

# QN-Mixer: A Quasi-Newton MLP-Mixer Model for Sparse-View CT Reconstruction

Anonymous CVPR submission

Paper ID 15740

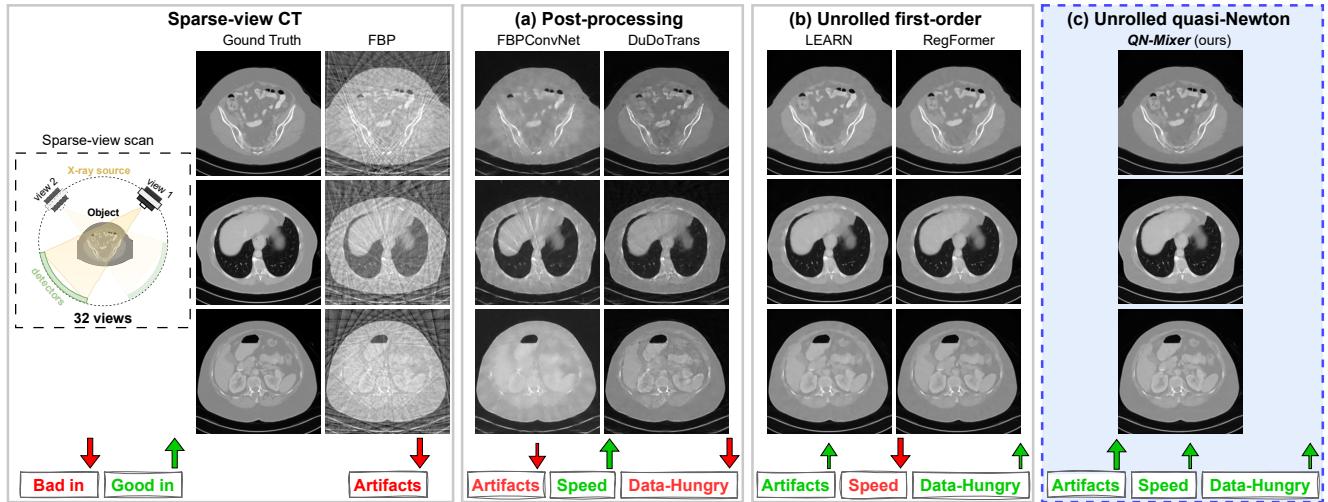


Figure 1. **CT Reconstruction of State-of-the-Art Methods.** We present the results of sparse-view CT reconstruction employing our QN-Mixer approach with 32 views. A comparative analysis against state-of-the-art post-processing and first-order unrolling networks reveals the superiority of QN-Mixer in artifact removal, training time, and data efficiency.

## Abstract

001 *Inverse problems span across diverse fields. In medical contexts, computed tomography (CT) plays a crucial*  
 002 *role in reconstructing a patient's internal structure, presenting challenges due to artifacts caused by inherently ill-*  
 003 *posed inverse problems. Previous research advanced image quality via post-processing and deep unrolling algo-*  
 004 *rithms but faces challenges, such as extended convergence*  
 005 *times with ultra-sparse data. Despite enhancements, resulting images often show significant artifacts, limiting their*  
 006 *effectiveness for real-world diagnostic applications. We*  
 007 *aim to explore deep second-order unrolling algorithms for*  
 008 *solving imaging inverse problems, emphasizing their faster*  
 009 *convergence and lower time complexity compared to com-*  
 010 *mon first-order methods like gradient descent. In this paper,*  
 011 *we introduce QN-Mixer, an algorithm based on the quasi-*  
 012 *Newton approach. We use learned parameters through the*  
 013 *BFGS algorithm and introduce Incept-Mixer, an efficient*  
 014 *neural architecture that serves as a non-local regularization*  
 015 *term, capturing long-range dependencies within images.*

020 *To address the computational demands typically associated*  
 021 *with quasi-Newton algorithms that require full Hessian*  
 022 *matrix computations, we present a memory-efficient alterna-*  
 023 *tive. Our approach intelligently downsamples gradient*  
 024 *information, significantly reducing computational require-*  
 025 *ments while maintaining performance. The approach is val-*  
 026 *idated through experiments on the sparse-view CT problem,*  
 027 *involving various datasets and scanning protocols, and is*  
 028 *compared with post-processing and deep unrolling state-*  
 029 *of-the-art approaches. Our method outperforms existing*  
 030 *approaches and achieves state-of-the-art performance in*  
 031 *terms of SSIM and PSNR, all while reducing the number*  
 032 *of unrolling iterations required.*

## 1. Introduction

033 Computed tomography (CT) is a widely used imaging  
 034 modality in medical diagnosis and treatment planning, de-  
 035 livering intricate anatomical details of the human body  
 036 with precision. Despite its success, CT is associated with  
 037 high radiation doses, which can increase the risk of can-  
 038

cer induction [46]. Adhering to the ALARA principle (As Low As Reasonably Achievable) [34], the medical community emphasizes minimizing radiation exposure to the lowest level necessary for accurate diagnosis. Numerous approaches have been proposed to reduce radiation doses while maintaining image quality. Among these, sparse-view CT emerges as a promising solution, effectively lowering radiation doses by subsampling the projection data, often referred to as the sinogram. Nonetheless, reconstructed images using the well-known Filtered Back Projection (FBP) algorithm [31], suffer from pronounced streaking artifacts (see Fig. 1), which can lead to misdiagnosis. The challenge of effectively reconstructing high-quality CT images from sparse-view data is gaining increasing attention in both the computer vision and medical imaging communities.

With the success of deep learning spanning diverse domains, initial image-domain techniques [5, 17, 23, 25, 54] have been introduced as post-processing tasks on the FBP reconstructed images, exhibiting notable accomplishments in artifact removal and structure preservation. However, the inherent limitations of these methods arise from their constrained receptive fields, leading to challenges in effectively capturing global information and, consequently, sub-optimal results.

To address this limitation, recent advances have seen a shift toward a dual-domain approach [16, 24, 26, 45], where post-processing methods turn to the sinogram domain. In this dual-domain paradigm, deep neural networks are employed to perform interpolation tasks on the sinogram data [14, 22], facilitating more accurate image reconstruction. Despite the significant achievements of post-processing and dual-domain methods, they confront issues of interpretability and performance limitations, especially when working with small datasets and ultra-sparse-view data, as shown in Fig. 1. To tackle these challenges, deep unrolling networks have been introduced [1, 6, 7, 10, 15, 18, 47, 50]. Unrolling networks treat the sparse-view CT reconstruction problem as an optimization task, resulting in a first-order iterative algorithm like gradient descent, which is subsequently unrolled into a deep recurrent neural network in order to learn the optimization parameters and the regularization term. Like post-processing techniques, unrolling networks have been extended to the sinogram domain [48, 52] to perform interpolation task.

Unrolling networks, as referenced in [11, 33, 41], exhibit remarkable performance across diverse domains. However, they suffer from slow convergence and high computational costs, as illustrated in Fig. 1, necessitating the development of more efficient alternatives [13]. More specifically, they confront two main issues: *Firstly*, they frequently grapple with capturing long-range dependencies due to their dependence on locally-focused regularization terms using CNNs.

This limitation results in suboptimal outcomes, particularly evident in tasks such as image reconstruction. *Secondly*, the escalating computational costs of unrolling methods align with the general trend of increased complexity in modern neural networks. This escalation not only amplifies the required number of iterations due to the algorithm’s iterative nature but also contributes to their high computational demand.

To tackle the aforementioned issues, we introduce a novel second-order unrolling network for sparse-view CT reconstruction. *In particular*, to enable the learnable regularization term to apprehend long-range interactions within the image, we propose a non-local regularization block termed **Incept-Mixer**. Drawing inspiration from the multi-layer perceptron mixer [43] and the inception architecture [42], it is created to combine the best features from both sides: capturing long-range interactions from the attention-like mechanism of MLP-Mixer and extracting local invariant features from the inception block. This block facilitates a more precise image reconstruction. *Second*, to cut down on the computational costs associated with unrolling networks, we propose to decrease the required iterations for convergence by employing second-order optimization methods such as [19, 27]. We introduce a novel unrolling framework named **QN-Mixer**. Our approach is based on the quasi-Newton method that approximate the Hessian matrix using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update [9, 12, 53]. Furthermore, we reduce memory usage by working on a projected gradient (latent gradient), preserving performance while reducing the computational cost tied to Hessian matrix approximation. This adaptation enables the construction of a deep unrolling network, showcasing superlinear convergence. Our contributions are summarized as follows:

- We introduce a novel second-order unrolling network coined **QN-Mixer** where the Hessian matrix is approximated using a latent BFGS algorithm with a deep-net learned regularization term.
- We propose **Incept-Mixer**, a neural architecture acting as a non-local regularization term. Incept-Mixer integrates deep features from inception blocks with MLP-Mixer, enhancing multi-scale information usage and capturing long-range dependencies.
- We demonstrate the effectiveness of our proposed method when applied to the sparse-view CT reconstruction problem on an extensive set of experiments and datasets. We show that our method outperforms state-of-the-art methods in terms of quantitative metrics while requiring less iterations than first-order unrolling networks.

## 143 2. Related Works

144 In this section, we present prior work closely related to  
 145 our paper. We begin by discussing the general framework  
 146 for unrolling networks in Sec. 2.1, which is based on the  
 147 gradient descent algorithm. Subsequently, in Sec. 2.2 and  
 148 Sec. 2.3, we delve into state-of-the-art methods in post-  
 149 processing and unrolling networks, respectively.

### 150 2.1. Background

151 **Inverse Problem Formulation for CT.** Image reconstruction  
 152 problem in CT can be mathematically formalized as the  
 153 solution to a linear equation in the form of:

$$154 \quad \mathbf{y} = \mathbf{A}\mathbf{x} + \epsilon, \quad (1)$$

155 where  $\mathbf{x} \in \mathbb{R}^n$  is the (unknown) object to reconstruct with  
 156  $n = h \times w$ ,  $\mathbf{y} \in \mathbb{R}^m$  is the data (i.e. sinogram), where  
 157  $m = n_v \times n_d$ ,  $n_v$  and  $n_d$  denote the number of projec-  
 158 tion views and detectors, respectively.  $\mathbf{A} \in \mathbb{R}^{n \times m}$  is the  
 159 forward model (i.e. discrete Radon transform [37]), and  
 160  $\epsilon \in \mathbb{R}^m$  is the system noise. The goal of CT image re-  
 161 construction is to recover the (unknown) object,  $\mathbf{x}$ , from  
 162 the observed data  $\mathbf{y}$ . As the problem is ill-posed due to the  
 163 missing data, the linear system in Eq. (1) becomes under-  
 164 determined and may have infinite solutions. Hence, recon-  
 165 structed images suffer from artifacts, blurring, and noise.  
 166 To address this issue, iterative reconstruction algorithms are  
 167 utilized to minimize a regularized objective function with a  
 168  $L^2$  norm constraint:

$$169 \quad \hat{\mathbf{x}} = \arg \min_{\mathbf{x}} J(\mathbf{x}) = \frac{\lambda}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \mathcal{R}(\mathbf{x}), \quad (2)$$

170 where  $\mathcal{R}(\mathbf{x})$  is the regularization term, balanced with the  
 171 weight  $\lambda$ . Those ill-posed problems were initially addressed  
 172 using optimization techniques, such as the truncated sin-  
 173 gular value decomposition (SVD) algorithm [39], or iter-  
 174 ative approaches like the algebraic reconstruction technique  
 175 (ART) [4], simultaneous ART (SART) [2], conjugate gra-  
 176 dient for least squares (CGLS) [20], and total generalized  
 177 variation regularization (TGV) [40]. Additionally, tech-  
 178 niques such as total variation [44] and Tikhonov regulariza-  
 179 tion [8] can be employed to enhance reconstruction results.  
 180 **Deep Unrolling Networks.** By assuming that the regular-  
 181 ization term in Eq. (2) (i.e.  $\mathcal{R}$ ) is differentiable and convex,  
 182 a simple gradient descent scheme can be applied to solve  
 183 the optimization problem:

$$184 \quad \mathbf{x}_{t+1} = \mathbf{x}_t - \alpha \nabla_{\mathbf{x}} J(\mathbf{x}_t), \\ \text{where } \nabla_{\mathbf{x}} J(\mathbf{x}_t) = \lambda \mathbf{A}^\dagger (\mathbf{A}\mathbf{x}_t - \mathbf{y}) + \nabla_{\mathbf{x}} \mathcal{R}(\mathbf{x}_t). \quad (3)$$

185 Here,  $\alpha$  represents the step size (i.e. search step), and  $\mathbf{A}^\dagger$  is  
 186 the pseudo-inverse of  $\mathbf{A}$ .

187 Previous research [15, 49] has emphasized the limita-  
 188 tions of optimization algorithms, such as the manual selec-  
 189 tion of the regularization term and the optimization hyper-  
 190 parameters, which can negatively impact their performance,  
 191 limiting their clinical application. Recent advancements  
 192 in deep learning techniques have enabled automated pa-  
 193 rameter selection directly from the data, as demonstrated  
 194 in [6, 10, 21, 30, 35, 52]. By allowing the terms in Eq. (3)  
 195 to be dependent on the iteration, the gradient descent itera-  
 196 tion becomes:

$$197 \quad \mathbf{x}_{t+1} = \mathbf{x}_t - \lambda_t \mathbf{A}^\dagger (\mathbf{A}\mathbf{x}_t - \mathbf{y}) + \mathcal{G}(\mathbf{x}_t), \quad (4)$$

198 where  $\mathcal{G}$  is a learned mapping representing the gradient of  
 199 the regularization term. It is worth noting that the step size  
 200  $\alpha$  in Eq. (3) is omitted as it is redundant when considering  
 201 the learned components of the regularization term. Finally,  
 202 Eq. (4) is unrolled into a deep recurrent neural network in  
 203 order to learn the optimization parameters.

### 204 2.2. Post-processing Methods

205 Recent advances in sparse-view CT reconstruction lever-  
 206 age two main categories of deep learning methods: post-  
 207 processing and dual-domain approaches. Post-processing  
 208 methods, including RedCNN [5], FBPCConvNet [17], and  
 209 DDNet [54], treat sparse-view reconstruction as a denoising  
 210 step using FBP reconstructions as input. While effective in  
 211 addressing artifacts and reducing noise, they often struggle  
 212 with recovering global information from extremely sparse  
 213 data. To overcome this limitation, dual-domain methods in-  
 214 tegrate sinograms into neural networks for an interpolation  
 215 task, recovering missing data [14, 22]. Dual-domain meth-  
 216 ods, surpassing post-processing ones, combine information  
 217 from both domains. DuDoNet [26], an initial dual-domain  
 218 method, connects image and sinogram domains through  
 219 a Radon inversion layer. Recent Transformer-based dual-  
 220 domain methods, such as DuDoTrans [45] and DDPTrans-  
 221 former [24], aim to capture long-range dependencies in the  
 222 sinogram domain, demonstrating superior performance to  
 223 CNN-based methods.

### 224 2.3. Advancements in Deep Unrolling Networks

225 Unrolling networks constitute a line of work inspired by  
 226 popular optimization algorithms used to solve Eq. (2).  
 227 Leveraging the iterative nature of optimization algorithms,  
 228 as presented in Eq. (4), unrolling networks aim to directly  
 229 learn optimization parameters from data. These meth-  
 230 ods have found success in various inverse problems, in-  
 231 cluding sparse-view CT [6, 18, 48, 50, 52], limited-angle  
 232 CT [7, 10, 47], low-dose CT [1, 15], and compressed sens-  
 233 ing MRI [11, 41].

234 **First-order.** One pioneering unrolling network, Learned  
 235 Primal-Dual reconstruction [1], replaces traditional proximal  
 236 operators with CNNs. In contrast, LEARN [6] and

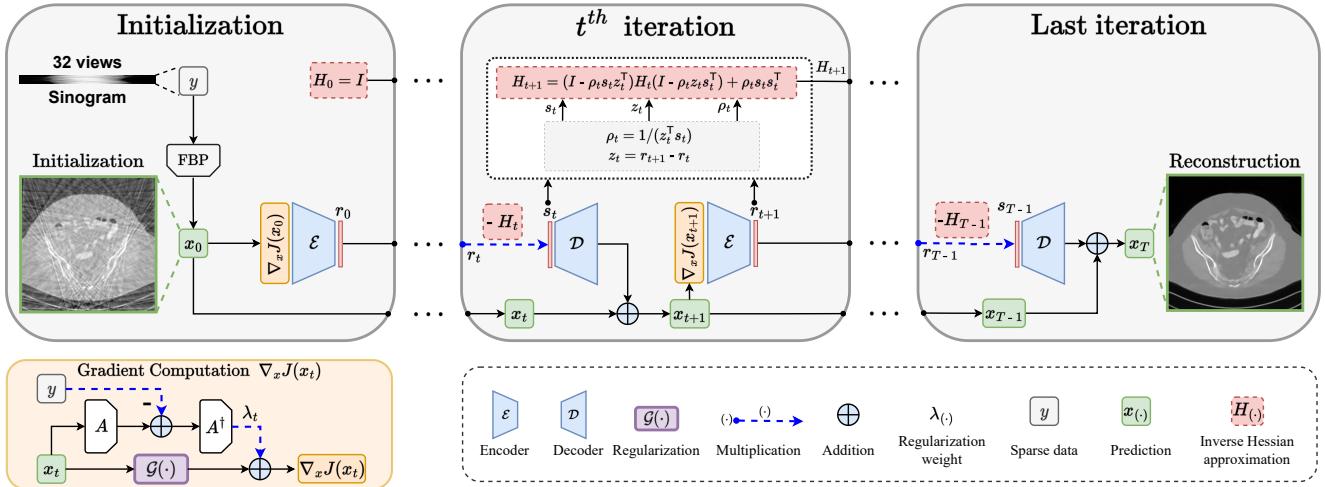


Figure 2. Overall structure of the proposed QN-Mixer for sparse-view CT reconstruction, unrolled from Algorithm 2. The method leverages the advantages of the quasi-Newton method for faster convergence while incorporating a latent BFGS update.

237 LEARN++ [52] directly unroll the optimization algorithm  
 238 from Eq. (4) into a deep recurrent neural network. More  
 239 recently, Transformers [3, 28] have been introduced into  
 240 unrolling networks, such as RegFormer [50] and HUMUS-  
 241 Net [11]. While achieving commendable performance,  
 242 these methods require more computational resources than  
 243 traditional CNN-based unrolling networks and incur a sig-  
 244 nificant memory footprint due to linear scaling with the  
 245 number of unrolling iterations.

246 **Second-order.** To address this, a new category of unrolling  
 247 optimization methods has emerged [13], leveraging second-  
 248 order techniques like the quasi-Newton method [9, 12, 19].  
 249 These methods converge faster, reducing computational de-  
 250 demands, but struggle with increased memory usage due to  
 251 Hessian matrix approximation and their application is lim-  
 252 ited to small-scale problems [27, 53]. In contrast our  
 253 method propose a memory-efficient approach by oper-  
 254 ating within the latent space of gradient information (i.e.  
 255  $\nabla_x J(x)$  in Eq. (3)).

### Algorithm 1: Quasi-Newton for sparse-view CT

**Data:**  $y$  (sparse sinogram)  
 Manual choice of the regularization term  $\mathcal{R}$ ;  
 $H_0 \leftarrow I^{n \times n}$ ;  
 $x_0 \leftarrow A^\dagger y$ ;  
**for**  $t \in \{0, \dots, T-1\}$  **do**  

$$\begin{aligned} s_t &\leftarrow -H_t \nabla_x J(x_t) \\ x_{t+1} &\leftarrow x_t + s_t \\ z_t &\leftarrow \nabla_x J(x_{t+1}) - \nabla_x J(x_t) \\ \rho_t &\leftarrow 1/(z_t^\top s_t) \\ H_{t+1} &\leftarrow (I - \rho_t s_t z_t^\top) H_t (I - \rho_t z_t s_t^\top) + \rho_t s_t s_t^\top \end{aligned}$$

## 3. Methodology

QN-Mixer is a novel second-order unrolling network inspired by the quasi-Newton (Sec. 3.1) method. It approximates the inverse Hessian matrix with a latent BFGS algorithm and includes a non-local regularization term, Incept-Mixer, designed to capture non-local relationships (Sec. 3.2). To cope with the significant computational burden associated with the full approximation of the inverse Hessian matrix, we use a latent BFGS algorithm (Sec. 3.3). An overview of the proposed method is depicted in Fig. 2, and the complete algorithm is presented in Sec. 3.4.

### 3.1. Quasi-Newton method

The quasi-Newton method can be applied to solve Eq. (2) and the iterative optimization solution is expressed as:

$$x_{t+1} = x_t - \alpha_t H_t \nabla_x J(x_t), \quad (5)$$

where  $H_t \in \mathbb{R}^{n \times n}$  represents the inverse Hessian matrix approximation at iteration  $t$ , and  $\alpha_t$  is the step size. The BFGS method updates the Hessian matrix approximation in each iteration. This matrix is crucial for understanding the curvature of the objective function around the current point, guiding us to take more efficient steps and avoiding unnecessary zigzagging. In the classical BFGS approach, the line search adheres to Wolfe conditions [9, 12]. A step size of  $\alpha_t = 1$  is attempted first, ensuring eventual acceptance for superlinear convergence [19]. In our approach, we adopt a fixed step size of  $\alpha_t = 1$ . The algorithm is illustrated in Algorithm 1.

### 3.2. Regularization term: Incept-Mixer

Recent research on unrolling networks has often focused on selecting the representation of the regularization term gradi-

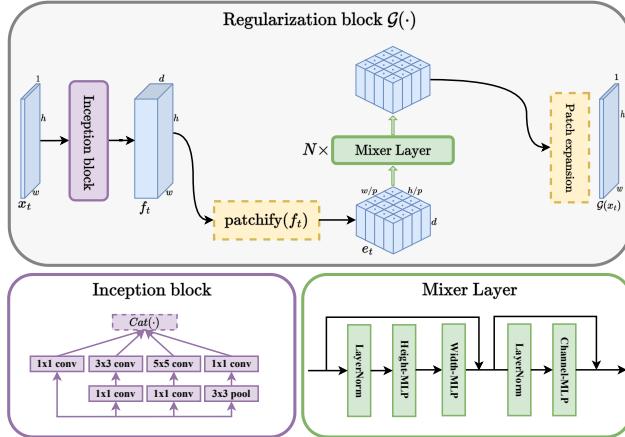


Figure 3. Architecture of our regularization block. It is referred to as “Incept-Mixer” and denoted as  $\mathcal{G}$  in Eq. (4)

ent (i.e.  $\mathcal{G}$  in Eq. (4)), ranging from conv-nets [6, 41, 52] to more recent attention-based nets [11, 50]. In alignment with this trend, we introduce a non-local regularization block named **Incept-Mixer** and depicted in, Fig. 3. This block is crafted by drawing inspiration from both the multi-layer perceptron mixer [43] and the inception architecture [42], leveraging the strengths of each: capturing long-range interactions through the attention-like mechanism of MLP-Mixer and extracting local invariant features from the inception block. This design choice is evident in the ablation study (see Tab. 6) where Incept-Mixer outperforms both alternatives.

Starting from an image  $x_t \in \mathbb{R}^{h \times w \times c}$  at iteration  $t$ , we pass it through an Inception block to create a feature map  $f_t \in \mathbb{R}^{h \times w \times d}$ , where  $d$  is the depth of features. Subsequently,  $f_t$  undergoes patchification using a CNN with a kernel size and stride of  $p$ , representing the patch size. This process yields patch embeddings,  $e_t = \text{patchify}(f_t) \in \mathbb{R}^{\frac{h}{p} \times \frac{w}{p} \times d}$ . These embeddings are then processed through a **Mixer Layer** with token and channel MLPs, layer normalization, and skip connections for inter-layer information flow, following [43]:

$$\begin{aligned} \text{MixerLayer}(e_t) &= \text{Mix}(\text{MLP}_{\text{channel}}, \\ &\quad \text{Mix}([\text{MLP}_{\text{height}}, \text{MLP}_{\text{width}}], e_t), \end{aligned} \quad (6)$$

where  $\text{Mix}(\text{Layer}, e_t) = \text{Layer}(\text{LN}(e_t)) + e_t$ , with LN as layer normalization.  $\text{MLP}_{\text{height}}$ ,  $\text{MLP}_{\text{width}}$  are applied to height and width features, respectively, and  $\text{MLP}_{\text{channel}}$  to rows and shared. Finally, after  $N$  such mixer layers, the regularized sample is transformed back to an image through a patch expansion step to obtain  $\mathcal{G}(x_t)$ . Consequently, the iterative optimization solution is as follows:

$$\begin{aligned} x_{t+1} &= x_t - H_t \nabla_x J(x_t), \\ \text{where } \nabla_x J(x_t) &= \lambda_t A^\dagger (Ax_t - y) + \mathcal{G}(x_t). \end{aligned} \quad (7)$$

Here,  $\mathcal{G}(x_t)$  denotes the Incept-Mixer model, representing the learned gradient of the regularization term.

### 3.3. Latent BFGS update

We propose a memory-efficient latent BFGS update. Drawing inspiration from LDMs [38], At step  $t$ , given the gradient value  $\nabla_x J(x_t) \in \mathbb{R}^{h \times w \times c}$ , the encoder  $\mathcal{E}$  encodes it into a latent representation  $r_t = \mathcal{E}(\nabla_x J(x_t)) \in \mathbb{R}^{l_h \cdot l_w}$ . Importantly, the encoder downsamples the gradient by a factor  $f_\mathcal{E} = \frac{h}{h_l} = \frac{w}{w_l}$ . Throughout the paper, we explore different downsampling factors (see Tab. 5)  $f_\mathcal{E} = 2^k$ , where  $k \in \mathbb{N}$  is the number of downsampling stacks. Encoding the gradient reduces the optimization variable size of BFGS (i.e.  $H_t \in \mathbb{R}^{(l_h \cdot l_w) \times (l_h \cdot l_w)}$ ), thereby decreasing the computational cost associated with high memory demand. The direction is then computed in the latent space  $s_t = -H_t r_t$ , and finally, the decoder  $\mathcal{D}$  reconstructs the update from the latent direction, giving  $\mathcal{D}(s_t) = \mathcal{D}(-H_t \mathcal{E}(\nabla_x J(x_t))) \in \mathbb{R}^{h \times w \times c}$ . It is noteworthy that  $\mathcal{E}$  and  $\mathcal{D}$  are shared across the algorithm iterations, as shown in Fig. 2.

### 3.4. Proposed algorithm of QN-Mixer

---

#### Algorithm 2: QN-Mixer (latent BFGS update)

---

**Data:**  $y$  (sparse sinogram)  
 $H_0 \leftarrow I^{(l_h \cdot l_w) \times (l_h \cdot l_w)}$ ;  
 $x_0 \leftarrow A^\dagger y$ ;  
 $r_0 \leftarrow \mathcal{E}(\nabla_x J(x_0))$ ;  
**for**  $t \in \{0, \dots, T-1\}$  **do**  

$$\begin{aligned} s_t &\leftarrow -H_t r_t \\ x_{t+1} &\leftarrow x_t + \mathcal{D}(s_t) \\ r_{t+1} &\leftarrow \mathcal{E}(\nabla_x J(x_{t+1})) \\ z_t &\leftarrow r_{t+1} - r_t \\ \rho_t &\leftarrow 1/(z_t^\top s_t) \\ H_{t+1} &\leftarrow (I - \rho_t s_t z_t^\top) H_t (I - \rho_t z_t s_t^\top) + \rho_t s_t s_t^\top \end{aligned}$$

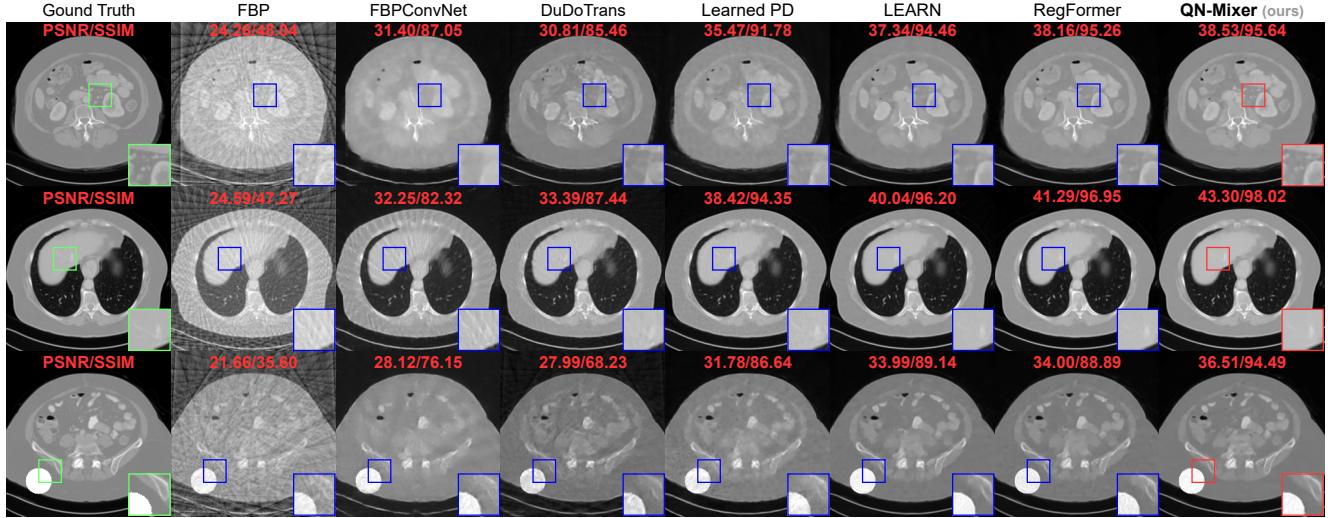
---

Our method, builds on the BFGS update [9, 12] rank-one approximation for the inverse Hessian. This approximation serves as a preconditioning matrix, guiding the descent direction. In contrast to [13], which directly learns the inverse Hessian approximation from data, our approach incorporates the mathematical equations of the BFGS algorithm for more accurate approximations. The full QN-Mixer algorithm is illustrated in Algorithm 2.

## 4. Experiments

In this section, we initially present our experimental settings, followed by a comparison of our approach with other state-of-the-art CT reconstruction methods. Finally, we delve into the contribution analysis of each component in our model.

Method	No noise ( $N_0 = 0$ )						Low noise ( $N_1 = 10^6$ )						High noise ( $N_2 = 5 \times 10^5$ )					
	$n_v = 32$		$n_v = 64$		$n_v = 128$		$n_v = 32$		$n_v = 64$		$n_v = 128$		$n_v = 32$		$n_v = 64$		$n_v = 128$	
	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑	PSNR ↑ SSIM ↑
FBP	22.65	40.49	27.29	57.94	33.04	79.50	22.09	32.73	26.51	49.56	31.69	71.09	19.05	15.56	22.71	25.74	26.52	40.87
FBPConvNet [17]	30.32	85.11	35.42	90.15	39.71	94.64	30.20	84.46	35.09	89.72	39.06	94.08	29.91	82.52	34.13	87.85	36.89	91.28
DuDoTrans [45]	30.48	84.70	35.37	91.87	40.62	96.41	30.34	83.72	35.36	91.42	39.75	95.49	30.09	81.83	34.09	88.67	37.08	93.44
Learned PD [1]	35.88	92.09	41.03	96.28	43.33	97.31	35.78	92.21	39.03	94.79	41.65	96.44	33.80	89.23	37.34	93.23	39.17	94.69
LEARN [6]	37.58	94.65	42.26	97.25	43.11	97.57	36.95	93.63	39.91	95.82	42.17	97.11	34.38	90.51	37.15	93.53	39.38	95.18
RegFormer [50]	38.71	95.42	43.56	97.76	47.95	98.98	37.21	94.73	41.65	96.92	44.38	98.02	35.93	92.78	38.53	94.84	40.52	96.19
QN-Mixer (ours)	<b>39.51</b>	<b>96.11</b>	<b>45.57</b>	<b>98.48</b>	<b>50.09</b>	<b>99.32</b>	<b>37.50</b>	<b>94.92</b>	<b>42.46</b>	<b>97.70</b>	<b>44.27</b>	<b>98.11</b>	<b>35.91</b>	<b>92.49</b>	<b>38.73</b>	<b>94.92</b>	<b>40.51</b>	<b>96.27</b>

Table 1. Quantitative evaluation on AAPM of state-of-the-art methods (PSNR in dB and SSIM in %). **Bold**: Best, under: second best.Figure 4. **Visual comparison on AAPM.** From top to bottom: the results under the following conditions: first ( $n_v = 32, N_1$ ), second ( $n_v = 64, N_1$ ), third ( $n_v = 32, N_0$ ). The last row presents out-of-distribution (OOD) results with a randomly overlaid circle on a test image. The display window is set to  $[-1000, 800]$  HU.

## 4.1. Experimental Setup

**Datasets.** We evaluate our method on two widely used datasets: the “2016 NIH-AAPM-Mayo Clinic Low-Dose CT Grand Challenge” dataset (AAPM) [32] and the DeepLesion dataset [51]. The AAPM dataset comprises 2378 full-dose CT images from 10 patients, while DeepLesion is the largest publicly accessible multi-lesion real-world CT dataset, including 4427 unique patients.

**Implementation details.** For AAPM, we select 1920 training images from 8 patients, 244 validation images from 1 patient, and 214 testing images from the last patient. For DeepLesion, we select a subset of 2000 training images and 300 testing images randomly from the official splits. All images are resized to  $256 \times 256$  pixels. To simulate the forward and backprojection operators, we use the Operator Discretization Library (ODL) [36] with a 2D fan-beam geometry (512 detector pixels, source-to-axis distance of 600 mm, axis-to-detector distance of 290 mm). Sparse-view CT images are generated with  $n_v \in \{32, 64, 128\}$  projection views, uniformly sampled from a full set of 512 views covering  $[0, 2\pi]$ . To mimic real-world CT images, we intro-

duce mixed noise to the sinograms, combining 5% Gaussian noise and Poisson noise with an intensity of  $1 \times 10^6$ .

**Training details.** For each set of  $n_v$  views, we train our model for 50 epochs using 4 Nvidia Tesla V100 (32GB RAM). We employ the AdamW optimizer [29] with a learning rate of  $1 \times 10^{-4}$ , weight decay  $1 \times 10^{-2}$ , and utilize the mean squared error loss with a batch size of 1. Additionally, we incorporate a learning rate decay factor of 0.1 after 40 epochs. Unrolling iterations for QN-Mixer are set to  $T = 14$ . Incept-Mixer uses a patch size of  $p = 4$ ,  $d = 96$  embedding dimension, and  $N = 2$  mixer layers. The inverse Hessian size is  $64^2 \times 64^2$  with  $k = 2$  downsampling blocks. Following [50],  $A^\dagger$  is implemented using the FBP algorithm for the pseudo-inverse of  $A$ .

**Evaluation metrics.** Consistent with standard evaluation practices [1, 45, 50], we use the structural similarity index measure (SSIM) with parameters set to: level = 5, a Gaussian kernel of size 11, and a standard deviation of 1.5 as our primary performance indicator. Additionally, we complement our evaluation with the peak signal-to-noise ratio (PSNR).

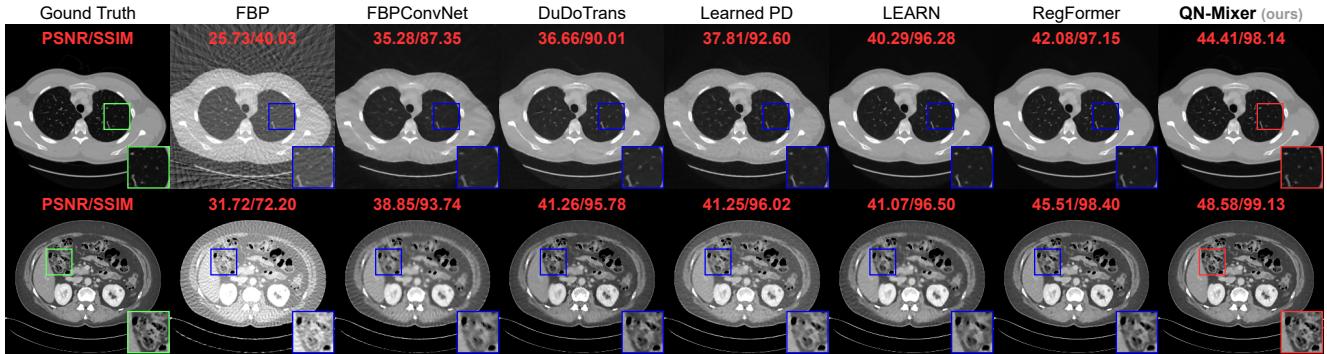


Figure 5. **Visual comparison on DeepLesion** of state-of-the-art methods. Rows display results under different conditions: ( $n_v = 64, N_1$ ) and ( $n_v = 128, N_1$ ). Display windows are set to  $[-1000, 800]$  HU for the first row and  $[-200, 300]$  HU for the second row.

Method	$n_v = 32$		$n_v = 64$		$n_v = 128$	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
FBP	21.55	31.65	26.07	47.17	31.49	69.63
FBPConvNet [17]	30.74	80.41	34.64	87.36	38.69	92.94
DuDoTrans [45]	32.11	79.86	36.02	88.14	40.47	93.81
Learned PD [1]	34.02	88.44	37.56	92.46	40.79	95.32
LEARN [6]	35.76	92.12	39.83	95.66	41.34	96.21
RegFormer [50]	37.38	93.89	41.70	96.78	46.10	98.39
QN-Mixer (ours)	<b>39.39</b>	<b>95.67</b>	<b>43.75</b>	<b>97.73</b>	<b>48.62</b>	<b>98.64</b>

Table 2. **Quantitative evaluation on DeepLesion** for state-of-the-art methods (PSNR in dB and SSIM in %). With Poisson noise level of  $N_1 = 10^6$ . **Bold**: Best, under: second best.

**State-of-the-art baselines.** We compare QN-Mixer to multiple state-of-the-art competitors: (1) *post-processing* based denoising methods, i.e., FBPConvNet [17], and DuDoTrans [45]; (2) *first-order unrolling* reconstruction networks, i.e., Learned Primal-Dual [1], LEARN [6], and RegFormer [50]. Note that we replace the pseudo-inverse operator used by LEARN with the FBP algorithm, as it has been demonstrated to be more effective according to [50]. To ensure a fair comparison, we utilize the code-base released by the authors when possible or meticulously implement the methods based on the details provided in their papers. All approaches undergo training and testing on the same datasets, as elaborated in implementation details.

## 4.2. Comparison with state-of-the-art methods

**Quantitative comparison.** We compared our model with state-of-the-art baselines on two public datasets. For AAPM, models were trained and tested across three projection views ( $n_v \in \{32, 64, 128\}$ ) and three noise levels, namely no noise  $N_0 = 0$ , low noise  $N_1 = 10^6$ , and high noise  $N_2 = 5 \times 10^5$  (see Tab. 1). For DeepLesion, models were trained and tested on the same three projection views and a noise level of  $N_1 = 10^6$  (see Tab. 2). Visual results are provided in Fig. 4 (AAPM) and Fig. 5 (DeepLesion). Impressively, our method achieves state-of-the-art results on DeepLesion across all projection views. It outperforms the second-best baseline, RegFormer, with an average im-

Method	$n_v = 32$		$n_v = 64$		$n_v = 128$	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
FBP	21.38	33.36	26.08	50.29	31.43	73.06
FBPConvNet [17]	28.05	75.96	32.50	82.90	35.45	88.14
DuDoTrans [45]	28.11	68.17	32.71	83.26	36.41	90.36
Learned PD [1]	31.96	87.10	36.40	92.57	37.63	93.17
LEARN [6]	34.48	<u>90.15</u>	36.89	<u>91.85</u>	<u>38.32</u>	<u>94.67</u>
RegFormer [50]	<u>34.49</u>	89.98	<u>36.95</u>	91.48	38.02	92.44
QN-Mixer (ours)	<b>36.84</b>	<b>94.84</b>	<b>42.11</b>	<b>97.78</b>	<b>45.69</b>	<b>98.82</b>

Table 3. **Quantitative evaluation on out-of-distribution (OOD) AAPM test dataset** of state-of-the-art methods (PSNR in dB and SSIM in %). **Bold**: Best, under: second best.

provements of +2.23 dB in PSNR and +1.02% in SSIM. On AAPM without noise, we achieve state-of-the-art results across all projection views and improve the second best by an average +1.65 dB and +0.58%. In the presence of low noise, QN-Mixer achieves state-of-the-art results performance in all cases except  $n_v = 128$  with -0.11 dB and shows an average improvements of +0.33 dB and +0.35% over RegFormer. With high noise, our method performs nearly on par in  $n_v = 32$  (-0.02 dB and -0.29%), achieves state-of-the-art in  $n_v = 64$  (+0.2 dB and +0.08%), and competes closely in  $n_v = 128$  (-0.01 dB and +0.08%). As noise increases, we attribute the decline in improvement to the compressed gradient information in the latent BFGS, influenced by sinogram changes, and the utilization of the FBP algorithm instead of the pseudo-inverse.

**Performance comparison on OOD textures.** In medical imaging, where training data predominantly consists of normal patient images, it is imperative to develop methods that generalize to scans with lesions or anomalies. To address this challenge, we conducted an experiment to evaluate model performance on out-of-distribution (OOD) textures. The frozen models were tested with CT images featuring a randomly positioned white circle with no noise added to the sinograms, as illustrated in the third row of Fig. 4. In Tab. 3, QN-Mixer attains state-of-the-art results across all  $n_v$  views. First-order unrolling networks such as LEARN and RegFormer exhibit significant PSNR degrada-

446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458

tion of  $-3.1$  dB and  $-4.22$  dB, respectively, for  $n_v = 32$ , while our method demonstrates a milder degradation of  $-2.67$  dB.

**Visual comparison.** As it can be seen on Fig. 4 and Fig. 5, FBPCovNet and DuDoTrans exhibit noticeable blurry images with severe artifacts when  $n_v = 32$ . While Learned PD and LEARN show satisfactory performance, they struggle with intricate details, like in the liver and spine. In contrast, RegFormer produces high-quality images but faces challenges in generalizing to OOD data. QN-Mixer excels in producing high-quality images with fine details, even under challenging conditions such as  $n_v = 32$  views and OOD data.

Method	#Iters	Epoch time (s)	#Params (M)	Memory (GB)
FBPCovNet [17]	-	68	31.1	1.30
DuDoTrans [45]	-	92	15.0	1.38
Learned PD [1]	10	82	0.25	0.81
LEARN [6]	30	780	4.50	1.85
RegFormer [50]	18	700	5.00	10.19
QN-Mixer (ours)	14	594	8.50	7.83

Table 4. **Comparison of computational efficiency.** Training epoch time is reported in seconds, #Params in M and memory costs for state-of-the-art methods on AAPM with  $n_v = 32$  views.

459  
460  
461  
462  
463  
464  
465  
466  
467

**Efficiency comparison.** The results in Tab. 4 show that QN-Mixer is more computationally efficient than RegFormer, with a  $1.3\times$  reduction in memory usage. Furthermore, our training time demonstrates a significant enhancement, realizing a speed improvement of 106 seconds per epoch compared to first-order unrolling methods like LEARN and RegFormer. Additionally, our method requires only 14 iterations, in contrast to the 30 and 18 iterations needed by LEARN and RegFormer, respectively.

Hessian size	PSNR $\uparrow$	SSIM $\uparrow$
$8^2 \times 8^2$	35.69	93.71
$16^2 \times 16^2$	38.11	95.31
$32^2 \times 32^2$	39.37	96.01
$64^2 \times 64^2$	<b>39.51</b>	<b>96.11</b>

Table 5. **Ablation on the inverse Hessian approximation size.**

### 4.3. Ablation Study

468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478

In this section, we leverage the AAPM dataset with  $n_v = 32$  views by default, and no noise is introduced to the sinogram. **Inverse Hessian approximation size.** The results in Tab. 5 emphasize the significant impact of the inverse Hessian approximation size on our performance. When too small, a notable degradation is observed (e.g.,  $8^2 \times 8^2$ ), while larger sizes result in performance improvements as the approximation approaches the full inverse Hessian. However, exceeding  $64^2 \times 64^2$  was unfeasible in our experiments due to increasing memory costs and memory constraints.

Method	PSNR $\uparrow$	SSIM $\uparrow$
QN+Inception	31.65	85.28
QN+MLP-Mixer	36.89	93.87
QN-Mixer+ $A^\dagger$	38.94	95.85
Incept-Mixer + first-order	37.45	94.25
QN-Mixer (ours)	<b>39.51</b>	<b>96.11</b>

Table 6. **Ablation for the regularization term.**  $A^\dagger$  denotes the pseudo-inverse.

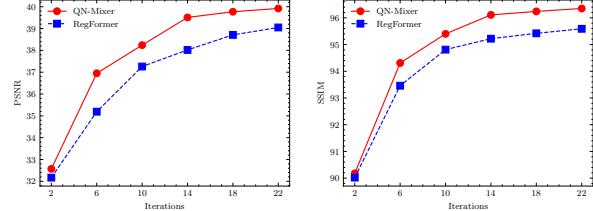


Figure 6. **Ablation on the number of unrolling iterations.** We compare our QN-Mixer against RegFormer. Left: PSNR (dB); Right: SSIM (%)

**Number of unrolling iterations.** In Fig. 6, we visually depict the influence of the number of unrolling iterations on the performance of QN-Mixer and RegFormer. Notably, the performance of both methods shows improvement with an increase in the number of iterations. When subjected to an equal number of iterations, our method consistently surpasses RegFormer in performance. Remarkably, we achieve comparable results to RegFormer even with only 10 iterations, demonstrating the efficiency of our approach.

**Regularization term.** We evaluate the impact of the regularization term in our framework. Our Incept-Mixer is compared against various learned alternatives, including the Inception block [42] and MLP-Mixer block [43]. Additionally, we explore the use of the pseudo-inverse  $A^\dagger$  instead of the FBP. The results presented in Tab. 6 show that employing the pseudo-inverse results in a less pronounced degradation ( $-0.57$  dB and  $-0.28\%$ ), enhancing the interpretability of QN-Mixer. Finally, we test our Incept-Mixer in the first-order framework, highlighting the significance of the second-order latent BFGS approximation with a significant improvement ( $+2.06$  dB and  $+1.86\%$ ).

## 5. Conclusion

In this paper, we investigate the application of deep second-order unrolling networks for tackling imaging inverse problems. To this end, we introduce QN-Mixer, a quasi-Newton inspired algorithm where a latent BFGS method approximates the inverse Hessian, and our Incept-Mixer serves as the non-local learnable regularization term. Extensive experiments confirm the successful sparse-view CT reconstruction by our model, showcasing superior performance with fewer iterations than state-of-the-art methods. In summary, this research offers a fresh perspective that can be applied to any iterative reconstruction algorithm. A limitation of our work is the memory requirements associated with quasi-Newton algorithm. We introduced a memory-efficient alternative by projecting the gradient to a lower dimension, successfully addressing the CT reconstruction problem. However, its applicability to other inverse problems may be limited. In future work, we aim to extend our approach to handle larger Hessian sizes, broadening its application to a range of problems.

520 **References**

- [1] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE TMI*, 37:1322–1332, 2018. 2, 3, 6, 7, 8, 521 4
- [2] Andersen Arie and Kak Avinash. Simultaneous algebraic 522 reconstruction technique (SART): a superior implementation 523 of the art algorithm. *Ultrasonic imaging*, 6:81–94, 1984. 3
- [3] Vaswani Ashish, Shazeer Noam, Parmar Niki, Uszkoreit 524 Jakob, Jones Llion, Gomez Aidan N, Kaiser Lukasz, and 525 Polosukhin Illia. Attention is all you need. In *NeurIPS*, 2017. 526 4
- [4] Kak Avinash and Slaney Malcolm. *Principles of Computerized 527 Tomographic Imaging*. Society for Industrial and Applied 528 Mathematics, 2001. 3
- [5] Hu Chen, Yi Zhang, Mannudeep K. Kalra, Feng Lin, Yang 529 Chen, Peixi Liao, Jiliu Zhou, and Ge Wang. Low-dose CT 530 with a residual encoder-decoder convolutional neural network. 531 *IEEE TMI*, 36:2524–2535, 2017. 2, 3
- [6] Hu Chen, Yi Zhang, Yunjin Chen, Junfeng Zhang, Weihua 532 Zhang, Huaiqiang Sun, Yang Lv, Peixi Liao, Jiliu Zhou, and 533 Ge Wang. LEARN: Learned experts’ assessment-based 534 reconstruction network for sparse-data CT. *IEEE TMI*, 37: 535 1333–1347, 2018. 2, 3, 5, 6, 7, 8, 4
- [7] Weilin Cheng, Yu Wang, Hongwei Li, and Yuping Duan. 536 Learned full-sampling reconstruction from incomplete data. 537 *IEEE TCI*, pages 945–957, 2020. 2, 3
- [8] Peng Chengbin, Rodi William L., and M. Nafi Toksoz. A 538 *Tikhonov Regularization Method for Image Reconstruction*, 539 pages 153–164. Springer US, 1993. 3
- [9] William C. Davidon. Variable metric method for minimization. 540 *SIAM Journal on Optimization*, 1:1–17, 1991. 2, 4, 5, 541 1
- [10] Hu Dianlin, Zhang Yikun, Liu Jin, Luo Shouhua, and Chen 542 Yang. DIOR: Deep iterative optimization-based residual- 543 learning for limited-angle CT reconstruction. *IEEE TMI*, 544 pages 1778–1790, 2022. 2, 3
- [11] Zalan Fabian, Berk Tinaz, and Mahdi Soltanolkotabi. 545 HUMUS-Net: Hybrid unrolled multi-scale network architecture 546 for accelerated MRI reconstruction. In *NeurIPS*, 2022. 2, 3, 4, 5
- [12] Roger Fletcher. *Practical Methods of Optimization*. John 547 Wiley & Sons, New York, NY, USA, 1987. 2, 4, 5, 1
- [13] Erik Gartner, Luke Metz, Mykhaylo Andriluka, C. Daniel 548 Freeman, and Cristian Sminchisescu. Transformer-based 549 Learned Optimization. In *CVPR*, pages 11970–11979, 2023. 550 2, 4, 5
- [14] Muhammad Usman Ghani and W. Clem Karl. Deep 551 learning-based sinogram completion for low-dose CT. In 552 *2018 IEEE 13th Image, Video, and Multidimensional Signal 553 Processing Workshop*, pages 1–5, 2018. 2, 3
- [15] Gupta Harshit, Jin Kyong Hwan, Nguyen Ha Q., McCann Michael T., and Unser Michael. CNN-based projected 554 gradient descent for consistent CT image reconstruction. 555 *IEEE TMI*, 37:1440–1453, 2018. 2, 3
- [16] Dianlin Hu, Jin Liu, Tianling Lv, Qianlong Zhao, Yikun 556 Zhang, Guotao Quan, Juan Feng, Yang Chen, and Limin 557
- Luo. Hybrid-domain neural network processing for sparse- 558 view CT reconstruction. *IEEE TRPMS*, 5:88–98, 2020. 2
- [17] Kyong Hwan Jin, Michael T. McCann, Emmanuel Froustey, 559 and Michael Unser. Deep convolutional neural network for 560 inverse problems in imaging. *IEEE TIP*, 26:4509–4522, 561 2017. 2, 3, 6, 7, 8, 4
- [18] Xiang Jinxi, Dong Yonggui, and Yang Yunjie. FISTA-Net: 562 Learning a fast iterative shrinkage thresholding network for 563 inverse problems in imaging. *IEEE TMI*, 40:1329–1339, 564 2021. 2, 3
- [19] Nocedal Jorge and Wright Stephen J. Quasi-Newton meth- 565 ods. *Numerical optimization*, 75:135–163, 2006. 2, 4, 1
- [20] Satoshi Kawata and Orhan Nalcioglu. Constrained iterative 566 reconstruction by the conjugate gradient method. *IEEE TMI*, 567 4:65–71, 1985. 3
- [21] Erich Kobler, Alexander Effland, Karl Kunisch, and Thomas 568 Pock. Total deep variation for linear inverse problems. In 569 *CVPR*, pages 7546–7555, 2020. 3
- [22] Hoyeon Lee, Jongha Lee, Hyeongseok Kim, Byungchul 570 Cho, and Seungryong Cho. Deep-neural-network-based 571 sinogram synthesis for sparse-view CT image reconstruc- 572 tion. *IEEE TRPMS*, 3:109–119, 2018. 2, 3
- [23] Minjae Lee, Hyemi Kim, and Hee-Joung Kim. Sparse-view 573 CT reconstruction based on multi-level wavelet convolution 574 neural network. *Physica Medica*, 80:352–362, 2020. 2
- [24] Runrui Li, Qing Li, Hexi Wang, Saize Li, Juanjuan Zhao, 575 Yan Qiang, and Long Wang. DDPTransformer: Dual- 576 domain with parallel transformer network for sparse view CT 577 image reconstruction. *IEEE TCI*, pages 1–15, 2022. 2, 3
- [25] Zilong Li, Chenglong Ma, Jie Chen, Junping Zhang, and 578 Hongming Shan. Learning to distill global representation 579 for sparse-view ct. In *ICCV*, pages 21196–21207, 2023. 2
- [26] Wei-An Lin, Cheng Liao, Haofu Peng, Xiaohang Sun, Jing- 580 dan Zhang, Jiebo Luo, Rama Chellappa, and Zhou S. Kevin. 581 DuDoNet: Dual domain network for CT metal artifact reduc- 582 tion. In *CVPR*, pages 10512–10521, 2019. 2, 3
- [27] Chengchang Liu and Luo Luo. Quasi-newton methods for 583 saddle point problems. In *NeurIPS*, 2022. 2, 4
- [28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng 584 Zhang, and Stephen Lin Baining Guo. Swin transformer: 585 Hierarchical vision transformer using shifted windows. In 586 *ICCV*, pages 9992–10002, 2021. 4
- [29] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay 587 Regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6
- [30] Sebastian Lunz, Ozan Öktem, and Carola-Bibiane 588 Schönlieb. Adversarial regularizers in inverse problems. In 589 *NeurIPS*, 2018. 3
- [31] Cormack Allan Macleod. Representation of a function by its 590 line integrals, with some radiological applications. *Journal 591 of Applied Physics*, 34:2722–2727, 1963. 2
- [32] C. McCollough. TU-FG-207A-04: Overview of the low dose 592 CT grand challenge. *Medical Physics*, 43:3759–3760, 2016. 593 6, 7
- [33] Luke Metz, C. Daniel Freeman, James Harrison, Niru Mah- 594 eswaranathan, and Jascha Sohl-Dickstein. Practical tradeoffs 595 between memory, compute, and performance in learned op- 596 timizers. *arXiv preprint arXiv:2203.11860*, 2022. 2

- 633 [34] Donald L. Miller and David Schauer. The alara principle in  
634 medical imaging. *Philosophy*, 44:595–600, 1983. 2
- 635 [35] Subhadip Mukherjee, Marcello Carioni, Ozan Öktem, and  
636 Carola-Bibiane Schönlieb. End-to-end reconstruction meets  
637 data-driven regularization for inverse problems. In *NeurIPS*,  
638 2021. 3
- 639 [36] Ozan Öktem, Jonas Adler, Holger Kohr, and The ODL Team.  
640 Operator discretization library (ODL), 2014. 6
- 641 [37] Johann Radon. über die bestimmung von funktionen  
642 durch ihre integralwerte längs gewisser mannigfaltigkeiten.  
643 *Berichte über die Verhandlungen der Königlich-Sächsischen*  
644 *Akademie der Wissenschaften zu Leipzig*, 69:262–277, 1917.  
645 3
- 646 [38] Rombach Robin, Blattmann Andreas, Lorenz Dominik,  
647 Esser Patrick, and Ommer Björn. High-resolution image  
648 synthesis with latent diffusion models. In *CVPR*, pages  
649 10684–10695, 2022. 5
- 650 [39] Liu Rui, He Lu, Luo Yan, and Yu Hengyong. Singular value  
651 decomposition-based 2D image reconstruction for computed  
652 tomography. *Journal of X-ray science and technology*, 25:  
653 113–134, 2017. 3
- 654 [40] Niu Shaohua, Gao Yan, Bian Zhaoying, Huang Jing, Chen  
655 Wufan, Yu Hengyong, Liang Zhengrong, and Ma Jianhua.  
656 Sparse-view x-ray CT reconstruction via total generalized  
657 variation regularization. *PMB*, 59:2997–3017, 2014. 3
- 658 [41] Anuroop Sriram, Jure Zbontar, Tullie Murrell, Aaron De-  
659 fazio, C. Lawrence Zitnick, Nafissa Yakubova, Florian  
660 Knoll, and Patricia Johnson. End-to-End variational net-  
661 works for accelerated MRI reconstruction. In *MICCAI*,  
662 pages 64–73, 2020. 2, 3, 5
- 663 [42] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet,  
664 Scott Reed, Dragomir Anguelov, Dumitru Erhan, and Vin-  
665 cent Vanhoucke Andrew Rabinovich. Going deeper with  
666 convolutions. In *CVPR*, pages 1–9, 2015. 2, 5, 8
- 667 [43] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas  
668 Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung,  
669 Andreas Peter Steiner, Daniel Keysers, Jakob Uszkoreit,  
670 Mario Lucic, and Alexey Dosovitskiy. MLP-mixer: An all-  
671 MLP architecture for vision. In *NeurIPS*, 2021. 2, 5, 8
- 672 [44] Gomi Tsutomu and Koibuchi Yukio. Use of a Total Varia-  
673 tion minimization iterative reconstruction algorithm to eval-  
674 uate reduced projections during digital breast tomosynthesis.  
675 *BioMed Research International*, 2018:1–14, 2018. 3
- 676 [45] Ce Wang, Kun Shang, Haimiao Zhang, Qian Li, and S. Kevin  
677 Zhou. DuDoTrans: Dual-domain transformer for sparse-  
678 view CT reconstruction. In *Machine Learning for Medical*  
679 *Image Reconstruction*, pages 84–94. Springer International  
680 Publishing, 2022. 2, 3, 6, 7, 8, 4
- 681 [46] Ge Wang, Hengyong Yu, and Bruno De Man. An outlook on  
682 X-ray CT research and development. *Medical Physics*, 35:  
683 1051–1064, 2008. 2
- 684 [47] Jiaxi Wang, Li Zeng, Chengxiang Wang, and Yumeng Guo.  
685 ADMM-based deep reconstruction for limited-angle CT.  
686 *PMB*, 64, 2019. 2, 3
- 687 [48] Wu Weiwen, Hu Dianlin, Niu Chuang, Yu Hengyong, Vard-  
688 hanabhuti Varut, and Wang Ge. DRONE: Dual-domain  
689 residual-based optimization network for sparse-view CT re-  
690 construction. *IEEE TMI*, 40:3002–3014, 2021. 2, 3
- 691 [49] Dufan Wu, Kyungsang Kim, Georges El Fakhri, and  
692 Quanzheng Li. Iterative low-dose ct reconstruction with pri-  
693 ors trained by artificial neural network. *IEEE TMI*, 36:2479–  
694 2486, 2017. 3
- 695 [50] Wenjun Xia, Ziyuan Yang, Qizheng Zhou, Zexin Lu,  
696 Zhongxian Wang, and Yi Zhang. Transformer-based itera-  
697 tive reconstruction model for sparse-view CT reconstruction.  
698 In *MICCAI*, 2022. 2, 3, 4, 5, 6, 7, 8
- 699 [51] Ke Yan, Xiaosong Wang, Le Lu, and Ronald M. Summers.  
700 DeepLesion: Automated mining of large-scale lesion an-  
701 notations and universal lesion detection with deep learning.  
702 *Journal of Medical Imaging*, 5:036501, 2018. 6
- 703 [52] Zhang Yi, Chen Hu, Xia Wenjun, Chen Yang, Liu Baodong,  
704 Liu Yan, Sun Huaiqiang, and Zhou Jiliu. LEARN++: Re-  
705 current dual-domain reconstruction network for compressed  
706 sensing CT. *IEEE TRPMS*, 7:132–142, 2023. 2, 3, 4, 5
- 707 [53] Tsai Yu-Jung, Bousse Alexandre, Ehrhardt Matthias J.,  
708 Stearns Charles W., Ahn Sangtae, Hutton Brian F., Arridge  
709 Simon, and Thielemans Kris. Fast quasi-newton algorithms  
710 for penalized reconstruction in emission tomography and  
711 further improvements via preconditioning. *IEEE TMI*, 37:  
712 1000–1010, 2018. 2, 4
- 713 [54] Zhicheng Zhang, Xiaokun Liang, Xu Dong, Yaoqin Xie,  
714 and Guohua Cao. A sparse-view CT reconstruction method  
715 based on combination of DenseNet and deconvolution. *IEEE*  
716 *TMI*, 37:1407–1417, 2018. 2, 3