
1 MATHEMATICAL FOUNDATIONS

1.1 FUNCTIONAL ANALYSIS

Definition 1.1 (Lower Semicontinuity). *A functional $F : X \rightarrow \mathbb{R} \cup \{+\infty\}$ is lower semicontinuous at u_0 if $F(u_0) \leq \liminf_{n \rightarrow \infty} F(u_n)$ for every sequence $\{u_n\}$ converging to u_0 .*

Definition 1.2 (Coercivity). *A functional $F : X \rightarrow \mathbb{R} \cup \{+\infty\}$ is coercive if $F(u) \rightarrow +\infty$ as $\|u\|_X \rightarrow \infty$.*

Theorem 1.3 (Direct Method). *Let $F : X \rightarrow \mathbb{R} \cup \{+\infty\}$ be a proper functional on a Banach space X . If F is coercive and sequentially lower semicontinuous in a topology where bounded sets are sequentially compact, then F attains its minimum.*

1.2 BV SPACE THEORY

Definition 1.4 (Functions of Bounded Variation). *A function $u \in L^1(\Omega)$ belongs to $BV(\Omega)$ if*

$$TV(u) = \sup \left\{ \int_{\Omega} u \operatorname{div} \varphi \, dx : \varphi \in C_c^1(\Omega; \mathbb{R}^n), \|\varphi\|_{\infty} \leq 1 \right\} < \infty$$

Theorem 1.5 (BV Compactness). *If $\{u_n\} \subset BV(\Omega)$ with $\|u_n\|_{BV} \leq M$, then there exists a subsequence and $u \in BV(\Omega)$ such that $u_{n_k} \rightarrow u$ in $L^p(\Omega)$ for all $p < \frac{n}{n-1}$.*

Theorem 1.6 (Lower Semicontinuity of TV). *The total variation functional is lower semicontinuous with respect to $L^1(\Omega)$ convergence.*

1.3 GAMMA-CONVERGENCE

Definition 1.7 (Γ -Convergence). *A sequence $\{F_n\}$ Γ -converges to F if for every $u \in X$:*

1. (liminf inequality) *For every $u_n \rightarrow u$: $F(u) \leq \liminf_{n \rightarrow \infty} F_n(u_n)$*
2. (recovery sequence) *There exists $u_n \rightarrow u$ with $F(u) = \lim_{n \rightarrow \infty} F_n(u_n)$*

Theorem 1.8 (Chambolle-Lions). *The sequence $\{E_{\varepsilon}\}$ with smooth approximations ϕ_{ε} Γ -converges to the relaxed TV functional as $\varepsilon \rightarrow 0$. Combined with equi-coercivity, minimizers u_{ε} converge to the TV minimizer in $L^1(\Omega)$.*

1.4 NEURAL NETWORK APPROXIMATION

Theorem 1.9 (Meyers-Serrin). *$C^{\infty}(\Omega) \cap W^{1,2}(\Omega)$ is dense in $W^{1,2}(\Omega)$ for bounded domains.*

Theorem 1.10 (Universal Approximation in Sobolev Spaces). *Neural networks with smooth non-polynomial activations are dense in $W^{1,2}(\Omega)$ on bounded domains. Specifically, they can approximate both function values and derivatives simultaneously in the Sobolev norm.*

1.5 EULER-LAGRANGE EQUATIONS

Proposition 1.11 (Optimality Conditions for E_{ε}). *The minimizer u_{ε} of $E_{\varepsilon}(u) = \frac{1}{2}\|Ru - u_0\|^2 + \lambda \int_{\Omega} \phi_{\varepsilon}(|\nabla u|) dx$ satisfies:*

$$R^*(Ru - u_0) - \lambda \operatorname{div} \left(\phi'_{\varepsilon}(|\nabla u|) \frac{\nabla u}{|\nabla u|} \right) = 0$$

2 IMPLEMENTATION DETAILS

2.1 PINN IMPLEMENTATION DETAILS

2.1.1 NETWORK ARCHITECTURE AND TRAINING CONFIGURATION

For single-image denoising experiments, we employ fully connected networks with configurations tailored to the problem dimension:

Parameter	2D (MNIST)	3D (Heart Cone)
<i>Model Architecture</i>		
Network type	Fully connected	Fully connected
Number of layers	4	4
Hidden units per layer	256	512
Activation function	SiLU	SiLU
<i>Training</i>		
Training epochs	500	300
Batch size	32	50
Learning rate	1×10^{-4}	5×10^{-4}
<i>Regularization</i>		
TV weight (λ)	1×10^{-3}	4×10^{-3}
<i>Dataset</i>		
Noise std (σ)	0.5 ($\sigma^2 = 0.25$)	0.2
Data dimensions	28×28	$64 \times 64 \times 64$

Table 1: PINN hyperparameters for 2D and 3D denoising experiments.

2.1.2 LOSS FUNCTION AND TRAINING PROCEDURE

For both 2D and 3D experiments, we minimize the smoothed functional:

$$E_\varepsilon(u_\theta) = \frac{1}{2} \int_{\Omega} (u_\theta - f_{\text{noisy}})^2 d\Omega + \lambda \int_{\Omega} \phi_\varepsilon(|\nabla u_\theta|) d\Omega$$

where $\Omega = [0, 1]^d$ and ϕ_ε is the smoothed TV functional with progressively reduced ε during training.

Coordinate sampling differs by dimension:

- 2D: Coordinates $(\frac{i-1}{27}, \frac{j-1}{27})$ for $i, j \in \{1, \dots, 28\}$
- 3D: Coordinates $(\frac{i-1}{63}, \frac{j-1}{63}, \frac{k-1}{63})$ for $i, j, k \in \{1, \dots, 64\}$

The network takes these normalized coordinates as input and outputs the denoised intensity value at that location. During training, we progressively reduce ε following Algorithm ?? to encourage sharper edge preservation while maintaining stability.

2.2 DEEPONET IMPLEMENTATION DETAILS

2.2.1 NETWORK ARCHITECTURES

For the experiments presented in this work, we use the following DeepONet architecture:

Branch Network: The branch network encodes the noisy input image using fully connected layers:

- Architecture: 4 fully connected layers
- Hidden dimension: 7500 units per layer
- Activation: ReLU (no gradient computation required with respect to input)
- Output dimension: 7500

Trunk Network: The trunk network maps spatial coordinates to features:

- Input dimension: 2 for 2D images, 3 for 3D volumes
- Architecture: 4 fully connected layers
- Hidden dimension: 5500 units per layer
- Activation: SiLU (smooth activation for computing divergence terms)
- Output dimension: 7500

The combined output is computed as:

$$G_\theta(u_0)(\mathbf{x}) = \sum_{i=1}^{7500} B_i(u_0) \cdot T_i(\mathbf{x})$$

2.2.2 TRAINING CONFIGURATION

Parameter	Value
<i>Regularization</i>	
TV weight (λ)	4.3×10^{-4}
<i>Training</i>	
Epochs	350
Batch size	20
Optimizer	AdamW
Initial learning rate	10^{-4}
LR scheduler	MultiStepLR
Schedule milestones	[30, 35, 40, 50, 60, 70, 100, 250]
Schedule gamma	0.8
<i>ε-reduction</i>	
Initial ε_0	0.5
Decay factor α	0.7
Minimum ε	10^{-3}

Table 2: Training hyperparameters for operator learning experiments.

2.2.3 LOSS FUNCTION

Following Algorithm ??, we minimize the physics-informed loss:

$$\mathcal{L}(\theta) = \frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M \left\| R^*(RG_\theta(u_0^{(i)}))(\mathbf{x}_j) - u_0^{(i)}(\mathbf{x}_j) - \lambda \mathcal{D}_\varepsilon[G_\theta(u_0^{(i)})](\mathbf{x}_j) \right\|^2$$

where N is the number of images in each batch, M is the number of sampled coordinates per image, and \mathcal{D}_ε is the divergence term from the Euler-Lagrange equation with smoothing parameter ε .

2.3 DATA GENERATION PROTOCOL

All experiments use synthetic noisy data generated as follows:

1. Start with clean images/volumes from standard datasets
2. Add i.i.d. Gaussian noise: $u_0 = u_{\text{clean}} + \eta$ where $\eta \sim \mathcal{N}(0, \sigma^2)$
3. Normalize to $[0, 1]$ range: $u_0 \leftarrow (u_0 - \min(u_0)) / (\max(u_0) - \min(u_0))$
4. For operator learning: all instances use the same σ value
5. Noise levels tested: $\sigma \in \{0.05, 0.1, 0.15, 0.2\}$ relative to normalized intensity

While we use clean images to generate noisy data for quantitative evaluation, the methods themselves never access the clean images during training, maintaining the unsupervised nature of the approach.

3 ADDITIONAL EXPERIMENTAL RESULTS

3.1 2D PINN DENOISING ON MNIST

We validate our PINN approach on 2D images using the MNIST dataset. A single digit (28x28 pixels) is corrupted with Gaussian noise ($\sigma = 0.5$, corresponding to $\sigma^2 = 0.25$) to test edge preservation in a simpler setting before scaling to 3D.

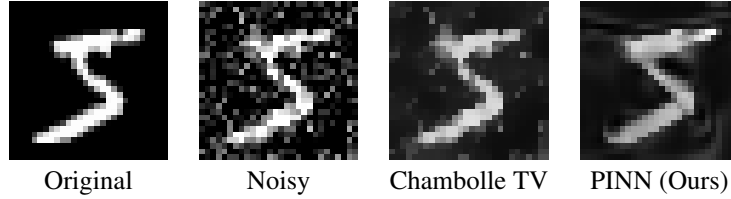


Figure 1: 2D denoising on MNIST digit with $\sigma^2 = 0.25$. Our PINN preserves sharp edges while effectively removing noise.

Method	RMSE \downarrow	PSNR \uparrow	SSIM \uparrow
Chambolle TV	0.022	16.40	0.626
PINN (Ours)	0.012	19.32	0.725

Table 3: Quantitative results for 2D MNIST denoising. TV uses optimal λ from 10 candidates.

3.2 2D OPERATOR LEARNING ON CHESTMNIST

We demonstrate operator learning on the ChestMNIST dataset, training a single DeepONet to denoise 100 chest X-ray images simultaneously. This experiment validates our operator approach on medical imaging data with varying anatomical structures and noise levels.

3.2.1 HIGH NOISE REGIME AT $\sigma^2 = 0.3$

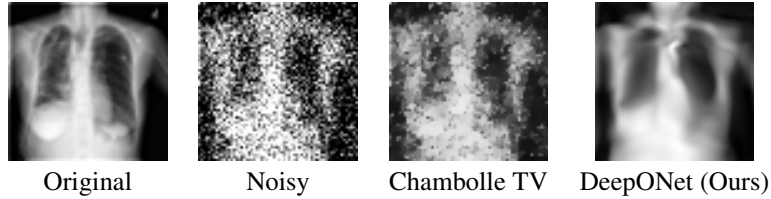


Figure 2: ChestMNIST denoising with $\sigma^2 = 0.3$. Our method maintains structural integrity under severe noise.

Method	RMSE \downarrow	PSNR \uparrow	SSIM \uparrow
Chambolle TV	0.010	18.1	0.560
DeepONet (Ours)	0.007	22.1	0.760

Table 4: Quantitative results on ChestMNIST with $\sigma^2 = 0.3$. Our operator shows robust performance under high noise.

The ChestMNIST experiments demonstrate robust operator performance across noise levels. At moderate noise ($\sigma^2 = 0.1$), our operator achieves marginal but consistent improvements. The advantages become dramatic at high noise ($\sigma^2 = 0.3$): 4 dB PSNR gain and 36

The operator uses a branch network (4 layers, 7500 hidden units, ReLU) to encode noisy images and a trunk network (4 layers, 5500 hidden units, SiLU) for spatial coordinates. Training with AdamW optimizer over 350 epochs takes approximately 2 hours for the full dataset. Once trained, the operator processes new chest X-rays in seconds, making it practical for clinical workflows requiring rapid batch processing.

These 2D experiments validate that our operator learning framework successfully generalizes across different anatomical structures and noise conditions, providing consistent improvements over classical TV methods while maintaining computational efficiency.

216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269

4 LLM USAGE

Large language models were used as assistive tools in the preparation of this paper. Claude was used to help refine mathematical exposition, improve paragraph flow, improving grammar, refining sentence and paragraph structure and strengthening the overall quality of writing and ensure consistency in notation throughout the paper.