

Article

Implicit Solutions of the Electrical Impedance Tomography Inverse Problem in the Continuous Domain with Deep Neural Networks

Thilo Strauss ^{1,†} and Taufiqar Khan ^{2,*,†}

¹ Research Department at ETAS GmbH, Robert Bosch GmbH, 70469 Stuttgart, Germany

² Department of Mathematics and Statistics, University of North Carolina at Charlotte, Charlotte, NC 28223, USA

* Correspondence: tkhan13@uncc.edu

† These authors contributed equally to this work.

Abstract: Electrical impedance tomography (EIT) is a non-invasive imaging modality used for estimating the conductivity of an object Ω from boundary electrode measurements. In recent years, researchers achieved substantial progress in analytical and numerical methods for the EIT inverse problem. Despite the success, numerical instability is still a major hurdle due to many factors, including the discretization error of the problem. Furthermore, most algorithms with good performance are relatively time consuming and do not allow real-time applications. In our approach, the goal is to separate the unknown conductivity into two regions, namely the region of homogeneous background conductivity and the region of non-homogeneous conductivity. Therefore, we pose and solve the problem of shape reconstruction using machine learning. We propose a novel and simple jet intriguing neural network architecture capable of solving the EIT inverse problem. It addresses previous difficulties, including instability, and is easily adaptable to other ill-posed coefficient inverse problems. That is, the proposed model estimates the probability for a point of whether the conductivity belongs to the background region or to the non-homogeneous region on the continuous space $\mathbb{R}^d \cap \Omega$ with $d \in \{2, 3\}$. The proposed model does not make assumptions about the forward model and allows for solving the inverse problem in real time. The proposed machine learning approach for shape reconstruction is also used to improve gradient-based methods for estimating the unknown conductivity. In this paper, we propose a piece-wise constant reconstruction method that is novel in the inverse problem setting but inspired by recent approaches from the 3D vision community. We also extend this method into a novel constrained reconstruction method. We present extensive numerical experiments to show the performance of the architecture and compare the proposed method with previous analytic algorithms, mainly the monotonicity-based shape reconstruction algorithm and iteratively regularized Gauss–Newton method.

Keywords: inverse problems; electrical impedance tomography; implicit solutions; deep learning; neural networks; continuous domains



Citation: Strauss, T.; Khan, T. Implicit Solutions of the Electrical Impedance Tomography Inverse Problem in the Continuous Domain with Deep Neural Networks. *Entropy* **2023**, *25*, 493. <https://doi.org/10.3390/e25030493>

Academic Editor: Jaroslaw Krzywanski, Yunfei Gao, Marcin Sosnowski, Karolina Grabowska, Dorian Skrobek, Ghulam Moeen Uddin, Anna Kulakowska, Anna Zylka and Bachil El Fil

Received: 21 December 2022

Revised: 7 March 2023

Accepted: 8 March 2023

Published: 13 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The electrical impedance tomography (EIT) inverse problem is a severely ill-posed inverse problem that attempts to reconstruct the unknown conductivity of an object Ω from boundary electrode measurements. EIT is a great alternative to CT scans, particularly for neonatal brain imaging, where CT can be harmful. EIT has been successfully used in medical imaging [1,2], monitoring soil [3], or crack detection [4]. The EIT forward problem is an elliptic differential equation:

$$-\nabla \cdot (\sigma \nabla u) = 0 \text{ in } \Omega, \quad (1)$$

where $u \in H^1(\Omega)$ is the electric potential, $\sigma \in L^\infty(\Omega)$ is the known conductivity with $0 < \sigma_1 \leq \sigma \leq \sigma_2 < \infty$, and there are no current sources inside Ω . We assume the electric conductivity σ is a bounded, positive, and isotropic conductivity distribution with $\sigma \geq \sigma_0 > 0$ for some $\sigma_0 \in \mathbb{R}$. Before solving (1), we must formulate a forward model that incorporates boundary conditions about the EIT forward problem. Once we have the EIT forward model, we can proceed to solve its inverse problem. In theory, the EIT inverse problem is exponentially ill posed and highly nonlinear. Even if a solution exists for (1), it will be highly sensitive to any noise or perturbations in the data. Therefore, in practice, we must impose a priori knowledge through some form of regularization. In the past, research topics in EIT mainly focused on analytical and numerical methods. Typically, the conductivity is discretized into triangles (2D) or tetrahedrons (3D). This discretization allows solving the PDE of the forward problem via the finite element method (FEM).

To address the poorly posed EIT inverse problem, significant research has been dedicated to developing deterministic and statistical reconstruction methods to improve the recovered tomographic images [5–9]. Deterministic and statistical methods enforce regularization to provide reasonable image reconstructions. However, in many applications, tomographic images of decent quality do not suffice. Often, high-resolution tomographic reconstructions are required for accurate diagnostic analyses or the detection of anomalies.

1.1. Contributions

We propose a new formulation of the EIT inverse problem. In particular, instead of reconstructing the conductivity function at each point, we find a function that predicts if a point $p \in \Omega$ belongs to the background conductivity or not. Here, the background conductivity refers to the known conductivity that the homogeneous portion of Ω should have. The assumption that such conductivity is known is also common in the analytical/numerical setting. Some regularization use this conductivity as an a priori assumption for parametrization [7]. Furthermore, most optimization algorithms require some knowledge of the background conductivity as an initialization of the unknown conductivity [10].

Our goal is to define a function $f : \mathbb{R}^d \rightarrow \{0, 1\}$ with $d \in \{2, 3\}$ that for any point $p \in \mathbb{R}^d$ estimates if p is a background conductivity. We parametrize f with a neural network. In detail, this neural network requires three network blocks. We call the first neural network the measure encoder $f_{measure}$. It takes the measurements m of an EIT experiment and maps them to a measurement embedding $m_e \in M \subseteq \mathbb{R}^{dim(m_e)}$. Here, $dim(m_e)$ refers to the dimension of the embedding space. A second neural network called the point encoder f_{point} takes a point $p \in \mathbb{R}^d$ and maps it to a point embedding p_e . We call the third network the decoder network $f_{decoder}$. It takes the point embedding conditioned on measure embedding to predict if a point p is from the background conductivity. In summary, we write the novel network architecture as $f : \mathbb{R}^d \times M \rightarrow \{0, 1\}$. See Figure 1 for a visualization of the method. Observe that for solving one inverse problem, the measurement embedding m_e only has to be computed once. On the other hand, we compute the point encoder f_{point} and the decoder network $f_{decoder}$ for every point $p \in \mathbb{R}^d$ on that for which we solve the inverse problem.

To show the use of the shape estimation method for reconstructing the unknown conductivity distribution σ , we combine it with gradient-based optimization methods. In detail, we introduce a constraint and a piecewise optimization problem.

In summary, we make the following contributions:

- We present a novel method to solve the EIT shape estimation problem in a continuous domain using machine learning.
- We introduce a constraint and a piece-wise gradient-based optimization problem by using the shape reconstruction from the proposed machine learning algorithm.
- We show that the proposed method using machine learning outperforms the benchmark approaches which do not involve machine learning.

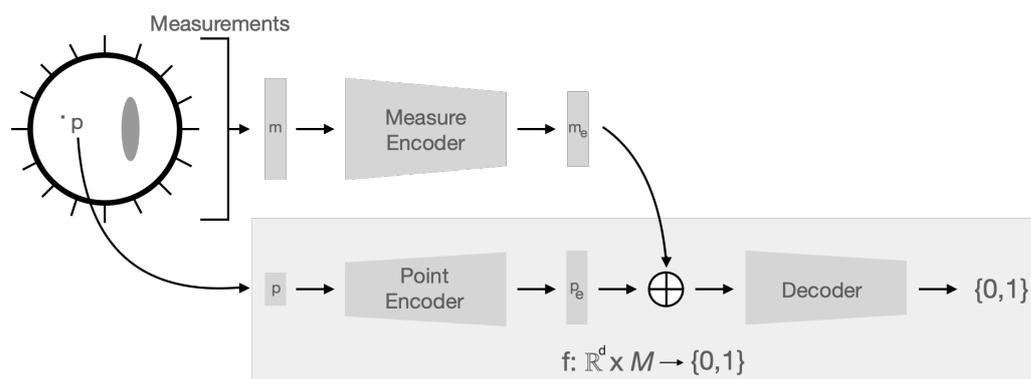


Figure 1. The image represents a schematic description of the proposed network architecture. Here, $d \in \{2, 3\}$ is the dimensionality of the EIT inverse problem. The implicit architecture is based on two main fully connected networks, a measurement encoder taking measurements as input and a point encoder that has a point in \mathbb{R}^d as input. The outputs are added. Then a fully connected decoder network predicts the probability of a point (the input of the point encoder) being of the background or anomaly type.

1.2. Related Work

An EIT experiment involves applying an electrical current (Neumann data) on $\partial\Omega$, the boundary of the body Ω , to measure the corresponding electrical potential differences on $\partial\Omega$. Hereby, the information on the Neumann-to-Dirichlet (NtD) operator Λ_σ is obtained. The estimate of the unknown conductivity σ is then reconstructed from a set of EIT experiments [11–14].

There are many applications of EIT, including medical imaging [1,2], monitoring soil [3], or crack detection [4]. There is also work on hardware design in related problems [15]. It is well known that EIT is highly nonlinear and strongly ill posed. Therefore, EIT requires regularization for proper reconstruction [5,6]. In the literature, there are abundant deterministic approaches for solving the EIT inverse problem, including the factorization method [16], d-bar method [17], or variational methods for least-square fitting [5,6,10,18]. There also exists previous work on statistical methods for solving the EIT problem [7,10,19,20]. There are deterministic approaches to reconstructing the shape of the abnormalities, for example, the monotonicity shape estimate [21] or the level set method [22,23]. Due to the simplicity and computational speed, we choose to compare our method in this paper with the monotonicity shape estimate [21] and iteratively regularized Gauss–Newton (IRGN) method [10].

With the rise of deep learning, it has become state of the art in many tasks, such as image recognition [24], strategy board games [25], or protein folding problems [26]. While these results have been impressive, deep learning did not become state of the art in EIT or other ill-posed inverse problems. However, solving the EIT inverse problem with deep learning is receiving increasing attention, for example, with radial bases functions [27], fully connected neural networks [28], and in particular by using convolutional neural networks [29–31]. While convolutional neural networks are attractive in the 2D inverse problem, they start having a limitation in resolution when solving the 3D problem. This limitation is because the number of voxels grows cubically as does the computational cost of the voxel representation. In 3D computer vision, many applications work in a voxelization of usually around 32^3 or 64^3 [32]. It is questionable whether this is sufficient for applications of EIT. Since 2019, in the computer vision community, implicit representations of 3D objects via neural networks replaced voxel-based and mesh-based methods as the state of the art for several applications. Some of these results include single image 3D reconstructions [33,34], representing texture on 3D objects [32], representing surface light fields in 3D [35], or 3D reconstructions from many images [36].

1.3. Rationale for Proposed Shape Reconstruction Using Machine Learning

It is well known that the EIT shape reconstruction inverse problem for piecewise-constant conductivity suffers from logarithmic instability as shown in [37]. The stability of the shape reconstruction for the EIT inverse problem also depends on the type of assumptions on the parameter space and the analytical technique used, for example, if we consider conductivity that belongs to a finite-dimensional set of piecewise-analytic functions that are bounded from above and below by a priori known constants and use theoretical technique using monotonicity. Then the stability is of the type Lipschitz [21]. There is no constructive estimate on C , and the assumption is that infinite information on the Dirichlet-to-Neumann map Λ is known. Here, C refers to the right hand side constant used in the following Lipschitz stability inequality $\|\sigma_1 - \sigma_2\|_{L^\infty(\Omega)} \leq C \|\Lambda_{\sigma_1} - \Lambda_{\sigma_2}\|_*$, where $*$ represents the operator norm of $\Lambda : L^2_\diamond(\partial\Omega) \rightarrow L^2_\diamond(\partial\Omega)$ and $L^2_\diamond(\partial\Omega)$ is the $L^2(\partial\Omega)$ space with vanishing integral mean on the boundary [38].

Using a completely different approach using the Bayesian formulation, one can obtain the (locally) Lipschitz stability of the probability distribution of unknown domains for the general shape identification inverse problem for a heat cavity problem that in principle should also be applicable to EIT [39]. Therefore, the proposed machine learning approach combining both statistical and deterministic shape reconstruction techniques should provide better stability. The focus of this manuscript is to demonstrate the efficacy of the proposed approach computationally.

2. Materials and Methods

In this section, we first describe the preliminary neural network blocks. Hereby, we aim to unify the basic knowledge of the terms needed to describe our network architecture. Then we describe the proposed network architecture in detail. This is followed by an explanation of how the training data are generated, what noise setting is used, and how the neural network is trained.

2.1. Leaky Rectified Linear Units

Leaky rectified linear units (Leaky ReLu) [40] is a nonlinear function used as an activation function in our architecture. We write Leaky ReLu as

$$\text{LeakyReLU}(x) := \begin{cases} 0.01x & \text{if } x < 0 \\ x & \text{else.} \end{cases}$$

2.2. Residual Neural Network Blocks

A residual neural network (ResNet) is a neural network with shortcuts between layers [41]. We use fully connected ResNet blocks with Leaky ReLu activations. It is among the most common building block in state-of-the-art networks. Here, we use the following variant: if the input and output sizes differ, we use a single fully connected network as a shortcut instead of copying the input data. Find a visualization of ResNet in Figure 2.

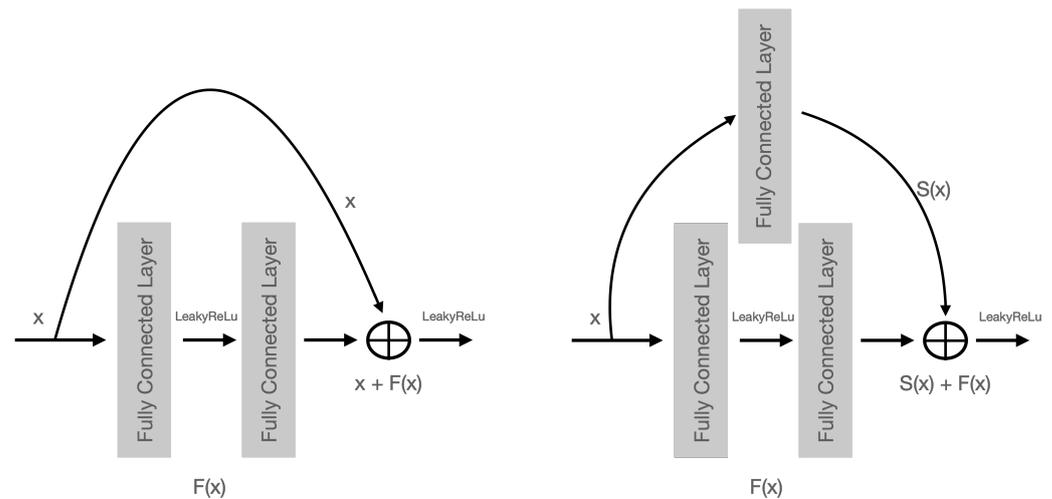


Figure 2. A schematic picture of the residual neural network (ResNet) versions that we use as the main building block in our implicit architecture. On the left-hand side is the standard ResNet block with Leaky ReLu activation. On the right-hand is a modified ResNet block that can handle different input–output sizes.

2.3. Softmax

One uses the softmax function to convert an array $x = (x_1, \dots, x_n)$ into a probability distribution. We write softmax as

$$\text{softmax}(x) := \left(\frac{\exp(x_1)}{\sum_i \exp(x_i)}, \dots, \frac{\exp(x_n)}{\sum_i \exp(x_i)} \right).$$

2.4. Implicit Network Architecture for EIT

The proposed network architecture requires three network blocks that operate on each other. Find the visualization of the network architecture in Figure 1.

When we perform an EIT experiment, we obtain the boundary electrode measurements m . We represent these measurements in the form of an array. Then, we use a network architecture called the measure encoder network $f_{measure}$ to map m to its corresponding embedding $m_e \in M \subseteq \mathbb{R}^{dim(m_e)}$. Here, the embedding m_e is an array that stores the relevant information of the measurements. We write the $f_{measure}$ as a series of ResNet blocks. See Table 1 for the exact setup.

For us, to solve the EIT inverse problem implicitly means to estimate if a point $p \in \mathbb{R}^d \cap \Omega$ with $d \in \{2, 3\}$ belongs to the background conductivity. To find this estimate, we first define a point encoder f_{point} . The point encoder takes a point $p \in \mathbb{R}^d \cap \Omega$ and maps it to a point embedding p_e . The point embedding is an array of the same dimensionality as the measure embedding m_e . We describe the architecture of the point encoder f_{point} in Table 2.

We call the last part of the implicit architecture the decoder network $f_{decoder}$. The input array of $f_{decoder}$ is computed by adding the point and measure embedding $p_e + m_e$. Hence, we condition the point embedding p_e on the measure embedding m_e . With the conditioning term, we want to emphasize that the measure embedding only needs to be computed once, while we compute the point embedding for every point at which we solve the inverse problem. The decoder network consists of a series of ResNet blocks mapping to a two-dimensional array. We apply the softmax function to the output to obtain a probability. Then the network predicts that the point is in the background if the first probability is larger than the second in the array. Otherwise, it predicts that the point is anomalous. The network architecture is summarized in Table 3.

Table 1. Measure encoder architecture.

Network Block	Input Dimension	Output Dimension
ResNet	208	128 * scale
ResNet	128 * scale	128 * scale
ResNet	128 * scale	128 * scale
ResNet	128 * scale	64 * scale
ResNet	64 * scale	64 * scale
ResNet	64 * scale	64 * scale

Table 2. Point encoder architecture.

Network Block	Input Dimension	Output Dimension
ResNet	2 or 3	32 * scale
ResNet	32 * scale	64 * scale

Table 3. Decoder architecture.

Network Block	Input Dimension	Output Dimension
ResNet	64 * scale	32 * scale
ResNet	32 * scale	2
Softmax	2	2

2.5. Generating Training Data

In this section, we briefly describe the well-studied EIT forward problem. We emphasize that, here, our goal is not to provide a complete description of the forward problem. Refer to [11–13] for a theoretical discussion and [42] for a description of the the finite element method. Based on the EIT forward problem, we describe how to generate data suitable for training the network architecture in Figure 1.

2.6. The EIT Forward Problem

The EIT forward problem finds the boundary electrode measurements for a known conductivity distribution σ on Ω . We write the partial differential equation as

$$\begin{aligned} \nabla \cdot (\sigma \nabla u) &= 0 \text{ on } \Omega \\ \sigma \frac{\partial u}{\partial n} &= g_1 \text{ on } \partial\Omega \quad (\text{Neumann Input Current}) \\ u &= g_2 \text{ on } \partial\Omega \quad (\text{Dirichlet Measured Voltage}) \\ \int_{\partial\Omega} u &= 0. \end{aligned}$$

where g_1 is the injected current and g_2 is the measured voltage.

Our experiments use a simple electrode model, where current flows into Ω through a set of boundary electrodes. Specifically, we use a point electrode model, where we use one electrode as a source and its right-hand neighbor as a sink and repeat this for each possible setup. For one source-sink setting, we compute the voltage difference between each electrode and its right-hand neighbor if none of the electrodes is a source nor a sink electrode. Our experiments used 16 electrodes on $\partial\Omega$. Hence, we have 13 measurements per source–sink setting, which results in 208 voltage differences for all source–sink settings that we call the measurements m .

As described in [42], the finite element method solves the EIT forward problem for a known into a mesh discretized conductivity distribution σ . However, to train the proposed network architecture, one needs the measurements m , a point $p \in \mathbb{R}^d \cap \Omega$, and whether or not the p belongs to the background conductivity.

2.7. Sampling Point Clouds

To train our neural network, we need to generate a data set of points and their labels. One can sample a point cloud from $\mathbb{R}^d \cap \Omega$ in many different ways, for example, by taking uniform distributed points. We found that uniform distributed points do not produce the best performance for training the network. For abnormalities with a small area, few elements of the point cloud will be within that anomaly. Hence, the resulting network is not strong in detecting these abnormalities.

The object Ω is separated into the background and abnormal strata. We sample uniformly the same number of points from each group. In the experimental part, we used 512 points per group of a given conductivity σ . This sampling procedure lets the neural network better perform for detecting small abnormal areas correctly.

2.8. Data Sets

We generate a training dataset consisting of 16,384 discretized conductivities and their corresponding measurements $\{m_i\}_{i=1}^{16,384}$. The training data set contains 4096 conductivities with one, two, three, and four circular abnormalities, respectively. We randomly chose the radius and the conductivities of each circular object. The background conductivities are constant 1 in the entire dataset. Similarly, we build an analog validation data set of 1024 conductivities and their measurements m . Finally, we handcraft a small test data set to compare our methods with baseline approaches.

2.9. Noise

In this document, Gaussian measurement noise of a specific noise level of $\delta\%$ refers to

$$m_{noisy} = m + \mathcal{N}_{Size(m)}(0, \max(m)) * \frac{\delta}{100}.$$

Note that m refers to the measurement array of length 208.

2.10. Training

In this section, we describe the training procedure of the proposed neural network architecture. We use the cross-entropy loss [43],

$$loss(y, q) = -(y \log(q) + (1 - y) \log(1 - q)),$$

where y is the binary indicator of a point being anomalous and q the computed probability of being anomalous. All neural networks are trained for 1000 epochs. One epoch refers to moving once over the entire training data set. We use batch size 64. During training for each batch element, we sample a point cloud of 1024 points. Hence, one batch requires 64 forward operations of the measure encoder $f_{measure}$, while it requires 65,536 forward operations of the point encoder f_{point} and the decoder $f_{decoder}$.

All measurements m are 0–1 normalized with the maximum–minimum from the training data. We also 0–1 normalize the points p . During training, different results, some with and some without noise, are presented. Training with noise implies that we add new Gaussian noise to each measurement m in each epoch.

For training, the ADAM optimizer [44] is used with a starting learning rate of $5e^{-4}$. We reset the optimizer every ten epochs and multiply the learning rate by 2/3. All parameters provided in this section aim to make the method reproducible. Most parameters were chosen ad hoc, but some key parameters, such as the learning rate, were iteratively improved to increase the training speed/stability. We trained all networks on a single Nvidia GeForce RTX 2060.

3. Results

In this section, we present our experimental results. We first show some numerical results on the training noise and an ablation study on the architecture choice. Then the network architecture is compared with baseline methods.

3.1. Noise Study

In a first experiment, we evaluate what level of training noise lets the network architecture perform best on the validation data set. Hence, we train multiple different models by adding different levels of noise to the training data. The noise is generated independently in every training epoch. For simplicity, we consider the network architecture with a scale of 2. Please see Tables 1–3 for the meaning of the scale value.

In Table 4, we compute the accuracy and standard deviation by evaluating the network on the center of the triangles of the original mesh. Hereby, we do not commit an inverse crime because the network is capable of solving the inverse problem in the entire domain $\mathbb{R}^2 \cap \Omega$. The accuracy for one measurement m is the ratio of points correctly labeled as either background or inclusion, i.e., $\text{accuracy} = \frac{\sum_{i \in \text{points}} I(\text{Prediction}_i == \text{Truth}_i)}{\text{Number of points}}$, where I is the indication function. We observe that the network has the best accuracy with a training noise of around 0.1%. For validation noise levels over 3%, larger training noise increases the performance by reducing the standard deviation. See Figure 3 for several reconstructions of the validation data set.

Table 4. Noise study.

Noise	Accuracy: Mean \pm Standard Deviation			
	Training 0.0%	Training 0.1%	Training 1.0%	Training 2.0%
Validation 0.0%	0.9737 \pm 0.0143	0.9778 \pm 0.0123	0.9623 \pm 0.0170	0.9591 \pm 0.0182
Validation 1.0%	0.9711 \pm 0.0160	0.9754 \pm 0.0139	0.9617 \pm 0.0175	0.9584 \pm 0.0187
Validation 2.0%	0.9643 \pm 0.0189	0.9686 \pm 0.0168	0.9590 \pm 0.0184	0.9566 \pm 0.0194
Validation 3.0%	0.9571 \pm 0.0219	0.9612 \pm 0.0199	0.9540 \pm 0.0199	0.9529 \pm 0.0206
Validation 4.0%	0.9503 \pm 0.0241	0.9545 \pm 0.0223	0.9485 \pm 0.0217	0.9476 \pm 0.0221
Validation 5.0%	0.9444 \pm 0.0261	0.9487 \pm 0.0247	0.9430 \pm 0.0234	0.9423 \pm 0.0236

For the rest of this document, we train all networks with 0.1% training noise. Note that we considered 5% as the largest noise level for the validation and test data because the reconstructions tend to become unstable, both the location and shape of the inclusions, for larger noise levels.

3.2. Ablation Study

Here, we provide a comparison of several architecture choices. For simplicity, we only present experiments on different scale values (see Table 1–3) for the measurement noise level of 0.1%. Hence, we trained a network independently on each scale value of 1, 2, and 4.

Table 5. Ablation study.

Noise	Accuracy: Mean \pm Standard Deviation		
	Scale 1	Scale 2	Scale 4
Validation 0.0%	0.9638 \pm 0.0203	0.9778 \pm 0.0123	0.9546 \pm 0.0274
Validation 1.0%	0.9614 \pm 0.0219	0.9754 \pm 0.0139	0.9530 \pm 0.0292
Validation 2.0%	0.9548 \pm 0.0244	0.9686 \pm 0.0168	0.9478 \pm 0.0316
Validation 3.0%	0.9477 \pm 0.0279	0.9612 \pm 0.0199	0.9423 \pm 0.0344
Validation 4.0%	0.9420 \pm 0.0307	0.9545 \pm 0.0223	0.9378 \pm 0.0371
Validation 5.0%	0.9372 \pm 0.0332	0.9487 \pm 0.0247	0.9343 \pm 0.0394

In Table 5, we observe that the architecture with scale 2 outperforms the other scale values for all validation noise levels. Hence, for all further experiments, we use the scale value 2. See Figure 3 for several reconstructions of the validation data set with scale 2. Observe that the reconstructions are relatively robust. As expected, they become less accurate in positioning and shape of the anomalies for larger noise levels.

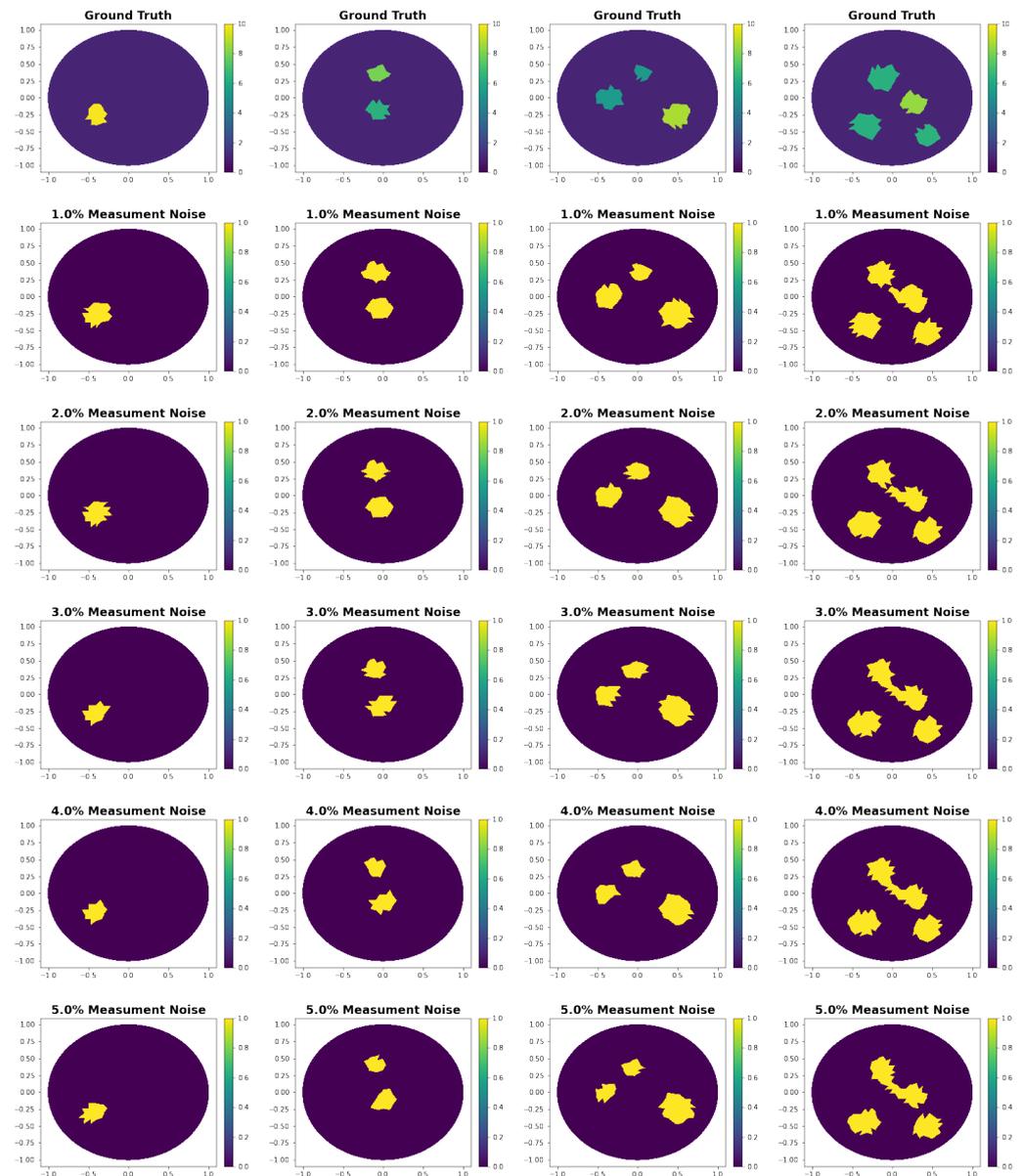


Figure 3. Shape reconstruction using the proposed algorithm with scale 2 and training noise 0.1%. The ground truth images are randomly chosen from the validation data set with 1, 2, 3, and 4 inclusion. The reconstructions are performed at different noise levels. They are evaluated at points at the center of the triangles of the ground truth mesh. Hereby, we do not commit an inverse crime because our method solves the problem on the continuous domain.

3.3. Comparison with Baseline Approaches

In this section, we compare our method with several benchmark algorithms on simulated test conductivities. Note that the validation data set contains conductivities generated from the same distribution as the training data set. This does not hold for the handcrafted test conductivities. Hence, in this section, we also test the generalization capabilities of the network.

3.3.1. Comparison with Monotonicity-Based Shape Reconstruction

In this paper, we compare our method with the algorithm proposed in [21]. It uses a monotonicity test that comes from the simple relation that for two conductivities σ_1 and σ_2 we have that if $\sigma_1 \leq \sigma_2$ implies $\Lambda_{\sigma_1} \geq \Lambda_{\sigma_2}$ where Λ is the Neumann-to-Dirichlet (NtD) operator. The second term is to be understood as $\Lambda_{\sigma_1} - \Lambda_{\sigma_2}$ being positive semi-definite. The idea is to test if the equation holds for many different small balls. Thus, we tested which balls are inside the anomalous region of the unknown conductivity distribution of interest. See [21] for a detailed description of the theory and algorithm. To speed up the method, we precomputed the NtD operator of all small balls on that we perform the test.

In Figure 4, it can be seen that our proposed method outperforms the monotonicity approach in all examples. The monotonicity-based shape reconstruction approach was only evaluated on 0% measurement noise. This is because even for small noise there is typically no positive semi-definite matrix in the tests, thus predicting that there is no anomalous area anywhere. From the numerical results, the proposed method also outperforms the monotonicity test in almost all examples (see Table 6). The computational speed is about 25 times faster using the deep learning approach. In detail, our approach took about 0.007 s while solving the monotonicity test, which about 0.178 s.

Table 6. Comparison with monotonicity.

Image	Accuracy		
	Monotonicity: 0% Noise	Ours: 0% Noise	Ours: 3% Noise
Figure 4, Row 1	0.9843	0.9929	0.9954
Figure 4, Row 2	0.9705	0.9871	0.9754
Figure 4, Row 3	0.9501	0.9695	0.9675
Figure 4, Row 4	0.8975	0.8910	0.8966
Figure 4, Row 5	0.7535	0.9538	0.9511

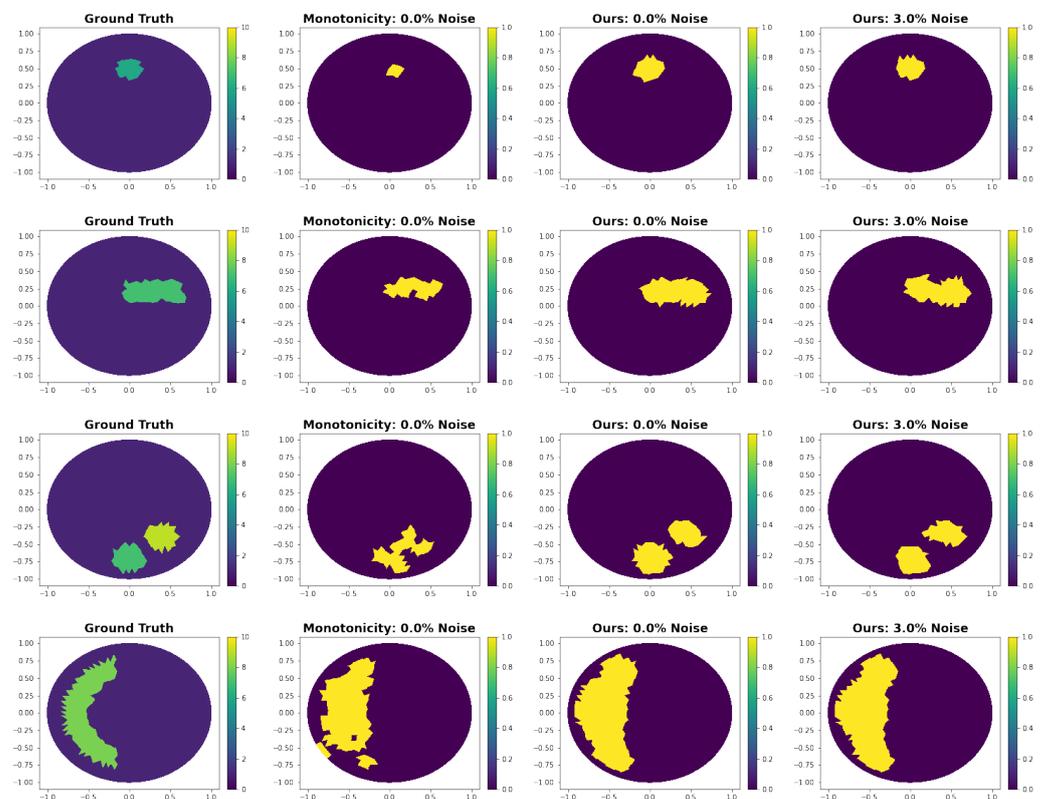


Figure 4. Cont.

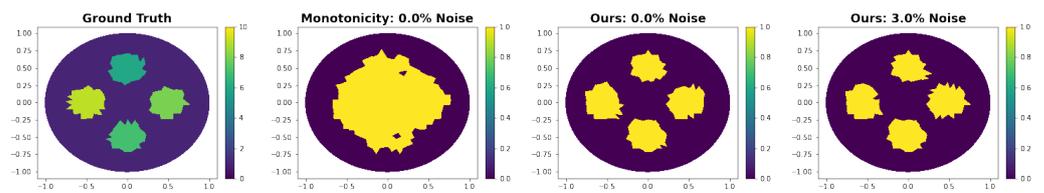


Figure 4. Shape reconstruction comparing our method with the monotonicity-based shape reconstruction algorithm. Due to the noise sensitivity of the monotonicity method, we only evaluated it at a noise level of 0.0% and with the same simulation and reconstruction mesh.

3.3.2. Comparison with Iteratively Regularized Gauss-Newton Method

In this section, we compare with the iteratively regularized Gauss–Newton (IRGN) method as described in [10] with IRGN methods that make use of our shape reconstructions. In detail, we reconstruct the true conductivity on a reconstruction mesh. First, we use the standard Tikhonov-regularized IRGN method, see [10] for a good description of the method. Then, we use a constrained IRGN method. Here, constrained means that we first reconstruct the problem with our shape reconstruction approach. Then, the IRGN method is used to reconstruct the mesh only on the abnormal areas of the predicted shape. Lastly, we use the piecewise IRGN method. By piecewise, we mean that we use the proposed shape reconstruction algorithm and optimize all abnormal areas as separate conductivity parameters, thus obtaining a piecewise constant reconstruction.

Some reconstructions can be seen in Figure 5 and the corresponding numerical results in Table 7. We see that both visually and numerically, it is advantageous to incorporate the shape reconstruction into the IRGN method. The constraint method outperforms the other methods numerically. Visually, the piecewise IRNG method seems to be the most informative.

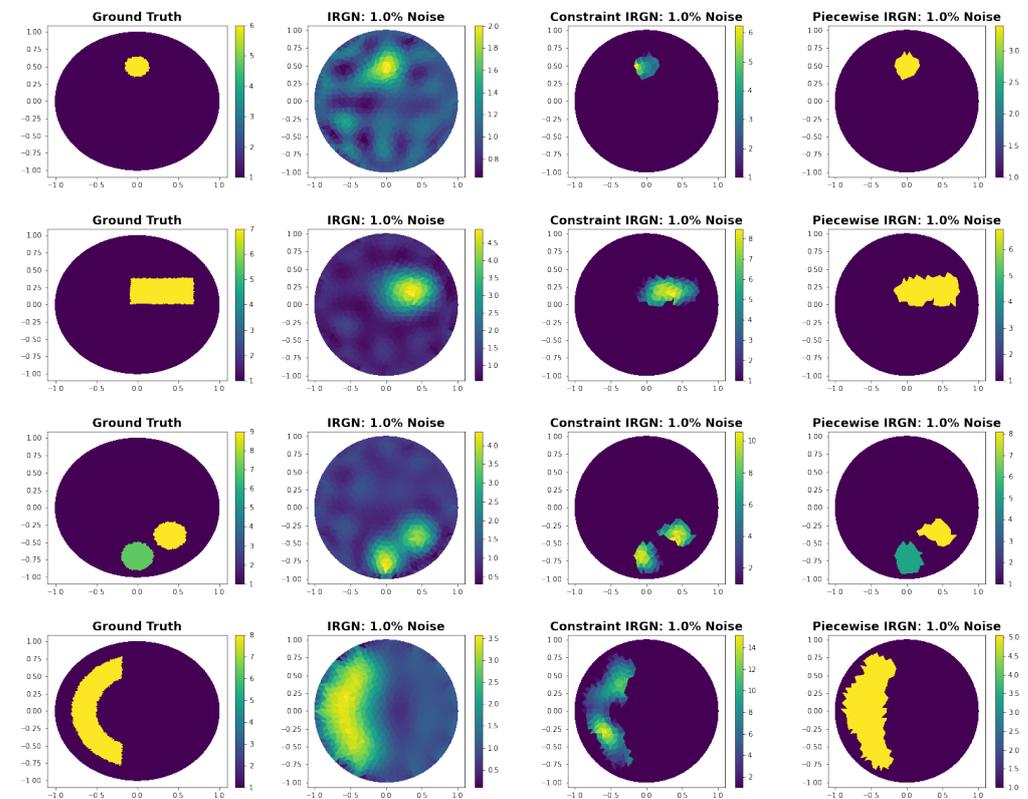


Figure 5. Cont.

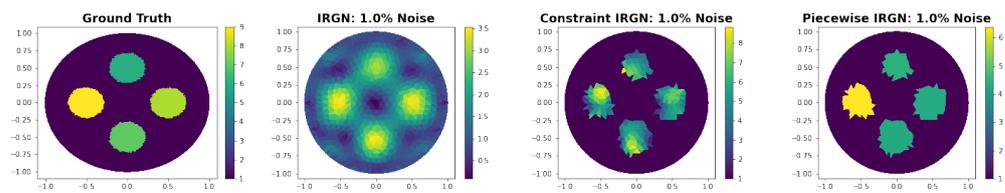


Figure 5. Comparison of the IRGN method with several IRGN methods that are using our shape reconstruction.

Table 7. Comparison with iteratively regularized Gauss–Newton method.

L_2 Norm			
Image	IRGN: 1% Noise	Constraint IRGN: 1% Noise	Piecewise IRGN: 1% Noise
Figure 5, Row 1	1.1635	0.7413	0.7690
Figure 5, Row 2	1.8013	1.3332	1.8467
Figure 5, Row 3	2.5513	2.0178	2.0959
Figure 5, Row 4	3.0121	2.8924	3.1042
Figure 5, Row 5	3.8958	2.8328	2.9130

3.4. Super-Resolution Shape Reconstruction

In the previous experiments, we only evaluated our method on the reconstruction mesh for providing a fair comparison. Here we use, similarly to the IRGN experiments in Figure 5, a much finer mesh than the one on that which our method is trained. We then evaluate our method on the same mesh. We note that this is not an inverse crime because our method directly solves the problem in the continuous domain.

In Figure 6, we see that even though the method was trained on data from a less fine mesh. It performs very well on data on the finer mesh. While for large noise levels the shape becomes less accurate, it still finds the correct position of the anomaly. This indicates that one could train the network with easily accessible simulated data and still obtain good performance on real experimental data.

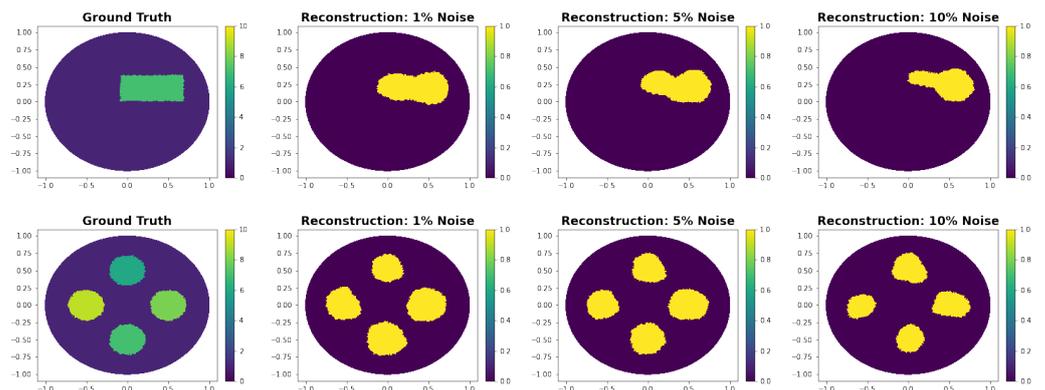


Figure 6. Super-resolution experiments are evaluations of the model trained on a less fine mesh. Here, the data are generated in a fine mesh and reconstructed on the same mesh. This is not an inverse crime because our method directly solves the problem in the continuous domain.

3.5. Difficulties of the Proposed Method

The main difficulty of the proposed method is that it assumes that the background conductivity is known. However, we note that many other methods also assume this to some extent. Further, problems with this and any other method based on neural networks are that they require an extensive training set to perform well. This is, in particular, problematic when one wants to use real training data because it is costly and time intensive

to create a large number of phantoms and measurements for such an approach. Hence, our training is entirely on simulated training data.

4. Discussion and Conclusions

We presented a novel neural network-based approach for reconstructing the shape of anomalies for the EIT inverse problem. The method is capable of solving the problem implicitly in the continuous domain. We discussed the theoretical results on the level of ill posedness of the shape reconstruction algorithm to provide some context for the proposed method. The proposed design flow can be used for any coefficient inverse problem, such as EIT. It can be easily applied to any inverse problem that aims to reconstruct unknown images or physical properties from some measurements. The only requirement for using this method is the existence of a properly defined forward model.

Numerical experiments showed that low measurement noise during training the neural network improved the performance on noisy test data. Further numerical experiments with different network parameters were performed to find a good network architecture. Our shape reconstruction algorithm outperformed the monotonicity-based shape reconstruction numerically, visually, and in terms of computational speed. We also proposed using the reconstructed shapes to improve gradient-based methods. In particular, we presented a piecewise and a constrained iteratively regularized Gauss–Newton (IRGN) method. We found that the constrained method outperforms the piecewise and the standard IRGN algorithm numerically. However, the piecewise method was visually the most informative and provided a piecewise constant reconstruction of the conductivity. Furthermore, we presented high-resolution reconstruction with different noise levels and found that the reconstructions using the proposed algorithm are very robust.

In summary, we demonstrate the efficacy of our approach. The proposed approach is simple, easy to transfer to other inverse problems, efficient in terms of computational speed, and able to solve the EIT problem directly in the continuous domain.

Author Contributions: Conceptualization, T.S. and T.K.; methodology, T.S. and T.K.; software, T.S.; validation, T.S. and T.K.; formal analysis, T.K.; investigation, T.S.; resources, T.K.; data curation, T.S.; writing—original draft preparation, T.S.; writing—review and editing, T.K.; visualization, T.S.; supervision, T.K.; project administration, T.K.; funding acquisition, T.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Holder, D.S. *Electrical Impedance Tomography: Methods, History and Applications*; CRC Press: Boca Raton, FL, USA, 2004.
2. Bayford, R.H. Bioimpedance tomography (electrical impedance tomography). *Annu. Rev. Biomed. Eng.* **2006**, *8*, 63–91. [[CrossRef](#)]
3. Daily, W.; Ramirez, A.; LaBrecque, D.; Nitao, J. Electrical resistivity tomography of vadose water movement. *Water Resour. Res.* **1992**, *28*, 1429–1442. [[CrossRef](#)]
4. Hou, T.C.; Lynch, J.P. Electrical impedance tomographic methods for sensing strain fields and crack damage in cementitious structures. *J. Intell. Mater. Syst. Struct.* **2009**, *20*, 1363–1379. [[CrossRef](#)]
5. Jin, B.; Khan, T.; Maass, P. A reconstruction algorithm for electrical impedance tomography based on sparsity regularization. *Int. J. Numer. Methods Eng.* **2012**, *89*, 337–353. [[CrossRef](#)]
6. Jin, B.; Maass, P. An analysis of electrical impedance tomography with applications to Tikhonov regularization. *ESAIM Control. Optim. Calc. Var.* **2012**, *18*, 1027–1048. [[CrossRef](#)]
7. Kaipio, J.P.; Kolehmainen, V.; Somersalo, E.; Vauhkonen, M. Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography. *Inverse Probl.* **2000**, *16*, 1487. [[CrossRef](#)]
8. Nissinen, A.; Heikkinen, L.; Kaipio, J. The Bayesian approximation error approach for electrical impedance tomography—experimental results. *Meas. Sci. Technol.* **2007**, *19*, 015501. [[CrossRef](#)]

9. Nissinen, A.; Heikkinen, L.; Kolehmainen, V.; Kaipio, J. Compensation of errors due to discretization, domain truncation and unknown contact impedances in electrical impedance tomography. *Meas. Sci. Technol.* **2009**, *20*, 105504. [[CrossRef](#)]
10. Ahmad, S.; Strauss, T.; Kupis, S.; Khan, T. Comparison of statistical inversion with iteratively regularized Gauss Newton method for image reconstruction in electrical impedance tomography. *Appl. Math. Comput.* **2019**, *358*, 436–448. [[CrossRef](#)]
11. Borcea, L. Electrical impedance tomography. *Inverse Probl.* **2002**, *18*, R99. [[CrossRef](#)]
12. Cheney, M.; Isaacson, D.; Newell, J.C. Electrical impedance tomography. *SIAM Rev.* **1999**, *41*, 85–101. [[CrossRef](#)]
13. Hanke, M.; Brühl, M. Recent progress in electrical impedance tomography. *Inverse Probl.* **2003**, *19*, S65. [[CrossRef](#)]
14. Lionheart, W.R. EIT reconstruction algorithms: pitfalls, challenges and recent developments. *Physiol. Meas.* **2004**, *25*, 125. [[CrossRef](#)] [[PubMed](#)]
15. Kassanos, P.; Triantis, I.F.; Demosthenous, A. A CMOS magnitude/phase measurement chip for impedance spectroscopy. *IEEE Sens. J.* **2013**, *13*, 2229–2236. [[CrossRef](#)]
16. Kirsch, A.; Grinberg, N. *The Factorization Method for Inverse Problems*; Number 36; Oxford University Press: Oxford, UK, 2008.
17. Isaacson, D.; Mueller, J.L.; Newell, J.C.; Siltanen, S. Reconstructions of chest phantoms by the D-bar method for electrical impedance tomography. *IEEE Trans. Med. Imaging* **2004**, *23*, 821–828. [[CrossRef](#)] [[PubMed](#)]
18. Khan, T.; Smirnova, A. 1D inverse problem in diffusion based optical tomography using iteratively regularized Gauss–Newton algorithm. *Appl. Math. Comput.* **2005**, *161*, 149–170. [[CrossRef](#)]
19. Kaipio, J.; Seppänen, A.; Somersalo, E.; Haario, H. Posterior covariance related optimal current patterns in electrical impedance tomography. *Inverse Probl.* **2004**, *20*, 919. [[CrossRef](#)]
20. Strauss, T.; Khan, T. Statistical inversion in electrical impedance tomography using mixed total variation and non-convex ℓ_p regularization prior. *J. Inverse -Ill-Posed Probl.* **2015**, *23*, 529–542. [[CrossRef](#)]
21. Harrach, B.; Ullrich, M. Monotonicity-based shape reconstruction in electrical impedance tomography. *SIAM J. Math. Anal.* **2013**, *45*, 3382–3403. [[CrossRef](#)]
22. Liu, D.; Khambampati, A.K.; Du, J. A parametric level set method for electrical impedance tomography. *IEEE Trans. Med. Imaging* **2017**, *37*, 451–460. [[CrossRef](#)]
23. Liu, D.; Smyl, D.; Du, J. A parametric level set-based approach to difference imaging in electrical impedance tomography. *IEEE Trans. Med. Imaging* **2018**, *38*, 145–155. [[CrossRef](#)] [[PubMed](#)]
24. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
25. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of go without human knowledge. *Nature* **2017**, *550*, 354–359. [[CrossRef](#)]
26. Senior, A.W.; Evans, R.; Jumper, J.; Kirkpatrick, J.; Sifre, L.; Green, T.; Qin, C.; Židek, A.; Nelson, A.W.; Bridgland, A.; et al. Improved protein structure prediction using potentials from deep learning. *Nature* **2020**, *577*, 706–710. [[CrossRef](#)]
27. Dimas, C.; Uzunoglu, N.; Sotiriadis, P.P. An efficient point-matching method-of-moments for 2D and 3D electrical impedance tomography using radial basis functions. *IEEE Trans. Biomed. Eng.* **2021**, *69*, 783–794. [[CrossRef](#)]
28. Fernández-Fuentes, X.; Mera, D.; Gómez, A.; Vidal-Franco, I. Towards a fast and accurate eit inverse problem solver: A machine learning approach. *Electronics* **2018**, *7*, 422. [[CrossRef](#)]
29. Hamilton, S.J.; Hauptmann, A. Deep D-bar: Real-time electrical impedance tomography imaging with deep neural networks. *IEEE Trans. Med. Imaging* **2018**, *37*, 2367–2377. [[CrossRef](#)]
30. Fan, Y.; Ying, L. Solving electrical impedance tomography with deep learning. *J. Comput. Phys.* **2020**, *404*, 109119. [[CrossRef](#)]
31. Hamilton, S.J.; Hänninen, A.; Hauptmann, A.; Kolehmainen, V. Beltrami-net: domain-independent deep D-bar learning for absolute imaging with electrical impedance tomography (a-EIT). *Physiol. Meas.* **2019**, *40*, 074002. [[CrossRef](#)]
32. Oechsle, M.; Mescheder, L.; Niemeyer, M.; Strauss, T.; Geiger, A. Texture fields: Learning texture representations in function space. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4531–4540.
33. Mescheder, L.; Oechsle, M.; Niemeyer, M.; Nowozin, S.; Geiger, A. Occupancy networks: Learning 3d reconstruction in function space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4460–4470.
34. Hu, P.; Shuai, B.; Liu, J.; Wang, G. Deep level sets for salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2300–2309.
35. Oechsle, M.; Niemeyer, M.; Reiser, C.; Mescheder, L.; Strauss, T.; Geiger, A. Learning implicit surface light fields. In Proceedings of the 2020 International Conference on 3D Vision (3DV), Virtual, 25–28 November 2020; pp. 452–462.
36. Mildenhall, B.; Srinivasan, P.P.; Tancik, M.; Barron, J.T.; Ramamoorthi, R.; Ng, R. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 405–421.
37. Di Cristo, M.; Ren, Y. Stable determination of an inclusion in a layered medium. *Math. Methods Appl. Sci.* **2018**, *41*, 4602–4611. [[CrossRef](#)]
38. Harrach, B. Uniqueness and Lipschitz stability in electrical impedance tomography with finitely many electrodes. *Inverse Probl.* **2019**, *35*, 024005. [[CrossRef](#)]
39. Kawakami, H. Stabilities of shape identification inverse problems in a Bayesian framework. *J. Math. Anal. Appl.* **2020**, *486*, 123903. [[CrossRef](#)]

40. Xu, B.; Wang, N.; Chen, T.; Li, M. Empirical evaluation of rectified activations in convolutional network. *arXiv* **2015**, arXiv:1505.00853.
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 26 June–1 July 2016; pp. 770–778.
42. Liu, B.; Yang, B.; Xu, C.; Xia, J.; Dai, M.; Ji, Z.; You, F.; Dong, X.; Shi, X.; Fu, F. pyEIT: A python based framework for Electrical Impedance Tomography. *SoftwareX* **2018**, *7*, 304–308. [[CrossRef](#)]
43. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; Volume 1.
44. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.