Differentiable Uncalibrated Imaging

Sidharth Gupta, Konik Kothari, Valentin Debarnot, and Ivan Dokmanić

***Abstract*—We propose a differentiable imaging framework to address uncertainty in *measurement coordinates* such as sensor locations and projection angles. We formulate the problem as measurement interpolation at unknown nodes supervised through the forward operator. To solve it we apply implicit neural networks, also known as neural fields, which are naturally differentiable with respect to the input coordinates. We also develop differentiable spline interpolators which perform as well as neural networks, require less time to optimize and have well-understood properties. Differentiability is key as it allows us to jointly fit a measurement representation, optimize over the uncertain measurement coordinates, and perform image reconstruction which in turn ensures consistent calibration. We apply our approach to 2D and 3D computed tomography, and show that it produces improved reconstructions compared to baselines that do not account for the lack of calibration. The flexibility of the proposed framework makes it easy to extend to almost arbitrary imaging problems.**

***Index Terms*—Differentiable imaging, differentiable splines, implicit neural representations, operator uncertainty, uncalibrated imaging.**

I. Introduction

In COMPUTATIONAL imaging, a physical process, $\mathcal{A}$, such as 2D computed tomography (CT) relates the object we want to image, x, to an observable field, y. Both x and y are naturally functions of continuous coordinates: x could be a density over 2D spatial coordinates (e.g., $[0,1]^2$), and y a sinogram over angles and 1D projection coordinates (e.g., $[0,\pi) \times [-1,1]$).

In practice, however, we have finite sensors. Denoting the space of continuous measurement coordinates by Ω, the sensors sample the field y at $\widetilde{\boldsymbol{\mu}} = (\widetilde{\mu}_1, \ldots, \widetilde{\mu}_M) \in \Omega^M$, such that the observed measurements $\boldsymbol{y} \in \mathbb{R}^M$ follow

$$\boldsymbol{y} = \boldsymbol{A}_{\widetilde{\boldsymbol{\mu}}}(x) + \boldsymbol{\eta}, \tag{1}$$

where

$$\boldsymbol{y} \overset{\text{def}}{=} \boldsymbol{y}(\widetilde{\boldsymbol{\mu}}) \overset{\text{def}}{=} \left[y\left(\widetilde{\mu}_1\right), \ldots, y\left(\widetilde{\mu}_M\right) \right]^\top, \tag{2}$$

and $\boldsymbol{\eta} \in \mathbb{R}^M$ is the measurement noise. The discrete forward operator $\boldsymbol{A}_{\widetilde{\boldsymbol{\mu}}}$ parameterized by $\widetilde{\boldsymbol{\mu}}$ samples the output of $\mathcal{A}$. In computational imaging we work with a discretization or some other finite-dimensional approximation of x denoted by $\boldsymbol{x} \in \mathbb{R}^N$. Hereafter we let $\boldsymbol{A}_{\widetilde{\boldsymbol{\mu}}}$ act on $\boldsymbol{x}$ rather than on x.

In this paper we address the scenario where the true measurement coordinates $\widetilde{\boldsymbol{\mu}}$ are only approximately known: the imaging system is out of calibration. Consequently, the true operator $\boldsymbol{A}_{\widetilde{\boldsymbol{\mu}}}$ is unknown, and we work with an operator $\boldsymbol{A}_{\boldsymbol{\mu}}$ for measurement coordinates $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_M)$, which *would* be correct if the system was calibrated. The assumed measurement coordinates $\boldsymbol{\mu}$ are related to the *unknown* true measurement coordinates $\widetilde{\boldsymbol{\mu}}$ by small perturbations.

Not accounting for the mismatch between $\widetilde{\boldsymbol{\mu}}$ and $\boldsymbol{\mu}$ can lead to a poor reconstruction. The gist of our method is to learn a representation of the measurement space that can be evaluated and differentiated at arbitrary measurement coordinates $\boldsymbol{\mu}$. This gives us measurements $\boldsymbol{y}(\boldsymbol{\mu})$ that are then well-suited for a reconstruction method that uses $\boldsymbol{\mu}$. Our proposed framework enables us to

1) Jointly reconstruct the image $\boldsymbol{x}$ and learn the true unknown measurement coordinates $\widetilde{\boldsymbol{\mu}}$ by using gradient-based optimization.
2) Learn continuous measurement representations that take measurement coordinates as input. These representations can be evaluated at $\boldsymbol{\mu}$ and are in essence interpolations of the discrete measurements with unknown interpolation knots.
3) Leverage standard differentiable image reconstruction methods that exist for the assumed $\boldsymbol{\mu}$ even though the observations correspond to unknown $\widetilde{\boldsymbol{\mu}}$. This helps learn consistent measurement representations.

A strength of our approach is that we can use any continuous interpolation method that admits backpropagation to input coordinates. To showcase this, we use implicit neural networks (neural fields) but also develop fully-differentiable variable-knot splines which allow us to optimize spline control points and control point weights. We show that splines perform as well as implicit neural networks while being faster to fit and simpler to interpret. A key property of both these representation types is that we can perform automatic differentiation with respect to their input coordinates and recover $\widetilde{\boldsymbol{\mu}}$ by gradient-based optimization.

A. Example: Computed Tomography

We illustrate our framework with 2D CT, where we measure parallel beam projections of an image at different view angles in a detector plane. Let $\Omega = [0,\pi) \times \mathbb{R}$, $x \in L^2([-1,1]^2)$, and $y \in$

Manuscript received 12 March 2023; revised 16 August 2023 and 25 October 2023; accepted 7 December 2023. Date of publication 22 December 2023; date of current version 12 January 2024. This work was supported in part by by the European Research Council Starting under Grant 852821—SWING and in part by the donation of GPUs from NVIDIA Corporation to the University of Illinois at Urbana-Champaign. Additionally, calculations were performed at sciCORE (http://scicore.unibas.ch/) scientific computing center at University of Basel. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jan Sijbers. *(Corresponding author: Sidharth Gupta.)*

Sidharth Gupta and Konik Kothari are with the University of Illinois at Urbana-Champaign, Champaign, IL 61820 USA (e-mail: gupta67@illinois.edu; kkothar3@illinois.edu).

Valentin Debarnot is with the University of Basel, 4001 Basel, Switzerland (e-mail: valentin.debarnot@unibas.ch).

Ivan Dokmanić is with the University of Basel, 4001 Basel, Switzerland, and also with the University of Illinois at Urbana-Champaign, Champaign, IL 61820 USA (e-mail: ivan.dokmanic@unibas.ch).

Digital Object Identifier 10.1109/TCI.2023.3346294

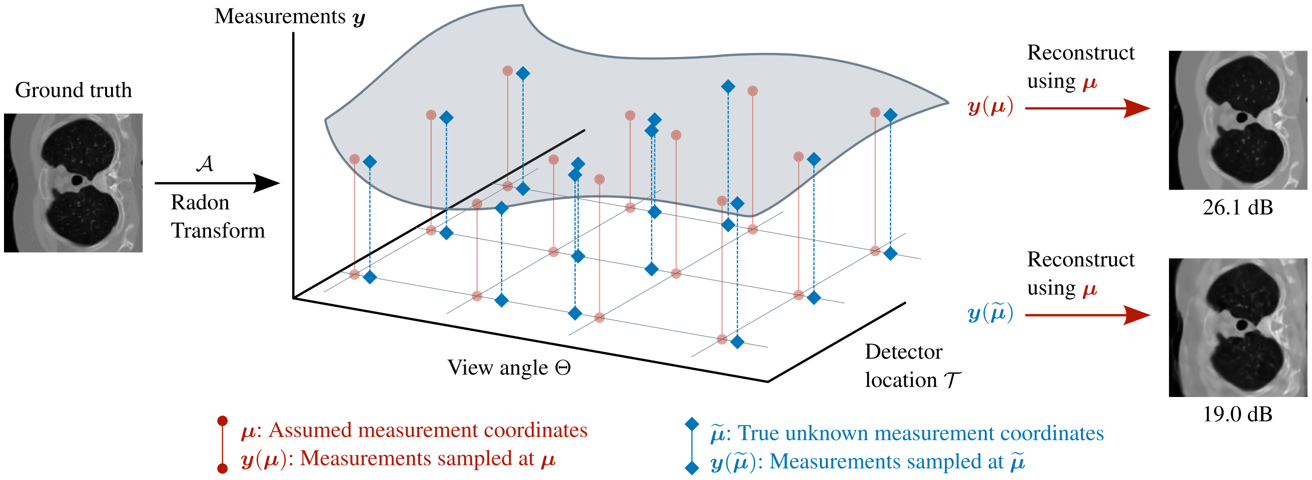

Measurements $\boldsymbol{y}$
Ground truth
$\mathcal{A}$
Radon Transform
View angle Θ
Detector location $\mathcal{T}$
$\boldsymbol{y}(\boldsymbol{\mu})$
Reconstruct using $\boldsymbol{\mu}$
26.1 dB
$\boldsymbol{y}(\widetilde{\boldsymbol{\mu}})$
Reconstruct using $\boldsymbol{\mu}$
19.0 dB
$\boldsymbol{\mu}$: Assumed measurement coordinates
$\boldsymbol{y}(\boldsymbol{\mu})$: Measurements sampled at $\boldsymbol{\mu}$
$\widetilde{\boldsymbol{\mu}}$: True unknown measurement coordinates
$\boldsymbol{y}(\widetilde{\boldsymbol{\mu}})$: Measurements sampled at $\widetilde{\boldsymbol{\mu}}$

Fig. 1. Visualization of the parameter mismatch between $\widetilde{\boldsymbol{\mu}}$ and $\boldsymbol{\mu}$ in 2D CT imaging that is described in Section I-A. A mismatch in the view angles between $\boldsymbol{\mu}$ and $\tilde{\boldsymbol{\mu}}$ can produce a significant drop in reconstruction quality.

$L^2(\Omega)$.[1] The measured projections $\boldsymbol{y}$ are obtained from a finite set of discrete view angles $\widetilde{\Theta} = (\widetilde{\theta}_1, \dots, \widetilde{\theta}_J)$, and are sampled at a finite set of discrete detector locations $\widetilde{\mathcal{T}} = (\widetilde{t}_1, \dots, \widetilde{t}_K)$. The true set of unknown measurement coordinates are then $(\widetilde{\mu}_m)_{m=1}^M = \widetilde{\Theta} \times \widetilde{\mathcal{T}}$ with $M = J \cdot K$. Furthermore, $\boldsymbol{A}_{\widetilde{\boldsymbol{\mu}}}$ is given by the discrete Radon transform for view angles $\widetilde{\Theta}$ and detector locations $\widetilde{\mathcal{T}}$. Many other computational imaging applications such as electron cryotomography (CryoET), magnetic resonance imaging and optical microscopy can be parameterized similarly.

The parameter mismatch between $\widetilde{\boldsymbol{\mu}}$ and $\boldsymbol{\mu}$ and its severe impact is illustrated in Fig. 1. We use a state-of-the-art reconstruction method that computes a filtered backprojection estimate with $\boldsymbol{\mu}$ and feeds it into a UNet deep neural network to obtain a reconstruction of $\boldsymbol{x}$ [1], [2]. The neural network is trained in a supervised manner using training data that is also generated using $\boldsymbol{\mu}$. When there is parameter mismatch, the true view angles differ from the assumed view angles and therefore $\widetilde{\boldsymbol{\mu}} \neq \boldsymbol{\mu}$. Fig. 1 shows that in this case, the reconstruction method produces a degraded reconstruction from both a visual and SNR evaluation. Fig. 2 shows this for another ground truth image and number of view angles. The last column of Fig. 2 also shows that when there is no parameter mismatch ($\widetilde{\boldsymbol{\mu}} = \boldsymbol{\mu}$), the same reconstruction method performs strongly.

B. Related Work

Many of the state-of-the-art methods for solving imaging inverse problems are based on deep learning [3], [4]. Popular supervised approaches use the forward operator to obtain an initial estimate which is then enhanced by a neural network (reconstruction method in Fig. 2 for example) [2], [5], [6], [7]. Unrolling iterative methods or replacing optimization components with deep networks is another established approach [8], [9], [10], [11], [12], [13]. A different direction involves requiring the inverse problem solution to lie in the range of a generative neural network [14], [15], [16]. In general all these methods utilize a forward operator in some way but do not account for measurement coordinate uncertainty. This can severely impact the reconstruction as shown in Fig. 2. In this paper we propose to address this problem by reevaluating the measurements at the measurement coordinates that deep neural network reconstruction methods are designed for.

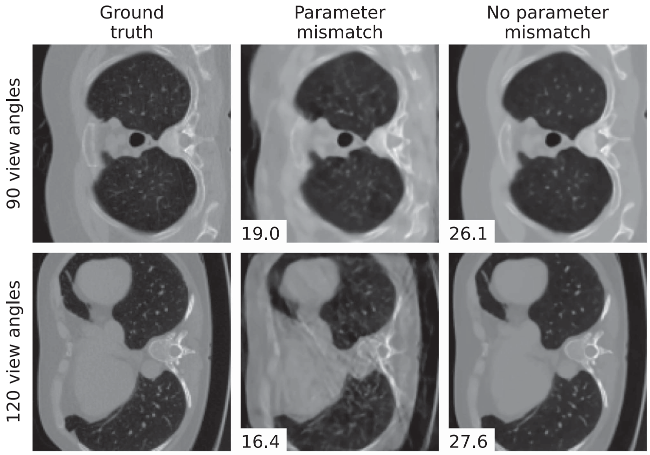


Fig. 2. 2D CT reconstructions with 90 and 120 view angle. The SNRs (in dB) are written in the bottom-left corners. When there is parameter mismatch because the true and assumed view angles differ, the reconstruction method performs poorly and there is a clear degradation.

While much less common, there are some recent deep learning approaches which address measurement coordinate uncertainty. Gilton et al. explored fine-tuning a neural network that was trained with measurements sampled at $\boldsymbol{\mu}$ to work well with test measurements sampled at unknown $\widetilde{\boldsymbol{\mu}}$ [17]. Our method is different: we reevaluate the measurements via a measurement representation for a single example so we do not need to fine-tune. Another appealing approach is to train a neural network on a family of operators with different parameterizations [18]. These methods always induce a tradeoff between the reconstruction quality and the variety of forward maps they are trained on, and they only work well for the distribution of perturbations seen at training time. There is also the challenge of dataset generation, especially in compute-intensive problems.

[1] We denote by $L^2(\mathcal{M})$ the space of square integrable functions on $\mathcal{M}$.

We mitigate this tradeoff by using consistency to identify the true measurement coordinates.

Continuous representations have been used to solve a variety of science and engineering problems. To the best of our knowledge no prior work has used them for imaging with measurement coordinate uncertainty. Splines have long been used to model surfaces and curves in geometry [19], [20], [21], [22]. Moreover, differentiable splines with parameters that can be fitted using automatic differentiation have also been developed [23]. Recently deep learning methods called neural fields or implicit neural networks have proven to be extremely good and efficient continuous representations [24]. They have been employed in a broad spectrum of applications ranging from representing geometry via signed distance functions [25], [26] to solving partial differential equations [27]. Implicit neural networks have also been used to solve tomographic imaging inverse problems [28], [29]. In particular, Sun et al. also used them to represent 2D CT measurements [30]. Rather than for calibration, they use implicit networks to upsample measurements for downstream reconstruction by a deep neural network. While upsampling may also be performed by traditional tools such as splines, our framework relies on the differentiability of the used representations—both neural and spline-based—with respect to the input coordinates.

In this paper, we learn forward model parameters and continuous measurement representations through a differentiable imaging framework [31]. We use automatic differentiation to optimize the parameters and inputs of implicit neural network and differentiable spline representations. Prior differentiable imaging works for microscopy [32], holography, ptychography and ptychographic tomography [33] also learn forward model parameters via automatic differentiation. However, these works do not learn a continuous measurement representation, and so do not reevaluate measurements for use with state-of-the-art reconstruction methods that assume different parameters.

There are alternative approaches for handling uncertainty in measurement coordinates. Bundle adjustment is an example from computer vision where scene coordinates, camera coordinates and system coordinates are jointly optimized [34]. Similar to our framework, bundle adjustment uses a good starting estimate to optimize the coordinates. However, a key difference is that we obtain a final solution by reevaluating measurements at the assumed measurement coordinates, and use a reconstruction method designed for the assumed measurement coordinates. Imaging approaches based on alternating minimization optimization [35], and designing handcrafted regularizers to manage ill-posedness [36], [37] have also been employed when there is miscalibration. In this work we jointly minimize our objective function. Furthermore, we use a consistency loss which can be interpreted as a regularizer when learning the measurement representation.

Our work is related to the broader theme of inverse problems with "noisy" forward operators. On one end of the spectrum there are inverse problems where the operator (and $\widetilde{\boldsymbol{\mu}}$) is perfectly known, and at the other extreme there are blind inverse problems where the operator and $\widetilde{\boldsymbol{\mu}}$ are completely unknown. Related blind imaging problems are tomography with unknown view angles [38], [39] and cryo-electron microscopy with unknown projection angles [40]. In between the extremes, there are semi-blind inverse problems where the operator and $\widetilde{\boldsymbol{\mu}}$ are approximately known. Total least squares approaches that perturb an assumed operator [41], [42], [43] and our proposed measurement coordinate-based framework fall under this category.

C. *Paper Organization*

In Section II we present the optimization problem which models joint calibration and image reconstruction. We first model measurements as continuous functions and then show how to learn these measurement representations. We show how implicit neural networks and splines can both be used as representations. Section III numerically verifies that when there is measurement coordinate uncertainty, our proposed method yields significantly improved reconstructions. The simulated experiments are performed on 2D CT in the main paper and 3D CT in Appendix B. We conclude the paper and motivate future work directions in Section IV.

II. Differentiable Framework

We wish to learn a continuous representation of the measurement space so that we can sample it at specific measurement coordinates and obtain the corresponding measurements. We model the measurement representations as continuous functions $r_{\boldsymbol{\varphi}}(\cdot)$ that take measurement coordinates as input and map them to the corresponding sampled measurements. The learnable parameters, $\boldsymbol{\varphi} \in \Phi$ where Φ is the space of feasible parameters, are optimized so that

$$r_{\boldsymbol{\varphi}}(\omega) \approx \boldsymbol{y}(\omega) \tag{3}$$

where $\omega \in \Omega$ is a measurement coordinate and $\boldsymbol{y}(\omega) \in \mathbb{R}$ is the sample from the measurement space at measurement coordinate ω. For convenience we also denote a batch evaluation of $r_{\boldsymbol{\varphi}}(\cdot)$ as

$$\begin{aligned} R_{\boldsymbol{\varphi}}(\boldsymbol{\omega}) &\overset{\text{def}}{=} \left[r_{\boldsymbol{\varphi}}\left(\omega_1\right), \ldots, r_{\boldsymbol{\varphi}}\left(\omega_Q\right)\right]^\top \\ &\approx \left[\boldsymbol{y}\left(\omega_1\right), \ldots, \boldsymbol{y}\left(\omega_Q\right)\right]^\top \end{aligned} \tag{4}$$

where $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_Q) \in \Omega^Q$ and $R_{\boldsymbol{\varphi}}(\boldsymbol{\omega}) \in \mathbb{R}^Q$. We also require $r_{\boldsymbol{\varphi}}(\cdot)$ to be differentiable with respect to $\boldsymbol{\varphi}$ and its input so that we can use gradient-based optimization to estimate $\boldsymbol{\varphi}$ and the unknown measurement coordinates.

A. *Joint Optimization Objective*

We want $r_{\boldsymbol{\varphi}}(\cdot)$ to accurately produce samples from the space of measurements. Since we only observe measurements $\boldsymbol{y}$ at measurement coordinates $\widetilde{\boldsymbol{\mu}}$ in (1), we require

$$R_{\boldsymbol{\varphi}}\left(\widetilde{\boldsymbol{\mu}}\right) \approx \boldsymbol{y} \tag{5}$$

However, since $\widetilde{\boldsymbol{\mu}}$ is unknown we cannot use it to verify the accuracy of $r_{\boldsymbol{\varphi}}(\cdot)$. This motivates us to jointly learn the representation parameters $\boldsymbol{\varphi}$ and the unknown measurement coordinates $\widetilde{\boldsymbol{\mu}}$ by minimizing a measurement fitting loss,

$$\mathcal{L}_{\text{fitting}}\left(\boldsymbol{\nu}, \boldsymbol{\varphi}\right) \overset{\text{def}}{=} \left\|\boldsymbol{y} - R_{\boldsymbol{\varphi}}\left(\boldsymbol{\nu}\right)\right\|_2^2, \tag{6}$$

with respect to the input $\boldsymbol{\nu} \in \Omega^M$ and $\boldsymbol{\varphi}$. Recall $\boldsymbol{y} \in \mathbb{R}^M$ is defined by (1).

Learning $r_{\boldsymbol{\varphi}}(\cdot)$ by minimizing only $\mathcal{L}_{\text{fitting}}(\boldsymbol{\nu}, \boldsymbol{\varphi})$ with respect to both the representation's parameters and input would not in general result in an accurate measurement representation because there are too many degrees of freedom. Therefore we regularize and control $\boldsymbol{\varphi}$ by enforcing $r_{\boldsymbol{\varphi}}(\cdot)$ to be consistent with reconstructions that could be obtained by using its output. This is done by minimizing a consistency loss with respect to $\boldsymbol{\varphi}$,

$$\mathcal{L}_{\text{consistency}}\left(\boldsymbol{\varphi}\right) \overset{\text{def}}{=} \left\| R_{\boldsymbol{\varphi}}\left(\boldsymbol{\mu}\right) - \boldsymbol{A}_{\boldsymbol{\mu}} G_{\boldsymbol{\mu}}\left(R_{\boldsymbol{\varphi}}\left(\boldsymbol{\mu}\right)\right)\right\|_2^2, \tag{7}$$

where $G_{\boldsymbol{\mu}} : \mathbb{R}^M \rightarrow \mathbb{R}^N$ is a differentiable reconstruction method that was designed using measurement coordinates $\boldsymbol{\mu}$. Putting everything together, our complete joint optimization objective is

$$\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\varphi}} = \underset{\boldsymbol{\nu} \in \Omega^M, \boldsymbol{\varphi} \in \Phi}{\arg\min} \; \mathcal{L}_{\text{fitting}}\left(\boldsymbol{\nu}, \boldsymbol{\varphi}\right) + \lambda \, \mathcal{L}_{\text{consistency}}\left(\boldsymbol{\varphi}\right) \tag{8}$$

where $\lambda \in \mathbb{R}_{\geq 0}$ is a tunable weight that controls the relative importance of the consistency and fitting losses. As the assumed coordinates $\boldsymbol{\mu}$ are close to the true unknown coordinates $\widetilde{\boldsymbol{\mu}}$, we initialize $\boldsymbol{\nu}$ to $\boldsymbol{\mu}$. After completing the optimization (8), the learned coordinates $\widehat{\boldsymbol{\mu}}$ are close to $\widetilde{\boldsymbol{\mu}}$ (verified in Section III), and the final reconstruction and estimate of $\boldsymbol{x} \in \mathbb{R}^N$ is given by

$$\widehat{\boldsymbol{x}} = G_{\boldsymbol{\mu}}(R_{\boldsymbol{\varphi}}(\boldsymbol{\mu})). \tag{9}$$

One of the advantages of the objective (8) is that the same operator, $\boldsymbol{A}_{\boldsymbol{\mu}}$ in (7), is used in each optimization iteration. A formulation that uses the continually updating learned measurement coordinates $\boldsymbol{\nu}$ in (7), would require the operator to be rebuilt after each optimization iteration. If there is no efficient implementation to change the measurement coordinates due to the complexity of the forward process, and if the operator is large, this can be severely time consuming and resource intensive.

B. Leveraging Reconstruction Methods

A key aspect of our framework is that consistency and the final reconstruction are obtained by using reconstruction method $G_{\boldsymbol{\mu}}(\cdot)$ that was designed using measurement coordinates $\boldsymbol{\mu}$ even though the observations in (1) are sampled at measurement coordinates $\widetilde{\boldsymbol{\mu}}$. This provides significant flexibility and allows us to incorporate a variety of reconstruction methods. For example, the reconstruction method may be a relatively straightforward adjoint or pseudoinverse operation. Alternatively, it can be a more complex neural network that provides state-of-the-art reconstructions with measurements from $\boldsymbol{\mu}$ (reconstruction method in Fig. 2 is one example). Our framework is particularly advantageous in this case because it may be cumbersome to retrain a neural network for different measurement coordinates.

C. Measurement Representations

In order to better understand how to use the framework to solve real imaging problems, we now pick implicit neural networks and splines and explain how they are suitable measurement representations. While we consider these, we emphasize that our framework is general and is not restricted to these representation types.

1) Implicit Neural Representations: Implicit neural representations are deep feedforward neural networks that represent discrete signals as continuous functions. When used as the measurement representation $r_{\boldsymbol{\varphi}}(\cdot)$ in (3), $\boldsymbol{\varphi}$ are the trainable network parameters. The input to the network are measurement coordinates and the output are the corresponding measurements. Implicit neural networks have previously been used to represent measurements when there is no measurement uncertainty [30]. In this case the network input was not optimized and consistency (7) was not enforced.

As implicit neural representations are neural networks, we can use automatic differentiation to calculate their gradients with respect to $\boldsymbol{\varphi}$ and their input coordinates to solve (8). In this paper we use an architecture comprising a Fourier feature mapping layer followed by standard fully-connected layers [26], [30].

2) Differentiable Splines: Splines use locally supported basis functions to represent signals as a continuous surface. In this paper we focus on Non-uniform Rational Basis Splines (NURBS) because of their ability to model complex surfaces [21], [44], [45]. To aid understanding, we explain NURBS using the 2D CT imaging example that was introduced in Section I-A. The measurement coordinates $\boldsymbol{\mu}$ are the Cartesian product of assumed view angles and assumed detector locations, $\Theta \times \mathcal{T}$ where $\Theta = (\theta_1, \ldots, \theta_J)$ and $\mathcal{T} = (t_1, \ldots, t_K)$. The NURBS surface $s_{\boldsymbol{\varphi}}(\cdot)$ with parameters $\boldsymbol{\varphi}$ evaluated at measurement coordinate $\mu' = [\theta', t']^\top \in [0, \pi] \times \mathbb{R}$ is then

$$s_{\boldsymbol{\varphi}}(\mu') = \sum_{j=0}^{J-1} \sum_{k=0}^{K-1} b_{j,k}(\mu') \, \boldsymbol{p}_{j,k} \tag{10}$$

where $b_{j,k}(\cdot) \in \mathbb{R}$ are scalar-valued rational basis functions with local support and $\boldsymbol{p}_{j,k} \in \mathbb{R}^3$ are control point vectors. The NURBS are parameterized by weight parameters $w_{j,k} \in \mathbb{R}$ of the rational basis functions, and the control point vectors. This gives $\boldsymbol{\varphi} = \{w_{j,k}\}_{j=0,\, k=0}^{J-1,\, K-1} \cup \{\boldsymbol{p}_{j,k}\}_{j=0,\, k=0}^{J-1,\, K-1}$. Note that the control point vectors are three-dimensional because for each two-dimensional measurement coordinate μ', there is a corresponding scalar measurement. Consequently, according to (10), $s_{\boldsymbol{\varphi}}(\mu')$ is also three-dimensional. We then establish the following relationship between NURBS surfaces and our measurement representations (3) to get a spline measurement representation,

$$r_{\boldsymbol{\varphi}}(\omega) = s_{\boldsymbol{\varphi}}^{\diamond}(\omega), \tag{11}$$

where $s_{\boldsymbol{\varphi}}^{\diamond}(\cdot)$ denotes the value of the measurement dimension of $s_{\boldsymbol{\varphi}}(\cdot)$.

It has recently been shown that automatic differentiation can be used to learn spline parameters [23]. Hence, we develop differentiable spline representations and use gradient-based optimization to learn the NURBS parameters $\boldsymbol{\varphi}$ and their input measurement coordinates. These differentiable splines fit into our framework straightforwardly as measurement representations (11), and we can use them to solve (8).

In our numerical experiments, we carefully initialize the spline parameters $\boldsymbol{\varphi}$ and then learn them: $w_{j,k}$ is initialized to one and control point vectors $\boldsymbol{p}_{j,k}$ are initialized using the assumed measurement coordinates and their corresponding observed measurements, $\boldsymbol{p}_{j,k} = [\theta_j, t_k, \boldsymbol{y}([\widetilde{\theta}_j, \widetilde{t}_k]^\top)]^\top \in \mathbb{R}^3$. Additionally, in our simulated experiments we also extend (10) to higher dimensional measurement coordinates (see Appendix B). Further details on NURBS and their implementation are provided in Appendices C and D.

III. Numerical Verification With 2D CT Imaging

We experimentally verify our framework by solving 2D CT imaging problems in the main paper. Further simulation results for 3D CT imaging are provided in Appendix B. The imaging forward operators are implemented using the Operator Discretization Library (ODL) [46]. These simulated experiments demonstrate that our framework can be used to solve imaging problems with different measurement coordinate dimensions. Furthermore, we exhibit the flexibility of our method by using implicit neural networks and splines as measurement representations.[2]

We use SNR (in dB) to quantify the measurement noise and measurement coordinate uncertainty. SNR is calculated by

$$\mathrm{SNR}(\boldsymbol{c},\, \boldsymbol{d}) = -20 \log_{10}\left(\frac{\|\boldsymbol{c}-\boldsymbol{d}\|_2}{\|\boldsymbol{d}\|_2}\right). \tag{12}$$

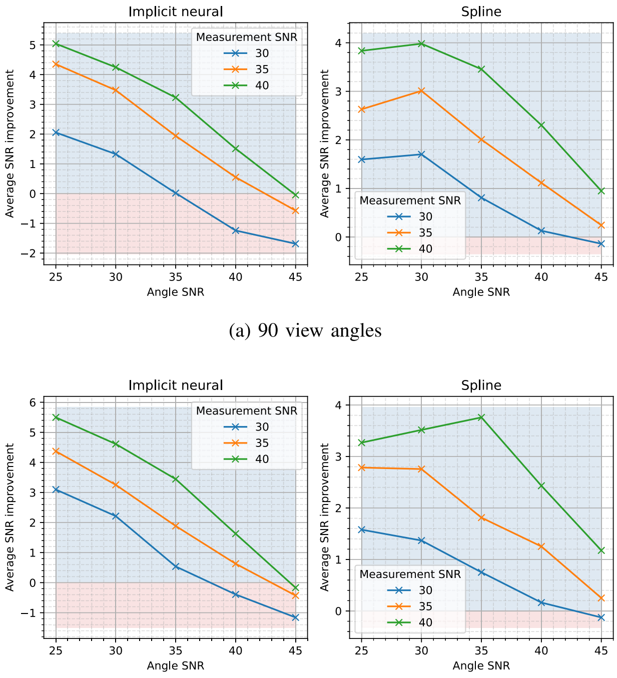
If we let $\boldsymbol{y} \in \mathbb{R}^M$ denote the observed measurements as in (1) and let $\widetilde{\boldsymbol{y}}$ denote the unobserved noiseless measurements, the measurement noise level is $\mathrm{SNR}(\boldsymbol{y},\, \widetilde{\boldsymbol{y}})$. The measurement noise is simulated with zero-mean iid Gaussian noise with variance adjusted to achieve a target SNR level.

Due to their state-of-the art performance when there is no measurement coordinate uncertainty, we use deep neural networks for the reconstruction method, $G_{\boldsymbol{\mu}}(\cdot)$, in (8). For 2D CT we use a 2D Unet [2], and for 3D CT in Appendix B, we use a 3D Unet [47]. Following standard practice, a preprocessing step applies the pseudoinverse of the imaging operator to the measurements to produce an initial image estimate [2]. These networks are then trained in a supervised manner to map the initial estimates to ground truth images. The training data generation and preprocessing step are done with the assumed measurement coordinates $\boldsymbol{\mu}$. The measurements in the training data are noisy and in each experiment we use a Unet whose training measurement noise level matches the measurement noise level of the obtained measurements.

As mentioned in Section II, the solution, $\widehat{\boldsymbol{x}}$, is given by (9). We use SNR to evaluate the solution quality, $\mathrm{SNR}(\widehat{\boldsymbol{x}},\, \boldsymbol{x})$. We compare $\widehat{\boldsymbol{x}}$ against baseline reconstructions that are obtained by directly using the obtained measurements with the reconstruction Unets,

$$\boldsymbol{x}_{\text{baseline}} = G_{\boldsymbol{\mu}}(\boldsymbol{y}). \tag{13}$$

Recall, the obtained measurements correspond to measurement coordinates $\widetilde{\boldsymbol{\mu}}$. Appendix D contains further hyperparameter and implementation details.



(a) 90 view angles

(b) 120 view angles

Fig. 3. SNR improvement (dB) when solving (8) for 2D CT imaging. There are different combinations of measurement noise SNR and view angle uncertainty SNR. We consider 90 and 120 view angles.

In 2D CT imaging, one-dimensional projections of a two-dimensional object at different view angles are collected by an array of detectors. Our goal is to reconstruct an image of the object from these projections when there is uncertainty in only the view angles, only the detector locations, or in both the view angles and the detector locations at the same time.

For these simulated experiments, the assumed view angles are uniformly spaced on the interval $[0, \pi]$, and the true view angles can have an unknown perturbation from these assumed view angles to simulate experimental incertitude. We keep the view angle measurement coordinate uncertainty level the same for each view angle—each true view angle is independently perturbed from the assumed view angle by zero-mean Gaussian noise with variance adjusted to obtain a target SNR level. If we let $\widetilde{\mu}_m \in \mathbb{R}$ and $\mu_m \in \mathbb{R}$ denote the mth true and assumed view angles, the view angle measurement coordinate uncertainty level is $\mathrm{SNR}(\widetilde{\mu}_m,\, \mu_m)$.

Similarly, the assumed detector locations are uniformly spaced on the normalized interval $[0, 1]$ and the true detector locations can have an unknown perturbation from the assumed detector locations. To simulate detector uncertainty, we first independently perturb the first and last detectors from their assumed locations with a uniform perturbation scaled so that their location uncertainty meets a target SNR. The remaining detectors are then perturbed so that that the final true unknown detector array is uniformly spaced between the perturbed first and last detectors. Unlike the view angle measurement coordinate uncertainty model, this detector location coordinate uncertainty model is not iid. This enables us to evaluate the performance of our framework when there are non-iid measurement coordinate uncertainties.

[2] Code available at https://github.com/swing-research/differentiable_uncalibrated_imaging.

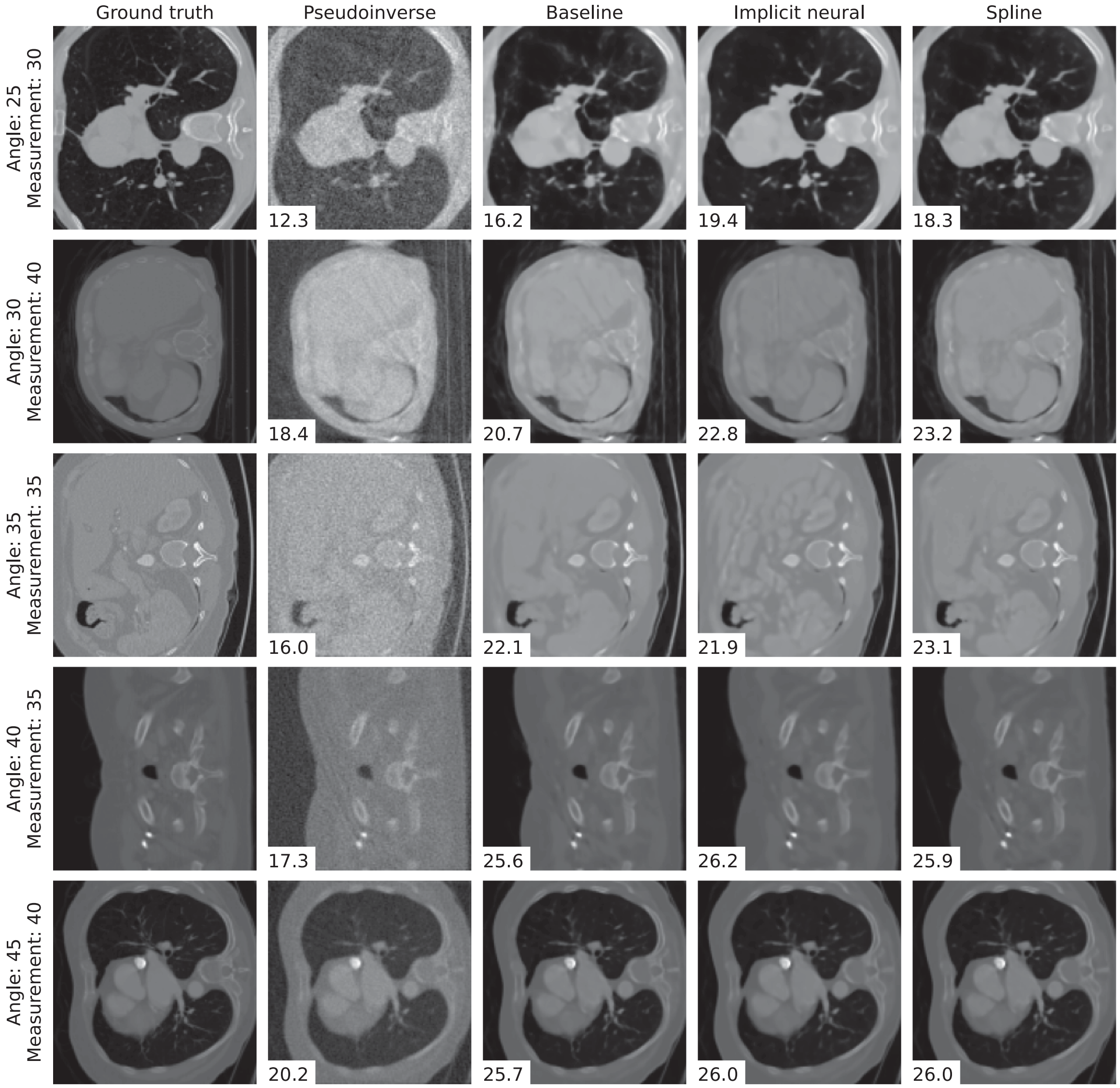


Fig. 4. Example reconstructions for different measurement noise SNR and view angle uncertainty SNR combinations for 2D CT imaging. There are 90 view angles. The reconstruction SNRs are shown for each reconstruction.

Combining these spaces gives the measurement coordinates space as $\Omega = [0, \pi] \times [0, 1]$. We explore the performance of our method compared to the baseline which does not account for measurement coordinate uncertainty. We use images from the the LoDoPaB-CT tomography dataset resized to 128×128 [48]. From this dataset, 35,000 samples were used to train the image reconstruction 2D Unet $G_{\boldsymbol{\mu}}(\cdot)$. Test images from this dataset are used in this section to verify our framework.

A. View Angle Uncertainty

In the first group of experiments we consider the case where there is only view angle uncertainty and no uncertainty in the detector locations. As there is only uncertainty in the view angle dimension and not in the detector location dimension, we only optimize the view angle dimension of the measurement coordinates in (8).

1) Combinations of Measurement Noise and View Angle Error: To determine how our framework performs under different settings, we consider different combinations of measurement noise and view angle measurement coordinate uncertainty. We do trials over 25 different test images, and in each trial, different measurement noise and view angle perturbations are used.

We evaluate the performance of our framework relative to the baseline by calculating the average reconstruction SNR improvement over the baseline. Fig. 3(a) shows the performance for 90 view angles. The performance trends are similar for both implicit neural and spline measurement representations. For a given measurement noise SNR, our method shows increasing improvements as the view angle SNR decreases (measurement coordinate uncertainty increases). This shows that our method handles measurement coordinate uncertainty well, especially when the uncertainties begin to dominate over measurement noise. We can also see that for a fixed angle SNR, the performance gains decrease as measurement SNR decreases and measurement errors becomes more dominant. Fig. 3(b) shows that the same trends hold when there are 120 view angles. The slight SNR drops seen with splines in Fig. 3(a), when the measurement SNR is 30 dB and 35 dB, do not appear in Fig. 3(b). This is because there are more view angles which makes the imaging problem less ill-posed.

In Fig. 4, we show some randomly chosen example ground truth, pseudoinverse filtered backprojection (FBP) and baseline reconstructions with their SNRs when there are 90 view angles. The reconstructions using our framework with implicit neural and spline measurement representations are also shown. Compared to the baseline, solutions obtained using our framework have fewer artifacts. Note that Fig. 4 shows specific examples, and that the average performance for different combinations of measurement noise and view angle uncertainty is shown in Fig. 3(a). Fig. 11 in the Appendix shows these same reconstructions when there are 120 view angles instead.

2) Learned View Angles Accuracy: In the next numerical experiment we verify that the learned measurement coordinates, $\widehat{\boldsymbol{\mu}}$ in (8), are close to the true unknown measurement coordinates $\widetilde{\boldsymbol{\mu}}$. As the detector locations have no error, we verify the learned view angles only. We denote the set of true unknown view angles and learned view angles as $\widetilde{\Theta}$ and $\widehat{\Theta}$. We quantitatively measure the average angle error in degrees as

$$\text{Average angle error} = \frac{1}{J}\|\widetilde{\Theta} - \widehat{\Theta}\|_1, \tag{14}$$

where J is the number of view angles.

Fig. 5 shows how the average angle error changes as the optimization iterations of (8) progress. This is shown for one of the test images with different combinations of measurement noise and view angle uncertainty when there are 90 view angles. The solid lines are for implicit neural representations and the dashed lines are for spline representations. For both representation types, the angle error reduces as (8) is solved which confirms that our framework learns measurement coordinates that are more accurate than the assumed measurement coordinates which they were initialized with. The average angle error when using the assumed measurement coordinates for reconstruction, as is done in the baseline, is the initial point on the plots. In some instances, the average angle error may increase slightly as the optimization of (8) progresses. This behavior can be explained as overfitting of the measurement representations. Fig. 12 in the Appendix shows that the same trends hold when when there are 120 view angles.

3) Reconstruction With More Measurements: Next we investigate a variant of the main problem considered in this paper: in addition to the M true measurement coordinates being unknown, the reconstruction method $G_{\boldsymbol{\mu}}(\cdot)$ is now designed for M' measurements where $M' \geq M$. In this case $\widetilde{\boldsymbol{\mu}} = (\widetilde{\mu}_1, \ldots, \widetilde{\mu}_M)$ as before and now $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_{M'})$. As the measurement representation $r_{\boldsymbol{\varphi}}(\cdot)$ can be evaluated at any measurement coordinate, we can evaluate (7) at M' measurement coordinates.

In this simulated experiment we obtain measurements from 90 view angles. As in the previous experiments, the true view angles are unknown and we initialize the angles of measurements

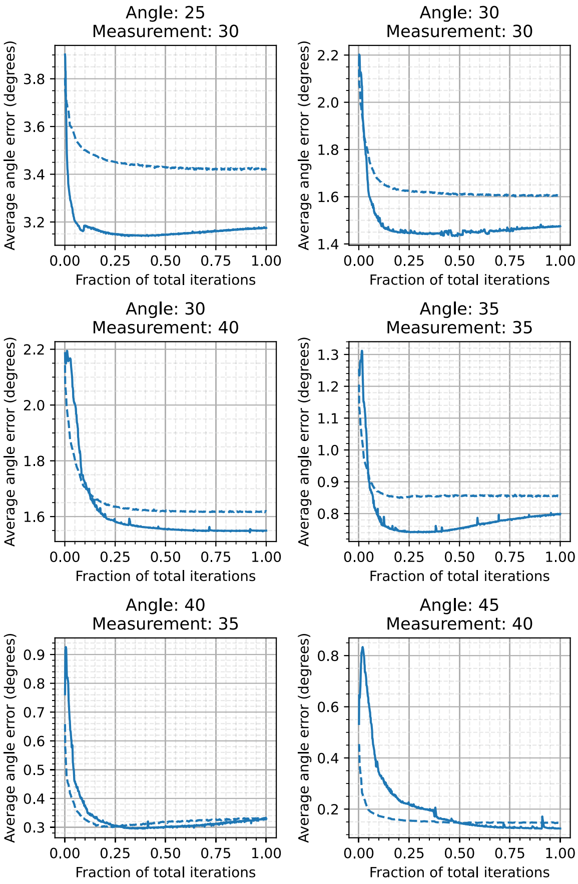


Fig. 5. Average angle error for one test image with different combinations of measurement noise and view angle uncertainty when there are 90 2D CT view angles. The solid lines are for implicit neural representations and the dashed lines are for spline representations.

coordinates $\boldsymbol{\nu}$ in (6) to be 90 uniformly spaced view angles in the interval $[0, \pi]$. The detector locations have no uncertainty. We consider different values of M' by varying the number of view angles J'. The number of detectors K are not varied which gives $M' = J' \cdot K$. Again, to obtain the strongest performance, we use state-of-the-art reconstruction Unets. We try two different reconstruction Unets: 1) $G^{J'}_{\boldsymbol{\mu}}(\cdot)$ which was trained with training data having J' view angles uniformly spaced on $[0, \pi]$ and, 2) $G^{90}_{\boldsymbol{\mu}}(\cdot)$ which was trained with training data having 90 view angles uniformly spaced on $[0, \pi]$.[3]

Table I shows the average reconstruction SNR over 25 test images. There is 35 dB measurement noise and the unknown true view angles are perturbed by 35 dB from 90 uniformly spaced view angles. With Unet $G^{J'}_{\mu}(\cdot)$, the performance fluctuates. The performance is stable with Unet $G^{90}_{\boldsymbol{\mu}}(\cdot)$. With both reconstruction methods, the best performance is seen when $J' = 90$ ($M' = M$) and evaluating more measurements does not help. This is because the fitting loss (6) ensures that $r_{\boldsymbol{\varphi}}(\cdot)$ accurately represents the M obtained measurements which were

[3] The number of Unet training view angles and evaluated view angles J' can be different. This is because the Unet input is computed by a filtered backprojection for J' view angles which always results in an Unet input with the dimensions of the image being reconstructed.

TABLE I
AVERAGE 2D CT RECONSTRUCTION SNR WHEN RECONSTRUCTING WITH MORE MEASUREMENTS

	$G^{J'}_{\boldsymbol{\mu}}(\cdot)$		$G^{90}_{\boldsymbol{\mu}}(\cdot)$	
J'	Implicit	Spline	Implicit	Spline
90	**23.5**	**23.6**	**23.5**	**23.6**
110	22.6	23.1	**23.7**	**23.6**
120	23.4	23.1	**23.8**	**23.6**
130	23.2	22.8	**23.7**	**23.5**
150	22.9	22.5	**23.8**	**23.6**
180	22.3	22.0	**23.9**	**23.6**

sampled from the measurement space at coordinates $\widetilde{\boldsymbol{\mu}}$. Then when $M' = M$, because the measurement coordinates used for reconstruction, $\boldsymbol{\mu}$, are a small perturbation away from $\widetilde{\boldsymbol{\mu}}$, they are also represented accurately which results in good reconstructions. Furthermore, when $M' = M$, there are enough measurement coordinates that densely cover the measurement space. The accuracy of the measurement coordinates for the extra measurements when $M' > M$ is not enforced by the fitting loss (6).

It has been shown that reconstruction with more measurements can help when there is no measurement coordinate uncertainty, and when the input to the reconstruction neural network combines the observed measurements with the measurement representation output [30]. When there is measurement coordinate uncertainty, using the observed measurements in the input to the reconstruction neural network can reduce performance as shown by Fig. 2 and the baseline reconstructions in Fig. 4.

B. Detector Location Uncertainty

The previous experiments demonstrate that our framework improves over the baseline when there is uncertainty in the view angles. Next we consider the case where there is only uncertainty in the detector location and no uncertainty in the view angles. In this set of experiments there are always 90 uniformly spaced view angles. As there is only uncertainty in detector locations, we only optimize the detector location dimension of the measurement coordinates in (8).

1) Combinations of Measurement Noise and Detector Location Error: Similar to Section III-A1, we consider different combinations of measurement noise and detector location uncertainty. We do trials over the same 25 test images. In each trial, different measurement noise and detector location perturbations are used.

The average reconstruction SNR improvement over the baseline is shown in Fig. 6. For a given measurement noise SNR, our method shows increasing improvements as the detector location SNR decreases (measurement coordinate uncertainty increases). This is consistent with the previous results for view angle measurement coordinate uncertainty shown in Fig. 3. Our method handles measurement coordinate uncertainty well, especially when the uncertainties begin to dominate over measurement noise. Note that our framework does not assume a specific uncertainty model—we have used different uncertainty simulation models for view angle uncertainty in Fig. 3 and detector location uncertainty in Fig. 6. Compared to Fig. 3, the curves in Fig. 6 for high measurement SNR and measurement coordinate uncertainty SNR almost overlap at times because the baseline SNR is higher and so the average SNR improvement is lower.

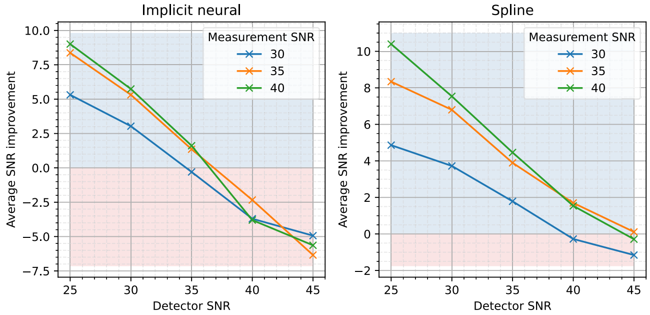


Fig. 6. Average SNR improvement (dB) when solving (8) for 2D CT imaging. There are different combinations of measurement noise SNR and detector location uncertainty SNR. There are 90 view angles.

In Fig. 7, we show randomly chosen example reconstructions with their SNRs.

2) Learned Detector Locations Accuracy: Similar to Section III-A2, in the next simulation we verify that the learned detector locations are close to the true unknown detector locations. We denote the set of true unknown detector locations and learned detector locations as $\widetilde{\mathcal{T}}$ and $\widehat{\mathcal{T}}$. We measure the average detector location error as

$$\text{Average detector error} = \frac{1}{K}\|\widetilde{\mathcal{T}} - \widehat{\mathcal{T}}\|_1, \tag{15}$$

where K is the number of detectors.

Fig. 8 shows how the average detector error changes as the optimization progresses for different combinations of measurement noise and detector location uncertainty. Again, the solid lines are for implicit neural representations and the dashed lines are for spline representations. Compared to the assumed detector locations, for both representation types, the final learned detector locations are closer to the true detector locations.

C. View Angle and Detector Location Uncertainty

The numerical experiments in Sections III-A and III-B demonstrates that our proposed framework performs well when there is view angle or detector location measurement coordinate uncertainty. We now perform simulations when there is view angle and detector location uncertainty at the same time. There are 90 view angles in these experiments. As there is uncertainty in both the view angles and the detector locations, we will optimize both the view angle and detector location dimensions of the measurement coordinates in (8).

1) Combinations of Measurement Noise and Measurement Coordinate Error: Following the experiments of Figs. 3 and 6, we consider different combinations of measurement noise and measurement coordinate uncertainty over the same 25 test images. The view angle and detector location SNRs are the same in these experiments.

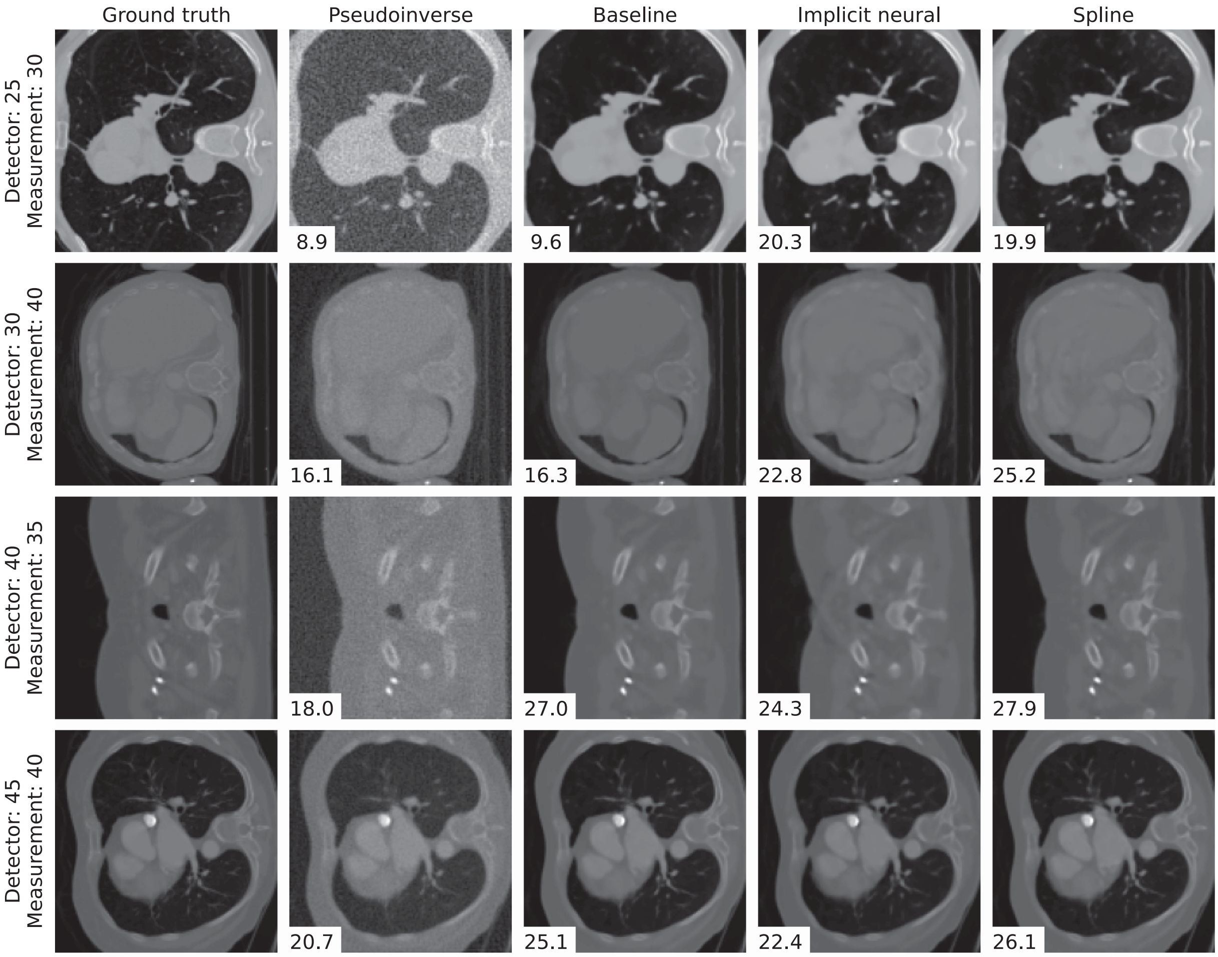


Fig. 7. Example reconstructions for 2D CT imaging. There are different measurement noise SNR and detector location uncertainty SNR combinations and 90 view angles. The reconstruction SNRs are shown for each reconstruction.

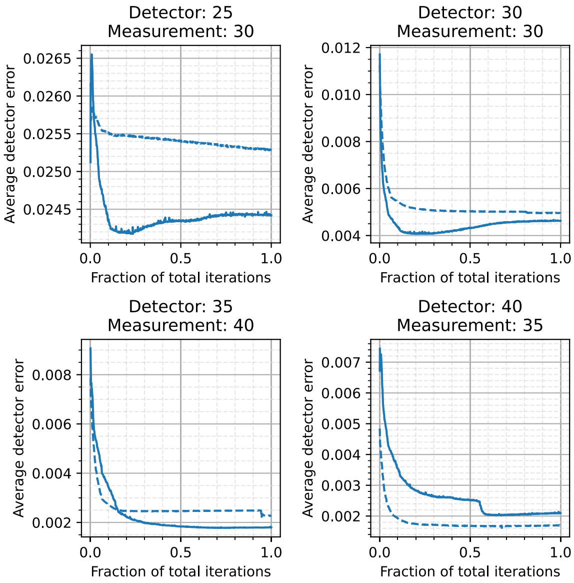


Fig. 8. Average detector error for one test image with different combinations of measurement noise and measurement detector location uncertainty when performing 2D CT imaging. The solid lines are for implicit neural representations and the dashed lines are for spline representations.

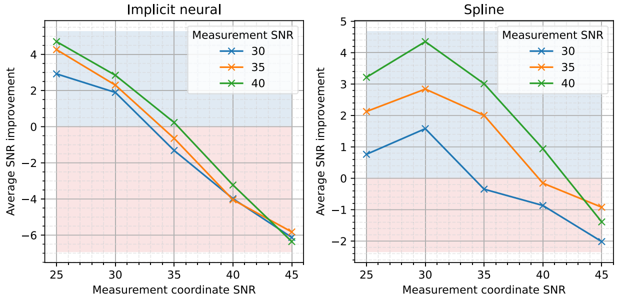


Fig. 9. Average SNR improvement (dB) for 2D CT imaging when solving (8) for different combinations of measurement noise SNR and measurement coordinate uncertainty. There is uncertainty in both the view angles and detector locations.

The average reconstruction SNR improvement over the baseline is shown in Fig. 9. Consistent with previous experiments, Fig. 9 shows that we are able to improve upon the baseline when there is measurement coordinate uncertainty. In Fig. 10, we also show randomly chosen example reconstructions with their SNRs when there is measurement coordinate uncertainty in both the view angles and the detector locations.

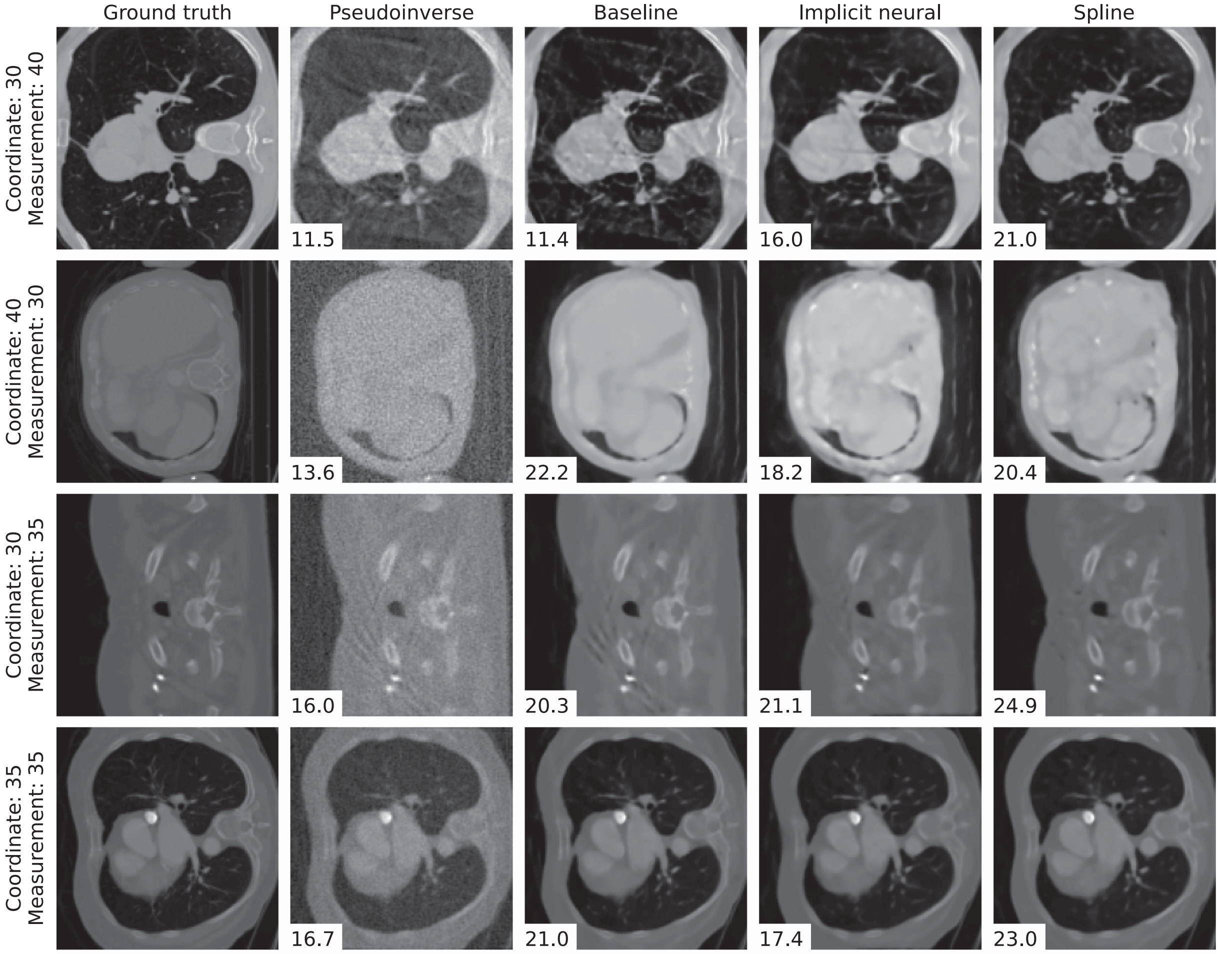


Fig. 10. Example reconstructions for 2D CT imaging. There are different measurement noise SNR and measurement coordinate uncertainty SNR combinations. There is uncertainty in both the view angles and detector locations. The reconstruction SNRs are shown for each reconstruction.

IV. Conclusion

We presented a differentiable imaging inverse problem framework to jointly reconstruct the unknown image and learn unknown measurement coordinates when they are approximately known. There are two major elements in our proposed method. Firstly, we learn continuous representations of the measurements whose input are measurement coordinates and output are the corresponding measurements. By optimizing with respect to their parameters and their input, we jointly learn the measurement representation parameters and the unknown measurement coordinates. The second aspect of our method is that because these representations can be evaluated at any input coordinate, we can leverage reconstruction methods that are designed for measurement coordinates that are different from the ones of the observations. Our 2D and 3D CT imaging simulations show that the benefit of using our framework increases with the level of measurement coordinate uncertainty.

As our framework does not assume a particular measurement representation, we use both implicit neural networks and splines to represent measurements. Splines are generally viewed as interpolation tools, however, our work demonstrates that they can also be learnable differentiable representations that perform comparably to implicit neural representations. Differentiable splines may provide a viable solution for current research directions that have been focusing on using implicit neural networks which can have significantly more parameters and complexity [24].

A strength of our framework is that no extra training data is required to learn the measurement representations. However, a drawback is that it can be time consuming if there are multiple test images because we have to learn separate measurement representations and measurement coordinates for each new set of observations. Therefore, extending our framework to jointly recover a batch of images and the shared unknown measurement coordinates is an important step towards helping practitioners adopt our framework. Batch imaging also introduces robustness which can help learn more accurate measurement representations.

Another important endeavor is to adapt our framework to account for operator uncertainties due to reasons other than measurement coordinate uncertainty which this paper studied. For example, there may be approximation uncertainties if the true operator is approximated to enable faster computations and facilitate analysis. Not accounting for the approximation can lead to degraded solutions [49]. Another source of operator

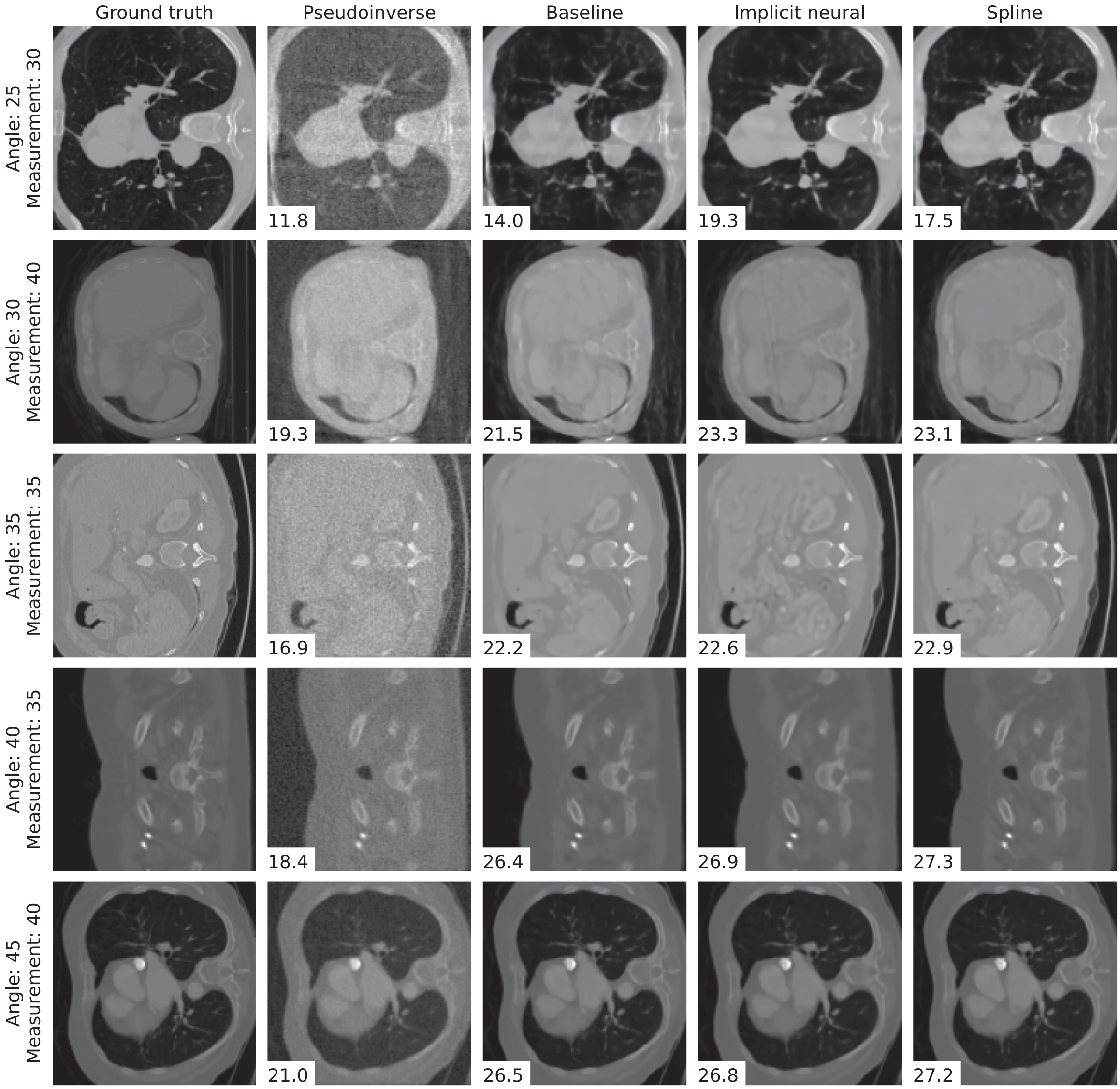


Fig. 11. Example reconstructions for different measurement noise and 2D CT view angle uncertainty. There are 120 view angles. The SNRs are shown for each reconstruction.

uncertainty arises when the object being imaged is altered in an unknown manner during the measurement acquisition process. For example, in CryoET imaging, the sample can translate and deform during imaging which needs to be taken into account [50], [51], [52].

Appendix A
2D CT Simulations With 120 View Angles

In Fig. 11 we show 2D CT imaging sample reconstructions for different combinations of measurement noise and measurement coordinate uncertainty when there are 120 view angles. The simulated experiment is performed as described in Section III-A1. The reconstructions for when there are 90 view angles is shown in Fig. 4.

Fig. 12 shows average angle error plots for 2D CT imaging when there are 120 view angles. The experimental details are in Section III-A2 and the equivalent results for 90 view angles are shown in Fig. 5.

Appendix B
Three-Dimensional CT Imaging

Similar to 2D CT imaging, in 3D CT imaging, two-dimensional projections of a three-dimensional volume are collected by tilting it at different tilt angles. The goal is to reconstruct the volume from the the series of projections. Inspired by CryoET imaging, we obtain projections at 60 tilt angles which we assume to be uniformly spaced on the interval $[-\pi/3, \pi/3]$. The true unknown tilt angles are perturbed from these. This setup is challenging because there is a 'wedge' of tilt angles for which we do not have projections. The detectors are uniformly spaced on a two-dimensional unit square $[0,1]^2$ and have no uncertainty in these experiments. Combining these spaces gives

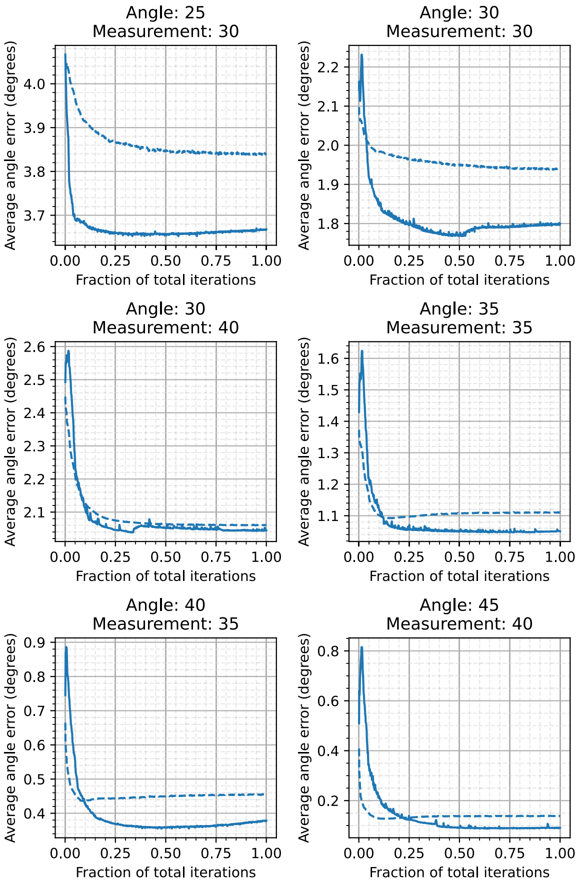


Fig. 12. Average angle error for one test image with different measurement noise and measurement coordinate uncertainty combinations when there are 120 2D CT view angles. The solid lines are for implicit neural representations and the dashed lines are for spline representations.

the measurement coordinates space as $\Omega = [-\pi/3, \pi/3] \times [0,1]^2$. The tilt angle uncertainty is simulated in the same way as the 2D CT imaging view angle uncertainty. Furthermore, we only optimize the tilt angles dimension of the measurement coordinates in (8) because the detector locations have no uncertainty in these simulations.

To train the reconstruction 3D Unets, we use 3D volumes from the 2019 Brain Tumor Segmentation (BraTS) Challenge dataset [53], [54], [55]. The volumes are resized to be $64 \times 64 \times 64$. We use 478 volumes for training and 15 separate volumes to test and verify our framework.

A. *Combinations of Measurement and Coordinate Error*

We consider different combinations of measurement noise and measurement coordinate uncertainty (in tilt angles) in the same way as was done for 2D CT. The trials are done over 15 test volumes and the results are shown in Fig. 13. We see that the performance trends for both implicit neural and spline representations are similar. Furthermore, the performance is consistent with what was seen for the 2D CT problem (Fig. 3)—as measurement coordinate uncertainty increases, our framework provides increasing gains.

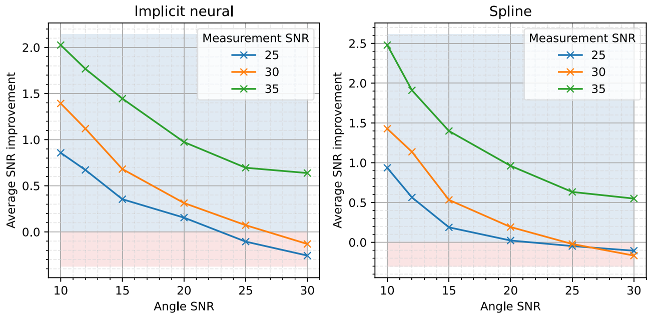


Fig. 13. Average SNR improvement (dB) when using our framework (8) for 3D CT. There are different combinations of measurement noise SNR and measurement coordinate uncertainty SNR. There are 60 tilt angles on the interval $[-\pi/3, \pi/3]$.

Fig. 14 shows two-dimensional slices through some of the test volumes. The ground truth, psuedoinverse, baseline and our reconstructions are shown with the SNRs of the entire volumes. Similarly, Fig. 15 shows 3D reconstructions of the same test volumes. Two central orthogonal volume slices have been plotted. Reconstructions using our framework are better than the baseline at recovering the geometric structure of the volumes and have fewer artifacts.

B. *Learned Tilt Angles Accuracy*

Next we verify that the learned measurement coordinates are accurate. The uncertainty is only in the tilt angles and we follow the same procedure as when verifying the learned 2D CT view angles in Fig. 5. We use the metric defined in (14) and show the results in Fig. 16 for one test volume with different measurement noise and measurement coordinate uncertainty combinations. Again, the solid lines are for the implicit neural representations, and the dashed lines are for the spline representations. For both representation types, the average tilt angle error reduces as the optimization of (8) progresses. This demonstrates that our framework learns tilt angles (measurement coordinates) that are more accurate than the assumed ones.

Appendix C
Additional Spline Information

We continue using 2D CT as an example and build on Section II-C2 to provide further information on NURBS. Besides the learnable weights and learnable control point vectors explained in Section II-C2, there is a knot vector for each dimension of the measurement coordinates that is not learnable. The NURBS surface also has degrees $d_\Theta, d_\mathcal{T} \in \mathbb{Z}^+$ for the view angle and detector location measurement coordinate dimensions. The knot vector for the view angle dimension has $(J + d_\Theta + 1)$ elements which are arranged in ascending order. We design its uth element, k_u, to be zero when $0 \leq u < d_\Theta + 1$, uniformly spaced between zero and one when $d_\Theta + 1 \leq u \leq J$, and one when $J < u \leq J + d_\Theta$ [23]. The knot vector for the detector location dimension has $(J + d_\mathcal{T} + 1)$ elements and is made in a similar manner by using its degree $d_\mathcal{T}$.

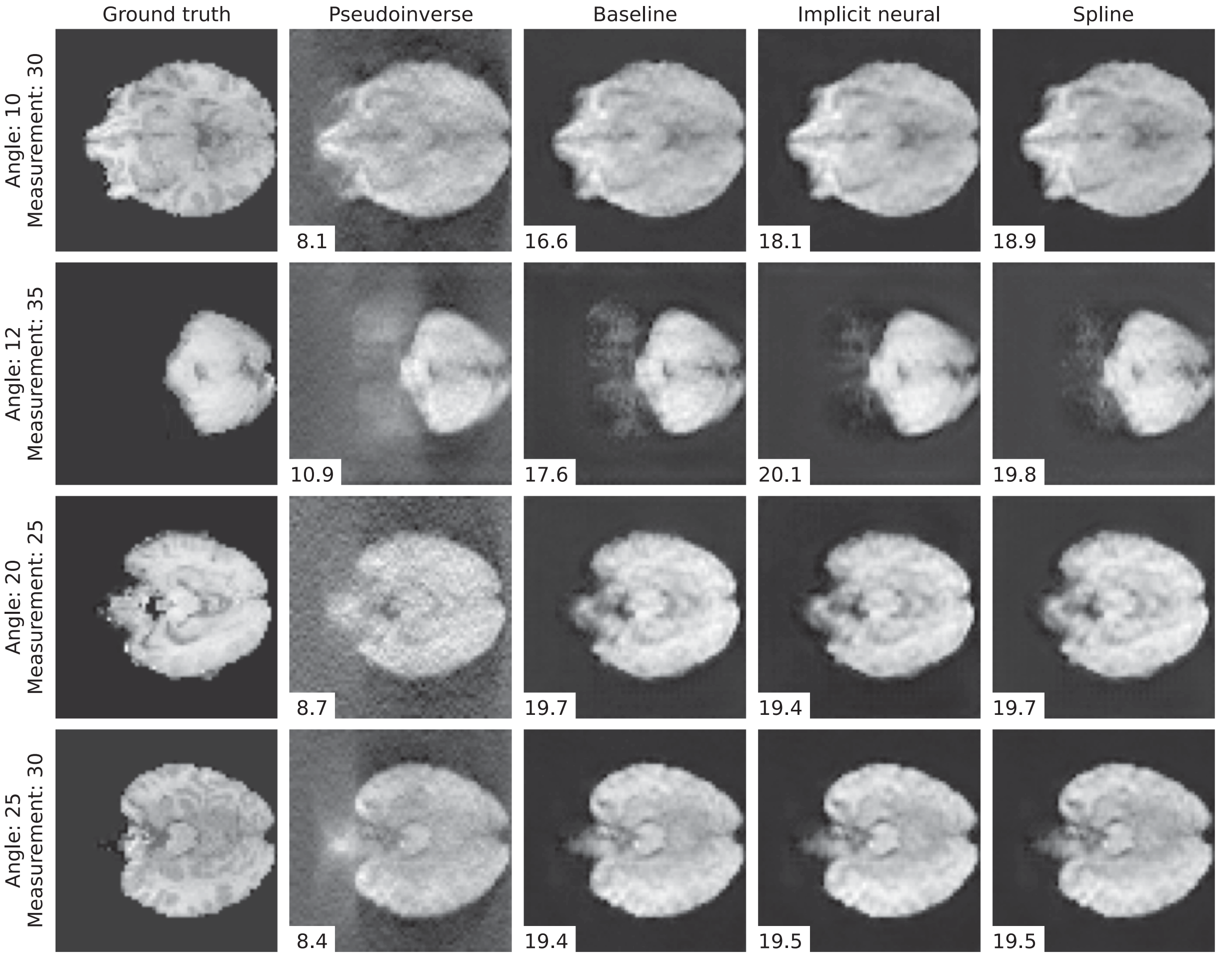


Fig. 14. Example reconstructions of slices of the 3D test volumes. Reconstructions for different measurement noise SNR and measurement coordinate uncertainty SNR are shown. The reconstruction SNRs for each volume are stated.

The rational basis functions defined in (10) are

$$b_{j,k}\left([\theta', t']^\top\right) = \frac{w_{j,k}\, Q_{j,d_\Theta}(\theta')\, Q_{k,d_\mathcal{T}}(t')}{\sum_{u=0}^{J-1}\sum_{v=0}^{K-1} w_{u,v}\, Q_{u,d_\Theta}(\theta')\, Q_{v,d_\mathcal{T}}(t')}. \tag{16}$$

The function $Q_{u,d}(\cdot)$ is the uth B-spline basis function of degree d. If k_u denotes the uth element of a knot vector, each basis function can be obtained using the Cox-de Boor recursion method,

$$Q_{u,0}(k) = \begin{cases} 1 & \text{if } k_u \le k \le k_{u+1} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

$$Q_{u,d}(k) = \frac{k - k_u}{k_{u+d} - k_u} Q_{u,d-1}(k) + \frac{k_{u+d+1} - k}{k_{u+d+1} - k_{u+1}} Q_{u+1,d-1}(k). \tag{18}$$

From (17), we can see that a degree zero NURBS is constructed from piecewise constant basis functions. The recursion (18) is then used to create higher degree basis functions with larger support. If the denominator in any term of (18) is zero, that term is taken to be zero.

The NURBS surface for two-dimensional measurement coordinates in (10) is the tensor product of two one-dimensional NURBS curves as shown in (16). To create NURBS surfaces for D-dimensional measurement coordinates, we take the tensor product of D one-dimensional NURBS curves. This is done for $D = 3$ in Appendix B.

Appendix D
Implementation Details

A. Framework Optimization

In this section we provide implementation details for solving our optimization problem (8). All parameters were tuned on a held out set of images for three randomly chosen measurement and operator error combinations.

To implement NURBS, we modified and extended the PyTorch source code released by the NURBS-Diff module authors [23]. The implicit neural representation and neural network for $G_{\boldsymbol{\mu}}(\cdot)$ in (8) are also implemented in PyTorch. This enables us to conveniently optimize the objective function (8) using automatic differentiation and the Adam optimizer.

When implicit neural networks are used, the optimization is run for at least 8,000 iterations and at most 20,000 iterations.

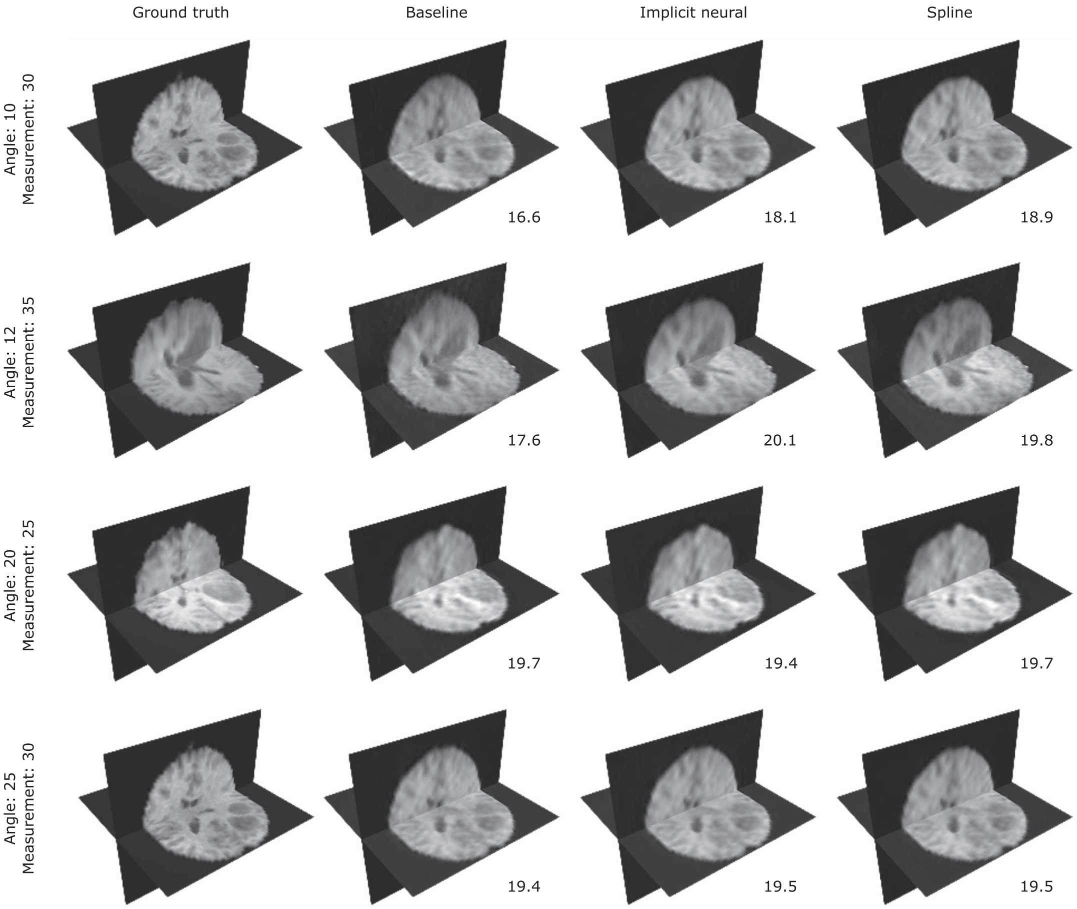


Fig. 15. Example reconstructions for two of the orthogonal central slices of each of the 3D test volumes in Fig. 14 for 3D CT imaging. The SNRs for the entire volume are stated.

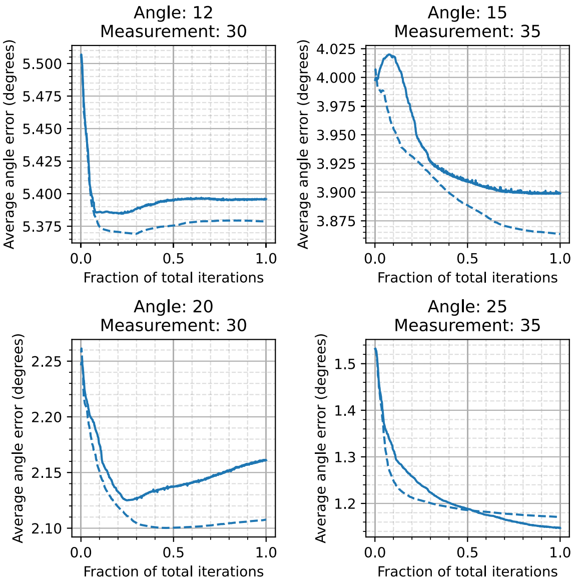


Fig. 16. Average tilt angle error for one 3D CT test volume with different measurement noise and measurement coordinate uncertainty combinations. The solid lines are for implicit neural representations and the dashed lines are for spline representations.

The optimization is terminated when the loss value between successive iterations is below 1×10^{-10} for 2D CT imaging and 1×10^{-11} for 3D CT imaging. When splines are used and there is any view angle or tilt angle uncertainty, the optimization runs for at least 2,000 iterations and at most 5,000 iterations. If there is only detector location uncertainty, the optimization runs for at least 6,000 iterations and at most 15,000 iterations. Additionally, when splines are used, the optimization for all imaging problems is terminated when the loss value between successive iterations is below 1×10^{-11}.

1) 2D CT Imaging: When implicit neural representations are used, $\lambda = 0.1$ in (8). The learning rate for both the neural network parameters and the input coordinates is 5×10^{-4}. When splines are used, $\lambda = 0.025$ if there is only view angle uncertainty, and $\lambda = 0.25$ if there is any detector location uncertainty. The learning rate for the neural network parameters is 5×10^{-2} and for the input coordinates is 2×10^{-4}.

For the implicit neural representation, we use the cosine of the view angle rather than the angle when creating the measurement coordinate. This encodes the circular nature of angular data and improves performance.

2) 3D CT Imaging: When implicit neural representations are used, $\lambda = 0.1$. The learning rate for the neural network parameters is 1×10^{-3} and for the input coordinates is 1×10^{-4}. When splines are used, $\lambda = 0.6$. The learning rate for the neural network parameters is 5×10^{-3} and for the input coordinates is 1×10^{-4}.

B. Reconstruction Unet Training

In this section we describe the implementation details for the Unet neural networks used for the reconstruction method $G_{\boldsymbol{\mu}}(\cdot)$ in (8).

A mean-squared error loss function is minimized using Adam during training. We train the Unets for 100 epochs where one epoch is a full pass through the training dataset.

For the 2D Unet, a batch size of 128 and learning rate of 1 $\times 10^{-3}$ is used. For the 3D Unet, a batch size of 16 and learning rate of 1×10^{-3} is used.

Publicly available Unet architectures were downloaded and trained. Unless mentioned here, the default parameters from the download sources were used. The 2D Unet model is from https://github.com/mateuszbuda/brain-segmentation-pytorch [56]. We used one input channel, one output channel, and 16 features in the first layer. The 3D Unet model is from https://github.com/ELEKTRONN/elektronn3. We used one input channel, one output channel, 16 features in the first layer and a depth of four blocks.

Due to the limited size of the 3D volume training dataset for 3D CT imaging, we use a data augmentation strategy. We perform random horizontal flips, random vertical flips, and random rotations by 90, 180 or 270 degrees.

C. NURBS

1) 2D CT Imaging: For 2D CT imaging, the NURBS degree along the measurement coordinate dimension being learned is 18. It is two in the measurement coordinate dimension that is not being learned. Furthermore, where there is view angle uncertainty, we create additional control points to ensure the spline measurement representations satisfy the Radon transform measurement consistency conditions [57]

$$r_{\boldsymbol{\varphi}}\left([\theta+\pi, t]^{\top}\right) = r_{\boldsymbol{\varphi}}\left([\theta, -t]^{\top}\right), \tag{19}$$

and

$$r_{\boldsymbol{\varphi}}\left([\theta-\pi, t]^{\top}\right) = r_{\boldsymbol{\varphi}}\left([\theta, -t]^{\top}\right), \tag{20}$$

where a detector location of $-t$ refers to the tth last detector. This gives the locally supported NURBS basis functions (16) a sufficient number of control points around 0 and π radians and ensures the NURBS is accurate in the interval $[0, \pi]$ radians. Specifically, if d_Θ is the degree of the spline in the view angle dimension, we use the consistency condition to create new control points for (10)

$$\boldsymbol{p}_{j+J-1,k} = \left[\theta_j + \pi, t_k, \boldsymbol{y}\left(\left[\widetilde{\theta}_j, -\widetilde{t}_k\right]^{\top}\right)\right]^{\top} \tag{21}$$

for $1 \leq j \leq d_\Theta$ and

$$\boldsymbol{p}_{j-J-1,k} = \left[\theta_j - \pi, t_k, \boldsymbol{y}\left(\left[\widetilde{\theta}_j, -\widetilde{t}_k\right]^{\top}\right)\right]^{\top} \tag{22}$$

for $J - d_\Theta < j \leq J$ where J is the number of view angles in the observed measurements. Essentially, if there are K detectors, we create $2d_\Theta K$ additional control points.

2) 3D CT Imaging: For 3D CT, the NURBS degree along the tilt angle measurement coordinate dimension is 15. It is two in the two detector location dimensions. Additionally, for this imaging technique, we fix the basis function weights to one and do not optimize them. This makes the NURBS surface a B-spline surface.

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*, 2015, pp. 234–241.

[2] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.

[3] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 85–95, Nov. 2017.

[4] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep learning techniques for inverse problems in imaging," *IEEE J. Sel. Areas Inf. Theory*, vol. 1, no. 1, pp. 39–56, May 2020.

[5] S. Antholzer, M. Haltmeier, and J. Schwab, "Deep learning for photoacoustic tomography from sparse data," *Inverse Problems Sci. Eng.*, vol. 27, no. 7, pp. 987–1005, 2019.

[6] K. Kothari, S. Gupta, M. v. de Hoop, and I. Dokmanic, "Random mesh projectors for inverse problems," in *Proc. Int. Conf. Learn. Representations*, 2019.

[7] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9446–9454.

[8] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.*, 2010, pp. 399–406.

[9] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1322–1332, Jun. 2018.

[10] H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1440–1453, Jun. 2018.

[11] J. R. Chang, C.-L. Li, B. Poczos, B. V. Kumar, and A. C. Sankaranarayanan, "One network to solve them all–solving linear inverse problems using deep projection models," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5888–5897.

[12] D. Gilton, G. Ongie, and R. Willett, "Neumann networks for linear inverse problems in imaging," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 328–343, 2020.

[13] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6360–6376, Oct. 2022.

[14] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 537–546.

[15] K. Kothari, A. Khorashadizadeh, M. de Hoop, and I. Dokmanić, "Trumpets: Injective flows for inference and inverse problems," in *Proc. Uncertainty Artif. Intell.*, 2021, pp. 1269–1278.

[16] Y. Song, L. Shen, L. Xing, and S. Ermon, "Solving inverse problems in medical imaging with score-based generative models," in *Proc. Int. Conf. Learn. Representations*, 2022.

[17] D. Gilton, G. Ongie, and R. Willett, "Model adaptation for inverse problems in imaging," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 661–674, 2021.

[18] A. Gossard and P. Weiss, "Training adaptive reconstruction networks for inverse problems," 2022, *arXiv:2202.11342*.

[19] R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. Burlington, MA, USA: Morgan Kaufmann, 1995.

[20] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*. New York, NY, USA: McGraw-Hill, 1989.

[21] L. Piegl, "On NURBS: A survey," *IEEE Comput. Graph. Appl.*, vol. 11, no. 01, pp. 55–71, Jan./Feb. 1991.

[22] E. Cohen, T. Lyche, and R. Riesenfeld, "Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics," *Comput. Graph. Image Process.*, vol. 14, no. 2, pp. 87–111, 1980.

[23] A. D. Prasad, A. Balu, H. Shah, S. Sarkar, C. Hegde, and A. Krishnamurthy, "NURBS-diff: A differentiable programming module for NURBS," *Comput.-Aided Des.*, vol. 146, 2022, Art. no. 103199.

[24] Y. Xie et al., "Neural fields in visual computing and beyond," *Comput. Graph. Forum*, vol. 41, no. 2, pp. 641–676, 2022.

[25] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 165–174.

[26] 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 *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 405–421.

[27] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, "Implicit neural representations with periodic activation functions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 7462–7473.

[28] A. W. Reed et al., "Dynamic CT reconstruction from limited views with implicit neural representations and parametric motion fields," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 2258–2268.

[29] L. Lozenski, M. Anastasio, and U. Villa, "Neural fields for dynamic imaging," in *Medical Imaging: Physics Medical Imaging*, vol. 12031. Bellingham, WA, USA: SPIE, 2022, pp. 231–238.

[30] Y. Sun, J. Liu, M. Xie, B. Wohlberg, and U. S. Kamilov, "CoIL: Coordinate-based internal learning for tomographic imaging," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1400–1412, 2021.

[31] N. Chen, L. Cao, T.-C. Poon, B. Lee, and E. Y. Lam, "Differentiable imaging: A new tool for computational optical imaging," *Adv. Phys. Res.*, vol. 2, no. 6, 2023, Art. no. 2200118.

[32] M. Du, Y. S. Nashed, S. Kandel, D. Gürsoy, and C. Jacobsen, "Three dimensions, two microscopes, one code: Automatic differentiation for X-ray nanotomography beyond the depth of focus limit," *Sci. Adv.*, vol. 6, no. 13, 2020, Art. no. eaay3700.

[33] M. Du et al., "Adorym: A multi-platform generic X-ray image reconstruction framework based on automatic differentiation," *Opt. Exp.*, vol. 29, no. 7, pp. 10000–10035, 2021.

[34] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—a modern synthesis," in *Proc. Vis. Algorithms: Theory Pract.: Int. Workshop Vis. Algorithms Corfu*, 2000, pp. 298–372.

[35] P. Campisi and K. Egiazarian, *Blind Image Deconvolution: Theory and Applications*. Boca Raton, FL, USA: CRC Press, 2017.

[36] T. F. Chan and C.-K. Wong, "Total variation blind deconvolution," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 370–375, Mar. 1998.

[37] N. A. B. Riis, Y. Dong, and P. C. Hansen, "Computed tomography reconstruction with uncertain view angles by iteratively updated model discrepancy," *J. Math. Imag. Vis.*, vol. 63, no. 2, pp. 133–143, 2021.

[38] S. Basu and Y. Bresler, "Uniqueness of tomography with unknown view angles," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1094–1106, Jun. 2000.

[39] R. R. Coifman, Y. Shkolnisky, F. J. Sigworth, and A. Singer, "Graph Laplacian tomography from unknown random projections," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1891–1899, Oct. 2008.

[40] T. Bendory, A. Bartesaghi, and A. Singer, "Single-particle cryo-electron microscopy: Mathematical theory, computational challenges, and opportunities," *IEEE signal Process. Mag.*, vol. 37, no. 2, pp. 58–76, Mar. 2020.

[41] G. H. Golub and C. F. Van Loan, "An analysis of the total least squares problem," *SIAM J. Numer. Anal.*, vol. 17, no. 6, pp. 883–893, 1980.

[42] I. Markovsky and S. Van Huffel, "Overview of total least-squares methods," *Signal Process.*, vol. 87, no. 10, pp. 2283–2302, 2007.

[43] S. Gupta and I. Dokmanić, "Total least squares phase retrieval," *IEEE Trans. Signal Process.*, vol. 70, pp. 536–549, 2022.

[44] L. Piegl and W. Tiller, *The NURBS Book*. Berlin, Germany: Springer Science & Business Media, 1996.

[45] D. F. Rogers, *An Introduction to NURBS: With Historical Perspective*. Burlington, MA, USA: Morgan Kaufmann, 2001.

[46] J. Adler, H. Kohr, and O. Öktem, "Operator discretization library (ODL)," Zenodo, 2017, doi: 10.5281/zenodo.249479.

[47] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*, 2016, pp. 424–432.

[48] J. Leuschner, M. Schmidt, D. O. Baguer, and P. Maaß, "The LoDoPaB-CT dataset: A benchmark dataset for low-dose CT reconstruction methods," 2019, *arXiv:1910.01113*.

[49] S. Lunz, A. Hauptmann, T. Tarvainen, C.-B. Schonlieb, and S. Arridge, "On learned operator correction in inverse problems," *SIAM J. Imag. Sci.*, vol. 14, no. 1, pp. 92–127, 2021.

[50] D. Tegunov and P. Cramer, "Real-time cryo-electron microscopy data preprocessing with warp," *Nature Methods*, vol. 16, no. 11, pp. 1146–1152, 2019.

[51] K. Naydenova, P. Jia, and C. J. Russo, "Cryo-EM with sub–1 Å specimen movement," *Science*, vol. 370, no. 6513, pp. 223–226, 2020.

[52] J.-J. Fernandez and S. Li, "TomoAlign: A novel approach to correcting sample motion and 3D CTF in CryoET," *J. Struct. Biol.*, vol. 213, no. 4, 2021, Art. no. 107778.

[53] B. H. Menze et al., "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[54] S. Bakas et al., "Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features," *Sci. Data*, vol. 4, no. 1, pp. 1–13, 2017.

[55] S. Bakas et al., "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge," 2018, *arXiv:1811.02629*.

[56] M. Buda, A. Saha, and M. A. Mazurowski, "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm," *Comput. Biol. Med.*, vol. 109, pp. 218–225, 2019.

[57] J. L. Prince and A. S. Willsky, "Constrained sinogram restoration for limited-angle tomography," *Opt. Eng.*, vol. 29, no. 5, pp. 535–544, 1990.