

A Tomographic Reconstruction Method using Coordinate-based Neural Network with Spatial Regularization

Jakeoung Koo^{*1}, Elise O. Brenne², Anders B. Dahl¹, and Vedrana A. Dahl¹

¹Department of Applied Mathematics and Computer Science, Technical University of Denmark

²Department of Energy Conversion and Storage, Technical University of Denmark

Abstract

Tomographic reconstruction is concerned with computing the cross-sections of an object from a finite number of projections. Many conventional methods represent the cross-sections as images on a regular grid. In this paper, we study a recent coordinate-based neural network for tomographic reconstruction, where the network inputs a spatial coordinate and outputs the attenuation coefficient on the coordinate. This coordinate-based network allows the continuous representation of an object. Based on this network, we propose a spatial regularization term, to obtain a high-quality reconstruction. Experimental results on synthetic data show that the regularization term improves the reconstruction quality significantly, compared to the baseline. We also provide an ablation study for different architecture configurations and hyper-parameters.

1 Introduction

Computed tomography (CT) is a versatile imaging technique allowing the study of interior structures of objects and has many applications, ranging from clinical to industrial applications [5]. In CT, a procedure known as tomographic reconstruction aims to reconstruct the material properties from measurements, called projections, based on the interaction of objects and penetrating radiation such as X-ray. Specifically, the projections are obtained by line integrals of an at-

tenuation coefficient function along straight lines and reconstructing an attenuation function is the goal of tomographic reconstruction.

Although the measurements are finite, we can consider different ways to represent the attenuation function. Among them, a discrete image on a regular grid has been popular in most of existing works including model-based [3, 14] or recent deep learning-based works [1, 8, 19]. In those works, the attenuation value within one pixel is typically assumed to be constant. From this image representation, conventional reconstruction works are based on solving a system of linear equations with some regularization terms.

Recently, there has been growing interest in coordinate-based neural networks to represent continuous functions [12, 16, 15]. Such networks input continuous spatial coordinate and output the signal on the coordinate. To represent high frequency features, a feature mapping of input coordinates was suggested in [12, 16] and another work called SIREN [15] used the sine function as the activation function with a specific initialization scheme.

In this paper, we study a coordinate-based neural network for tomographic reconstruction. We use the same architecture as in SIREN [15], to reconstruct a continuous attenuation function. We use the neural network, but our method does not require any training data. Our main contribution is to propose a regularization term to impose spatial smoothness. We also provide an ablation study with various configurations of the network and hyper-parameters.

Concurrent to our submission, there is a recent paper [16], which applied a coordinate-based network to

*Corresponding Author: jakoo@dtu.dk

different tasks including tomographic reconstruction. Our method differs in that we use another neural network architecture [15] and introduce a spatial regularization term. Our experimental results in Sec. 3 show that the regularization term improves the performance significantly.

1.1 Related works

Methods for tomographic reconstruction using deep neural networks can be classified into two categories: learning-based and learning-free approach. The learning-based approach utilizes the available data in a supervised or unsupervised manner. Zhu et al. [19] proposed a general method to train neural networks from pairs of projection data and ground truth reconstruction images. He et al. [8] considered two neural networks, where the first network simulate the filtered backprojection method [5] and the output of the first network is refined by the second convolutional neural network. The idea of filtered backprojection is also used in [10, 18]. Another approach is to combine conventional iterative methods and neural networks, called learned iterative methods [1, 2]. In this approach, convolutional layers replace some parts of the iterative methods and receive the forward projection operator and its adjoint as part of the inputs.

A learning-free approach does not need any training data, but still harnesses the power of neural networks. Gadelha et al. [7] extended Deep Image Prior (DIP) [17] to 2D tomographic reconstruction. DIP represents an image by the output of a learnable convolutional neural network on a fixed random input and have an implicit regularization effect. The disadvantage of [7] is that it requires early-stopping to prevent overfitting to noisy data.

As our method belongs to the learning-free approach, we compare our method to [7] shown in Sec. 3. The main difference of our work, compared to image-based methods, is that the outcome of our method is a continuous function, instead of discrete image, and the forward model does not depend on regular grid, which will be explained in the next section. Moreover, as we include the regularization term, our method does not need early-stopping.

2 Method

The aim of tomographic reconstruction is to reconstruct an attenuation coefficient function f from a finite number of projections. We first explain the representation of solution f based on a coordinate-based neural network and the forward model to connect the solution and measurements. Then, we introduce a regularization term and define our loss function.

2.1 Continuous representation of the attenuation function

To represent a continuous function, we employ a coordinate-based neural network, called SIREN [15]. As shown in Fig. 1, SIREN is a multi-layer perceptron whose input $\mathbf{x} = (x, y)$ is a spatial coordinate and the output $f_{\Theta}(\mathbf{x})$ represents the signal at that coordinate where Θ represents neural network parameters – the weight matrices and the bias vectors. In our case, the output corresponds to the attenuation coefficient. The key feature of SIREN is to use the sine function as the activation in the network and a principled initialization, which allows the representation of high frequency features.

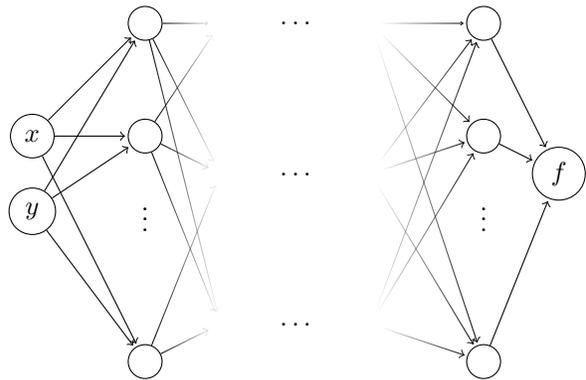


Figure 1: The coordinate-based neural network maps a spatial coordinate $\mathbf{x} = (x, y)$ to the attenuation coefficient $f_{\Theta}(\mathbf{x})$.

To be specific, each layer with the input \mathbf{z} from the previous layer is constructed as

$$\sin(\omega(W\mathbf{z} + \mathbf{b})) \tag{1}$$

where W is a weight matrix and \mathbf{b} is a bias vector in the layer and ω is a hyper-parameter to control the spatial frequency. We use Rectified Linear Unit (ReLU) as the last layer activation, to impose a non-negativity constraint.

2.2 Forward model

We model the process of obtaining tomographic measurements from the attenuation coefficient function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$. We follow a common assumption in tomographic reconstruction based on Lambert-Beer’s law [5], where radiation intensity is attenuated exponentially. After pre-processing, the projection data for a ray can be considered as the line integral of f along the path. The projection data consist of M measurements where M is the multiplication of the number of detector pixels and the number of projection angles. Each measurement can be assigned to a ray from the source of X-rays (or other waves) towards the corresponding detector pixel. Consider a ray i with an origin \mathbf{o} and a normalized direction vector \mathbf{d} . Then, the projection p_i can be computed as the line integral of f along the ray:

$$p_i = \int_a^b f_{\Theta}(\mathbf{o} + t \mathbf{d}) dt \quad (2)$$

where a and b denote the initial and the last value for t .

To numerically integrate Eq. (2), we employ the mid-point rule by dividing the limits of integration (a, b) into N subintervals:

$$p_i = \sum_{k=1}^N f_{\Theta}(\mathbf{x}_k) \delta \quad (3)$$

where $\delta = (b - a)/N$ and \mathbf{x}_k is the midpoint of the intervals defined by

$$\mathbf{x}_k = a + \frac{\delta}{2} + (k - 1)\delta. \quad (4)$$

The forward projection in Eq. (3) is differentiable with respect to the neural network parameters and automatic differentiation tools can be used.

2.3 Regularization and loss function

To deal with noisy data and obtain a more accurate reconstruction, we introduce a regularization term to impose spatial smoothness, inspired by total variation [13]. The total variation term computes a spatial gradient from both the x - and y -direction. Instead of two directions, we impose smoothness on one direction along the ray, for computational efficiency. Specifically, we aim to minimize the variation on each point along the ray i :

$$R_i = \frac{1}{\sqrt{\delta}} \sum_{k=1}^{N-1} \sqrt{|f_{\Theta}(\mathbf{x}_{k+1}) - f_{\Theta}(\mathbf{x}_k)|} \quad (5)$$

where we omit the dependency of i in each \mathbf{x} , for notational simplicity. We use the square root function to make the regularization term more robust [4].

From the computed projection p in Eq. (3), we aim to fit it to the projection data \hat{p} by minimizing the L_2 norm between them. With this data fitting term and the regularization term (5), we define the loss function to minimize with respect to neural network parameters Θ :

$$\mathcal{L}(\Theta) = \frac{1}{M} \sum_{i=1}^M \|p_i - \hat{p}_i\|_2^2 + \frac{\lambda}{M} \sum_{i=1}^M R_i, \quad (6)$$

where λ is a hyper-parameter to control the weighting between the data fidelity and regularization term.

To optimize Eq. (6), we use the mini-batch gradient descent method where at each iteration, we choose B random measurements among in total M available measurements.

3 Experimental Results

In this section, we compare our work to other image-based methods on simulated data qualitatively and quantitatively. We conduct an ablation study for different neural network settings and the regularization parameter. We also provide a result for real data.

3.1 Results on simulated data

We use phantom images with the size of 512×512 pixels shown in the first column of Fig. 2. These

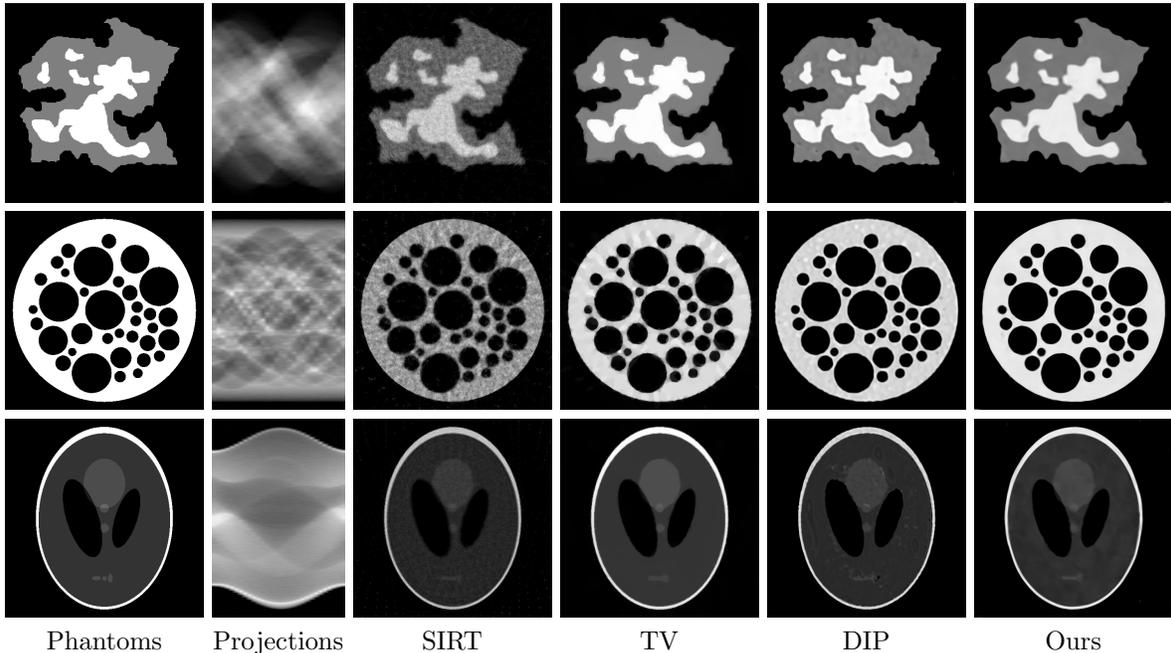


Figure 2: Qualitative results on noisy projection data. The first column shows the ground truth phantoms of Phantom 1, 2, 3 which are used to generate noisy projection data on the second column. The other columns show the reconstruction results.

Phantoms	PSNR					SSIM				
	SIRT	TV	DIP	Ours		SIRT	TV	DIP	Ours	
				$\lambda = 0$	$\lambda \neq 0$				$\lambda = 0$	$\lambda \neq 0$
Phantom 1	23.08	28.12	27.83	25.86	27.36	0.524	0.896	0.941	0.851	0.939
Phantom 2	16.52	19.70	21.20	20.88	22.04	0.468	0.764	0.881	0.820	0.933
Phantom 3	25.59	27.09	27.44	26.05	26.09	0.710	0.944	0.949	0.906	0.958

Table 1: Quantitative comparison to other methods on synthetic data from the phantom images in Fig. 2. We show two results for our method without regularization $\lambda = 0$ and with regularization $\lambda \neq 0$.

phantoms are used to generate projection data by an image-based linear forward model with parallel beam, shown in the second column. Each projection datum consists of 512 detector pixels and 30 projection angles. We impose Gaussian noise on the projection data with the relative noise level 0.02, which is chosen to reflect noise degree in real data.

Experimental details. Our method is implemented in the Julia language and based on a deep

learning library called Flux [9]. Unless explicitly mentioned, we use a neural network with 3 hidden layers where each layer has size 128. That is, the weight matrix for each hidden layer has size 128×128 , while the bias vector has 128 elements. We consider the reconstruction domain bounded into $(-1, 1) \times (-1, 1)$. In this bounded domain, the spatial frequency parameter $\omega = 30$ yields good performance overall. We initialize neural network parameters in

the same way as SIREN did [15] where the distributions of activations are preserved through the network. We use the Adam optimizer [11] with learning rate 0.0001, batch size $B = 512$ and iteration number 4,500 (150 epochs) as the solution changes very little after that. The batch size corresponds to the number of rays per iteration and the rays are sampled randomly. For the numerical integration (3), we divide the range of integration into $N = 256$ intervals.

Comparison to other image-based methods.

As our method belongs to a learning-free approach, we compare our method to conventional model-based approaches including: SIRT [3] and a Total Variation (TV)-based method [14]. We also compare to a method [7] based on Deep Image Prior (DIP) [17]. As mentioned in Sec. 1.1, DIP needs an early stopping. For DIP, we iterate 2000 times and save the results every 50th iteration and among those candidate solutions, we choose the best result.

To compare our result to image-based methods, we produce images from our implicit solutions by evaluating the network at each pixel position. For quantitative comparison, we employ two image-quality metrics: Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). We choose the same configuration for 3 phantoms except for the regularization parameter λ , which is set 0.0005 for Phantom 1 and 2 and 0.0001 for Phantom 3.

Fig. 2 provides a visual comparison of the results by SIRT, TV, DIP and our method and Table 1 shows the corresponding quantitative results. Due to the noisy data, the result images have some high values, which make the images look darker than the ground truth images. Overall, our method gives competitive results. Fig. 3 shows some intermediate results of our method during optimization for Phantom 2.

3.2 Ablation study

In this ablation study, we use projection data without noise and use the same settings as before.

Effect of network size. We vary the layer size, but fix the number of hidden layers. As shown in Table 2, as the layer size increases, the number of network parameters increases approximately 4 times. For reference, for an image of size 512×512 , image-

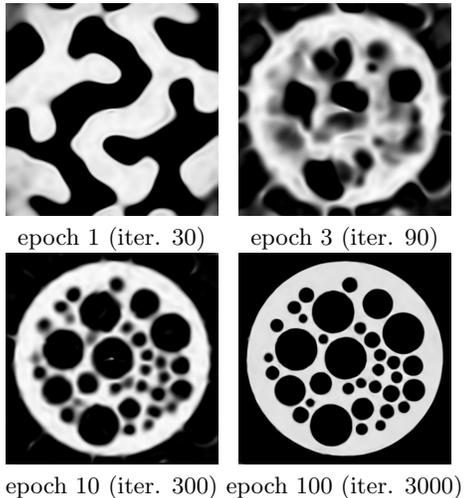


Figure 3: Intermediate results of our method during optimization.

based methods optimize 262,144 pixel values. When the layer size is 512 with a large number of parameters, the reconstruction quality is shown to degrade.

layer size	# params.	PSNR	SSIM
32	3,296	19.1	0.827
64	12,736	21.2	0.907
128	50,048	23.4	0.953
256	198,400	23.6	0.957
512	790,016	21.5	0.930

Table 2: Effect of network size for the reconstruction of Phantom 2.

Effect of sampling points N . The number of sampling points per ray affects both reconstruction quality and the computational cost. Table 3 provides the effect of N . As expected, as N increases, the computation time also increases.

Effect of regularization hyper-parameter λ . Fig. 4 shows the effect of λ . Without regularization, some artifacts are observed and, with large regularization $\lambda = 0.0005$, our method could not capture some fine details. The value $\lambda = 0.0001$ is shown to yield the best result in terms of both visual quality and PSNR.

N	time per iter.	PSNR	SSIM
64	0.031	20.03	0.846
128	0.065	22.00	0.927
256	0.110	23.37	0.953
512	0.293	23.61	0.961
1024	1.115	24.48	0.968

Table 3: Effect of the number of sampling points for the reconstruction of Phantom 2. The computation time in GPU per iteration is shown in seconds.

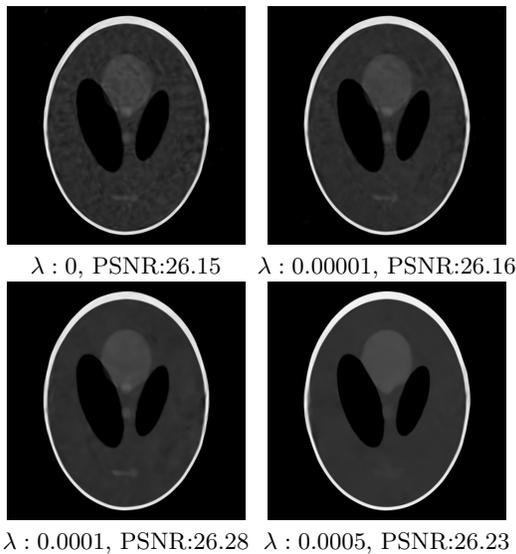


Figure 4: Effect of regularization parameter λ .

3.3 Results on real fan beam data

We test our method on cone-beam X-ray CT data from the SophiaBeads Dataset [6]. The scanned sample is a plastic tube filled with glass beads. Specifically, we use the provided central slice of the dataset “SophiaBeads_1024_averaged”, choose 64 projections at angles in $[0^\circ, 360^\circ]$ and down-sample with a factor 2 so that each projection is 680 pixels wide. As a pre-processing, we correct the center-of-rotation by shifting the projections 12 pixels and remove a high-intensity edge artifact by subtracting 0.0015 from the projection. As we use the central slice of a cone-beam dataset, a fan-beam geometry is used for the recon-

struction.

In Fig. 5, we provide the results of SIRT and our method. This demonstrates that our method can successfully be applied to a real data and that the performance is comparable to that of SIRT.

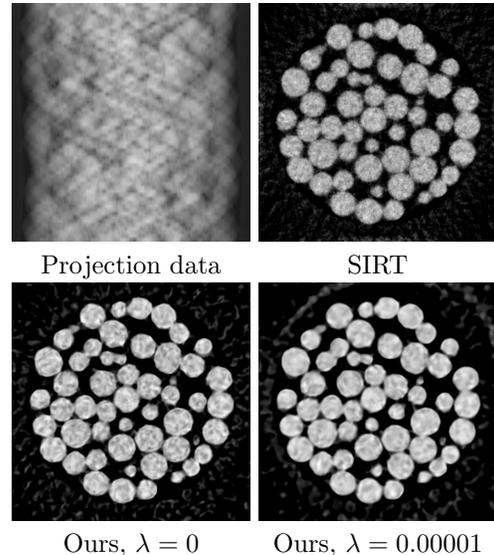


Figure 5: Reconstruction results for real fan beam data.

4 Conclusion and Discussion

We have proposed a spatial regularization term for tomographic reconstruction using a coordinate-based neural network. The experimental results show that the regularization term improves the reconstruction quality significantly, although choosing optimal regularization parameter is not trivial. For a practical use of our method, the major limitation is the high demand of memory requirement, which prevents us from choosing a large number of rays per iteration. This is an issue, especially in 3D reconstruction where projection data consist of a large number of measurements. We leave it as future work to improve the speed. Another future direction would be to include the encoding of projection data in the network, so that the network can infer to unseen projection data.

References

- [1] J. Adler and O. Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12), 2017.
- [2] J. Adler and O. Öktem. Learned Primal-dual Reconstruction. *IEEE Trans. Med. Imaging*, 37(6), 2018.
- [3] A. H. Andersen and A. C. Kak. Simultaneous Algebraic Reconstruction Technique (SART): A superior implementation of the ART algorithm. *Ultrason. Imaging*, 6(1), 1984.
- [4] T. Brox, A. Bruhn, N. Papenbergh, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision (ECCV)*, 2004.
- [5] T. M. Buzug. *Computed Tomography from Photon Statistics to Modern Cone Beam CT*. Springer, 2008.
- [6] S. B. Coban and S. A. McDonald. Sophiabeats dataset project, Mar. 2015.
- [7] M. Gadelha, R. Wang, and S. Maji. Shape Reconstruction Using Differentiable Projections and Deep Priors. In *International Conference on Computer Vision (ICCV)*, 2019.
- [8] J. He, Y. Wang, and J. Ma. Radon inversion via deep learning. *IEEE Transactions on Medical Imaging*, 39(6):2076–2087, 2020.
- [9] M. Innes. Flux: Elegant machine learning with julia. *J. Open Source Softw.*, 2018.
- [10] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep Convolutional Neural Network for Inverse Problems in Imaging. *IEEE Trans. Image Process.*, 26(9), 2017.
- [11] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.
- [12] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European Conference on Computer Vision (ECCV)*, 2020.
- [13] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. Nonlinear Phenom.*, 60(1-4), 1992.
- [14] E. Y. Sidky, J. H. Jørgensen, and X. Pan. Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle–Pock algorithm. *Phys. Med. Biol.*, 57(10), 2012.
- [15] V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein. Implicit Neural Representations with Periodic Activation Functions. *arXiv:2006.09661*, 2020.
- [16] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. *arXiv:2006.10739*, 2020.
- [17] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [18] T. Wurfl, M. Hoffmann, V. Christlein, K. Breininger, Y. Huang, M. Unberath, and A. K. Maier. Deep Learning Computed Tomography: Learning Projection-Domain Weights From Image Domain in Limited Angle Problems. *IEEE Trans. Med. Imaging*, 37(6), 2018.
- [19] B. Zhu, J. Z. Liu, S. F. Cauley, B. R. Rosen, and M. S. Rosen. Image reconstruction by domain-transform manifold learning. *Nature*, 555(7697), 2018.