IMPROVED TRAINING OF PHYSICS-INFORMED NEURAL NETWORKS USING ENERGY-BASED PRIORS: A STUDY ON ELECTRICAL IMPEDANCE TOMOGRAPHY

Anonymous authors
Paper under double-blind review

ABSTRACT

Physics-informed neural networks (PINNs) are attracting significant attention for solving partial differential equation (PDE) based inverse problems, including electrical impedance tomography (EIT). EIT is non-linear and especially its inverse problem is highly ill-posed. Therefore, successful training of PINN is extremely sensitive to interplay between different loss terms and hyper-parameters, including the learning rate. In this work, we propose a Bayesian approach through data-driven energy-based model (EBM) as a prior, to improve the overall accuracy and quality of tomographic reconstruction. In particular, the EBM is trained over the possible solutions of the PDEs with different boundary conditions. By imparting such prior onto physics-based training, PINN convergence is expedited by more than ten times faster to the PDE's solution. Evaluation outcome shows that our proposed method is more robust for solving the EIT problem.

1 INTRODUCTION

Physics-informed neural networks (PINNs) (Raissi et al., 2019) parameterize the solution of a partial differential equation (PDE) using a neural network and trains the neural network to predict the solution's scalar value for any given point inside the problem's domain by minimizing the residual PDE and associated boundary conditions (BCs). Various other factors such as gradient stiffness (Wang et al., 2021) and complex parameter settings, lack of constraints and regularization (Krishnapriyan et al., 2021), often cause major issues in training PINNs and makes them very sensitive to hyper-parameters and regularization. For example, depending on the PDE, the interplay between BCs residuals and PDEs residuals can result in invalid solutions that mostly satisfy one type of constraints over the others. In general, we believe that some of these problems are attributable to the lack of joint representation in PINNs, as these models are assumed to implicitly learn that two points in a close vicinity possibly have a similar solution. However, this representation is hard-coded in the numerical methods as they update the solution value of each mesh point based on the neighboring mesh points.

In this work, we augment PINNs with a joint representation via a Bayesian approach. We train a data-driven prior over joint solutions of PDEs on the entire domain and boundary points. This prior relates the predictions of PINNs via explicit joint representation and encourage learning a coherent and valid solution. Priors have been studied extensively in statistical approach to solve inverse problems (Kaipio et al., 2000; Ahmad et al., 2019; Abhishek et al., 2022; Strauss & Khan, 2015) and also recently have been used in data-driven approaches as well (Ramzi et al., 2020). More importantly, we focused on the non-linear and ill-posed electrical impedance tomography (EIT) inverse problem and experimentally show that using our data-driven Bayesian approach results in accurate, fast, and more robust training algorithms. We have the following contributions:

- We present various experimental studies which circumvent instability in training PINNs for the EIT problem.
- We introduce a data-driven prior in the form of an EBM for improving the training convergence and accuracy of PINNs.
- We propose a robust training framework for the EIT semi-inverse problem.

2 AN INVERSE PROBLEM WITH APPLICATION TO ELECTRICAL IMPEDANCE TOMOGRAPHY (EIT)

The inverse problem that we are interested in is inspired by applications in imaging paradigms such as electrical impedance tomography (EIT) and geophysical imaging for ground water flow. The EIT inverse problem reconstructs the unknown electrical conductivities σ of a body $\Omega \subset \mathbb{R}^d$ with $d \in \{2, 3\}$ from measurements of finite electrical potential differences of neighboring surface electrodes. The differential equation

$$-\nabla \cdot \sigma \nabla u = 0 \qquad \text{in } \Omega \tag{1}$$

governs the distribution of electric potential u in the body. Additionally, its accompanying BCs are given as following:

$$\begin{aligned} \sigma \left(\frac{\partial u}{\partial n} \right) &= g \qquad && \text{on } \partial\Omega \ \text{ Neumann BC}, \\ u &= f && \text{on } \partial\Omega \ \text{ Dirichlet BC} \end{aligned} \tag{2}$$

An EIT experiment involves applying an electrical current on the surface $\partial\Omega$ of the region Ω to be imaged, which produces a current density $\sigma \frac{\partial u}{\partial n}|_{\partial\Omega} = g$ (Neumann data) where n is a unit normal vector w.r.t u associated with Ω at its boundary. The current also induces an electric potential u in the body, whose surface value $u|_{\partial\Omega} = f$ can be measured. Thus by repeating several such experiments in which surface current is given and the corresponding surface voltage (Dirichlet data) is measured, we obtain the information on the Neumann-to-Dirichlet (NtD) operator which can be denoted by:

$$\Lambda_\sigma : g \mapsto f. \tag{3}$$

In the full EIT version, the problem is to reconstruct σ from just the surface current and voltage measurements. The estimate of the unknown conductivity σ can be reconstructed from a set of EIT experiments (Somersalo et al., 1992; Borcea, 2002; Hanke & Brühl, 2003). In a simplified version that we attempt to solve here using our novel method, we will be interested in recovering σ from the measurements of u in the interior of the medium. This problem formulation is a close cousin of the problem in groundwater flow wherein the source term on the right hand side of Eq. 1 is non-zero. It should also be noted, that a similar formulation was studied in Bar & Sochen (2021).

In our training paradigm, we need to simulate u for for the interior of the medium for any underlying σ. To achieve that we first solve for the forward problem by training neural network that can predict the value of the function $u(x)$ at any given point $x \in \Omega$, for any given σ. Once we have access to this pre-trained network that we will call u-Net, we will subsequently train another network to predict the value of $\sigma(x)$ for any $x \in \Omega$ from pointwise measurements of a function $u(x)$ that satisfies Eq. 1 and agrees with the Neumann and Dirichlet boundary data given from Eq. 3.

2.1 USING PINNS FOR EIT

Following the construction of physics-informed neural networks (PINNs), we can parameterize both σ and u using neural networks, called σ-Net and u-Net, respectively, and train them such that the values of σ-Net and u-Net for any provided point in the interior or boundary of the medium respect the PDE in Eq.1 or its boundary conditions[1] – depending on the position of the point.

The forward problem assumes that σ distribution is known everywhere inside the domain Ω. The u-Net takes mesh points in Cartesian space $(x, y) \in \mathbb{R}^2$ as input and correspondingly learns to recover correct potential values u at the points inside the domain Ω, given the voltages only at electrodes u_e. To train u-Net, we utilize Eq. 1 and, 2 to write them in a functional form as follows:

$$\mathcal{L}_{pde} = \nabla \cdot \big(\sigma \ \nabla u\big) \qquad \text{on } \Omega \tag{4}$$

$$\mathcal{L}_{bc} = \big[\sigma_b \frac{\partial u_b}{\partial n_b}\big] + \big[\sigma_e \frac{\partial u_e}{\partial n_e} - g_e\big] \quad \text{on } d\Omega_b, d\Omega_e. \tag{5}$$

[1]To the best of our knowledge only Bar & Sochen (2021) claim to train the EIT inverse problem using PINNs by jointly training u-Net and σ-Net, but unfortunately, their implementation is not open-source and we were unable to reproduce their results based on the details in the paper. Refer further to Appendix A.4.

Here, $\mathcal{L}_{pde}$ governs the potential u in forward and conductivity σ distributions in inverse problem respectively. $\mathcal{L}_{bc}$ are the combined BCs where we enforce Neumann BC twice, once on all boundary points $d\Omega_b$ except electrodes and separately on electrodes $d\Omega_e$. Thus, σ-Net and u-Net represent their respective distributions as parameterized functions $\sigma(x, y)$ and $u(x, y)$. We can then write a common objective function which is used to train both problems as follows:

$$\mathcal{L}_\theta = \frac{\alpha}{N_d}\sum_{i=1}^{N_d}\|\mathcal{L}_{pde}\|_2^2 + \frac{\beta}{M}\sum_{m\in \text{top}_M \mathcal{L}_{pde}}|\mathcal{L}_{pde}| + \frac{\gamma}{N_b}\frac{\delta}{N_e}\sum_{i=1}^{N_b}\sum_{i=1}^{N_e}|\mathcal{L}_{bc}|, \tag{6}$$

where α, β, and γ control the contribution of each terms to the overall loss. The required derivatives of u and σ with respect to coordinate x and y are calculated via reverse mode differentiation provided by autodiff frameworks such as PyTorch or Tensorflow. Following Bar & Sochen (2021), we penalize violations in $\mathcal{L}_{pde}$ twice using L_2 and L_∞ norm variants in order to provide a stronger approximate solution. Finally, in order to train u-Net more efficiently, we enforce additional regularizers:

$$\mathcal{L}_{\theta_{u\text{-Net}}} = \mathcal{L}_\theta + \frac{\epsilon}{N_e}\sum_{i=1}^{N_e}|u_i - f_i| + \zeta\,\|w_u\|^2 \tag{7}$$

The second term of Eq. 7 enforces Dirichlet BC only on electrodes while the last term controls the weights of u-Net using L_2 regularization on θ_u.

We train σ-Net (parameters are denoted as θ_σ) such that it learns the conductivity distribution over all given mesh points to solve the semi-inverse problem. Similar to the forward problem training, σ-Net incorporates the main PDE and the Nuemann BCs seen in Eq. 6 as a part of its training objective in order learn the conductivity inside Ω. However, the problem is known to be illposed and thus needs strong regularizers to improve the quality of reconstructions in conjunction with $\mathcal{L}_\theta$. For instance, $\sigma_{b,e}$ loss penalizes violations on $d\Omega$ which do not satisfy the assumption of uniform conductivity on boundary. Similarly, we regulate the norm of gradients $\nabla\sigma$ inside Ω to promote sparse edges in predictions. We also introduce a new hinge loss regularizer to penalize σ-Net predictions that are less than one. Thus, the combined σ-Net training objective assimilating the elliptical PDE, its BC and all the aforementioned mentioned regularizers is given as follows:

$$\mathcal{L}_{\theta_{\sigma\text{-Net}}} = \mathcal{L}_\theta + \frac{1}{N_b}\sum_{i=1}^{N_b}\sum_{i=1}^{N_e}|\sigma_i - \sigma_{b,e}| + \frac{\tau}{N_d}\sum_{i=1}^{N_d}|\nabla\sigma| + \zeta\,\|w_\sigma\|^2 + \upsilon\max(0, 1-\sigma) \tag{8}$$

3 Energy Based Priors

L_{pde} from Eq. 6 can be interpreted as a residual of violation (of Eq. 1), noted as r for simplicity. Assume $\mathbf{r}$ (residual of entire domain) follow a multi-variate Gaussian distribution with a zero mean and diagonal covariance matrix. Therefore, maximizing the likelihood of $p(\mathbf{r}|\boldsymbol{\sigma};\theta_\sigma)$ results in the similar optimization as minimizing $\sum_i^{N_d}\|\|L_{pde}\|_2^2$ with respect to θ_σ (parameters of σ-Net).

Now we can define our Bayesian approach but assuming $\boldsymbol{\sigma}$ follows the prior distribution $p(\boldsymbol{\sigma})$:

$$\max_{\theta_\sigma}\log p(\mathbf{r}|\boldsymbol{\sigma}) + \log p(\boldsymbol{\sigma}), \tag{9}$$

where $\boldsymbol{\sigma}$ is parameterized by θ_σ via σ-Net. We interpret $p(\boldsymbol{\sigma})$ as possible solutions to the PDE defined by Eq. 1 for different boundary conditions.

We define the prior distribution using energy-based models (EBMs) (LeCun et al., 2006): ($p(\boldsymbol{\sigma}) \propto \exp(-E_\phi(\boldsymbol{\sigma}))$. Several techniques have been proposed in literature for training EBMs including contrastive divergence (Gutmann & Hyvärinen, 2010), noise-contrastive estimation (Du & Mordatch, 2019), score matching (Hyvärinen & Dayan, 2005), and denoising score matching Vincent (2011). We found that denoising score matching (DSM) is more stable, less compute intensive as it avoids expensive second order partial derivatives by approximating the trace of Hessians involved in traditional score matching and ultimately generates more realistic σ solutions in our setting. DSM trains the energy function such that its vector field ($\nabla_{\boldsymbol{\sigma}_m}\log p_\phi(\boldsymbol{\sigma}_m)$) matches the vector field of the underlying σ distribution $p(\boldsymbol{\sigma}_m)$, which is approximate by perturbing the empirical data distribution

with Gaussian noise of different intensities. See Song & Ermon (2019), Song & Ermon (2020) for more detail.

We jointly estimate a noise conditional energy function $E_\phi(\sigma, \mu)$ for all noise-perturbed data distributions conditioned on various noise scales $\mu \in [1 \dots L]$, which are chosen such that they satisfy $\mu_1 > \mu_2 > \cdots > \mu_L$. In our work, we chose $L = 20$ linearly spaced noise scales between $\mu_i \in [2, 0.01]$. Essentially, then the training is to minimize the following objective:

$$\mathcal{L}(\phi; \mu_i) = \frac{1}{L} \sum_{i=1}^{L} \lambda(\mu_i) \left[\frac{1}{2} \mathbb{E}_{p(\sigma)} \, \mathbb{E}_{\mathrm{z} \sim \mathcal{N}(0,I)} \left\| \nabla_\sigma E_\phi(\hat{\sigma}, \mu_i) - \frac{\mathrm{z}}{\mu_i} \right\|_2^2 \right] \tag{10}$$

where $\lambda(\mu_i) > 0$ is a coefficient function chosen as μ^2 and finally, $\hat{\sigma} = \sigma + \mathrm{z}\mu$ is the noise perturbed version of conductivity distribution $\hat{\sigma} \sim \mathcal{N}(\sigma, \mu^2 I)$. In order to use E_ϕ as a prior, in contrast to the standard DSM training that trains a score network ($S(.) = \nabla_z E$), we directly train the energy network. See Appendix A.3 for the details on training E_ϕ. The generated $\boldsymbol{\sigma}$ from E_ϕ and closest training data point is shown Fig. 5 in Appendix.

Upon successful training of the E_ϕ using Eq. 10, the energy function $E^*_\phi(\sigma, \mu)$ will assign lower energy value to $\boldsymbol{\sigma}$ that are more likely to present in the real world (valid solution of Eq. 1) and higher energy value to unlikely assignments (invalid assignments that violates Eq. 1 greatly).

Now we can rewrite the training objective of σ-Net (Eq. 8) as :

$$\mathcal{L}_S = \frac{1}{K} \sum_{k=1}^{K} \mathcal{L}_{\theta_k} + \frac{1}{N_b} \sum_{i=1}^{N_b} \sum_{i=1}^{N_e} |\sigma_i - \sigma_{b,e}| + \frac{\tau}{N_d} \sum_{i=1}^{N_d} |\nabla \sigma| + \zeta \left\| w_\sigma \right\|^2 + \upsilon \max(0, 1 - \sigma) - \kappa E^*_\phi(\widetilde{\sigma}, \mu), \tag{11}$$

where κ is the weight of the prior in the overall loss.

4 Experiments

Our EIT data simulation setup primarily consists of phantom generation and forward solution construction via finite element solver. Initially, we construct various discretized solutions of σ on a 2D mesh-grid of size 128×128 to obtain phantoms $\Omega_{1 \dots Z}$. In particular, we randomly choose the anomaly location, radius, shape deformity, quantity and their conductivity on a circular mesh. The target σ values are chosen randomly between $\in [3, 15]$ for either $1, 2$ or 3 anomalies per mesh with restrictions on their locations, so as to not let the anomalies touch each other and the boundary. We also assume a uniform background conductivity of $\sigma = 1$ (simulating vacuum) on locations devoid of any anomaly and $\sigma = 0$ outside the circle mesh. Upon selection of phantom configuration followed by discretization, we smooth our solutions following Evans (2010) using a Gaussian low-pass filter of size 200 and standard deviation 3 to obtain smoother solutions. We then generate 6512 such smoothed phantoms for EBM prior training and 1628 for testing. Majority of these datasets contain circular anomalies with varying radii and a few thousand examples with minor shape deformity (ellipsoids). Furthermore, we standardize both train, test sets to $[0, 1]$ interval by dividing all samples using the maximum conductivity σ value obtained from training set. We additionally generate a few hand-crafted phantoms to train the u-Net and σ-Net for forward and inverse problems, seen in row 1 of Fig. 2. These configurations are specifically designed to be more challenging by setting higher shape deformity, larger variance in σ values for each anomaly and placing them in challenging locations inside mesh. This is to purposefully make the physics-based training and inference more challenging to ensure extreme robustness. Note that there are no phantoms in the EBM training set which resemble these hand-crafted samples. With regards to our forward problem simulation setup, we assume that Neumann condition is imposed on the entire boundary where the function g in Eq. 2 is given as a trigonometric pattern Siltanen et al. (2000) as follows:

$$g = \frac{1}{\sqrt{2\pi}} \cos(\eta\omega + \psi), \quad n \in \mathbb{Z} \tag{12}$$

where ω is the angle along $d\Omega$, η and ψ are the current frequency, phase respectively. We use one current pattern where $n = 1$ and $\psi = 0$ for all our experiments. Thus, to obtain the forward solution, we use finite element method (FEM) solvers by utilizing discretized hand-crafted phantoms profile (σ value, anomaly location) and current g as input. We then solve for u everywhere and obtain the solver based solution on circular mesh.

4.1 EXPEDITED SEMI-INVERSE PROBLEM EVALUATION

We now present results of our proposed framework. As discussed in prior sections, at each step of σ-Net training procedure, the current prediction $\hat{\sigma}$ is fed to the trained EBM prior $E^*_\phi(\hat{\sigma}, \mu_L)$ to obtain a scalar energy value. This energy value provides useful supervision to σ-Net and essentially expedites the convergence of PINNs. This phenomenon can be viewed in Fig. 1. Here, each sub-figure represents the learning curve of σ-Net with the choice of metric as mean squared error (MSE), separately trained to recover conductivity of various phantoms with and without the inclusion of EBM prior. Although the model is trained for 3000 epochs, for brevity, we only showcase the first 500 epochs of the curve due to more interesting properties here. Evidently, the PINNs with EBM prior converge much faster within the first few 100 epochs than the training without prior. In case of phantom 4, the convergence is more than ten times faster while also aiding the PINN to avoid getting stuck in a local minima. Corresponding to these learning curves, we present Fig. 2 showing the quality of reconstruction for each phantom.

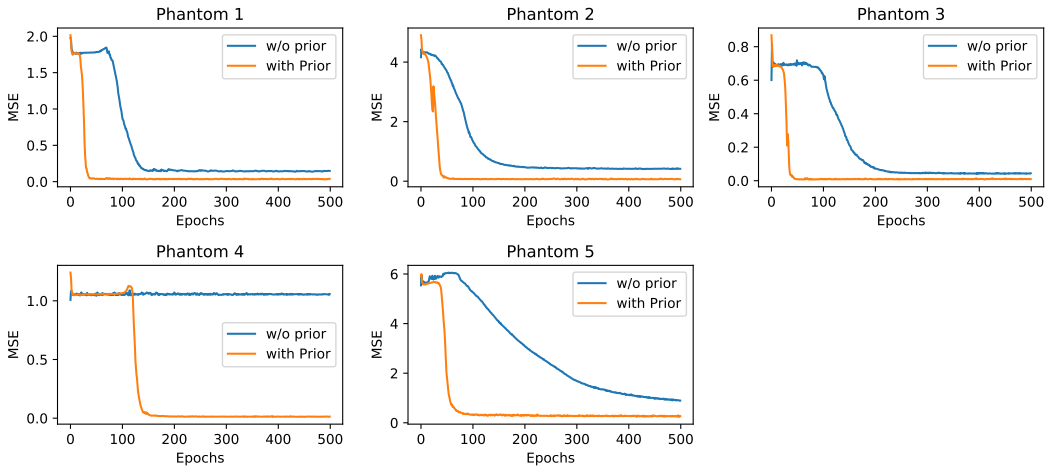

Figure 1: PDE solution convergence and accuracy analysis via σ MSE for various phantoms

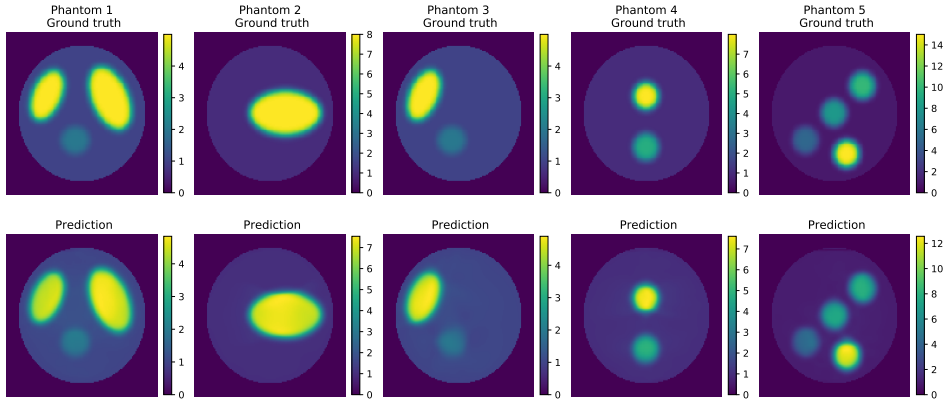

Figure 2: Comparison of ground truth σ and predictions from PINN with EBM prior

4.2 COMPARISON WITH BASELINES

We now compare the performance of our framework with baselines Strong sol. Bar & Sochen (2021), Deep Galerkin Methods (DGM) Sirignano & Spiliopoulos (2017) by using the hand-crafted phantoms. In order to introduce these works, there are a few differences that set our works apart. Firstly, both Bar & Sochen (2021), Sirignano & Spiliopoulos (2017) utilize high resolution mesh

which naturally enable PINNs to learn the problem more efficiently due to availability of large input data points to solve the PDE and BC. Additionally, Bar & Sochen (2021) utilizes a large number of randomly selected boundary voltages to enforce the Dirichlet BC in Eq. 2 during forward solution learning. In contrast, we train our u-Net and σ-Net in a much more challenging settings while using a low-resolution mesh and only 16 boundary voltages during the forward problem training. The driving point here is that, one can view Dirichlet BC as a L_1 regression loss over given boundary voltages. When more boundary points and their corresponding voltages are revealed for learning u, the more stronger this supervision signal, which enables learning the forward map and anomaly location more accurately. However from practical EIT system standpoint, it is unrealistic to collect such close proximity fine-grained information over many boundary points. Apart from these difference, we treat Bar & Sochen (2021) and Sirignano & Spiliopoulos (2017) as special cases of our work. Our work differs mainly in the incorporation of the energy scalar from EBM and the new hinge loss term in Eq. 8 to improve the performance. Given the high non-linearity and ill-posed nature of the EIT problem, even a slight change in one parameter can lead to highly varying solutions. Hence, we tune our loss penalties differently to obtain higher quality reconstructions. We show a list of common parameters among the considered baselines. We now present the semi-inverse

Table 1: List of common hyperparameters among various baselines

Baseline	α	β	M	γ	δ	ϵ	ζ	τ	ν	κ
DGM	1	0	0	1	1	1	0	0	0	0
Strong Sol.	0.01	0.01	40	1	1	1	1e-8	0.01	0	0
Ours	0.05	0.05	40	1	0.1	100	1e-6	0.01	10	0.0001

Table 2: Evaluation of PINNs augmented with EBM priors across various metrics and frameworks

Phantom	**Metric**	**DGM [paper]**	**DGM [paper] with EBM Prior**	**Strong sol. [paper]**	**Strong sol. [paper] with EBM Prior**	**Our Method**	**Our Method with EBM Prior**
Ω_1	MSE	3.26 ± 0.028	0.13 ± 0.002	2.57 ± 0.041	0.16 ± 0.026	0.13 ± 0.001	$\mathbf{0.03 \pm 0.001}$
	PSNR	8.84 ± 0.038	22.77 ± 0.079	9.89 ± 0.068	22.02 ± 0.645	22.79 ± 0.034	$\mathbf{29.94 \pm 0.146}$
	MDE	2.29 ± 0.024	0.36 ± 0.003	1.84 ± 0.028	0.23 ± 0.002	0.35 ± 0.002	$\mathbf{0.12 \pm 0.014}$
Ω_2	MSE	6.19 ± 0.021	0.3 ± 0.004	5.18 ± 0.039	0.2 ± 0.015	0.4 ± 0.002	$\mathbf{0.06 \pm 0.001}$
	PSNR	10.15 ± 0.014	23.3 ± 0.054	10.92 ± 0.033	25.00 ± 0.325	22.04 ± 0.023	$\mathbf{30.58 \pm 0.079}$
	MDE	2.85 ± 0.015	0.39 ± 0.003	2.2 ± 0.023	0.25 ± 0.002	0.43 ± 0.003	$\mathbf{0.1 \pm 0.009}$
Ω_3	MSE	1.56 ± 0.032	0.03 ± 0.001	1.37 ± 0.017	0.11 ± 0.013	0.04 ± 0.001	$\mathbf{0.01 \pm 0.0001}$
	PSNR	12.05 ± 0.089	29.03 ± 0.193	12.62 ± 0.056	23.45 ± 0.47	28.2 ± 0.061	$\mathbf{35.46 \pm 0.088}$
	MDE	1.38 ± 0.033	0.25 ± 0.002	1.09 ± 0.021	0.2 ± 0.002	0.25 ± 0.002	$\mathbf{0.11 \pm 0.009}$
Ω_4	MSE	2.00 ± 0.013	0.08 ± 0.002	1.64 ± 0.019	0.2 ± 0.036	0.07 ± 0.001	$\mathbf{0.01 \pm 0.0001}$
	PSNR	15.04 ± 0.028	29.07 ± 0.099	15.92 ± 0.052	25.23 ± 0.779	29.68 ± 0.034	$\mathbf{38.45 \pm 0.052}$
	MDE	1.47 ± 0.013	0.26 ± 0.004	1.1 ± 0.02	0.19 ± 0.001	0.25 ± 0.003	$\mathbf{0.12 \pm 0.011}$
Ω_5	MSE	7.39 ± 0.046	1.38 ± 1.25	6.68 ± 0.018	0.5 ± 0.084	0.57 ± 0.003	$\mathbf{0.22 \pm 0.004}$
	PSNR	14.83 ± 0.027	24.03 ± 1.99	15.27 ± 0.012	26.67 ± 0.694	25.95 ± 0.025	$\mathbf{30.17 \pm 0.082}$
	MDE	2.97 ± 0.012	0.73 0.439	2.47 ± 0.017	0.28 ± 0.004	0.44 ± 0.002	$\mathbf{0.08 \pm 0.016}$

problem evaluation results for considered baselines in Table 2 for all five phantoms. We primarily use three metrics namely, mean squared error (MSE), peak signal-to-noise ratio (PSNR) and mean difference error (MDE) to evaluate the quality of reconstruction of σ solution while learning the objective Eq. 8. Note that MDE is computed as the difference in means of σ over all $(x, y) \in \Omega$ and $d\Omega$. We then utilize the hyper-parameters shown in Table 1 to train σ-Net for our framework, while the neural network presented in Bar & Sochen (2021) is used on both Bar & Sochen (2021) and Sirignano & Spiliopoulos (2017) baselines. All these models are trained separately on all five phantoms while using the same settings, once without the EBM prior and once under the presence of EBM prior. These experiments are repeated for 10 runs and the averaged metrics along with their standard deviation error is presented. The main observation from Table 2 is, that by incorporating the EBM prior in PINN training enables lower error rates and higher quality reconstructions for the semi-inverse problem. Additionally, we can also see that the EBM prior can be easily plugged into any PINN-based framework and vastly improve the underlying training, showcasing our methods strong generalizability.

4.3 Learning rate convergence

We aim to show through this study that, the learning rate vastly effects PINN's solution accuracy and this inherent instability can alleviated easily by introducing the EBM priors, even while choosing

really high learning rates. For this study, we keep all the parameters of our proposed method shown in Table 1 fixed and vary only the learning rate during σ-Net training with Eq. 8. We choose 0.0001, 0.001, 0.01 and 0.1 learning rates to study their effects on the semi-inverse PINN training with and without the EBM prior. Note that, we still decay the all these chosen learning rates using exponential decay with rate 0.9 over every 200 epochs. The results of learning rate studies are presented in Fig.

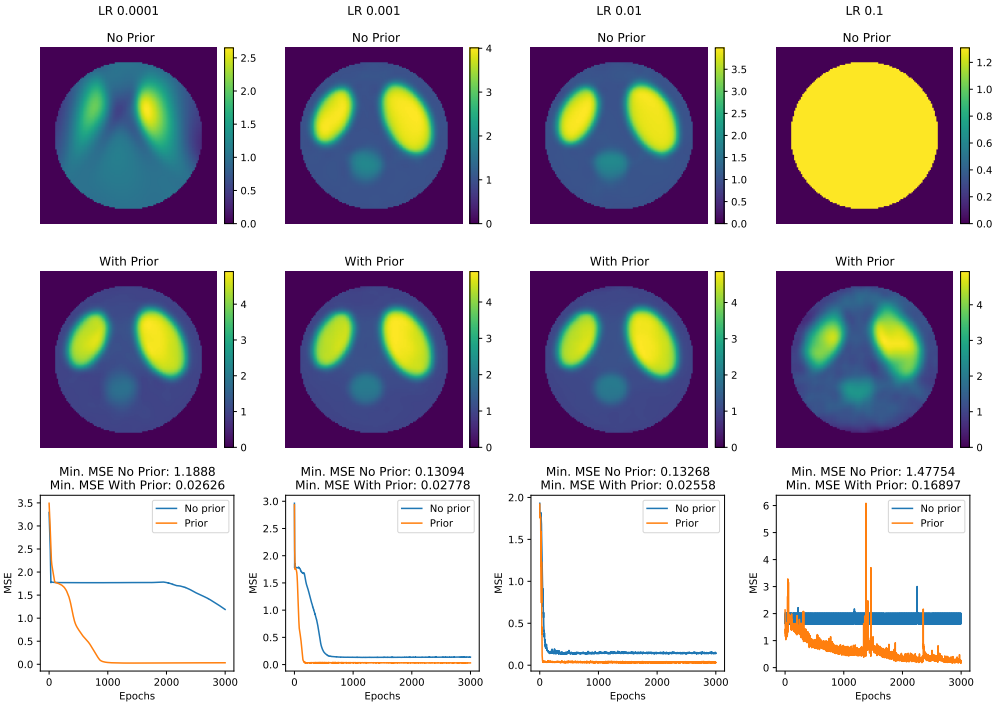

Figure 3: Training robustness of our proposed work with EBM prior for various learning rates on Phantom 1

3. As seen in this figure, the true σ value for the ellipsoid anomalies is 5 while the smaller circular anomaly in the bottom-middle has $\sigma = 2$. We can evidently see that the PINN without EBM prior fails to converge to an accurate solution with regards to row three. More importantly, the maximum σ values for row one and two's color-bars indicate the failure to reach an accurate conductivity prediction. Clearly, the PINN without EBM prior never matches the solution accurately, under all learning rates. More adversely, the PINN falls into a trivial solution mode in a training regime without our EBM prior while using a learning rate that is either too low or too high respectively. On the other hand, PINNs augmented with EBM prior always produce accurate solutions of slightly varying degree of MSE as seen in row three of the figure. More importantly, the training session with extremely large learning rate in $LR = 0.1$ still manages to produce a reconstruction although the accuracy might be lower. This proves the robustness of our framework and instigates clear evidence that PINN training is being improved by a very large margin while incorporating EBM priors. Other advantages such as, burden of choosing the optimal learning rate is also alleviated.

4.4 Sensitivity analysis via random parameter search

As discussed in these earlier sections, PINNs are extremely sensitive to the balance of interplay among different hyperparameters. The training procedure essentially becomes an optimization problem where one has to tune the right settings in order to converge to a highly accurate solution or fall into a failed trivial solution. The problem of choosing parameters is even more challenging in EIT as it is highly non-linear and ill-posed. We thus study the effect of changing the strength of various loss weighting penalties mentioned in Table 1. More precisely, we restrict the search space to α, β, γ and δ which are main terms involving the PDE and its BC, while solving the semi-inverse problem using Eq. 8. We then configure a limited valid set of weights for $\alpha \in [0.01, 0.05, 0.1, 0.5, 1]$, $\beta \in [0.01, 0.05, 0.1, 0.5, 1]$, $\gamma \in [0.01, 0.1, 0.5, 1.0, 2.0]$ and finally

$\delta \in [0.001, 0.01, 0.1, 0.5, 1.0, 2.0]$ and randomly choose a single weight from each set to perform training. We separately conduct 200 of such random search runs each on phantom 1 and 5, with and without the EBM prior. The results of this study can be seen in Fig. 4.

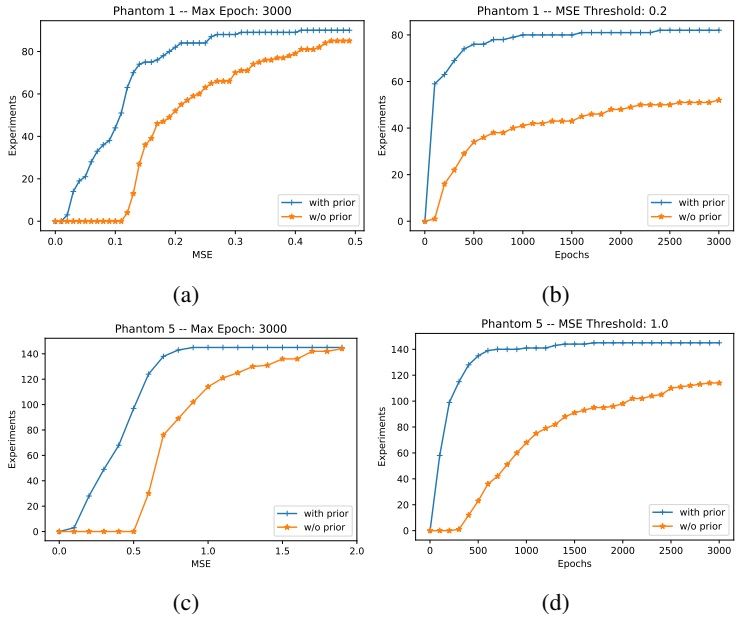

Figure 4: Sensitivity analysis of σ-Net while randomly searching the optimal weights in Eq. 6

Here, Fig. 4a corresponds to phantom 1 which indicates the number of random experimental runs that were successful to reach a set upper bound of 0.5 MSE after training for 3000 epochs. We can easily infer that the PINN training success rate with the EBM prior is much higher. Additionally, a large number of experiments reach a MSE of 0.1 or lower in comparison to experiments without prior, translating to high accuracy rates. Fig. 4b also corresponds to phantom 1, however we set a strict MSE threshold of 0.2 here and check for number of experiments that reach this threshold by the end of all 3000 training epochs. Here also, we can see clear evidence of PINN with EBM prior has much higher success rate in producing highly accurate σ reconstructions in the semi-inverse problem setting. Similar observations can be said about Figs. 4c and 4d which are studies related to phantom 5.

4.5 Semi-Inverse With Noisy Data

In the final set of experiments, we test the robustness of our framework while learning from noisy data. Initially, we add uniform noise of various standard deviation levels such as $0.1, 0.25, 0.5$, to the 16 boundary measurements. Then the u-Net learns the forward problem under these noisy settings and learns to approximate a noisy-forward solution (see Appendix A.2 for full evaluation of u-Net under noisy settings). While moving to the semi-inverse problem, we use data originating from the noisy u-Net predictions and evaluate σ-Net performance and test its ability to withstand measurement noise with and without the EBM prior. The evaluation results can be seen in Table 3. The trend of performance degradation is visible while the noise addition increases, more noticeable in some phantoms that others. Despite these noisy conditions, incorporating the EBM prior during PINN training greatly improves the σ predictions and proves robustness of the method.

5 Related Work

Inverse problems for elliptic are ill-posed and highly nonlinear, which makes it imperative to include a regularization method for obtaining a reasonable reconstruction Jin et al. (2012); Jin & Maass (2012). In the literature, there are many approaches for solving the EIT inverse problem with rigorous theoretical justifications of the methods used. These include variational methods for least-square

Table 3: Performance evaluation of σ-Net using noisy measurements

Noise	Metric	Phantom 1		Phantom 2		Phantom 3		Phantom 4		Phantom 5	
		w/o Prior	w/ Prior	w/o Prior	w/ Prior	w/o Prior	w/ Prior	w/o Prior	w/ Prior	w/o Prior	w/ Prior
0	MSE	0.13	**0.03**	0.4	**0.06**	0.04	**0.01**	0.07	**0.01**	0.57	**0.22**
	PSNR	22.79	**29.94**	22.04	**30.58**	28.20	**35.46**	29.68	**38.45**	25.95	**30.17**
	MDE	0.35	**0.12**	0.43	**0.1**	0.25	**0.11**	0.25	**0.12**	0.44	**0.08**
0.1	MSE	0.094	**0.03**	3.38	**1.64**	**0.02**	0.03	2.84	**0.01**	0.49	**0.11**
	PSNR	24.25	**29.91**	12.77	**15.91**	**31.50**	29.72	13.53	**36.78**	26.67	**33.09**
	MDE	0.29	**0.16**	1.09	**0.52**	0.21	**0.21**	0.02	**0.17**	0.42	**0.23**
0.25	MSE	0.17	**0.02**	0.58	**0.08**	0.03	**0.03**	0.09	**0.03**	0.67	**0.31**
	PSNR	21.55	**31.20**	20.40	**29.10**	28.95	**29.29**	28.46	**33.86**	25.24	**28.56**
	MDE	0.37	**0.23**	0.50	**0.29**	0.24	**0.16**	0.26	**0.17**	0.49	**0.37**
0.5	MSE	0.13	**0.06**	0.37	**0.13**	0.61	**0.57**	0.09	**0.02**	5.22	**4.39**
	PSNR	22.96	**26.05**	22.42	**26.84**	16.15	**16.41**	28.69	**35.43**	16.35	**17.09**
	MDE	0.34	**0.13**	0.42	**0.13**	0.14	**0.22**	0.25	**0.17**	0.90	**0.34**

fitting (Jin et al. (2012); Jin & Maass (2012); Ahmad et al. (2019)), the factorization method (Kirsch & Grinberg (2008)), the d-bar method (Mueller et al. (2002); Isaacson et al. (2004); Knudsen et al. (2009)), the monotonicity shape estimate (Harrach & Ullrich (2013)), the level set method (Liu et al. (2017; 2018)) and series inversion methods (Arridge et al. (2012); Abhishek et al. (2020)) to name a few. Besides, regularized Newton type methods(Haber (2005); Lechleiter & Rieder (2008)) are also popular. For the groundwater flow model, some important references can be found in Frühauf et al. (2005); Russell & Wheeler; Oliver et al. (2008).

Driven by the need to quantify uncertanities in the reconstruction, a trend that started around 2000 was to solve such inverse problems in a parametric Bayesian setting (Kaipio et al. (2000; 2004); Strauss & Khan (2015); Ahmad et al. (2019); Strauss et al. (2015)). However, the error in the parametric Bayes method have been shown to depend upon discretization and hence, alternative non-parametric Bayesian inversion methods have been proposed which are demonstrably mesh-independent. Following the celebrated work of Stuart (2010), a Bayesian level set method was proposed for reconstruction of piecewise smooth conductivity in Dunlop & Stuart (2016), for reconstruction of piecewise continuous permeability in groundwaterflow model in Iglesias et al. (2014) and for reconstruction of conductivity in certain Sobolev spaces in Abraham & Nickl (2019). Recall that in a Bayesian setting, the posterior distribution on the space of conductivities is proportional to the likelihood function and the prior distribution. The prior distribution; in addition to encapsulating our subjective belief about the functional space in which the conductivity is thought to lie; also introduces regularization in the problem. Great strides have been made in inventing efficient MCMC sampling methods which have increased computational efficiency of such Bayesian inversion methods (Cotter et al. (2013)), however increasing computational efficiency continues to be an active area of research in this field.

With the rise of deep learning, methods for solving the EIT with neural networks were proposed. In a supervised setting, this includes Hamilton & Hauptmann (2018); Fan & Ying (2020); Hamilton et al. (2019) that primarily use convolutional architectures to predict the conductivity σ based on boundary measurements. A related problem of classification of strokes with EIT was considered in Agnelli et al. (2020).

6 Conclusion

Physics-informed neural networks are an important category of data-driven PDE solvers. However, for more complicated problems PINNs are not stable and robust. In this work, we look at the EIT problem and show that we can improve the stability and robustness of PINNs training (with respect to noisy measurement and hyperparameter selection) via a Bayesian approach. We describe a data-driven prior using energy-based methods, which can easily be used with the other loss term to robustly train PINNs. In the EIT setting, our experimental result also show that PINNs converge faster and also to a more accurate solution when trained with prior. In our future work, we will study using priors for potential in order to the improve the training of PINN for full EIT inverse problem and other PDE problems.

REFERENCES

Anuj Abhishek, Marc Bonnet, and Shari Moskow. Modified forward and inverse Born series for the Calderon and diffuse-wave problems. *Inverse Problems*, 36(11):114001, 19, 2020. ISSN 0266-5611. doi: 10.1088/1361-6420/abae11. URL https://doi.org/10.1088/1361-6420/abae11.

Anuj Abhishek, Thilo Strauss, and Taufiquar Khan. An optimal bayesian estimator for absorption coefficient in diffuse optical tomography. *SIAM Journal on Imaging Sciences*, 15(2):797–821, 2022. doi: 10.1137/21M1462842. URL https://doi.org/10.1137/21M1462842.

Kweku Abraham and Richard Nickl. On statistical Calderón problems. *Math. Stat. Learn.*, 2(2): 165–216, 2019. ISSN 2520-2316.

J. P. Agnelli, A. Çöl, M. Lassas, R. Murthy, M. Santacesaria, and S. Siltanen. Classification of stroke using neural networks in electrical impedance tomography. *Inverse Problems*, 36(11): 115008, 26, 2020. ISSN 0266-5611. doi: 10.1088/1361-6420/abbdcd. URL https://doi.org/10.1088/1361-6420/abbdcd.

Sanwar Ahmad, Thilo Strauss, Shyla Kupis, and Taufiquar Khan. Comparison of statistical inversion with iteratively regularized gauss newton method for image reconstruction in electrical impedance tomography. *Applied Mathematics and Computation*, 358:436–448, 2019.

Simon Arridge, Shari Moskow, and John C. Schotland. Inverse Born series for the Calderon problem. *Inverse Problems*, 28(3):035003, 16, 2012. ISSN 0266-5611. doi: 10.1088/0266-5611/28/3/035003. URL https://doi.org/10.1088/0266-5611/28/3/035003.

Leah Bar and Nir Sochen. Strong solutions for pde-based tomography by unsupervised learning. *SIAM Journal on Imaging Sciences*, 14(1):128–155, 2021.

Liliana Borcea. Electrical impedance tomography. *Inverse problems*, 18(6):R99, 2002.

S. L. Cotter, G. O. Roberts, A. M. Stuart, and D. White. MCMC methods for functions: modifying old algorithms to make them faster. *Statist. Sci.*, 28(3):424–446, 2013. ISSN 0883-4237. doi: 10.1214/13-STS421. URL https://doi.org/10.1214/13-STS421.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

Matthew M. Dunlop and Andrew M. Stuart. The Bayesian formulation of EIT: analysis and algorithms. *Inverse Probl. Imaging*, 10(4):1007–1036, 2016. ISSN 1930-8337. doi: 10.3934/ipi.2016030. URL https://doi.org/10.3934/ipi.2016030.

Lawrence C Evans. *Partial differential equations*, volume 19. American Mathematical Soc., 2010.

Yuwei Fan and Lexing Ying. Solving electrical impedance tomography with deep learning. *Journal of Computational Physics*, 404:109119, 2020.

F. Frühauf, O. Scherzer, and A. Leitão. Analysis of regularization methods for the solution of ill-posed problems involving discontinuous operators. *SIAM Journal on Numerical Analysis*, 43(2): 767–786, 2005. doi: 10.1137/S0036142903430906. URL https://doi.org/10.1137/S0036142903430906.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

E. Haber. Quasi-Newton methods for large-scale electromagnetic inverse problems. *Inverse Problems*, 21(1):305–333, 2005. ISSN 0266-5611. doi: 10.1088/0266-5611/21/1/019. URL https://doi.org/10.1088/0266-5611/21/1/019.

Sarah J Hamilton, Asko Hänninen, Andreas Hauptmann, and Ville Kolehmainen. Beltrami-net: domain-independent deep d-bar learning for absolute imaging with electrical impedance tomography (a-eit). *Physiological measurement*, 40(7):074002, 2019.

Sarah Jane Hamilton and Andreas Hauptmann. Deep d-bar: Real-time electrical impedance tomography imaging with deep neural networks. *IEEE transactions on medical imaging*, 37(10): 2367–2377, 2018.

Martin Hanke and Martin Brühl. Recent progress in electrical impedance tomography. *Inverse Problems*, 19(6):S65, 2003.

Bastian Harrach and Marcel Ullrich. Monotonicity-based shape reconstruction in electrical impedance tomography. *SIAM Journal on Mathematical Analysis*, 45(6):3382–3403, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Marco A. Iglesias, Kui Lin, and Andrew M. Stuart. Well-posed Bayesian geometric inverse problems arising in subsurface flow. *Inverse Problems*, 30(11):114001, 39, 2014. ISSN 0266-5611. doi: 10.1088/0266-5611/30/11/114001. URL https://doi.org/10.1088/0266-5611/30/11/114001.

David Isaacson, Jennifer L Mueller, Jonathan C Newell, and Samuli Siltanen. Reconstructions of chest phantoms by the d-bar method for electrical impedance tomography. *IEEE Transactions on Medical Imaging*, 23(7):821–828, 2004.

Bangti Jin and Peter Maass. An analysis of electrical impedance tomography with applications to tikhonov regularization. *ESAIM: Control, Optimisation and Calculus of Variations*, 18(4):1027–1048, 2012.

Bangti Jin, Taufiquar Khan, and Peter Maass. A reconstruction algorithm for electrical impedance tomography based on sparsity regularization. *International Journal for Numerical Methods in Engineering*, 89(3):337–353, 2012.

Jari P Kaipio, Ville Kolehmainen, Erkki Somersalo, and Marko Vauhkonen. Statistical inversion and monte carlo sampling methods in electrical impedance tomography. *Inverse problems*, 16(5): 1487, 2000.

JP Kaipio, A Seppänen, E Somersalo, and H Haario. Posterior covariance related optimal current patterns in electrical impedance tomography. *Inverse Problems*, 20(3):919, 2004.

Andreas Kirsch and Natalia Grinberg. *The factorization method for inverse problems*. Number 36. Oxford University Press, 2008.

Kim Knudsen, Matti Lassas, Jennifer L. Mueller, and Samuli Siltanen. Regularized D-bar method for the inverse conductivity problem. *Inverse Probl. Imaging*, 3(4):599–624, 2009. ISSN 1930-8337. doi: 10.3934/ipi.2009.3.599. URL https://doi.org/10.3934/ipi.2009.3.599.

Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in Neural Information Processing Systems*, 34:26548–26560, 2021.

Armin Lechleiter and Andreas Rieder. Newton regularizations for impedance tomography: convergence by local injectivity. *Inverse Problems*, 24(6):065009, 18, 2008. ISSN 0266-5611. doi: 10.1088/0266-5611/24/6/065009. URL https://doi.org/10.1088/0266-5611/24/6/065009.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Dong Liu, Anil Kumar Khambampati, and Jiangfeng Du. A parametric level set method for electrical impedance tomography. *IEEE transactions on medical imaging*, 37(2):451–460, 2017.

Dong Liu, Danny Smyl, and Jiangfeng Du. A parametric level set-based approach to difference imaging in electrical impedance tomography. *IEEE transactions on medical imaging*, 38(1):145–155, 2018.

J.L. Mueller, S. Siltanen, and D. Isaacson. A direct reconstruction algorithm for electrical impedance tomography. *IEEE Transactions on Medical Imaging*, 21(6):555–559, 2002. doi: 10.1109/TMI.2002.800574.

Dean S. Oliver, Albert C. Reynolds, and Ning Liu. *Inverse Theory for Petroleum Reservoir Characterization and History Matching*. Cambridge University Press, 2008. doi: 10.1017/CBO9780511535642.

Maziar Raissi, Paris Perdikaris, and George 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*, 378:686–707, 2019.

Zaccharie Ramzi, Benjamin Rémy, Francois Lanusse, Jean-Luc Starck, and Philippe Ciuciu. Denoising score-matching for uncertainty quantification in inverse problems. *arXiv preprint arXiv:2011.08698*, 2020.

Thomas F. Russell and Mary Fanett Wheeler. *2. Finite Element and Finite Difference Methods for Continuous Flows in Porous Media*, pp. 35–106. doi: 10.1137/1.9781611971071.ch2. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611971071.ch2.

Samuli Siltanen, Jennifer Mueller, and David Isaacson. An implementation of the reconstruction algorithm of a nachman for the 2d inverse conductivity problem. *Inverse Problems*, 16(3):681, 2000.

Justin Sirignano and Konstantinos Spiliopoulos. Dgm: A deep learning algorithm for solving partial differential equations. 2017. doi: 10.48550/ARXIV.1708.07469. URL https://arxiv.org/abs/1708.07469.

Erkki Somersalo, Margaret Cheney, and David Isaacson. Existence and uniqueness for electrode models for electric current computed tomography. *SIAM J. Appl. Math.*, 52(4):1023–1040, 1992. ISSN 0036-1399. doi: 10.1137/0152060. URL https://doi.org/10.1137/0152060.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

Thilo Strauss and Taufiquar Khan. Statistical inversion in electrical impedance tomography using mixed total variation and non-convex ℓ_p regularization prior. *Journal of Inverse and Ill-posed Problems*, 23(5):529–542, 2015.

Thilo Strauss, Xiaolin Fan, Shuyu Sun, and Taufiquar Khan. Statistical inversion of absolute permeability in single-phase darcy flow. *Procedia Computer Science*, 51, 2015. doi: doi.org/10.1016/j.procs.2015.05.291. URL https://doi.org/10.1016/j.procs.2015.05.291.

A. M. Stuart. Inverse problems: a Bayesian perspective. *Acta Numer.*, 19:451–559, 2010. ISSN 0962-4929. doi: 10.1017/S0962492910000061. URL https://doi.org/10.1017/S0962492910000061.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

Yuxin Wu and Kaiming He. Group normalization, 2018. URL https://arxiv.org/abs/1803.08494.

A APPENDIX

The model architectures and training implementation details for u-Net, σ-Net and EBM are shared here.

A.1 SEMI-INVERSE IMPLEMENTATION DETAILS

The common loss weighting penalties used by both the u-Net, σ-Net models are: $\alpha = 0.05$, $\beta = 0.05$, $M = 40$, $\delta = 0.1$ and $\zeta = 1e^{-6}$. The Nuemann loss penalty $\gamma = 0.1$ and Dirichlet loss penalty $\epsilon = 100$ are specific to the forward problem only. While, $\gamma = 1$, $\tau = 0.01$, $\nu = 10$ and lastly $\kappa = 0.0001$ are specific inverse problem. Additionally, both models use the same multi-layer perceptron (MLP) architecture with residual connections Wang et al. (2021) and consist of 4 hidden layers with $\tanh$ activation and 64 neurons each. A single output neuron with no activation aids in predicting the solutions for given mesh points. We train both these models with a batch size of 1000 using ADAM optimizer for 3000 epochs with an initial learning rate of 0.005 and decay it exponentially with a rate of 0.9 over intervals of every 200 epochs. We use NVIDIA Titan RTX GPU for all our training purposes and the training time for forward and semi-inverse problems separately take around takes ~ 8 minutes for 3000 epochs.

A.2 FORWARD PROBLEM IMPLEMENTATION AND RESULTS

Table 4 shows the evaluation results of u-Net for the forward problem on the considered phantoms. We study the quality of forward problem solutions from u-Net while adding random uniform noise to the 16 electrode measurements of various levels. As seen in these experiments, the quality degradation is obvious due to increasing levels of noise. To counter this effect, we tune the penalty weights α, β, γ, δ on individual loss terms in Eq. 7 for increasing noise levels to improve the quality of reconstructions. The weights on Dirichlet BC $\epsilon = 100$, L_2 model parameter regularization $\zeta = 1e-6$ are fixed for all experiments and phantoms. Additionally, we use the same parameter set for all phantoms while only tuning them for a given noise level, which showcases the robustness of our method.

Table 4: Performance evaluation of u-Net for solving Forward problem with various noise levels

Noise Std	**Metric**	**Phantom 1**	**Phantom 2**	**Phantom 3**	**Phantom 4**	**Phantom 5**
		Parameters: $\alpha = 0.05$, $\beta = 0.05$, $\gamma = 0.1$, $\delta = 0.1$				
0	MSE	0.00046	0.00091	0.00027	0.00058	0.00036
	PSNR	39.41	33.40	46.90	47.02	40.51
		Parameters: $\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.1$, $\delta = 0.1$				
0.1	MSE	0.0090	0.070	0.02076	0.0208	0.026
	PSNR	26.51	14.55	28.13	31.48	21.91
		Parameters: $\alpha = 0.05$, $\beta = 0.05$, $\gamma = 1$, $\delta = 0.1$				
0.25	MSE	0.0426	0.1665	2.667	1.722	0.039
	PSNR	16.70	13.854	7.04	12.30	20.22
		Parameters: $\alpha = 0.1$, $\beta = 0.05$, $\gamma = 1$, $\delta = 0.1$				
0.5	MSE	1.041	2.25	10.59	2.46	4.70
	PSNR	5.89	0.52	1.05	3.01	0.587

A.3 EBM PRIOR IMPLEMENTATION AND EVALUATION

We train our EBM prior on σ solutions (as described in Section 4) by perturbing them with Gaussian noise of 20 noise scales $\mu_i \in [2, 0.01]$ in order to learn by DSM training. We create a deep convolutional neural network with multiple residual He et al. (2015) connections inspired by the architectures in Song & Ermon (2019) and Song & Ermon (2020). In particular, our model consists of a single 3×3 convolution layer followed by series of 16 convolutional residual and residual-downsampling blocks in the form of an encoder block of typical convolution encoders. Each of these blocks consist of convolutional and Group normalization Wu & He (2018) layers with a group size 32. We then learn the input features by gradually down-sampling them to smaller resolutions. Upon feature map reduction at the end of the last convolution layer, we introduce a single hidden dense layer of 256 units to jointly learn the compressed σ features along with the noise scales, followed by a final dense layer which outputs scalar energy values. We use ELU non-linearity for all

our EBM layers except the last energy layer which has no activation. We then train this EBM model to denoise the noise perturbed σ solutions via the objective seen in Eq. 10 for 2000 epochs, with a batch size of 64, using ADAM optimizer with a fixed learning rate of 0.0001. The training time takes around 15 hours while using a NVIDIA Titan RTX GPU. After the EBM $E^*_\phi(\sigma, \mu)$ is trained,

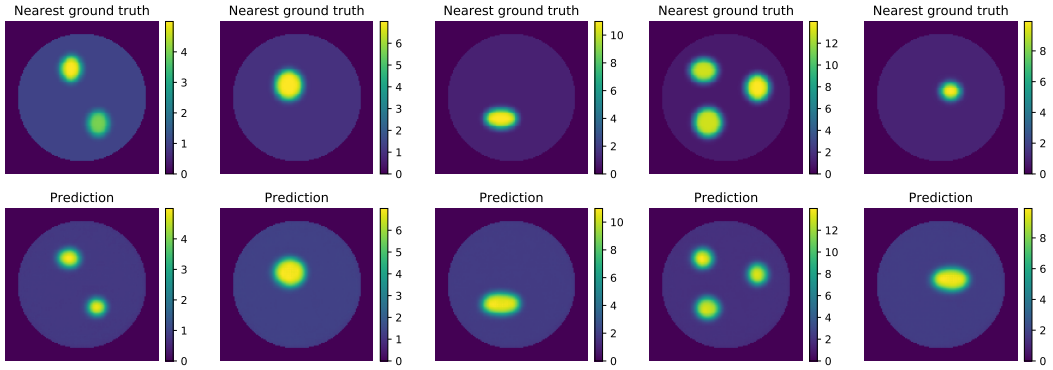

Figure 5: Comparison of training set samples to nearest generated σ using annealed Langevin dynamics

we can generate samples by using annealed Langevin dynamics (LD) [DSM] sampling. We start from a fixed prior distribution such as uniform noise and initially run LD for 100 steps with a step size of s_{n_1}, using first μ_1 noise scale and draw samples from $E_\phi(., \mu_1)$ by adding Gaussian noise. Next we draw samples from $E_\phi(., \mu_2)$ by reducing the step size and refine the samples. We continue running LD until all the noise scales are used to sample and we finally arrive at the final step size $s_{n_L} = a\mu_i^2/\mu_L^2$, where $a = 0.0002$ in our inference run. During the final step of last noise scale μ_L, we perform 1 step of gradient descent instead of LD to obtain an better denoised version of the σ generations. In Fig. 5, we display some of these curated σ solutions along with their nearest neighbors in the training set which are obtained by calculating the L_2 distance between all training set examples.

A.4 Full-Inverse EIT

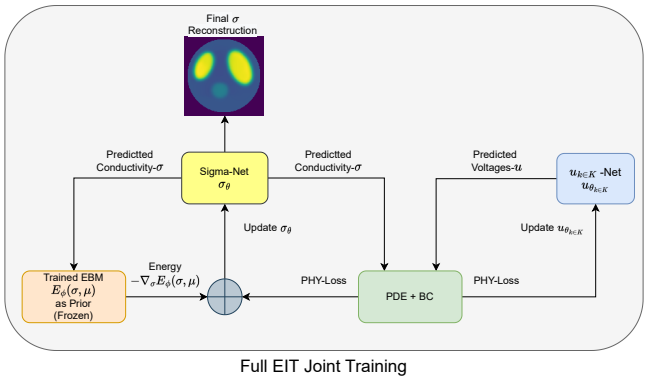

Figure 6: Framework for learning Full-inverse EIT problem with EBM priors

We share some implementation details and evaluations of a much harder and highly ill-posed inverse problem of EIT. By problem definition, given a set of boundary voltages at N_e electrode locations and its corresponding current g, the goal is to recover the target σ and u inside Ω jointly while all the networks are initiated from scratch. We train all PINN models based on the update routine shown in Fig. 6. Here, given the same set of collocation points $(x, y) \in \mathbb{R}^2$ for any random mini-batch, we first update all our K u-Net models based on the objective function shown in Eq. 13. We then

utilize these updated models to make predictions during the inverse problem update of σ-Net using Eq. 14.

$$\mathcal{L}_U = \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{\theta_{u_k\text{-Net}}} \tag{13}$$

$$\mathcal{L}_S = \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{\theta_k} + \frac{1}{N_b}\sum_{i=1}^{N_b}\sum_{i=1}^{N_e}|\sigma_i - \sigma_{b,e}| + \frac{\tau}{N_d}\sum_{i=1}^{N_d}|\nabla\sigma| + \zeta\,\|w_\sigma\|^2 + \upsilon\max(0, 1-\sigma) - \kappa E^*_\phi(\widetilde{\sigma}, 0.01) \tag{14}$$

As seen in Eq. 13 and Eq. 14, we use $K = 8$ which correspond to multiple current patterns g from Eq. 2 in our joint training scheme. Here, each current pattern is learnt a separate u-Net like model and one σ-Net model learns the conductivity distribution inside Ω. Essentially, we obtain K loss values from K current patterns which are then averaged and used to update every u-Net and σ-Net model simultaneously. We present some of the phantoms along with their σ-Net reconstructions used in evaluation of this joint training strategy in Fig. 7. As seen in this figure, we are able to

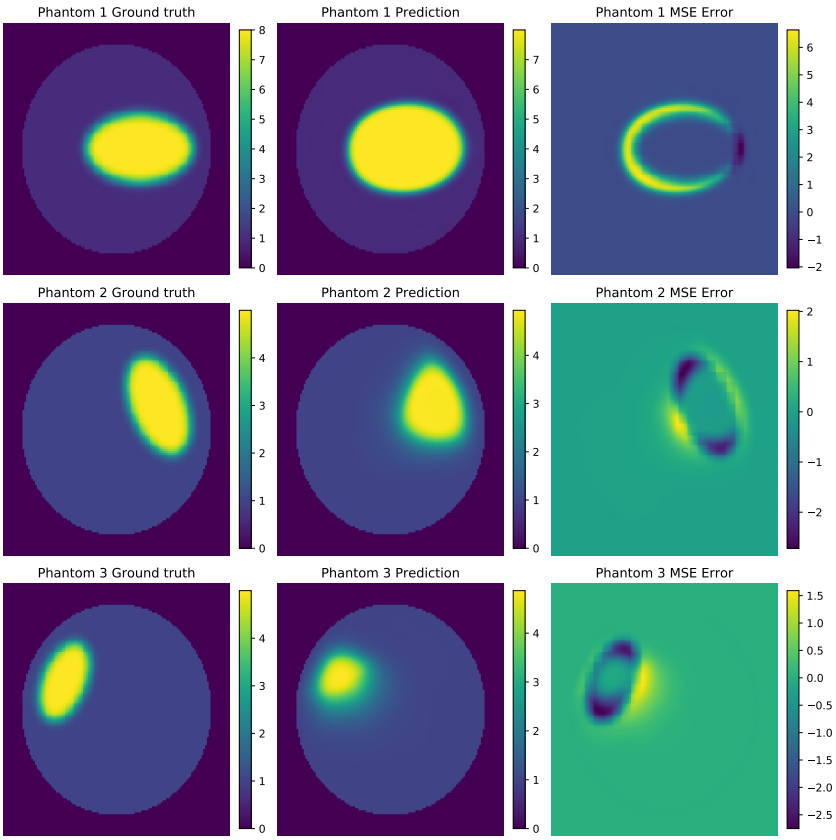

Figure 7: Unsupervised Full Inverse EIT evaluation on some considered phantoms

recover σ accurately for simple anomalies like row one. However, when the phantom configuration becomes complex, the joint training fails. We reserve this for our future work in scope of improving this method further.