Nonlinear Dynamics and Systems Theory, **26** (1) (2026) 69–78

Optimal Thermal Diffusivity via Deep Learning for Heat Equation Image Denoising

H. Kharro [1]*, S.M. Douiri [1] and M. Moumni [2]

[1] *IMIA Laboratory, A2MSDS Team, Department of Mathematics, Faculty of Sciences and Technics, Moulay Ismaïl University of Meknes, P.O. Box 509 Boutalamine, Errachidia 52000, Morocco.*

[2] *MAMCS Team, Department of Mathematics, Faculty of Sciences and Technics, Moulay Ismaïl University of Meknes, P.O. Box 509 Boutalamine, Errachidia 52000, Morocco.*

Received: December 16, 2024; Revised: January 27, 2026

Abstract: Modern cameras inevitably introduce noise into images, which impacts their visual quality. As a result, various noise reduction strategies are necessary. Researchers have proposed numerous techniques for reducing noise, including approaches based on the linear and nonlinear partial differential equations. The choice of parameter values in partial differential equations plays a significant role in image denoising. Accurate tuning of these parameters can balance noise reduction and detail preservation, leading to higher quality denoised image. On the other hand, misadjusted parameters can result in either excessive smoothing or insufficient noise removal. Given these reasons, in this paper, we will concentrate on denoising the image using the heat equation and aim to identify the optimal thermal diffusivity value by solving a nonlinear inverse problem, which will allow us to achieve the best possible image denoising results. Finally, for the numerical experiments, we will employ deep learning and the Physics-Informed Neural Networks method to find this optimal value.

Keywords: *image denoising; heat equation; heat inverse problem; deep learning; physics-informed neural networks.*

Mathematics Subject Classification (2020): 68U10, 35K05, 34A34, 34A55, 65D15, 68T07.

* Corresponding author: `mailto:hafida.kharro10@gmail.com`

1 Introduction

One of the main challenges in computer vision and image processing is image denoising, which aims to estimate the original image by reducing noise from the contaminated version. Intrinsic and extrinsic conditions, such as sensor and environmental factors, can lead to image noise, which is often unavoidable in practical situations. Image denoising is quite important in many different applications, including image registration, image restoration, image classification and image segmentation, where reclaiming the original image content is essential for achieving high performance in these fields. Although several algorithms have been put forth for denoising images, the challenge of suppressing image noise persists, especially when images are captured under suboptimal conditions with high noise levels.

Generally, there are two typical methods for reducing image noise. One approach involves acquiring the data multiple times and averaging it, but this results in an extended acquisition time. Another method is to use post-processing techniques to reduce noise in the image. Numerous techniques have been proposed in the literature for image denoising [1]; these include the classic spatial and temporal filters, the nonlocal means algorithm, methods using anisotropic diffusion filters, bilateral and trilateral filters, the transformations of the curvelet and the contourlet, the wavelet transform, deep learning based methods, the maximum likelihood approach, and models based on partial differential equations (PDEs) [2, 3].

Recently, there has been a surge of interest in segmentation schemes based on PDEs, due to their many advantages, such as the reduction in computational complexity and their simplicity, since the problem reduces to the form of a PDE solution, which could be solved by iterative methods like the Finite Difference Method [4, 5]. One of the most widely used PDEs is the heat diffusion equation, which is defined as

$$\begin{cases} v_t(x,y,t) - \rho\Delta v(x,y,t) = 0 & \text{in } \mathcal{D}\times J, \\ v(x,y,0) = v_0 & \text{on } \mathcal{D}, \\ v(x,y,t) = 0 & \text{on } \partial\mathcal{D}\times J, \end{cases} \tag{1}$$

where $\mathcal{D} \subset \mathbb{R}^2$ is a bounded domain, $J = [0, T]$ is the interval of time with $T > 0$ and ρ represents the thermal diffusivity. The solution of this equation (1) can be represented as a convolution with a Gaussian function $G_\sigma(x, y)$ [6], specifically

$$v(x,y,t) = G_\sigma * v_0(x,y).$$

Instead of using the original image's classical convolution with G_σ, it is possible to solve the linear heat equation using the original image as the initial condition.

One of the biggest challenges when trying to denoise an image is the optimal choice of ρ value. As shown in Figure 1, each choice of ρ results in different denoising, and it is clear that the image is denoised better by $\rho = 0.8$ than by $\rho = 0.2$ since the PSNR value in the first case is equal to 25.50 and in the second case to 21.51, The question remains whether we can find the value that provides the best denoising. Indeed, searching for these optimal values often leads to inverse problems [7], which can be quite challenging in mathematics, and requires careful consideration and analysis to find the best solution. One of the techniques used in solving an inverse problem is the Physics-Informed Neural Networks (PINNs) method [8]. Recently, PINNs have appeared as a simple alternative method to solve many problems in engineering and computational science. Specifically, they do not need meshes and may effectively resolve forward problems and ill-posed inverse problems [9, 10], which are typically challenging or unfeasible to resolve by employing conventional numerical techniques.

The primary advantage of the PINNs method is its capability to smoothly integrate all provided information, including initial/boundary conditions, experimental data, and governing equations, into the loss function. This effectively converts the original problem into an equivalent optimization problem. In the specific case of denoising using the heat equation, PINNs can learn the optimal value of ρ by minimizing the difference between the original and predicted denoised images. This approach can significantly reduce the computational cost and time required for denoising. The primary aim of this paper is to determine the optimal ρ to achieve the best image denoising using the PINNs method.

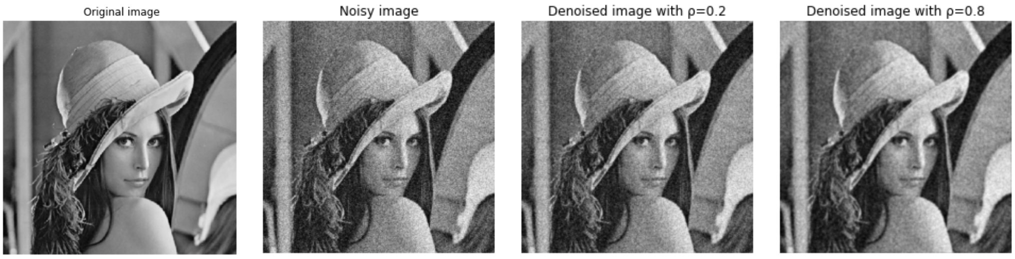

Figure 1: Image denoising using heat equation with various ρ values.

This paper is structured into several sections, which are outlined below. In Section 2, we demonstrate the existence of ρ. In Section 3, we use deep learning and the PINNs method to solve the heat inverse problem and find the optimal ρ. In Section 4, we perform a series of numerical experiments. The paper concludes with a summarization of key insights and findings.

2 Heat Inverse Problem

In this part, we are interested in the theoretical existence of ρ. The primary elements introduced in this section are Lemma 2.1 [11] and Theorem 2.1 [12,13]. Then we consider the problem (1) and we frequently receive the terminal condition observation given by

$$v(x,T) = r(x) \qquad \forall x \in \mathcal{D},$$

where $\mathcal{D}$ represents a bounded domain in $\mathbb{R}^d (d \geq 1)$ with a boundary Π that is piecewise smooth.

To establish the existence of ρ, we will require the following space:

$$GL(\mathcal{D}) = \{\rho \in L^1(\mathcal{D}); ||\rho||_{GL(\mathcal{D})} < \infty\},$$

where

$$||\rho||_{GL(\mathcal{D})} = ||\rho||_{L^1(\mathcal{D})} + \int_{\mathcal{D}} |D\rho|,$$

and

$$\int_{\mathcal{D}} |D\rho| = sup\Big\{ \int_{\mathcal{D}} \rho \, \mathrm{div} g dx;\ g \in (C_0^1(\mathcal{D}))^d \,;\, |g(x)| \leq 1 \quad \text{in}\, \mathcal{D}\Big\}.$$

We define the parameter identification problem as the subsequent constrained minimization process

$$\min_{\rho\in Q} J(\rho) = \frac{1}{2}\int_{\mathcal{D}} \rho|\nabla(w(\rho;T) - r)|^2 dx + \gamma P(\rho), \tag{2}$$

where $w = w(\rho; t) \in H_0^1(\mathcal{D})$ satisfying

$$w(x, 0) = v_0(x), \tag{3}$$

and for a.e $t \in (0, T)$,

$$\int_{\mathcal{D}} w_t\phi dx + \int_{\mathcal{D}} \rho\nabla w\nabla\phi dx = 0 \qquad \forall\phi \in H_1^0(\mathcal{D}). \tag{4}$$

We denote the solution to the variational problem (3) and (4) as $w(\rho; t)$ or $w(\rho)$. In (2), the function $r \in H_0^1(\mathcal{D})$ represents the measured data, and $P(\rho)$ serves as a regularization term with a weighting coefficient $\gamma > 0$. Specifically,

$$P(\rho) = \int_{\mathcal{D}} |D\rho|$$

defines the semi-norm in the GL-space or in $H^1(\mathcal{D})$. The Q is a subset of $H^1(\mathcal{D})$ or $GL(\mathcal{D})$ and is defined by

$$Q = \big\{\rho \in L^1(\mathcal{D});\ |||\rho||| < \infty \text{ and } \beta_1 \le \rho \le \beta_2 \text{ a.e in } \mathcal{D}\big\},$$

where $|||\rho||| = ||\rho||_{H^1(\mathcal{D})}$ or $|||\rho||| = ||\rho||_{GL(\mathcal{D})}$, β_1 and β_2 are two positive constants.

It is essential to recognize that evaluating the cost functional $J(\rho)$ necessitates having the terminal state value of the solution $w(\rho; t)$ for the system described by (3) and (4) at $t = T$. This implies the regularity $w \in C(0, T; H_0^1(\mathcal{D}))$.

For the purposes of this analysis, we will consider the following assumption about the initial data for the problem (1):

$$v_0 \in H^1(\mathcal{D}). \tag{5}$$

Assuming (5), classical parabolic theory assures that for every $\rho \in Q$, there exists a unique solution $w(\rho; t)$ to the parabolic problem, which corresponds to the variational problem defined in (3) and (4). This solution exhibits the following regularity properties:

$$w(\rho) \in L^2(0, T; H_0^1(\mathcal{D})), \quad w(\rho) \in H^1(0, T; L^2(\mathcal{D})), \quad w(\rho) \in C(0, T; L^2(\mathcal{D})).$$

Instead of using the system described in (2)-(4), we will adopt a simpler formulation as follows:

$$\min_{\rho\in Q} J(\rho) = \frac{1}{2}\int_{T-\sigma}^{T}\int_{\mathcal{D}} \rho|\nabla(w(\rho;t) - r)|^2 dxdt + \gamma P(\rho), \tag{6}$$

where σ is a small constant number and $w = w(\rho; t) \in H_0^1(\mathcal{D})$ satisfying

$$w(x, 0) = v_0(x) \quad \text{in } \mathcal{D}, \tag{7}$$

$$\int_{\mathcal{D}} w_t\phi dx + \int_{\mathcal{D}} \rho\nabla w\nabla\phi = 0 \qquad \forall\phi \in H_0^1(\mathcal{D}), \tag{8}$$

for a.e, $t \in (0, T)$.

We will now show that the problem (6)-(8) has a minimizer by employing the following lemma.

Lemma 2.1 *[11] Let $\{\rho_m\}$ be a sequence in Q that converges to $\rho \in L^1(\mathcal{D})$ as $m \to \infty$, then we have*

$$\lim_{m\to\infty} \int_{T-\sigma}^{T} \int_{\mathcal{D}} \rho_m |\nabla(w(\rho_m) - r)|^2 dxdt = \int_{T-\sigma}^{T} \int_{\mathcal{D}} \rho |\nabla(w(\rho) - r)|^2 dxdt.$$

Using Lemma 2.1, we obtain the following result.

Theorem 2.1 *[12, 13] For the optimization problem* (6)-(8)*, there is at least one minimizer.*

3 Using Deep Learning for Solving the Heat Inverse Problem and Finding ρ Optimal

We use heat equation (1) to denoise the image, knowing that the solution to the equation is the denoised image. The noisy image is denoted as v_0, the original image as v_T, and the Laplacian Δv is defined as the convolution product between the noisy image and the kernel I, which is defined as

$$I = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

The choice of ρ is very important, so we will look for an optimal ρ which is a solution to the heat inverse problem (1), and to solve it we use PINNs. To define the algorithm for PINNs, we can refer to [14].

Algorithm 3.1 *PINNs for Solving the Inverse Problem*

1. *Create a neural network (NN) with parameter ρ, initialized randomly.*
2. *Identify the training datasets.*
3. *Define the loss function as the weighted sum of the L^2 norms of: the initial condition residual, the PDE residual, and the final-time solution residual.*
4. *Train the NN to optimize the parameter ρ by minimizing the loss function.*

First, a NN denoted by $\hat{v}(X;\theta;\rho)$ with $X = (x, y, t)$ is constructed as a surrogate of the solution of the problem (1) $v(x, y, t)$, and it takes the coordinates (x, y, t) as the inputs, and outputs a vector that has the same dimension as v. Here, the NN parameters that will be tuned at the training stage are denoted by θ, namely, θ contains all of the biases b and weights w, and ρ is a specific parameter representing thermal diffusivity. One benefit of employing PINNs, where neural networks are utilized as approximators for v, is that it enables us to compute the derivatives of $\hat{v}$ with respect to its input X. This is achieved through the chain rule for differentiating function compositions, supported by automatic differentiation tools [14] that are commonly integrated into machine learning packages such as PyTorch and TensorFlow. In the following step, it is necessary to ensure that the neural network $\hat{v}$ conforms to the physical constraints dictated by the PDE and the boundary/initial conditions. Practically, this is accomplished by constraining $\hat{v}$ at a series of scattered points, which serve as the training data $\tau = \{X_1, X_2, \ldots, X_\tau\}$ with a size of $|\tau|$.

To evaluate the disparity between the constraints and the neural network $\hat{v}$, we define the loss function as

$$\chi(\theta;\rho) = ||v_t - q\Delta v||_{L^2} + ||v - v_0||_{L^2} + ||v - v_T||_{L^2}.$$

In the last step, we employ gradient-based optimization methods, including Adam, to minimize the loss function and find the optimal parameters θ (θ^*) and ρ (ρ^*). Refer to Figure 2 for a visual representation of a neural network.

Consider the noisy image as a given gray-scale intensity map $v_0 : \mathcal{D} \to [0, 255]$ for the image domain $\mathcal{D} \in \mathbb{R}^2$. The solution of the problem (1) denotes the restored image, and v_T is the original image. We can find ρ^* by following Algorithm 3.2.

Algorithm 3.2 *Finding ρ^* using deep learning*

1. *Read the image from a file using an image processing library (e.g., OpenCV, PIL, or skimage).*
2. *Add noise to the image.*
3. *Discretize a two-dimensional spatial domain (x, y) and a temporal domain (t) using* ***linspace*** *from the NumPy library.*
4. *Generate a meshgrid of coordinates X and Y from given 1D arrays x and y, and define initial time (t_0) and final time (t_f).*
5. *Define X_{star}, X_{starf}, and X_{star0} using* ***hstack().***
6. *Define the function* ***heat_loss****, which takes three parameters:* ***v****, ρ, and X_{star}, and returns the squared sum of $v_t - \rho\nabla v$.*
7. *Define the function* ***heat_loss0****, which takes three parameters:* ***v****, ρ, and X_{star0}, and returns the squared sum of the difference between the noisy image and the neural network's output.*
8. *Define the function* ***heat_lossT****, which takes three parameters:* ***v****, ρ, and X_{starf}, and returns the squared sum of the difference between the original image and the neural network's output.*
9. *Define a total loss function as the sum of* ***heat_loss****,* ***heat_loss0****, and* ***heat_lossT****.*
10. *Construct a neural network (NN) with parameter ρ.*
11. *Train the model and optimize the loss function using the Adam optimizer.*

4 Numerical Results

In this part, we perform our experiments using the well-known images Lena.png, Barbara.png and Mandrill.png noised with Gaussian additive noise ($\sigma = 50$). All tests were performed with PYTHON.
We use Algorithm 3.2 for three images Lena.png, Barbara.png, and Mandrill.png. The hyperparameters used are listed in Table 1. After determining ρ^*, the obtained values are presented in Table 2.

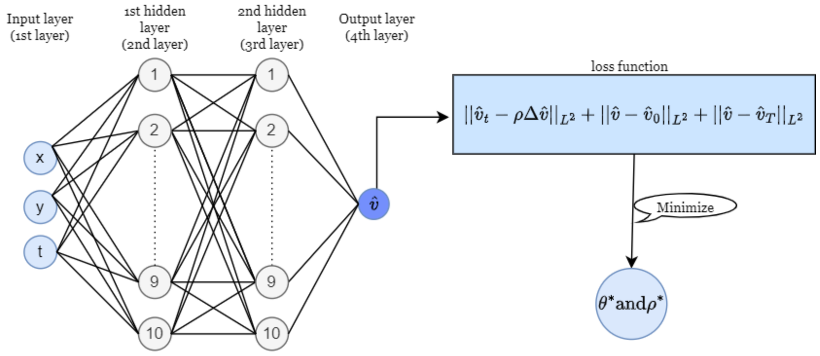

Figure 2: PINNs schematic for solving the heat inverse problem.

Activation function	Optimizer	Learning rate
tanh	Adam	0.0001

Table 1: Hyperparameters used for all experiments.

Image	Lena.png	Barbara.png	Mandrill.png
ρ^*	0.9634	0.9360	0.9752

Table 2: Optimal ρ values (ρ^*) for various images.

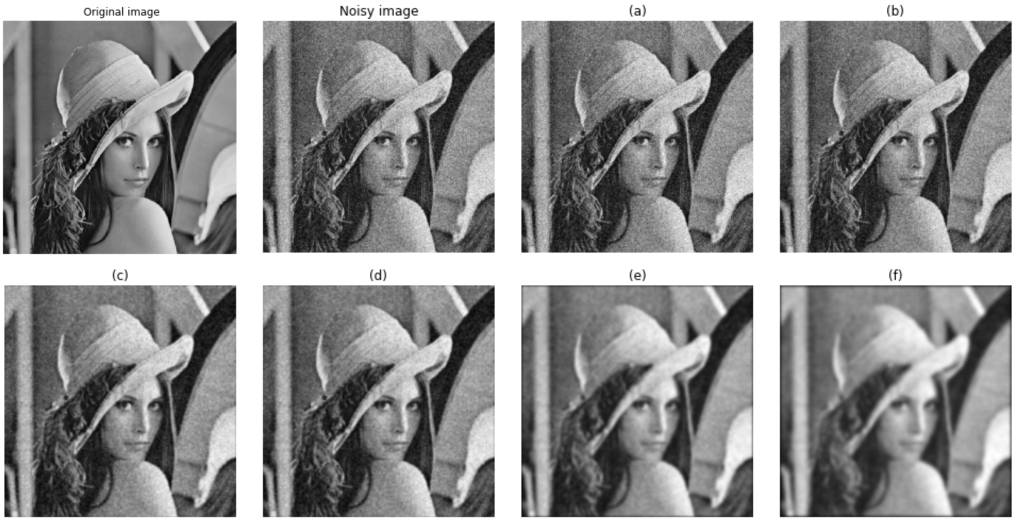

Figure 3: Denoising results for Lena.png image with various ρ values: (a) $\rho = 0.01$, (b) $\rho = 0.1$, (c) $\rho = 0.9634$, (d) $\rho = 1$, (e) $\rho = 4$, (f) $\rho = 8$.

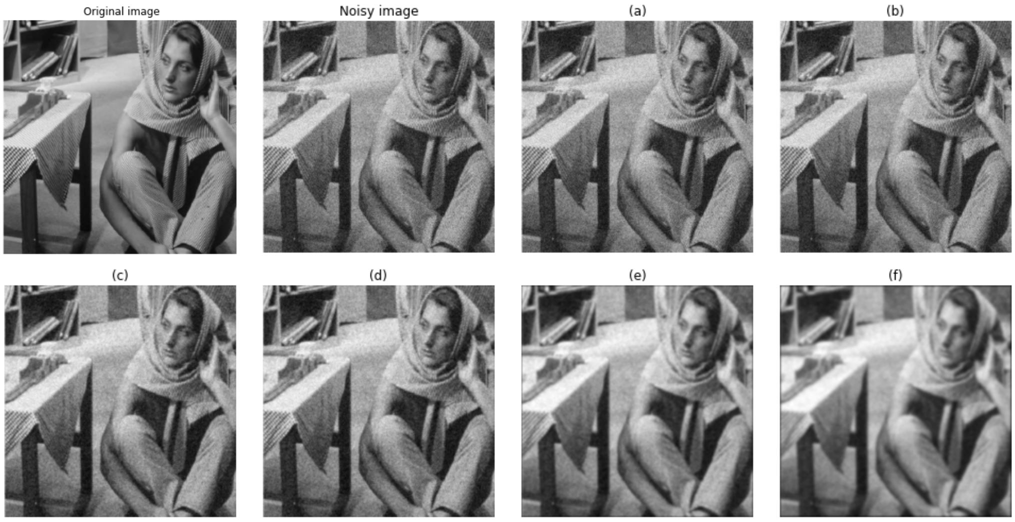

Figure 4: Denoising results for Barbara.png image with various ρ values: (a) $\rho = 0.01$, (b) $\rho = 0.1$, (c) $\rho = 0.9360$, (d) $\rho = 1$, (e) $\rho = 4$, (f) $\rho = 8$.

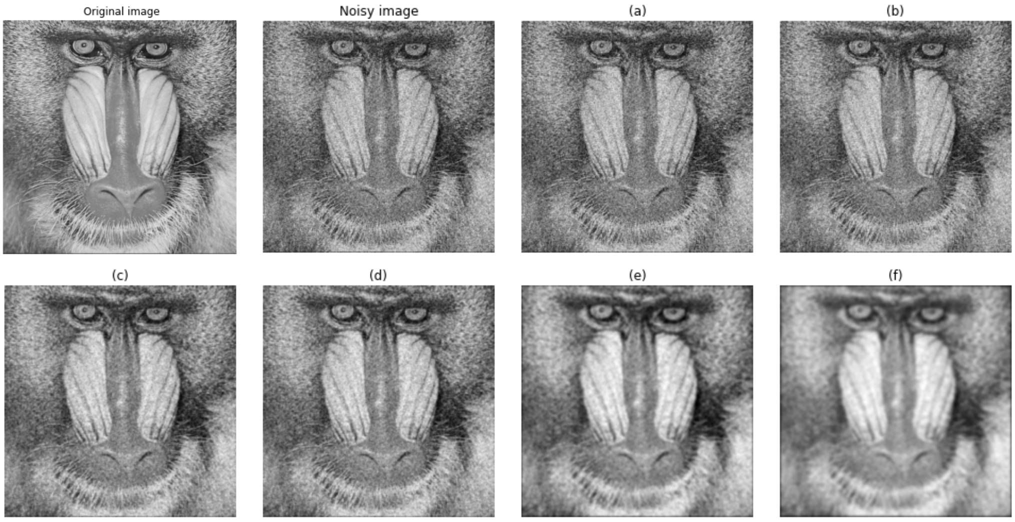

Figure 5: Denoising results for Mandrill.png image with various ρ values: (a) $\rho = 0.01$, (b) $\rho = 0.1$, (c) $\rho = 0.9752$, (d) $\rho = 1$, (e) $\rho = 4$, (f) $\rho = 8$.

Then restore the image using the heat equation with Δt equal to 0.001. The denoising results are illustrated in Figures 3, 4 and 5.

To evaluate image quality and perform comparisons, we will utilize the widely-used metric, peak-signal-to-noise ratio (PSNR) [15], which is defined as

$$PSNR = 10 \log_{10} \left(\frac{MN \times 255^2}{\sum_{i,j} (l_{ij} - m_{ij})^2} \right),$$

where $M \times N$ represents the image dimensions, and m_{ij} and l_{ij} are the values of the pixels of the restored and original images, respectively.

Tables 3, 4, 5 illustrate the PSNR values for each ρ value for the three images, and we observe that the highest PSNR value is the value corresponding to ρ^*.

ρ	0.01	0.1	0.7	0.8	**0.9634**	1	2	4	8
PSNR	15.26	18.80	25.40	25.48	**25.5079**	25.4934	24.71	23.21	21.52

Table 3: PSNR (dB) of different ρ values for Lena.png image.

ρ	0.01	0.1	0.7	0.8	**0.9360**	1	2	4	8
PSNR	10.07	18.58	25.59	25.70	**25.7434**	25.7341	25.01	24.28	23.45

Table 4: PSNR (dB) of different ρ values for Barbara.png image.

ρ	0.01	0.1	0.7	0.8	**0.9752**	1	2	4	8
PSNR	14.73	16.50	21.31	21.42	**21.4819**	21.4813	21.08	20.48	19.84

Table 5: PSNR (dB) of different ρ values for Mandrill.png image.

5 Conclusion

In this study, we investigated the use of the heat equation for image denoising and demonstrated the existence of the optimal value of the thermal diffusivity ρ by solving a nonlinear inverse problem. We also introduced a novel algorithm that combines deep learning and Physics-Informed Neural Networks to determine the optimal ρ for each image. To demonstrate the effectiveness of our proposed method, we conducted a numerical simulation and compared the denoising results for three images using the optimal $\rho^\star$ and various other ρ values. Then, in Tables 3, 4 and 5, we present the PSNR values for each ρ value, highlighting that the highest PSNR value occurs at the optimal $\rho^\star$. The results of our study show that the optimal choice of thermal diffusivity $\rho^\star$ plays a crucial role in achieving the best image denoising performance. The combination of deep learning techniques and Physics-Informed Neural Networks proves to be an effective and innovative approach to determining this optimal value. Our numerical experiments demonstrate that this method outperforms traditional techniques and offers a promising solution for image denoising in various practical applications. For future research, we would like to work with other types of noise, and we will focus on other models based on PDEs.

References

[1] A. Buades and B. Coll and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale modeling and simulation* **4** (2) (2005) 490–530.

[2] B. Kawar and M. Elad and S. Ermon and J. Song. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems* **35** (2022) 23593–23606.

[3] S. M. Douiri and H. Farjil and M. Moumni. Improved Perona-Malik Model with a Leray-Lions Operator for Image Denoising. *Bol. Soc. Paran. Mat* **43** (2025) 1–14.

[4] A. Z. Arifin and M. I. Joesidawati and T. Herlambang and S. Mizan and A. A. Suryanto. Simulation of Tsunami Wave Propagation Using the Finite Difference Method for Disaster Early Warning System. *Nonlinear Dynamics and Systems Theory* **25** (1) (2025).

[5] M. Khumalo and A. Dlamini. The Finite Element Method for Nonlinear Nonstandard Volterra Integral Equations. *Nonlinear Dynamics and Systems Theory* **20** (2) (2020) 191–202.

[6] Q. D. Ho and H. T. Huynh. An Image Denoising Model Based on Nonlinear Partial Diferential Equation Using Deep Learning. *Future Data and Security Engineering. Big Data, Security and Privacy, Smart City and Industry 4.0 Applications, Springer Nature* **1688** (2022) 407–418.

[7] A. Benguesmia and I. M. Batiha and T. E. Oussaeif and A. Ouannas and W. G. Alshanti. Inverse problem of a semilinear parabolic equation with an integral overdetermination condition. *Nonlinear Dynamics and Systems Theory* **23** (2) (2023) 249–260.

[8] M. Raissi and P. Perdikaris and G. E. Karniadakis. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* **378** (2019) 686–707.

[9] A. D. Jagtap and E. Kharazmi and G. E. Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering* **365** (2020) 113028.

[10] M. Moumni and M. Tilioua. A neural network approximation for a model of micromagnetism. *Nonlinear Dynamics and Systems Theory* **22** (4) (2022) 432–446.

[11] Y. L. Keung and J. Zou. Numerical identifications of parameters in parabolic systems. *Inverse Problems* **14** (1) (1998) 83.

[12] Y. L. Keung and J. Zou. Identifying Parameters in Parabolic Systems by Finite Element Methods. *Technical Report 97–10 126*, Department of Mathematics, The Chinese University of Hong Kong, (1997).

[13] K. Kunisch and W. Ring. Regularization of nonlinear illposed problems with closed operators. *Numerical functional analysis and optimization* **14** (3–4) (1993) 389–404.

[14] L. Lu and X. Meng and Z. Mao and G. E. Karniadakis. DeepXDE: A Deep Learning Library for Solving Differential Equations. *SIAM review* **63** (1) (2021) 208–228.

[15] S. I. Hong and K. R. Dong and Y. H. Ryu. SNR and PSNR measurements and analysis of median filtering for the removal of impulse noise from CR imaging. *International Journal of Contents* **5** (4) (2009) 7–12.