H^1_0(\Omega)$, $X_1 = H^2(\Omega) \cap H^1_0(\Omega)$, and $X_\beta \subset H^{2\beta}(\Omega)$, see [14].

3 Motivation of the proposed algorithm

To motivate the proposed regularization approach, let u be the solution of the parabolic initial-boundary value problem given by equations (1) and (2). Let us set $y^\delta(x) = u(x, t)$ for $x \in \Omega$, and think of t as a parameter. Then, on separating variables, we have the following representation

$$y^\delta(x) = \sum_{n=1}^{\infty} (f^\delta, \varphi_n)_{L^2(\Omega)} \exp(-\lambda_n t)\varphi_n(x). \tag{5}$$

A standard result [14] is that $y^\delta \in H^2(\Omega) \cap H^1_0(\Omega)$ for $t > 0$. From the expansion given in (5), we deduce that

$$(y^\delta, \varphi_n)_{L^2(\Omega)} = \exp(-\lambda_n t)(f^\delta, \varphi_n)_{L^2(\Omega)}, \quad n = 1, 2, \dots$$

and so, due to the exponential term, the Fourier coefficients of y^δ decay faster than those of f^δ. It is also evident that $y^\delta \approx f^\delta$ for small values of the parameter t. Thus, from this brief analysis, we infer that y^δ is an approximation of f^δ that is smoother than the input data f^δ.

As a consequence of the aforementioned discussion, it would be reasonable to take $y^\delta(x)$, $t > 0$, as smooth approximations to $f(x)$ for $x \in \Omega$. Therefore, we suggest the following **numerical differentiation algorithm**:

Step 1. Compute the smoothed version of the noisy data by

$$y^\delta(x) = \sum_{n=1}^{\infty}(f^\delta, \varphi_n)_{L^2(\Omega)}\exp(-\lambda_n t)\varphi_n(x), \quad x \in \Omega;$$

Step 2. Take the gradient $\nabla y^\delta(x)$ as an approximation of $\nabla f(x)$.

Few remarks are in order. First, from expansion (5), we see that the parameter t plays the role of a *smoothing parameter*, that is, sufficiently small or large values of t result in undersmoothing or oversmoothing, respectively. Second, we point out that the gradient in the second step can be computed exactly or using an appropriate difference scheme depending on the nature of the available data.

4 Stability and consistency analysis

Our goal next is to examine $\nabla y^\delta(x)$ computed by the proposed algorithm as a stable approximation to $\nabla f(x)$. In particular, we study the stability and consistency of the aforementioned approach and prove a convergence result. Throughout the sequel, we assume $t > 0$, and for convenience, we also let

$$y(x) = \sum_{n=1}^{\infty}(f, \varphi_n)_{L^2(\Omega)}\exp(-\lambda_n t)\varphi_n(x), \quad x \in \Omega,$$

that is, y is the solution of (1) and (2) corresponding to the noiseless data f.

We start off with the following stability result.

Lemma 4.1. *We have*

$$\|\nabla y^\delta - \nabla y\|_{L^2(\Omega)} \le \frac{\delta}{2\sqrt{t}}.$$

Proof. Set $g = f^\delta - f$ and $u = y^\delta - y$. Since $g \in L^2(\Omega)$, we have $u \in H_0^1(\Omega)$. Then, it follows from (4) and the uniqueness of the Fourier coefficients that

$$\|\nabla u\|_{L^2(\Omega)}^2 = \sum_{n=1}^{\infty}\lambda_n\left(\sum_{m=1}^{\infty}(g, \varphi_m)_{L^2(\Omega)}\exp(-\lambda_m t)\varphi_m, \varphi_n\right)_{L^2(\Omega)}^2 = \sum_{n=1}^{\infty}(g, \varphi_n)_{L^2(\Omega)}^2\exp(-2\lambda_n t)\lambda_n.$$

By using the fact that $4t\exp(-t) \le 1$ and the Parseval's identity, we see that

$$\|\nabla u\|_{L^2(\Omega)}^2 \le \frac{1}{4t}\|g\|_{L^2(\Omega)}^2 \le \frac{\delta^2}{4t},$$

which concludes the proof. □

Throughout the sequel, we shall assume that $\beta > 1/2$, and we set $\eta = \beta - 1/2$ if $\beta < 3/2$, and $\eta = 1$ if $\beta \ge 3/2$. We have the following consistency result:

Lemma 4.2. *If $f \in X_\beta$, then*

$$\|\nabla y - \nabla f\|_{L^2(\Omega)} \le C_\beta t^\eta,$$

where $C_\beta = 2\max\{1, \lambda_1^{3/2-\beta}\}\|f\|_{X_\beta}$.

Proof. Set $u(x) = y(x) - f(x)$, and so $u \in H_0^1(\Omega)$. Moreover, from (4), the definition of y, and the uniqueness of the Fourier coefficients, we have

$$\|\nabla u\|^2_{L^2(\Omega)} = \sum_{n=1}^{\infty}[(y,\varphi_n)_{L^2(\Omega)} - (f,\varphi_n)_{L^2(\Omega)}]^2\lambda_n = \sum_{n=1}^{\infty}(f,\varphi_n)^2_{L^2(\Omega)}(1-\exp(-\lambda_n t))^2\lambda_n.$$

The last equality together with the fact that $(1-\exp(-t)) \le 2t/(t+1)$ imply

$$\|\nabla u\|^2_{L^2(\Omega)} \le 4\sum_{n=1}^{\infty}(f,\varphi_n)^2_{L^2(\Omega)}\left(\frac{\lambda_n t}{1+\lambda_n t}\right)^2\lambda_n = 4\sum_{n=1}^{\infty}(f,\varphi_n)^2_{L^2(\Omega)}\lambda_n^{2\beta}\frac{\lambda_n^{3-2\beta}t^2}{(1+\lambda_n t)^2}.$$

Now, it is an easy matter to show that

$$\frac{t^2\lambda^{3-2\beta}}{(1+t\lambda)^2} \le \begin{cases} t^{2\beta-1}, & \beta < 3/2,\\ \lambda_1^{3-2\beta}t^2, & \beta \ge 3/2,\end{cases}$$

for all $\lambda \ge \lambda_1$, from which we see

$$\|\nabla u\|^2_{L^2(\Omega)} \le 4\max\{1,\lambda_1^{3-2\beta}\}\|f\|^2_{X_\beta}\begin{cases} t^{2\beta-1}, & \beta < 3/2,\\ t^2, & \beta \ge 3/2,\end{cases}$$

which ends the proof. □

Combining Lemmas 4.1, 4.2, and the triangle inequality, we obtain the main convergence result:

Theorem 4.1. *Suppose that $f \in X_\beta$ for some $\beta > 1/2$, then there exists a constant K independent of t and δ such that*

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} \le K\left(t^\eta + \frac{\delta}{\sqrt{t}}\right).$$

Remark 4.1. If we choose the parameter t so that $t \sim \delta^\gamma$ for some $\gamma \in (0,2)$, then we obtain the convergence result

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} \to 0$$

as $\delta \to 0$. The convergence rate is optimal when $\gamma = 2/(2\eta+1)$; in this case, we have

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} = O(\delta^{2\eta/(2\eta+1)}).$$

We obtain the fastest convergence when $\beta \ge 3/2$, in which case, $\eta = 1$ and

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} = O(\delta^{2/3}),$$

provided that $t = C\delta^{2/3}$ for some positive constant C.

5 Numerical experiments

Since data acquisition results in only discrete data in many scientific applications, we assume $\Omega = (0,L)^N$ and that the noisy data, which we denote by $\mathbf{f}^\delta$, is sampled at $n = (m+1)^N$ regular grid points $\mathbf{X} = \{\mathbf{x}_i = (i_1h,\dots,i_Nh) : i \in \mathbf{I}\}$ using the formula

$$\mathbf{f}_i^\delta = f(\mathbf{x}_i) + \mathbf{e}_i, \quad i \in \mathbf{I} = \{(i_1,\dots,i_N) : i_1,\dots,i_N = 1,\dots,m-1\},$$

where $\mathbf{e}$ is some Gaussian noise with mean 0, and $h = L/m$. Let us further assume that $f|_{\partial\Omega} = 0$ (refer to the attached appendix for general treatment). We will designate the symbol δ to denote the relative noise level in the data measured in the l_2 norm, that is,

$$\|\mathbf{f}^\delta - \mathbf{f}\| = \delta\|\mathbf{f}\|,$$

where $\|\cdot\|$ is the usual Euclidean norm. We shall assess the quality of the recovered gradient by the relative l_2 error given by

$$\frac{\|\nabla \mathbf{y}^\delta - \nabla \mathbf{f}\|}{\|\nabla \mathbf{f}\|}.$$

Regarding the choice of the parameter t, we employ the Morozov's discrepancy principle (e.g., [1]), which amounts to choose t such that

$$\|\mathbf{y}^\delta - \mathbf{f}^\delta\| = \delta\|\mathbf{f}\|.$$

For such domain Ω, the eigenvalues and eigenfunctions of (3) are

$$\lambda_i = \left(\frac{i_1\pi}{L}\right)^2 + \cdots + \left(\frac{i_N\pi}{L}\right)^2, \quad \varphi_i(\mathbf{x}) = \sin\left(\frac{i_1\pi x_1}{L}\right)\cdots\sin\left(\frac{i_N\pi x_N}{L}\right).$$

Therefore, the Fourier discrete sine transform $\widehat{\mathbf{f}^\delta}$ of the data $\mathbf{f}^\delta$ is expressed as follows:

$$\widehat{\mathbf{f}^\delta}_i = \left(\frac{2}{m}\right)^{N/2} \sum_{k\in\mathbf{I}} \mathbf{f}^\delta_k \varphi_i(\mathbf{x}_k), \quad \forall i \in \mathbf{I}.$$

Using the trapezoidal rule over the regular grid $\mathbf{X}$, the smoothed data given by (5) can be computed as follows:

$$\begin{aligned} y^\delta(\mathbf{x}_j) &= \sum_{i\in\mathbf{I}} \left\{ \left(\frac{2}{l}\right)^N \int_\Omega f^\delta(\mathbf{x})\varphi_i(\mathbf{x})d\mathbf{x} \right\} e^{-\lambda_i t}\varphi_i(\mathbf{x}_j) \\ &\doteq \sum_{i\in\mathbf{I}} \left\{ \left(\frac{2}{m}\right)^N \sum_{k\in I} \mathbf{f}^\delta_k \varphi_i(\mathbf{x}_k) \right\} e^{-\lambda_i t}\varphi_i(\mathbf{x}_j) \\ &= \left(\frac{2}{m}\right)^{N/2} \sum_{i\in\mathbf{I}} (\mathbf{E}_i\, \widehat{\mathbf{f}^\delta}_i)\varphi_i(\mathbf{x}_j), \quad \forall j \in \mathbf{I}, \end{aligned}$$

where $\mathbf{E}_i = e^{-\lambda_i t}$. Thus, the smoothed data can be computed by

$$\mathbf{y}^\delta = \widehat{\widehat{\mathbf{f}^\delta} * \mathbf{E}},$$

where the symbol $(*)$ denotes the elementwise multiplication operator, and the hat $(\hat{\cdot})$ stands for the Fourier discrete sine transform.

In the following examples, the derivatives of the smoothed data are computed via the central difference formula:

$$\frac{\partial y^\delta}{\partial x_k}(\mathbf{x}_i) \doteq \frac{\mathbf{y}^\delta_{i_1,\ldots,i_k+1,\ldots,i_N} - \mathbf{y}^\delta_{i_1,\ldots,i_k-1,\ldots,i_N}}{2h}, \quad i \in \mathbf{I}, \; k = 1,\ldots,N$$

and similarly for the true and noisy data.

Now, we examine examples in several space dimensions. We point out that the computations are extremely quick owing to the fast Fourier transform (FFT) routine, which is available in many computer algebra systems.

Example 5.1. The true data are given by the function

$$f(x) = \exp(-3(2x-1)^2)\sin(5\pi x), \quad 0 \le x \le 1.$$

For noise level $\delta = 20\%$ and sample size $n = 500$, the true, noisy, and smoothed data, and their corresponding derivatives are shown in Figure 1. Error results are summarized in Table 1.

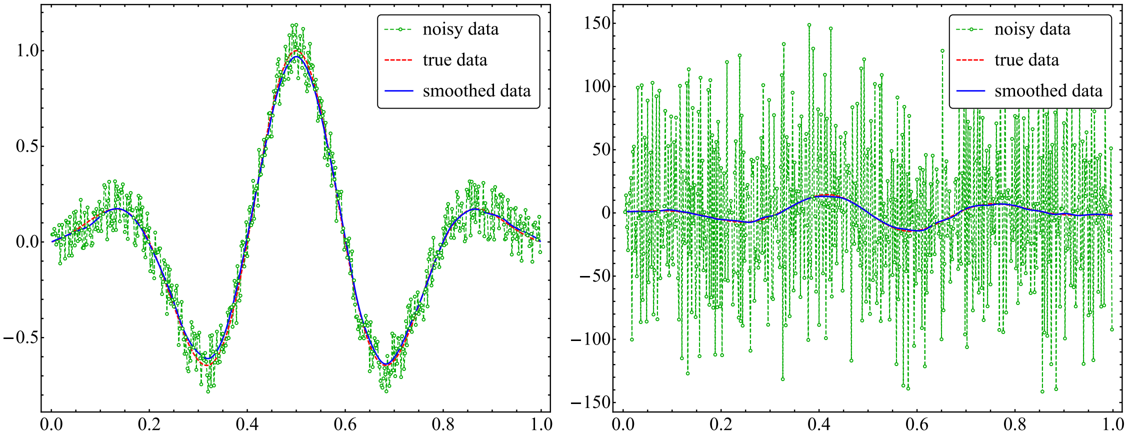

Figure 1: Data (left) and the corresponding derivative (right) for Example 5.1.

Table 1: l_2-relative errors in the estimated derivatives for Example 5.1 for several sample sizes n and noise levels δ. Numbers in parenthesis represent the relative errors computed from the noisy data

$\delta \backslash n$	50	500	5,000
0.2	0.190 (**0.809**)	0.085 (**9.084**)	0.043 (**89.13**)
0.02	0.045 (**0.080**)	0.030 (**0.908**)	0.020 (**8.913**)
0.002	0.007 (**0.008**)	0.012 (**0.090**)	0.008 (**0.891**)

Remark 5.1. The results in Example 5.1 suggest a convergence rate of $O(\varepsilon^{0.48})$. Since $f \in X_\beta$ for all $\beta < 1.25$, from Remark 4.1, we deduce the theoretic optimal order $O(\varepsilon^{0.6})$. The deterioration in the convergence rate is mainly due to discretization errors and the nonoptimal choice of the smoothing parameter t.

Example 5.2. We consider the recovery of the gradient of the function

$$f(x, y) = 5(6y - 3)\exp(-(6x - 3)^2 - (6y - 3)^2),$$

where $(x, y) \in [0, 1]^2$. For noise level $\delta = 10\%$ and sample size $n = 50 \times 50$, the noisy and smoothed data are depicted in Figure 2. The error results in the estimated gradients are summarized in Table 2.

Example 5.3. The true data are given by the function

$$f(x, y, z) = \sin((x - x^2)(y - y^2))(z - z^2),$$

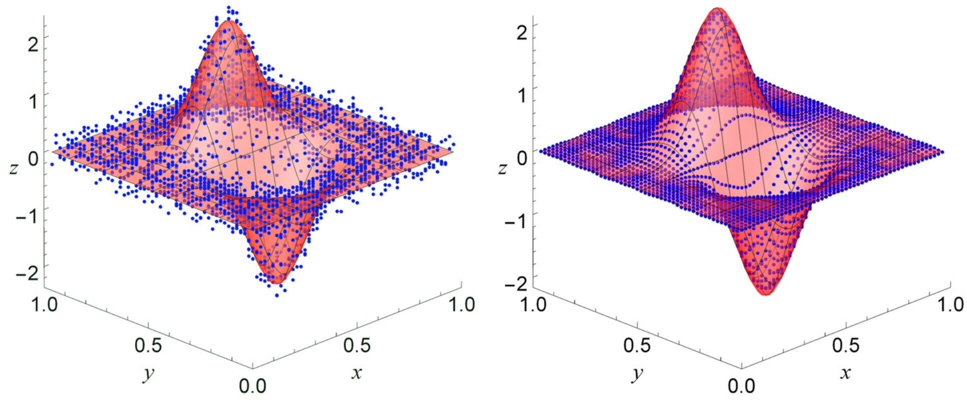

Figure 2: True data surface along with the noisy data (left) and smoothed data (right) for Example 5.2.

Table 2: l_2-relative errors in the recovered gradients for Example 5.2 for several sample sizes n and noise levels δ. Numbers in parenthesis represent the relative errors computed from the noisy data

$\delta \backslash n$	50 × 50	500 × 500	750 × 750
0.2	0.138 (**1.671**)	0.056 (**16.64**)	0.047 (**24.97**)
0.02	0.051 (**0.167**)	0.023 (**1.664**)	0.021 (**2.497**)
0.002	0.013 (**0.016**)	0.011 (**0.166**)	0.009 (**0.249**)

where $(x, y, z) \in [0, 1]^3$. A slice contour plot for the noisy and the corresponding smoothed data are shown in Figure 3. Relative errors in the estimated gradients are summarized in Table 3.

Remark 5.2. Contrary to the naive method, the error results in Tables 1–3 show that the proposed method is stable with respect to the sample size n, that is, as the grid is refined, the error gets smaller.

Now we present an application on edge detection (emphasizing) of digital images.

Example 5.4. Image gradient is an essential ingredient in common image processing and computer vision applications, particularly in edge detection algorithms. Since the gradient is sensitive to noise, noisy images must be denoised or smoothed prior to the gradient estimation. In common edge detection algorithms, such as the Canny method, a Gaussian filter is typically used for this purpose. In the following demonstration, we employ our method to smooth out the noisy images, and then the image gradient is approximated using the central difference formula. Consequently, the magnitude of the image gradient at pixel (x, y) is approximated by

$$\|\nabla f(x,y)\| \approx \frac{1}{2}\sqrt{[f(x+1,y)-f(x-1,y)]^2+[f(x,y+1)-f(x,y-1)]^2},$$

where f is the underlying image intensity function.

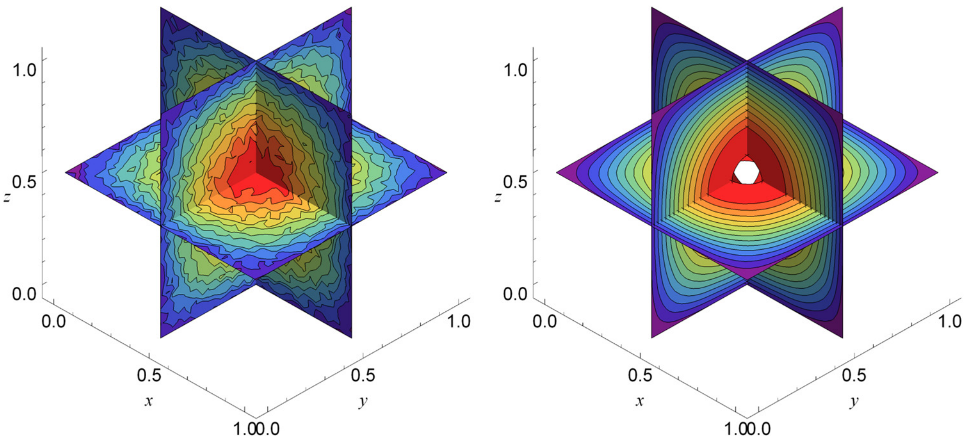

Figure 3: Sliced contours of noisy data (left) and smoothed data (right) for Example 5.3.

Table 3: l_2-relative errors in the recovered gradients for Example 5.3 for several sample sizes n and noise levels δ. Numbers in parenthesis represent the relative errors computed from the noisy data

$\delta \backslash n$	50 × 50 × 50	100 × 100 × 100	150 × 150 × 150
0.2	0.061 (**4.705**)	0.038 (**9.174**)	0.038 (**13.64**)
0.02	0.030 (**0.470**)	0.021 (**0.917**)	0.020 (**1.364**)
0.002	0.015 (**0.047**)	0.011 (**0.091**)	0.009 (**0.136**)

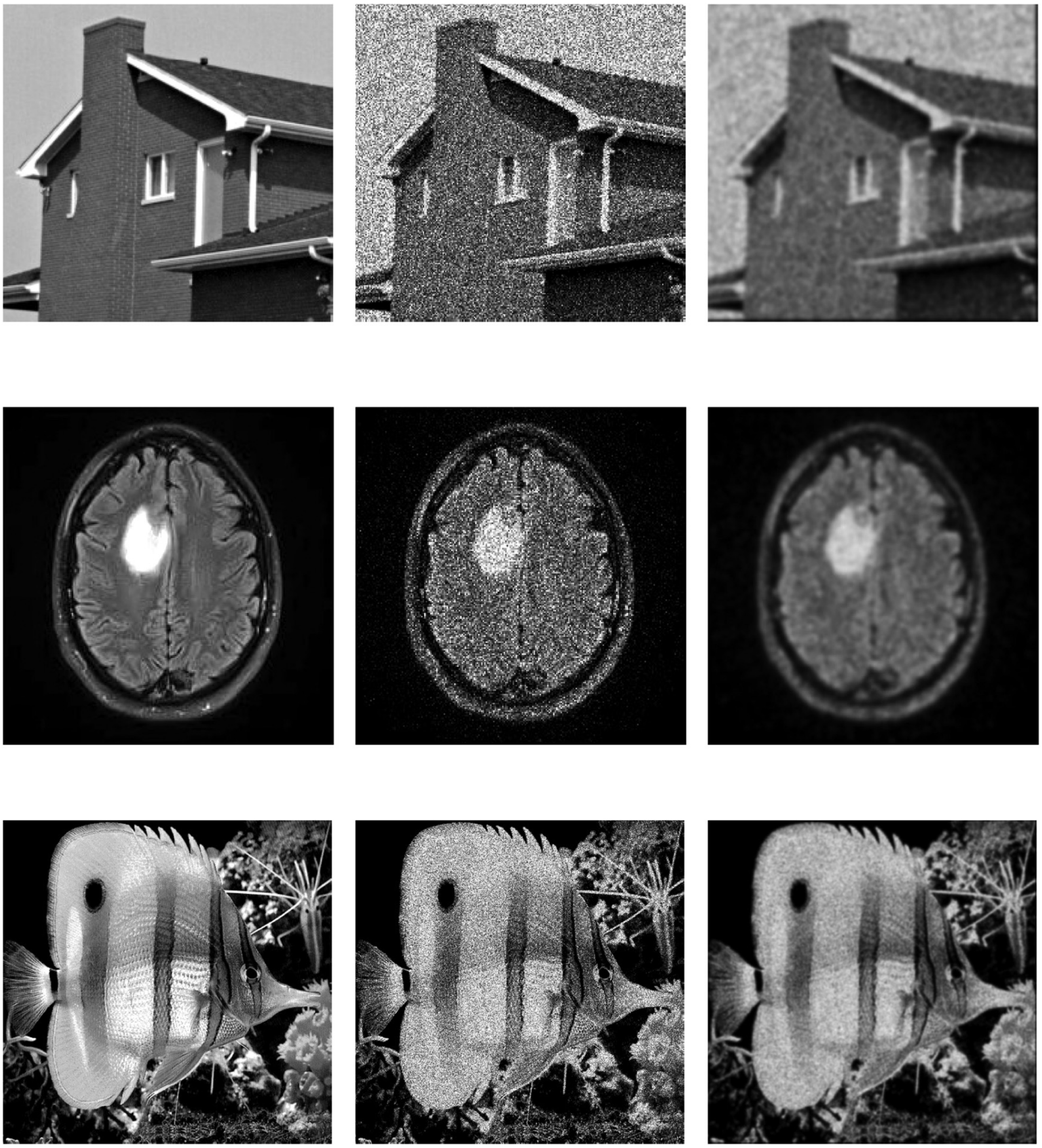

Figure 4: Original images (left), noisy images (middle), and smoothed images (right).

In this demonstration, we consider three noisy images that are obtained by adding a Poisson noise to the noise free images. The original, noisy, and smoothed images by the diffusion method are shown in Figure 4. To assess the quality of the smoothed (denoised) images, we computed the peak signal-to-noise ratio (PSNR) of the noisy and the smoothed images, given by

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{(255)^2}{\mathrm{MSE}}\right)(\mathrm{dB}),$$

$$\mathrm{MSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[f(i,j) - \tilde{f}(i,j)]^2,$$

where $f_{M\times N}$ and $\tilde{f}_{M\times N}$ are the original and denoised images, respectively. The PSNR results are summarized in Table 4.

Table 4: PSNR results for noisy and denoised images in Example 5.4

Image	Noisy image	Denoised image
House	15.129	21.530
Brain	18.092	25.216
Fish	15.568	21.164

The gradient magnitudes of the noisy and smoothed images are plotted in Figure 5. For the sake of comparison, we also plot the edges of the noisy images obtained by the popular Canny method [15], which consists of five main steps: Gaussian filtering, measuring intensity gradient, edge thinning, double thresholding to determine strong and weak edges, and edge tracking via blob analysis to extract weak edges.

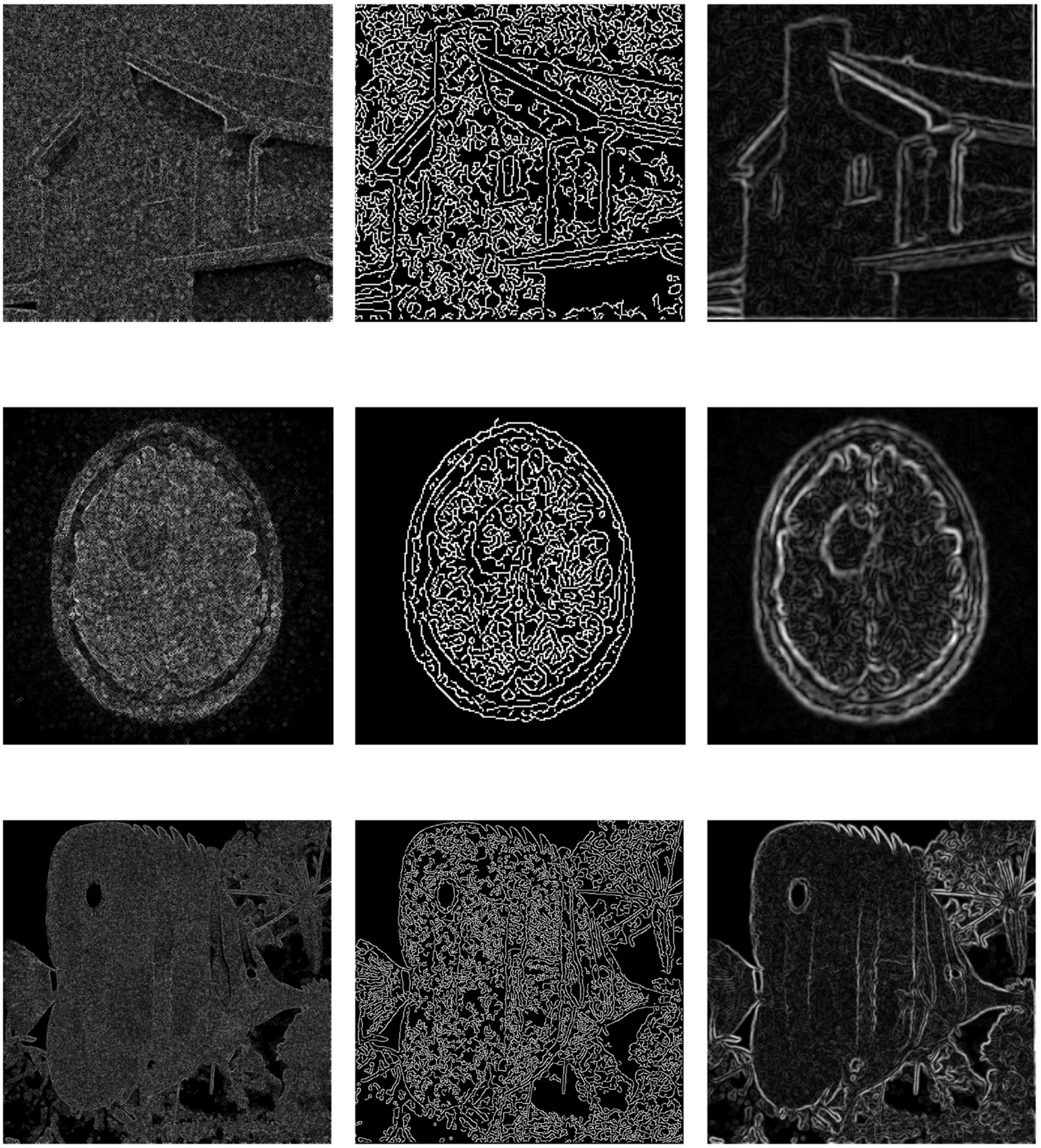

Figure 5: Detected edges: from noisy images (left), by Canny method (middle), and from smoothed images (right).

From Table 4, we see that the PSNR of the denoised images are significantly higher than those of noisy images, which indicates the denoised images by the proposed smoothing method have better qualities than the noisy images. Moreover, as it can be observed from Figures 4 and 5, the proposed technique can significantly enhance the quality of the recovered edges, and it seems that the smoothing stage has no bias toward certain direction; this is a major advantage since edge orientation is mostly unknown in real-life applications. It is evident from this experiment that diffusion smoothing gives plausible results, and it can be well suited for such type of applications.

From the presented examples, we see that the proposed diffusion technique is very suitable for data smoothing and numerical differentiation, and it can excel in various scientific applications due to its speed (in particular, when utilizing the fast Fourier transform), supporting theory behind it, and further, it does not pose any dimension limitations; the method can be extended easily to problems in any dimension, contrary to the most existing methods in which they are limited to one-dimensional domains.

6 Conclusion

The smoothing property of the diffusion equation is utilized to estimate the gradient of functions specified by noisy data. We proved stability and consistency results and carried out the convergence analysis for the proposed method. Several examples in one and higher dimensional domains demonstrate the feasibility and robustness of the proposed approach.

The main advantage of the method lies in multivariable problems, where some of the other smoothing approaches do not easily generalize. Moreover, the method shows decent results under mild assumptions on the true data even in the presence of a large amount of noise, while these could be limitations for other methods. We believe the method can excel in various scientific applications due to its speed, relative ease of implementation, and the supporting theory behind it; we look forward to extending this approach and the theoretical results for higher derivatives estimations. Furthermore, appropriate *a priori* and *a posteriori* parameter choice strategies for choosing t are under investigation.

Acknowledgements: The authors would like to thank the *Scientific Research and Graduate Studies* at Yarmouk University for supporting this paper, and they would like express their sincere gratitude to the anonymous referees for their helpful comments that helped us to improve the quality of the manuscript.

Conflict of interest: The authors state no conflict of interest.

References

[1] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer Academic, Dordrecht, 1996.

[2] L. Yang, *A perturbation method for numerical differentiation*, Appl. Math. Comput. **199** (2008), no. 1, 368–374, DOI: https://doi.org/10.1016/j.amc.2007.09.066.

[3] D. A. Murio, C. E. Mejia, and S. Zhan, *Discrete mollification and automatic numerical differentiation*, Comput. Math. Appl. **35** (1998), no. 5, 1–16, DOI: https://doi.org/10.1016/S0898-1221(98)00001-7.

[4] M. Hanke and O. Scherzer, *Inverse problems light: numerical differentiation*, Amer. Math. Monthly **108** (2001), no. 6, 512–521, DOI: https://doi.org/10.2307/2695705.

[5] C. W. Groetsch, *Lanczos' generalized derivative*, Amer. Math. Monthly **105** (1998), no. 4, 320–326, DOI: https://doi.org/10.2307/2589707.

[6] D. Y. Liu, O. Gibaru, and W. Perruquetti, *Differentiation by integration with Jacobi polynomials*, J. Comput. Appl. Math. **235** (2011), no. 9, 3015–3032, DOI: https://doi.org/10.1016/j.cam.2010.12.023.

[7] I. Knowles and R. J. Renka, *Methods for numerical differentiation of noisy data*, Electron. J. Differ. Equ. Conf. **21** (2014), 235–246.

[8] M. F. Al-Jamal, A. K. Alomari, and M. S. Gockenbach, *Smoothing via elliptic operators with application to edge detection*, Inverse Probl. Sci. Eng. **26** (2018), no. 5, 657–676, DOI: https://doi.org/10.1080/17415977.2017.1336552.

[9] Z. Meng, Z. Zhao, D. Mei, and Y. Zhou, *Numerical differentiation for two-dimensional functions by a Fourier extension method*, Inverse Probl. Sci. Eng. **28** (2020), no. 1, 126–143, DOI: https://doi.org/10.1080/17415977.2019.1661410.

[10] A. A. Yahya, J. Tan, B. Su, K. Liu, and A. N. Hadi, *Image edge detection method based on anisotropic diffusion and total variation models*, J. Eng. **2019** (2019), no. 2, 455–460, DOI: https://doi.org/10.1049/joe.2018.5345.

[11] C. Brito-Loeza, R. Legarda-Saenz, and A. Martin-Gonzalez, *A fast algorithm for a total variation based phase demodulation model*, Numer. Methods Partial Differential Equations **36** (2020), no. 3, 617–636, DOI: https://doi.org/10.1002/num.22444.

[12] K. Atkinson and W. Han, *Theoretical Numerical Analysis: A Functional Analysis Framework*, Springer-Verlag, New York, 2009.

[13] D. Braess, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, Cambridge, 2007.

[14] R. C. McOwen, *Partial Differential Equations: Methods and Applications*, Prentice Hall, New Jersy, 2003.

[15] J. Canny, *A Computational Approach To Edge Detection*, IEEE Tran. Pattern Anal. Machine Intell. **8** (1986), no. 6, 679–698, DOI: https://doi.org/10.1109/TPAMI.1986.4767851.

Appendix

If the true boundary data $h = f\,|_{\partial\Omega}$ is known approximately by the function h^ε satisfying

$$\|h^\varepsilon - h\|_{H^{1/2}(\partial\Omega)} \le \varepsilon,$$

then smoothed data can be defined as follows:

$$y^{\delta,\varepsilon}(x) = W^\varepsilon(x) + \sum_{n=1}^{\infty}(f^\delta - W^\varepsilon, \varphi_n)_{L^2(\Omega)}\exp(-\lambda_n t)\varphi_n(x), \quad x \in \Omega,$$

where the function W^ε satisfies (in the weak sense) the boundary-value problems:

$$-\Delta u + u = 0 \text{ in } \Omega, \quad u = h^\varepsilon \text{ on } \partial\Omega.$$

To derive stability and convergence results, we set

$$y(x) = W(x) + \sum_{n=1}^{\infty}(f - W, \varphi_n)_{L^2(\Omega)}\exp(-\lambda_n t)\varphi_n(x), \quad x \in \Omega,$$

where W solves the boundary-value problems

$$-\Delta u + u = 0 \text{ in } \Omega, \quad u = h \text{ on } \partial\Omega.$$

From the definition of $H^{1/2}(\partial\Omega)$ (e.g., [12]), we see that

$$\|W^\varepsilon - W\|_{H^1(\Omega)} = \|h^\varepsilon - h\|_{H^{1/2}(\partial\Omega)} \le \varepsilon.$$

Using the aforementioned inequality and similar arguments as in the proofs of Lemmas 4.1 and 4.2, it is easy to conclude that

$$\|\nabla y^{\delta,\varepsilon} - \nabla y\|_{L^2(\Omega)} \le \frac{\delta + \varepsilon}{2\sqrt{t}} + \varepsilon, \quad \|\nabla y - \nabla f\|_{L^2(\Omega)} \le C_\beta t^\eta,$$

where C_β depends on f and W, provided $f - W \in X_\beta$. The last inequalities together with triangle inequality imply the main error result:

$$\|\nabla y^{\delta,\varepsilon} - \nabla f\|_{L^2(\Omega)} \le K\left(\frac{\delta + \varepsilon}{2\sqrt{t}} + \varepsilon + t^\eta\right).$$

Moreover, if we let $\varepsilon = \max\{\delta, \varepsilon\}$, and choose $t \sim \varepsilon^\gamma$ for some $\gamma \in (0, 2)$, then we obtain the convergence result

$$\|\nabla y^{\delta,\varepsilon} - \nabla f\|_{L^2(\Omega)} \to 0$$

as $\varepsilon \to 0$. The convergence rate is optimal when $\gamma = 2/(2\eta + 1)$, and in this case, we have

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} = O(\varepsilon^{2\eta/(2\eta+1)}).$$

We obtain the fastest convergence when $f - W \in X_\beta$ for some $\beta \ge 3/2$, in which case $\eta = 1$ and

$$\|\nabla y^\delta - \nabla f\|_{L^2(\Omega)} = O(\varepsilon^{2/3}),$$

provided that $t = C\varepsilon^{2/3}$ for some positive constant C.