LRR-CED

Kandarpa, V. S. S.; Perelli, Alessandro; Bousse, Alexandre; Visvikis, Dimitris

Published in:
Physics in Medicine and Biology

DOI:
10.1088/1361-6560/ac7bce

Publication date:
2022

Licence:
CC BY-NC-ND

Document Version
Peer reviewed version

Physics in Medicine & Biology

IPEM Institute of Physics and Engineering in Medicine

ACCEPTED MANUSCRIPT

LRR-CED: Low-resolution reconstruction-aware convolutional encoder-decoder network for direct sparse-view CT image reconstruction

To cite this article before publication: V.S.S. Kandarpa *et al* 2022 *Phys. Med. Biol.* in press https://doi.org/10.1088/1361-6560/ac7bce

LRR-CED: Low-Resolution Reconstruction-Aware Convolutional Encoder-Decoder Network for Direct Sparse-View CT Image Reconstruction

V.S.S. Kandarpa[1], Alessandro Perelli[1,2], Alexandre Bousse[1] and Dimitris Visvikis[1]

[1] LaTIM, INSERM, UMR 1101, *Université de Bretagne Occidentale*, 29238 Brest, France

[2] School of Science and Engineering, University of Dundee, Scotland DD1 4HN, UK

E-mail: venkatasaisundar.kandarpa@etudiant.univ-brest.fr

June 2022

Abstract.

Objective: Sparse-view computed tomography (CT) reconstruction has been at the forefront of research in medical imaging. Reducing the total X-ray radiation dose to the patient while preserving the reconstruction accuracy is a big challenge. The sparse-view approach is based on reducing the number of rotation angles, which leads to poor quality reconstructed images as it introduces several artifacts. These artifacts are more clearly visible in traditional reconstruction methods like the filtered-backprojection (FBP) algorithm.

Approach: Over the years, several model-based iterative and more recently deep learning-based methods have been proposed to improve sparse-view CT reconstruction. Many deep learning-based methods improve FBP-reconstructed images as a post-processing step. In this work, we propose a direct deep learning-based reconstruction that exploits the information from low-dimensional scout images, to learn the projection-to-image mapping. This is done by concatenating FBP scout images at multiple resolutions in the decoder part of a convolutional encoder-decoder (CED).

Main Results: This approach is investigated on two different networks, based on Dense Blocks and U-Net to show that a direct mapping can be learned from a sinogram to an image. The results are compared to two post-processing deep learning methods (FBP-ConvNet and DD-Net) and an iterative method that uses a total variation (TV) regularization.

Significance: This work presents a novel method that uses information from both sinogram and low-resolution scout images for sparse-view CT image reconstruction. We also generalize this idea by demonstrating results with two different neural networks. This work is in the direction of exploring deep learning across the various stages of the image reconstruction pipeline involving data correction, domain transfer and image improvement.

1. Introduction

The impact of deep learning has been immense over the last few years in the field of medical imaging (Litjens et al. 2017, Greenspan et al. 2016). Medical image reconstruction has also benefited hugely from the various advances in neural network architectures (Wang et al. 2020, Yedder et al. 2021, Reader et al. 2020). In the specific case of CT image reconstruction, there has been active interest in sparse-view and low-dose reconstruction scenarios. In both cases, severe artifacts are introduced in reconstructed images either due to incomplete projections or low counts. Many established model-based iterative methods account for the low-dose and sparse-view settings to remove artifacts and noise from the reconstruction (Nuyts et al. 1998, Elbakri & Fessler 2002, Liu et al. 2013). However, these methods require the knowledge of the noise and artifacts statistics and generally have longer reconstruction times (Kim et al. 2014). Deep learning-based methods on the other hand are claimed to achieve reconstructed images with quality on par with iterative techniques and in a much shorter time frame (Leuschner et al. 2021). In this work, we predominantly focus on sparse-view CT image reconstruction.

A straight forward way to introduce deep learning architectures in the image reconstruction pipeline is to improve the images estimated by traditional reconstruction methods. While it is possible to train a convolutional neural network (CNN) to regress directly from the measurement (raw data) domain to the image domain, the use of CNN entirely in the image domain makes it fast and relatively easy to implement. Typically within this approach the training data consists of sparse-view reconstructions with FBP and the corresponding full-view reconstructed images. The authors in Jin et al. (2017) use U-Net with a residual connection for denoising and artefact removal in the sparse-view estimate, while the work in Zhang et al. (2018) called DD-Net, uses DenseNet with deconvolution for the same purpose. It is interesting to note that the networks have an encoder-decoder structure, wherein the encoder finds a compact representation of the input domain and the decoder learns to map this representation to the target domain. The dimensions of the input are reduced through the encoder as we go deeper into the layers. On the other hand, each of the decoder layers samples up these feature maps to eventually arrive at the output dimensions. Deep learning algorithms were also used for data correction in the projection domain. For example Lee et al. (2018) uses U-Net to map sparse-view sinograms to full-view sinograms and then reconstruct the images using FBP.

The hybrid methodology of unrolled iterative networks combines model-based and neural network approaches exploring the benefits of both methods. One example in this regard is Gupta et al. (2018) where a U-Net is used to encode the prior, i.e., to project the current estimate to the prior image set while gradient descent enforces measurement consistency. Neural networks can be also used to replace traditional operators in optimization strategies as shown by Adler & Öktem (2018). The reconstruction using these hybrid methods can be computationally expensive since it requires running an optimization procedure at test time. Another recent work specific to sparse-view CT

was proposed by Wu et al. (2021). Their methodology named DRONE, consists of three modules namely embedding, refinement and awareness. The first module estimates the missing views in the sparse-view sinogram through neural network. An image is then estimated with FBP, which goes through the refinement module for artifact removal and image enhancement. Finally, the awareness module using compresses sensing techniques, establishes data consistency with the measurement data.

An alternative approach is using deep learning-based methods to directly map from projection to image space. The challenge in this approach is the management of data and the number of parameters required for learning the mapping. In Li et al. (2019) the authors proposed an architecture termed iCT-Net consisting of 12 layers that are a combination of convolutions and modified fully-connected layers. The 12 layers are separated into segments and are trained separately before being combined for end-to-end training. To reduce the number of parameters in learning the mapping for full resolution CT reconstruction, Fu & De Man (2019) proposed a breakdown of the problem into smaller fragments that can be mapped onto a hierarchical network architecture. The approach proposed in Ye et al. (2018) converts the sinogram data into a stack of back projections for each angle, which are then fed into a CNN. The spatial in-variance of the CNN is exploited to learn the mapping from these single view stacked back projections onto reconstructed images. Currently, we observe that adversarial networks are increasingly used in scenarios with high-resolution images. In Thaler et al. (2018) a Wasserstein generative adversarial network is proposed for sparse-view CT image reconstruction (Arjovsky et al. 2017). The authors used a combination of L_1 loss and adversarial loss to train their network. The generator in their work is a U-Net and the discriminator a typical classification CNN. It is to be noted that the authors performed their experiments on down-sampled images of resolution 128×128. In our earlier work referred to as DUG-RECON (Kandarpa et al. 2020), we used a three-stage network to divide the image reconstruction problem into denoising, domain mapping and resolution improvement. We used a residual UNet for denoising the sinograms, then a double-UNet architecture to map the sinogram to image, and finally a super ResNet to improve image estimate. The approach was tested with both positron emission tomography (PET) and CT data.

1.1. Main Contribution

The main drawbacks of current deep learning-based direct image reconstruction algorithms are the tedious training process necessary to train large networks with large number of trainable parameters and the requirement of high memory in case of high-resolution CT images. In this work we propose a new method for direct deep learning based sparse-view CT image reconstruction with fully convolutional networks. We use two networks, namely Fully Convolutional Densenets and U-Net (Jégou et al. 2017, Ronneberger et al. 2015). An important characteristic of both these architectures is the presence of concatenation from the encoding layers to the decoding

layers that ensures the usage of features from the input for the reconstruction. Specifically, for application in sparse-CT image reconstruction, the network would have sparse-view sinograms as input and reconstructed images as output. The original application in the medical imaging field of both these architectures was in image segmentation, where the image-to-image mapping operates in the same image domain. Medical image reconstruction on the other hand involves mapping between two different domains (sinogram to image). In order to help the network to learn the mapping from sinogram to image, we propose the use of low-resolution FBP scout images and concatenate them with the feature maps of the decoder.

These custom concatenations enable architectures that were previously used for denoising/artifact removal to learn a mapping from sparse sinograms to full-resolution CT images for the same number of training examples. One characteristic feature of reconstructions generated by deep learning-based methods is the blurriness of the outputs. To counteract this we used perceptual loss involving features extracted from two different levels of VGG16 network (Block 1 and Block 3). Since the exclusive use of perceptual loss results in unrealistic artifacts we couple it with a L_1 loss. The proposed approach called Low-Resolution Reconstruction-Aware Convolutional Encoder-Decoder (LRR-CED), consists of a CED network with two blocks in both the encoder and the decoder that takes in as input a reshaped sparse sinogram which has the same dimensions as the output image. A concatenation of two resolutions $h_1 \times w_1$ and $h_2 \times w_2$ is incorporated in the decoder.

The main contributions of our work are summarized as follows:

- A new approach for sparse-view CT image reconstruction using fully-convolutional networks
- Use of lower resolution FBP scout images which enable the networks that are predominantly used for denoising to learn the more complex mapping from sinogram to image domain with the same number of training examples.
- An ablation study to highlight the impact of the combination of sinograms and the proposed concatenations.
- Two neural networks are implemented to test this approach using different levels of sparsity in the sinograms.

1.2. Paper Organization

This paper is organized as follows: Section 2 explains the proposed reconstruction approach; Section 3 presents the experimental results, hyperparameter selection and ablation study; Section 4 and Section 5 are the discussion and conclusion sections respectively.

2. Methods

2.1. CT Physical Model

Let an image be represented by $\boldsymbol{x} \in \mathbb{R}^m$ and the scanner measurement by $\boldsymbol{b} \in \mathbb{R}^n$ where m is the number of voxels and n is the number of measurements. In two-dimensional (2-D) CT imaging n depends on the number of detectors n_{d} and the number of angles n_{a}. The task of medical image reconstruction corresponds to finding a mapping from $\boldsymbol{b}$ to $\boldsymbol{x}$. The measurement $\boldsymbol{b}$ is a random vector modeling the number of detection (photon counting) at each of the n detector bins, and follows a Poisson distribution with independent entries, i.e.,

$$\boldsymbol{b} \sim \mathrm{Poisson}(\bar{\boldsymbol{b}}(\boldsymbol{x})) \tag{1}$$

where, $\boldsymbol{b} = [b_1(\boldsymbol{x}), \ldots, b_n(\boldsymbol{x})]^\top \in \mathbb{R}^n$ and $\bar{\boldsymbol{b}}(\boldsymbol{x}) = [\bar{b}_1(\boldsymbol{x}), \ldots, \bar{b}_n(\boldsymbol{x})]^\top \in \mathbb{R}^n$ is the expected number of counts (noiseless), which is a function of the image $\boldsymbol{x}$.

The image $\boldsymbol{x} \in \mathbb{R}^m$ is a vectorized input image (also referred to as attenuation) representing the measure of X-rays absorbed or scattered as they pass through the patient. In a monochromatic setting, the expected number of counts $\bar{\boldsymbol{b}}(\boldsymbol{x})$ is given by the Beer-Lambert law, i.e.,

$$\bar{b}_i(\boldsymbol{x}) = I \cdot \exp(-[\boldsymbol{P}\boldsymbol{x}]_i) \quad \forall i = 1, \ldots, n \tag{2}$$

where, I is the intensity and $\boldsymbol{P} \in \mathbb{R}^{n \times m}$ is a system matrix such that each entry $[\boldsymbol{P}]_{i,j}$ represents the contribution of the j-th image voxel to the i-th detector. Given the raw projections $\bar{\boldsymbol{b}}$, we take the logarithm as follows

$$y_i = \log\left(\frac{I}{\bar{b}_i}\right) \quad \forall i = 1, \ldots, n \tag{3}$$

where we assumed that the intensity I is sufficiently high so that $b_i > 0$ for all i. Image reconstruction is based on finding a suitable image $\hat{\boldsymbol{x}}$ that approximately solves

$$\boldsymbol{y} = \boldsymbol{P}\hat{\boldsymbol{x}} \tag{4}$$

where $\boldsymbol{y} = [y_1, \ldots, y_n]^\top \in \mathbb{R}^m$. The reconstruction can also be achieved with more sophisticated iterative techniques that account for the stochastic properties of the measurement (1) (Nuyts et al. 1998, Elbakri & Fessler 2002).

In a sparse-view setting, the number of rotation angles of the detector is decreased in order to reduce the radiation passing through the patient. This leads to a degradation in image quality due to artifacts caused by the reduction in the number of projection angles in the measurement $\boldsymbol{y}$.

2.2. Proposed Low Resolution Reconstruction aware CED Model

Supervised deep learning-based methods learn the mapping between the measurement $\boldsymbol{y}$ and the corresponding reconstructed image $\boldsymbol{x}$. In the case of direct deep learning-based

image reconstruction this mapping is typically learned via neural networks which can be represented as a function $\boldsymbol{F}_{\boldsymbol{\theta}}: \mathbb{R}^n \rightarrow \mathbb{R}^m$ with trainable parameters $\boldsymbol{\theta} \in \mathbb{R}^{n_{\mathrm{P}}}$:

$$\hat{\boldsymbol{x}} = \boldsymbol{F}_{\boldsymbol{\theta}}(\boldsymbol{y})\,. \tag{5}$$

where, $\hat{\boldsymbol{x}}$ is the predicted image.

Most of the works in direct reconstruction for sparse-view CT represent $\boldsymbol{F}$ with a neural network with fully-connected layers. These networks require huge memory and large datasets for training. As an alternative to this, we propose the use of fully convolutional encoder-decoder networks that have lesser trainable parameters and are faster to train.

The main idea is to enforce data consistency by providing estimates at different resolutions $\hat{\boldsymbol{x}}_r \in \mathbb{R}^{m_r}$, $m_r < m$, $r = 1, \ldots, R$:

$$\hat{\boldsymbol{x}} = \boldsymbol{F}_{\boldsymbol{\theta}}(\boldsymbol{y}, (\hat{\boldsymbol{x}}_r)_{r=1}^{R}) \tag{6}$$

where each $\hat{\boldsymbol{x}}_r \in \mathbb{R}^{m_r}$ is an approximate solution of

$$\boldsymbol{y} = \boldsymbol{P}\boldsymbol{U}_r\hat{\boldsymbol{x}}_r \tag{7}$$

with $\boldsymbol{U}_r \in \mathbb{R}^{m \times m_r}$ being an upsampling operator.

In a typical CED, the encoder learns the representation of the input domain and the decoder learns to map this representation to the corresponding image in the output domain. In the specific case of a CED for medical image reconstruction, the encoder operates in the sinogram space and the decoder in the image space. Based on this hypothesis, we propose to concatenate the estimates at different levels of the decoder part of the network. The function of these concatenations is to help the network learn the structure of the image. The feature maps at different levels of the decoder have different resolutions. Hence, concatenating the estimate $\hat{\boldsymbol{x}}_r$ at different levels requires the estimate to be of the appropriate resolution. The different convolutional layers in the decoder work towards arriving at a clear reconstructed image that is free of artifacts and noise. The estimate $\hat{\boldsymbol{x}}_r$ is obtained with a sparse sinogram, hence it is artifact-ridden and noisy. Therefore, concatenating the estimate $\hat{\boldsymbol{x}}_r$ at a level closer to the output resolution is counter productive as the network has lesser number of convolutional layers to correct the noise and artifacts. On the other hand the estimate at lower resolutions has lesser structural information compared to the estimates at higher resolution. The selection of $\hat{\boldsymbol{x}}_r$ should ensure a balance between aiding the network to learn the structure of the image and enabling it to correct the artifacts and noise.

Our method, namely Low-Resolution Reconstruction-Aware Convolutional Encoder-Decoder (LRR-CED), was implemented with $R = 2$ and the image estimates $\hat{\boldsymbol{x}}_r$ were obtained by FBP at lower resolution. With the help of a series of experiments, we determined the best possible configuration for concatenating $\hat{\boldsymbol{x}}_r$. In Section 3.4.1, we present quantitative evaluation of the effect of these concatenations on the reconstructed images. In Section 3.5, through an ablation study we establish the combined impact of sinogram and the proposed concatenations on the reconstructed image quality.

We investigate LRR-CED with two different variations for $\boldsymbol{F}$, LRR-CED(D) with Fully Convolutional DenseNets and LRR-CED(U) with U-Net, which are discussed in Section 2.2.1 and Section 2.2.2.

2.2.1. Fully Convolutional Dense Networks A fully convolutional dense network was used as first variation of LRR-CED. Dense networks Huang et al. (2017) are based on the hypothesis that connecting all the layers to each other in a feed forward fashion leads to higher accuracy and easier training of the network. A typical dense block of three layers is depicted in Figure 1(a). The extension of dense networks for image segmentation was proposed by Jégou et al. (2017). The three blocks involved in the construction of this network are Dense Block (DB) with l layers, Transition Up (TU) and Transition Down (TD). The combination of these three blocks helps in building an encoder-decoder structure suitable for tasks dealing with image-to-image domain transfer. Each layer consists of batch normalization, rectified linear unit (ReLU) activation and 3×3 convolution. TD includes: batch normalization, ReLU, 1×1 Convolution and 2×2 max pooling. Finally, TU includes a 3×3 transposed convolution with stride 2. The important modification to the architecture blocks in our work is the removal of the dropout layers. The fully convolutional dense network with proposed concatenations is represented in Figure 1(b). The complete architecture details are given in Figure 1(c).

2.2.2. U-Net One of the most established architectures for image-to-image translation is U-Net, which we used as second variation of LRR-CED (called from here on-wards as LRR-CED(U)). A typical U-Net consists of Convolution, Activation (ReLU) and Pooling layers in the encoder and Upsampling, Convolution and Activation in the decoder. We have used U-Net without the dropout, similar to the dense network. The U-Net is represented in Figure 2.

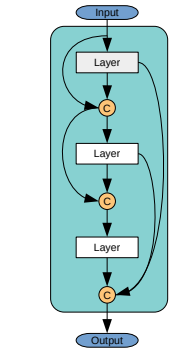
2.2.3. Loss Function The aim of a supervised data-driven image reconstruction task is to predict an image that is as close as possible to the ground truth (GT) image. The appropriate loss function to achieve this is the mean absolute error (MAE) which is defined as follows:

$$\mathrm{MAE}(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) = \frac{1}{m} \sum_{j=1}^{m} |x_j^\star - \hat{x}_j| \tag{8}$$

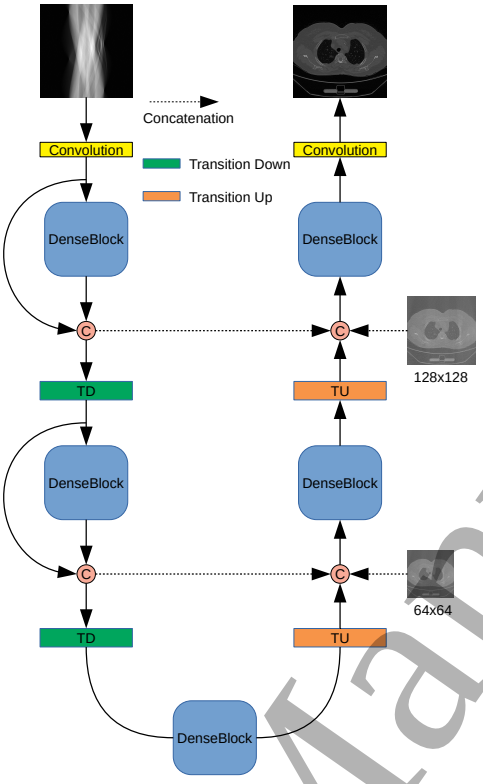
where $\boldsymbol{x}^\star = [x_1^\star, \ldots, x_m^\star]^\top \in \mathbb{R}^m$ and $\hat{\boldsymbol{x}} = [\hat{x}_1, \ldots, \hat{x}_m]^\top \in \mathbb{R}^m$ are respectively the true image and predicted image.

In order to improve the resolution of reconstructed images, many deep learning approaches have used the perceptual loss as proposed by Johnson et al. (2016). This loss uses a pre-trained neural network to extract features from the predicted image and the GT. It can be defined as follows:

$$P_k(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) = |[\mathrm{VGG}_{16}]_k(\boldsymbol{x}^\star) - [\mathrm{VGG}_{16}]_k(\hat{\boldsymbol{x}})|, \quad k = 1, \ldots, 5 \tag{9}$$



(a) Dense Block



(b) LRR-CED(D)

Input (Sinogram)
32, 3x3 Convolutions
DB (4 layers) + TD
DB (5 layers) + TD
DB (5 layers) + TD
DB (7 layers) + TD
DB (10 layers) + TD
DB (12 layers) + TD
DB (15 layers)
TU + DB (12 layers)
TU + DB (10 layers) Concatenate $\mathbf{x}_1$
TU + DB (7 layers) Concatenate $\mathbf{x}_2$
TU + DB (5 layers)
TU + DB (4 layers)
1x1 Convolution
Sigmoid Activation
Output (Image)

(c) Architecture Summary.

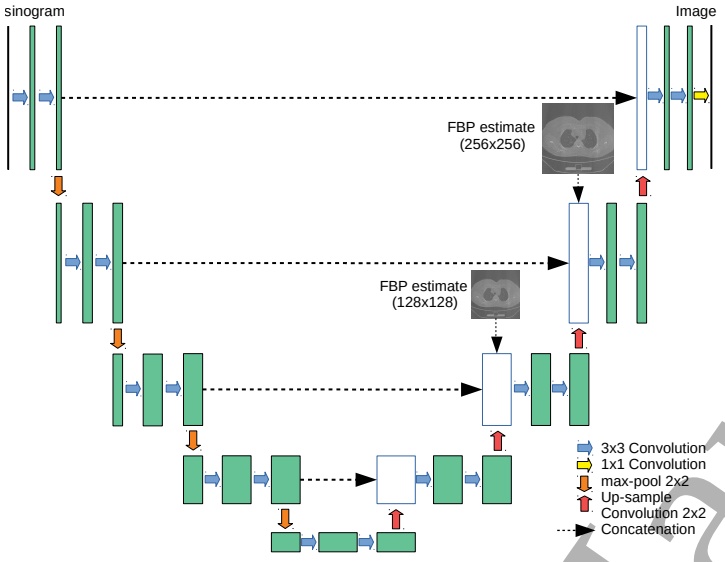
Figure 1: Different components of LRR-CED(D): (a) Representation of a dense block with three layers; (b) LRR-CED(D) with Fully convolutional dense network with $\boldsymbol{x}_1$ at 64×64 and $\boldsymbol{x}_2$ at 128×128 (for the sake of representation we included only 5 dense blocks in the figure); (c) Complete architecture summary

where $[\text{VGG}_{16}]_k(\boldsymbol{x}^\star)$ and $[\text{VGG}_{16}]_k(\hat{\boldsymbol{x}})$ are the features extracted from block k of the VGG_{16} neural network with respectively the GT and the predicted image as inputs (Simonyan & Zisserman 2014). The features extracted from higher layers of the neural network contain generic information (edges, contrast, etc.) while the deeper layers have finer task-specific details. The VGG_{16} network was pre-trained on Image-Net data Deng et al. (2009) which is far from a medical context. Hence, the higher-level generic features were found to be more relevant for the task of medical image reconstruction. We observed that using extracted features from two different levels, namely Block 1 and Block 3, of the VGG16 network proved to be most effective.

The final loss function that was used for training both the aforementioned networks is defined as follows:

$$\mathcal{L}(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) = \alpha \text{MAE}(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) + \beta(P_1(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) + P_3(\boldsymbol{x}^\star, \hat{\boldsymbol{x}})) \tag{10}$$

where P_1 and P_3 are perceptual loss from the extracted features of the two different blocks above-mentioned, α and β are weights which were set to 10 and 0.5 during the training phase.



(a) LRR-CED(U)

Input (Sinogram)
3 layers of 32, 3x3 Convolutions
Max pooling
3 layers of 64, 3x3 Convolutions
Max pooling
3 layers of 128, 3x3 Convolutions
Max pooling
3 layers of 256, 3x3 Convolutions
Max pooling
3 layers of 512, 3x3 Convolutions
256, 3x3 Transposed Convolutions
2 layers of 256, 3x3 Convolutions
128, 3x3 Transposed Convolutions
Concatenate $\mathbf{x}_1$
2 layers of 128, 3x3 Convolutions
64, 3x3 Transposed Convolutions
Concatenate $\mathbf{x}_2$
2 layers of 64, 3x3 Convolutions
32, 3x3 Transposed Convolutions
2 layers of 32, 3x3 Convolutions
1x1 Convolution
Output (Image)

(b) Architecture Summary

Figure 2: Different components of LRR-CED(U): (a) LRR-CED(U): U-Net with $\boldsymbol{x}_1$ at 64×64 and $\boldsymbol{x}_2$ at 128×128. (b) Complete architecture summary.

2.3. Dataset

The data used in this work is from the Large-Scale CT and PET/CT Dataset for Lung Cancer Diagnosis (Lung-PET-CT-Dx) (Li et al. 2020, Clark et al. 2013). Details of the dataset are given in Table 1. The images in this dataset were reconstructed using FBP on full-angular coverage measurement data. We used the ASTRA toolbox, for data processing to create the projection-image pairs (Van Aarle et al. 2016). A fan-beam geometry with a source to detector distance at 1500 mm and source to the center of the rotation at 1000 mm were considered. The number of detectors was set to 700 and the number of angles was varied to generate different levels of sparsity ($n_{\mathrm{a}} = 20$, 40, 60, 90 and 120). The source was rotated 360 degrees around the object, the angular sampling was adjusted to generate different sparsity configurations. The noise-free projection data were obtained using the Beer-Lambert law (2) with an input emission intensity of 10^5. The final projection data were obtained by adding Poisson noise (i.e., (1)) to the noise-free projection data. We finally generated the FBP estimates from the noise-added sparse-projections which were used in training the networks as explained previously. The estimates at different lower resolutions were obtained through nearest-neighbor

interpolation of the images at full-resolution (512×512). The sinograms which are inputs to the network are resized using the same interpolation technique. Sample images from the dataset used are shown in Figure 3.

Table 1: Dataset Description

Dataset Statistics	
Modalities	CT
Number of Participants	355
Number of Studies	436
Number of Series	1295
Number of 2-D Image slices	251,135
CT Matrix size	512

2.4. Training

We implemented the architectures described in the previous section using TensorFlow and Keras (Abadi et al. 2016, Chollet et al. 2015). A subset of the dataset consisting of 22,000 2-D CT images was used in this study. We then split the data into 30,000 images for training and 2,000 images for testing. The sinograms and FBP estimates were generated using the ASTRA toolbox as described above. The sinograms were resized to 512×512 to ensure symmetry with the images for easier training of the network. The FBP estimates $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$ were resized to the resolutions required for concatenation to the proposed networks. The neural networks were independently trained for each of the sparse-view settings with $N_{\mathrm{a}} = 20, 40, 60, 90$ and 120. The choice of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ were at 64×64 and 128×128 resolutions for LRR-CED(D) and 128×128 and 256×256 resolutions for LRR-CED(U). The networks were trained for 50 epochs with Adam optimizer with a decay of 10^{-4}.

2.5. Quantitative Analysis

The metrics used for evaluating the reconstructed images were structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR). They are defined as follows:

$$\mathrm{SSIM}(\boldsymbol{x}^\star, \boldsymbol{x}) = \frac{(2\mu_{\boldsymbol{x}^\star}\mu_{\boldsymbol{x}} + c_1)(2\sigma_{\boldsymbol{x}^\star\boldsymbol{x}} + c_2)}{(\mu_{\boldsymbol{x}^\star}^2 + \mu_{\boldsymbol{x}}^2 + c_1)(\sigma_{\boldsymbol{x}^\star}^2 + \sigma_{\boldsymbol{x}}^2 + c_2)} \tag{11}$$

where $\mu_{\boldsymbol{x}^\star}$ and $\mu_{\boldsymbol{x}}$ are the mean of $\boldsymbol{x}^\star$ and $\boldsymbol{x}$ respectively, $\sigma_{\boldsymbol{x}^\star}^2$ and $\sigma_{\boldsymbol{x}}^2$ are the variance of $\boldsymbol{x}^\star$ and $\boldsymbol{x}$, $\sigma_{\boldsymbol{x}^\star\boldsymbol{x}}$ is the covariance between $\boldsymbol{x}^\star$ and $\boldsymbol{x}$, $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ where $k_1 = 0.01$ and $k_2 = 0.03$ by default,

$$\mathrm{PSNR} = 20 \log_{10}\left(\frac{L-1}{\mathrm{RMSE}}\right) \tag{12}$$

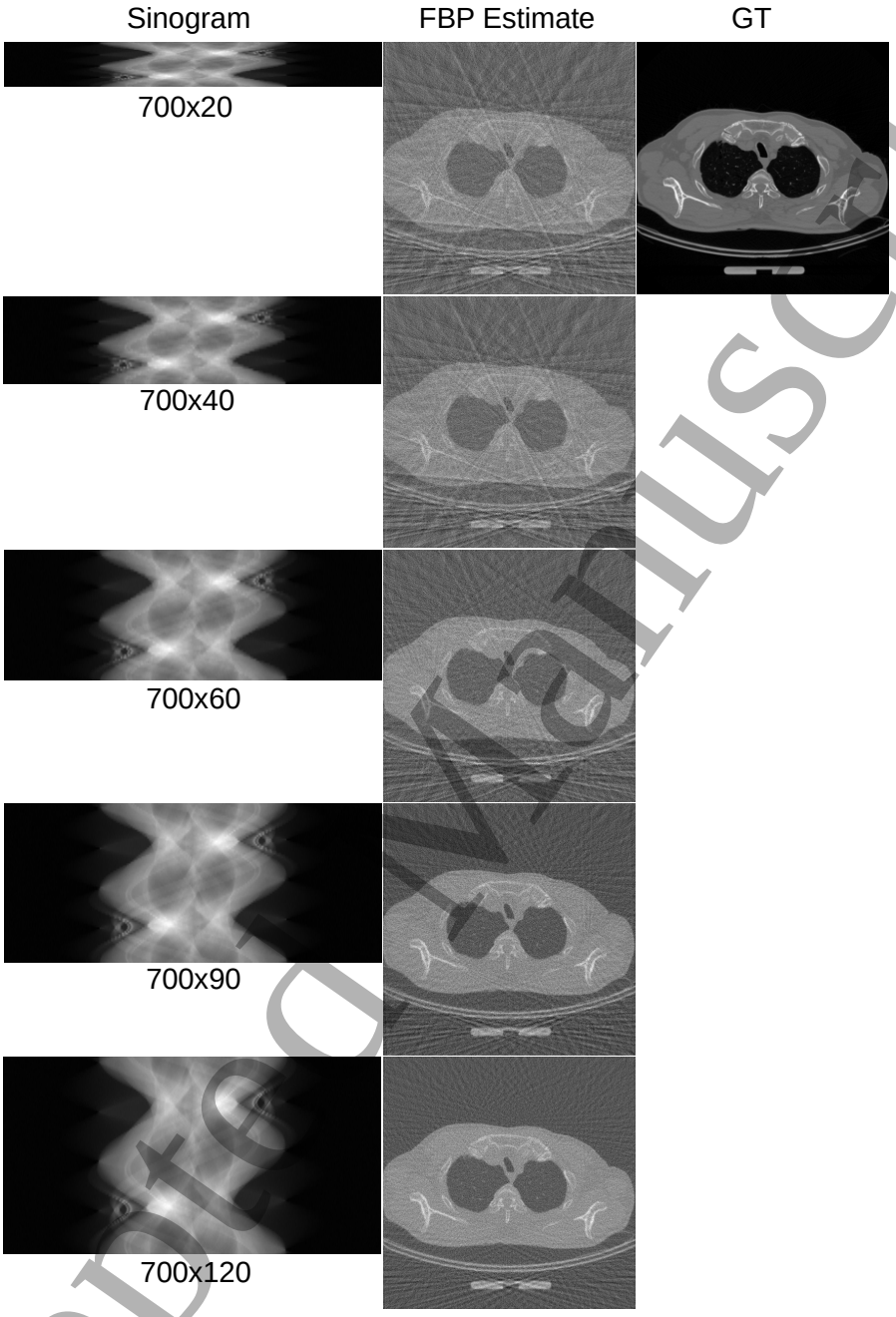


Figure 3: Samples from the dataset: Sinograms with different sparse-view configurations along with their corresponding FBP estimate.

where L is the maximum intensity in the image and root mean squared error (RMSE) is given by

$$\mathrm{RMSE}(\boldsymbol{x}^\star, \hat{\boldsymbol{x}}) = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(x_j^\star - \hat{x}_j)^2}\,. \tag{13}$$

3. Results

3.1. Experimental Results with Simulated Data

Fig. 4 shows the images reconstructed with LRR-CED(D) for various degrees of sparsity in the projections. Images from various parts of the patient volume are displayed at different HUT windows for clearer evaluation of the proposed approach. We observe the improvement in the reconstructed images with the decrease in sparsity in the views. The images reconstructed with $n_{\mathrm{a}} = 120$ appear closest to the GT. The soft tissue regions in the images reconstructed with less than 60 views show artifacts which are not present with the use of more projections. Similarly in Fig. 5, we show the images reconstructed with LRR-CED(U).

The reconstruction algorithms used for comparing with the proposed method were two post-processing deep learning methods namely FBP-ConvNet (Jin et al. 2017) and DD-Net (Zhang et al. 2018), and an iterative method (penalized weighted least-squares (PWLS)-TV). The FBP-ConvNet is based on U-Net and the DD-Net consists of dense blocks and deconvolution layers. For the PWLS-TV method we have used a FISTA solver with Prox TV to implement the TV regularizer (Beck & Teboulle 2009).

In Fig. 6 and Fig. 7 we present a comparison of reconstructed images using different algorithms with 60 and 90 views respectively. The top row consists of the GT and the reconstructed image by proposed LRR-CED(D) approach. The second row consists of images with LRR-CED(U) and the FBP-ConvNet. The third row consists of images reconstructed with PWLS-TV and DD-Net. Finally the last row has the image reconstructed with FBP. We have also performed the reconstruction using different regularization parameters to select the optimal parameter's setting. The region highlighted in yellow is zoomed and displayed alongside the corresponding image. These methods are quantitatively compared in Table 2 and Table 3.

We observe that the deep learning methods perform better than the iterative and analytical methods. The images reconstructed with U-Net based methods namely LRRCED(U) and FBP-ConvNet, have very similar characteristics: The contrast is higher and they perform better quantitatively. However, images reconstructed with DenseNet by comparison show less noise and streaking artifacts. These visual observations can be more clearly seen in the zoomed images shown in Fig. 6. This is further reiterated in the intensity plot profiles shown in Fig. 8 and Fig. 9, where the LRR-CED(D) results are closer to the GT. In accordance with the metrics tabulated in Table 2 and Table 3, we find that the plots of deep learning-based methods are very close to that of the GT. Even though the proposed approach with typical CEDs performs a task which is more complex than denoising, the metrics indicate that the quality has not deteriorated compared to a standard post-processing approach.

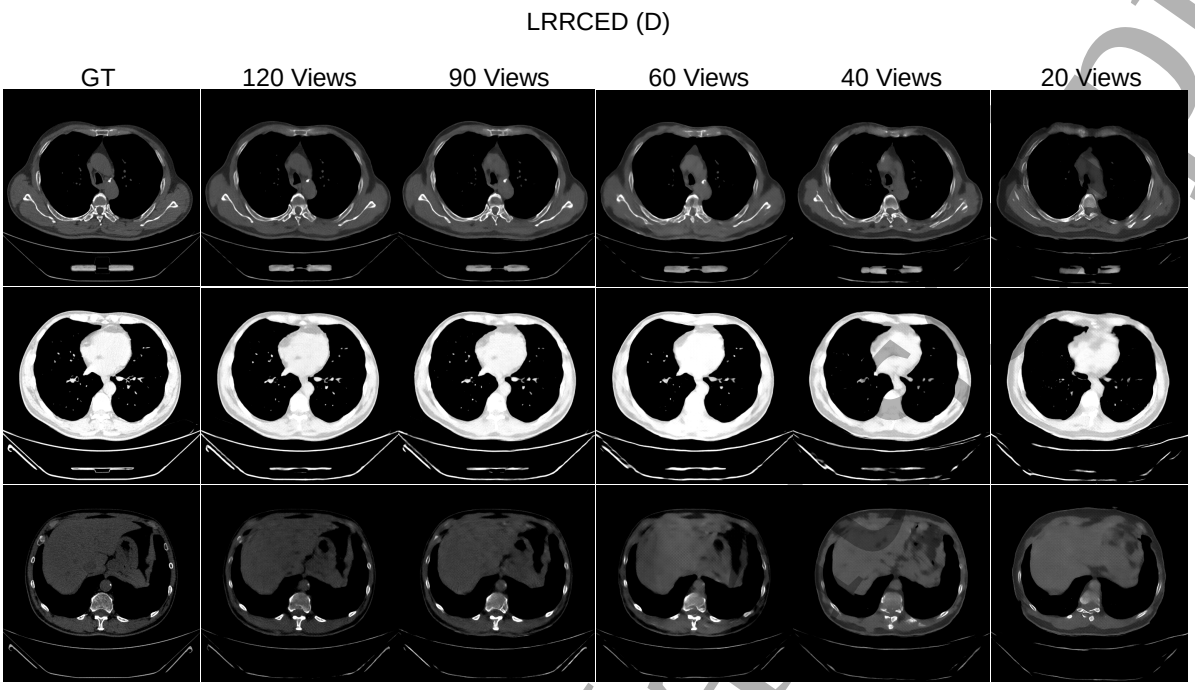


Figure 4: Images reconstructed with LRR-CED(D) approach with different sparse-view configurations, i.e., projections with $N_{\mathrm{a}} = 120, 90, 60, 40$ and 20. For better visual inspection images in first row are displayed in -40 ± 600 HUT window, the second row in -340 ± 400 HUT and the third in -150 ± 400 HUT.

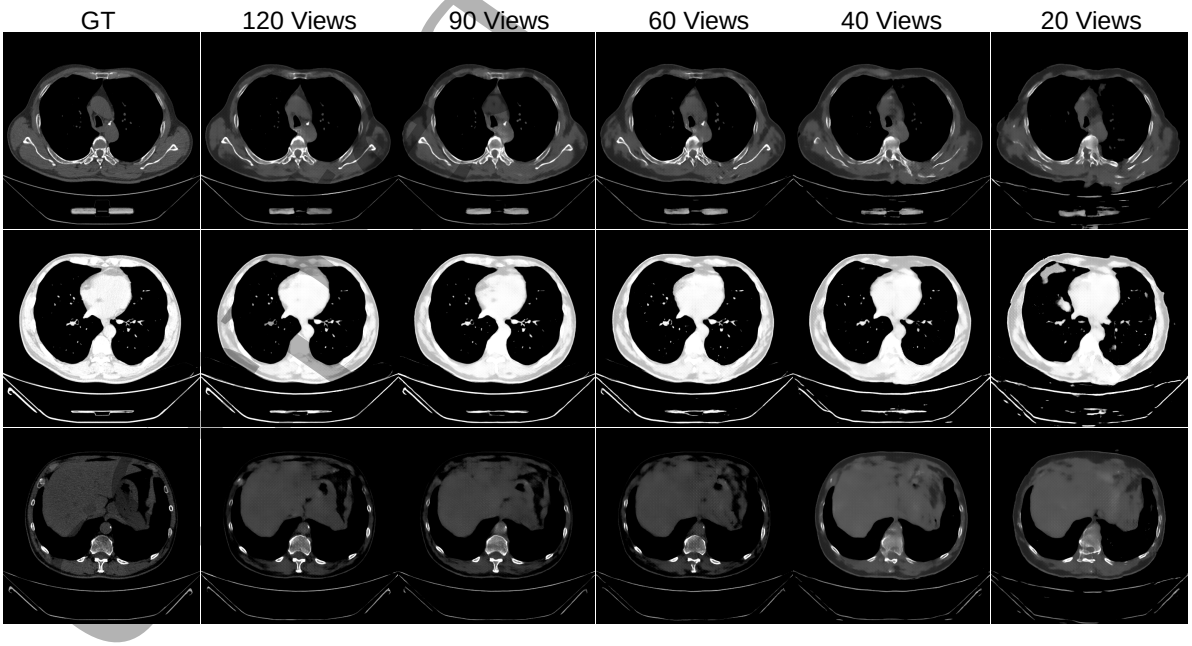


Figure 5: Images reconstructed with LRR-CED(U) approach with different Sparse-View configurations, i.e., projections with $N_{\mathrm{a}} = 120, 90, 60, 40$ and 20. Images in first row are displayed in -40 ± 600 HUT window, the second row in -340 ± 400 HUT and the third in -150 ± 400 HUT.

Table 2: Quantitative comparison of various reconstruction algorithms with SSIM and PSNR for projections with 60 views

Metric	FBP	PWLS-TV	FBP ConvNet	DD-Net	LRR-CED (D)	LRR-CED (U)
SSIM	0.16	0.89	0.90	0.88	0.89	0.90
PSNR	11.57	30.79	31.58	30.04	30.04	30.20

Table 3: Quantitative comparison of various reconstruction algorithms with SSIM and PSNR for projections with 90 views

Metric	FBP	PWLS-TV	FBP ConvNet	DD-Net	LRR-CED (D)	LRR-CED (U)
SSIM	0.19	0.91	0.93	0.90	0.91	0.92
PSNR	13.57	32.49	35.27	32.47	32.70	32.86

3.2. Experiments with Real Data

The proposed networks were initialized with the weights from the previous study and were then trained on the real data. The real data used in this study was part of the Low Dose CT grand challenge (McCollough 2016). The data constituted of 10 patients, acquired with flying spot technique and a helical scan. It was a subset of the larger Mayo CT clinic database (Moen et al. 2021). The data from nine patients constituting of 3,994 2-D slices was used for training and the trained network was tested on another patient data. The three-dimensional (3-D) sinograms obtained from the helical scan were converted into 2-D sinograms through the single slice re-binning method employed in (Kim et al. 2017). We further resampled the sinograms reducing the number of views to 64. The number of detector panels was 734. The FBP estimates were generated from these sparse-view sinograms and resized for training the LRR-CED.

We present the results for four different slices across the patient volume and their quantitative evaluation in Figure 10 and Table 4, respectively. We observe that the reconstructed images with the proposed networks have similar characteristics as the ones from the simulation study. The transfer learning strategy ensures that the quality of the reconstructed images is maintained even with very limited training data.

3.3. Stability Study

One of the major challenges to data-driven neural network approaches is the ability to generalize over different types of test data. The extent to which a neural network is stable when presented with data different from the training data is the focus of this study. This topic has been extensively evaluated in the article by Antun et al. (2020). The authors analyzed the impact of tiny perturbations and small structural changes in sampling and image domain on the reconstructed images. They also observed

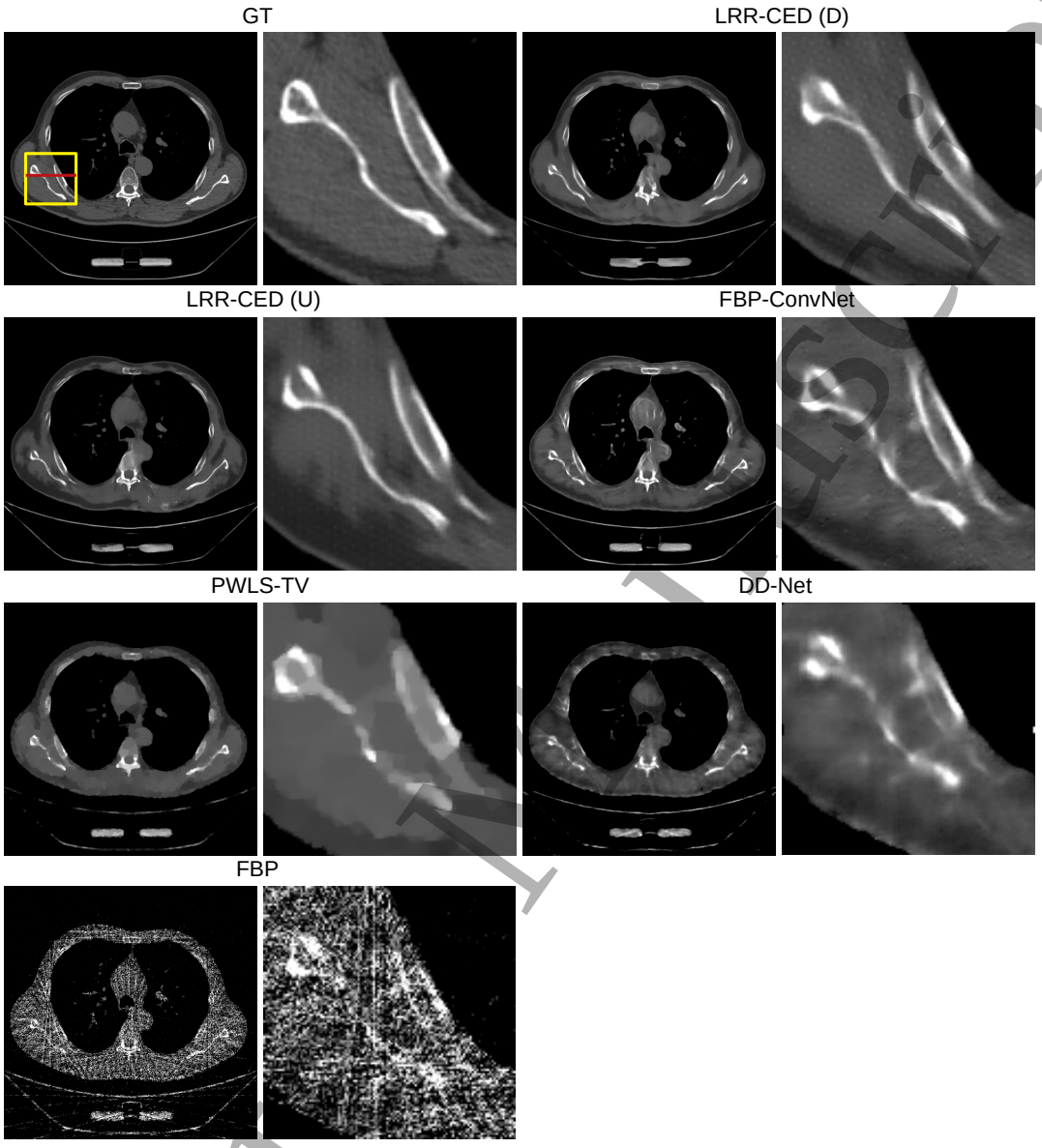


Figure 6: Comparative analysis for 60 views: From the top left corner, we have GT image, reconstructions with LRR-CED(D). In the second row reconstructed images with LRR-CED(U) and FBP-ConvNet. The third row consists of images reconstructed with PWLS-TV and DD-Net. Finally the last row has the image reconstucted with FBP.

the way in which a change in sampling (sparsity in CT for example) could influence performance. In our work centered around sparse-view CT image reconstruction, we performed a series of experiments with different levels of sparsity in the testing data. The proposed network LRR-CED(D) was trained separately on each of the sparsity configurations, ($N_{\mathrm{a}} = 20, 40, 60, 90$ and 120). It was then tested using the sinograms and the corresponding FBP estimates for all of the possible values of N_{a} considered.

The results are displayed in Fig 11. The top row corresponds to network trained with 20-view data, the second with 40-view data and so on. The trend is towards an

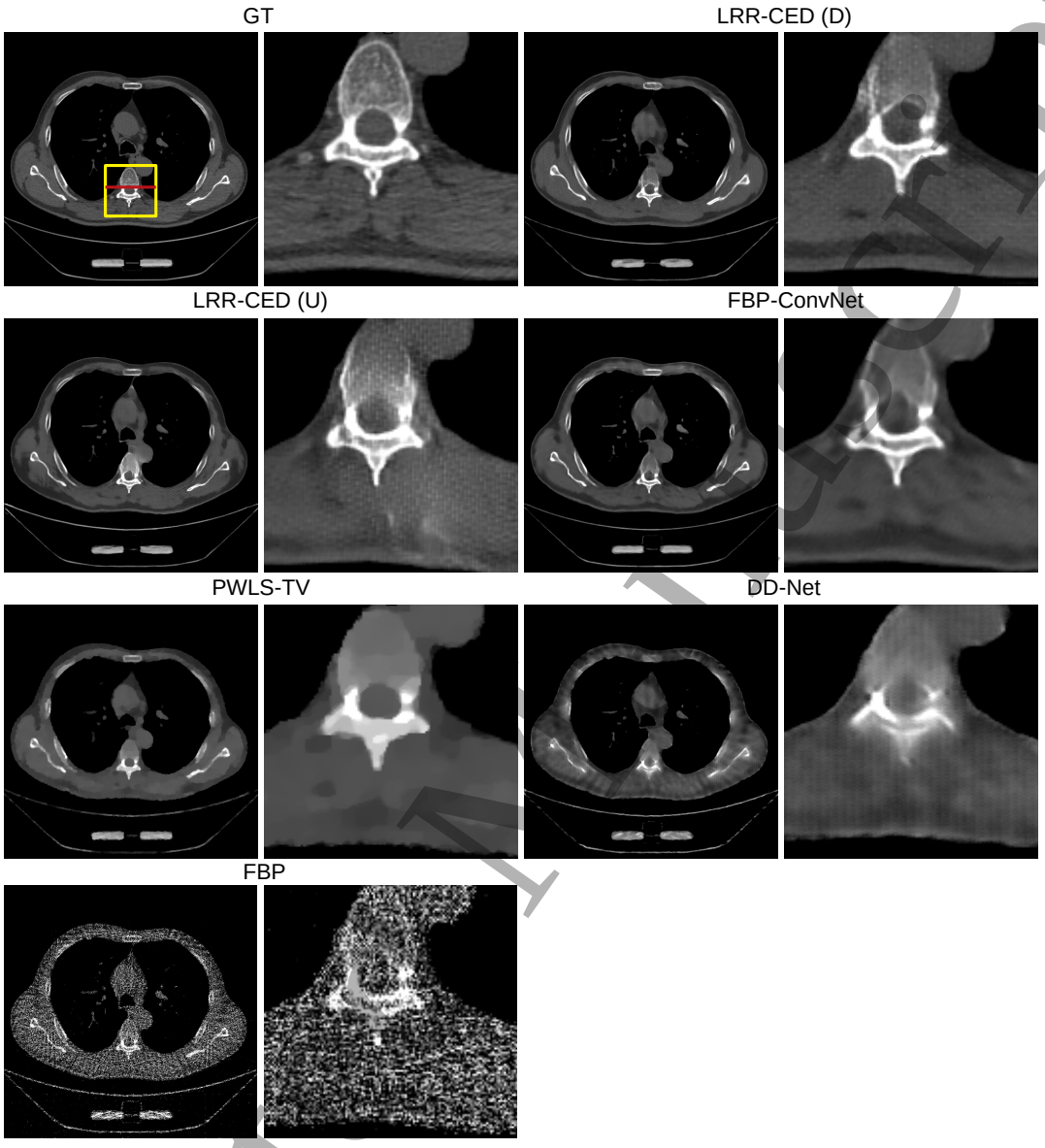
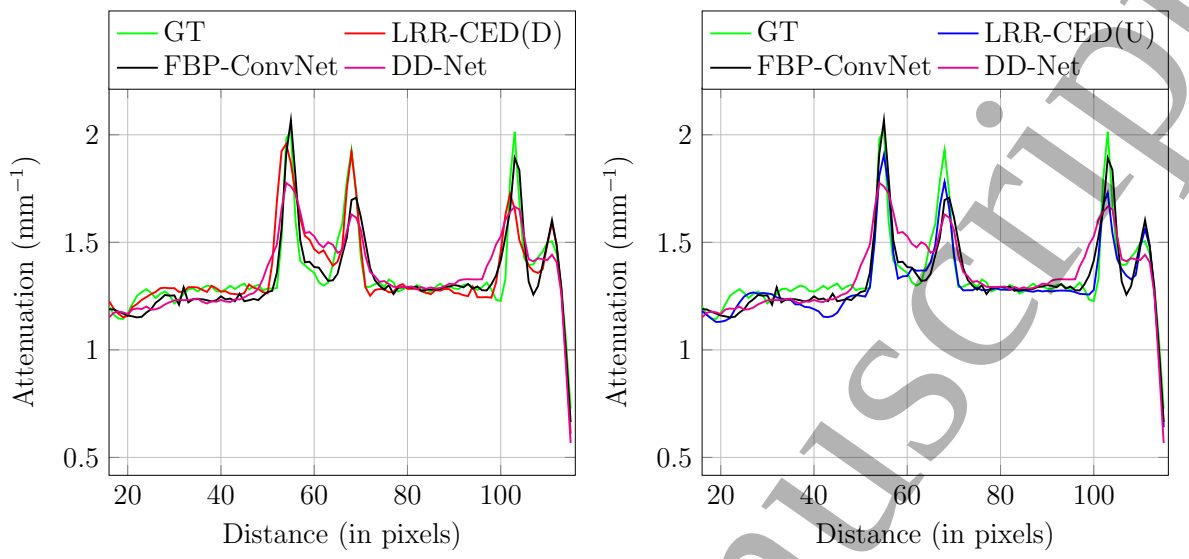


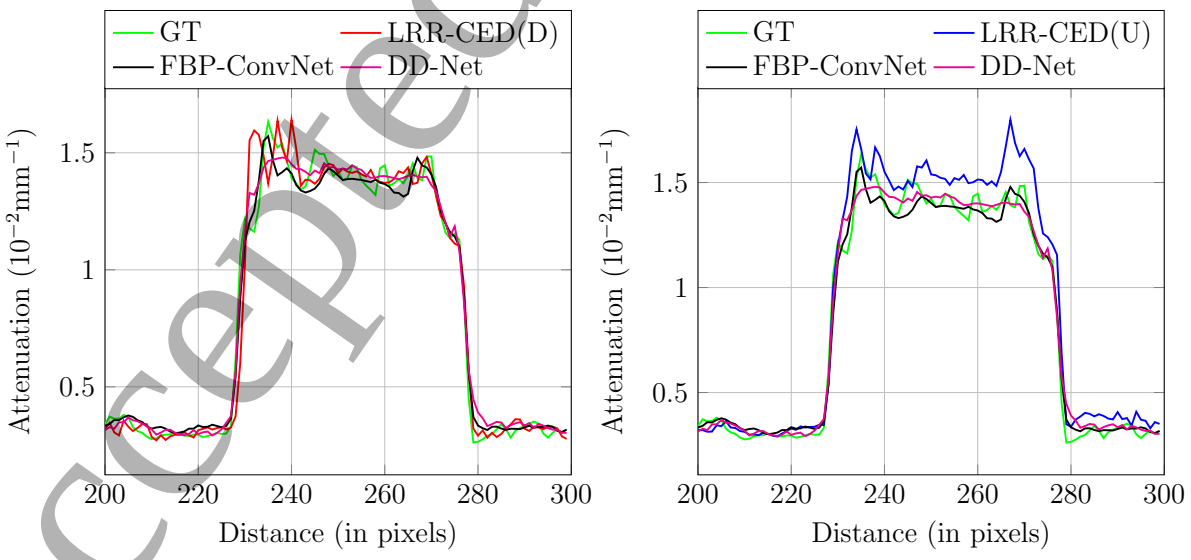
Figure 7: Comparative analysis for 90 views: From the top left corner, we have GT image, reconstructions with LRR-CED(D). In the second row reconstructed images with LRR-CED(U) and FBP-ConvNet. The third row consists of images reconstructed with PWLS-TV and DD-Net. Finally the last row has the image reconstucted with FBP.

improvement in overall image quality with reduced sparsity in the sinograms. On one hand, we observe that in the scenarios where the testing data has more sparsity than the training data, the artifacts in the reconstructed images are more clearly visible. This is clearly seen in the last two rows in Figure 11, where the network was trained on 90 views and 120 views data and the images reconstructed with lower N_{a} are ridden with artifacts. On the other hand, the image quality especially in the soft tissue regions is higher when the network is trained and tested on data with more views. The proposed network maintains stability in the reconstructed images with the increase in the sampling



(a) 60 views 1 (b) 60 views 2

Figure 8: Intensity plot profile for the region marked in red from Fig. 6 comparing LRR-CED(D), FBP-ConvNet and DD-Net to the GT in (a) and LRR-CED(U), FBP-ConvNet and DD-Net in (b)



(a) (b)

Figure 9: Intensity plot profile for the region marked in red from Fig. 7 comparing LRR-CED(D), FBP-ConvNet and DD-Net to the GT in (a) and LRR-CED(U), FBP-ConvNet and DD-Net in (b)

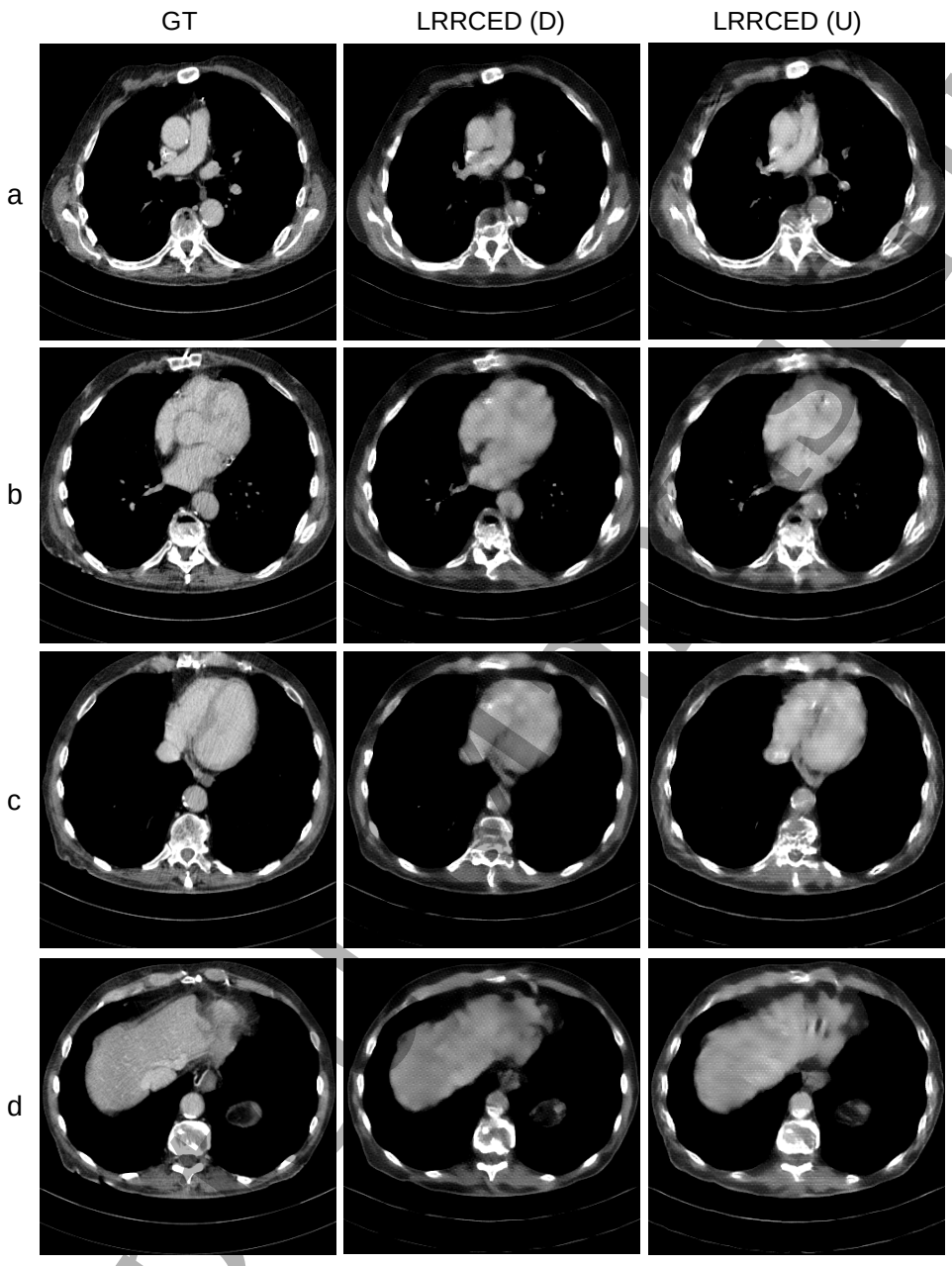


Figure 10: Real data study: Images reconstructed with the proposed approaches across 4 different slices displayed in the window 40 ± 200 HUT.

in the testing data. However, when the testing data has fewer views than the training data, artifacts are present in the reconstructed images.

3.4. Hyperparameter Optimization

Finding the optimal hyperparameters is an important aspect of training neural networks. The common hyperparameters in a typical CNN are number of filters, number of layers, etc. These interdependent hyperparameters determine the rate of convergence and require task-specific experimentation to arrive at the best possible configuration. The

Table 4: Quantitative comparison of images reconstructed with the proposed algorithms w.r.t. GT across different slices in the patient volume from the real dataset displayed in Fig. 10

Image	Metric	LRRCED(D)	LRRCED(U)
a	SSIM	0.89	0.92
	PSNR	35.70	36.64
b	SSIM	0.88	0.92
	PSNR	35.19	36.13
c	SSIM	0.94	0.92
	PSNR	40.86	42.04
d	SSIM	0.84	0.91
	PSNR	33.37	34.59

unique hyperparameters in our proposed approach are the resolutions of concatenated FBP estimates. The number of training examples is another important component that varies depending on the task and the trainable parameters of the neural network selected for the task. In this section we discuss our experiments that determined the selection of these two important hyperparameters.

3.4.1. Concatenation Resolution Selection To select the best possible configuration for concatenation in the proposed approach, we trained the networks with a fixed set of hyper-parameters and different combinations of concatenations. We discuss the results with LRR-CED(D) in this regard. The number of training samples were set to 10,000 for all the experiments. The training data were projections with 90 views, corresponding FBP reconstructed images and the GT. The training was done for 25 epochs. Each of the concatenation setting was evaluated on 5 test patients. The average SSIM for each patient was plotted for each of the experiment setting. In Fig. 12(a) we have the average SSIM vs Patient plot for single concatenation at a specific resolution. Similarly Fig. 12(b) consists of plots for double concatenation at two different resolutions. The double concatenation at $64 \times 64, 128 \times 128$ overall leads to the best metrics, thus becoming our choice for the experiments in this work. These results are tabulated in Table 5.

3.4.2. Training Examples Analysis One of the biggest challenges in any data driven algorithm is the selection of training examples required for the experiments. It is important to analyze this hyper-parameter as it serves as an important factor for the network to be reproducible and scalable. We varied the number of training examples for the best concatenation setting from the previous section and the 90-view scenario. The evaluation was similar to the previous experiment with the average SSIM for 5 patients. The results from these experiments are tabulated in Table 6. As seen in Fig. 13(a), the performance of the network improves along with the increase in the number of training

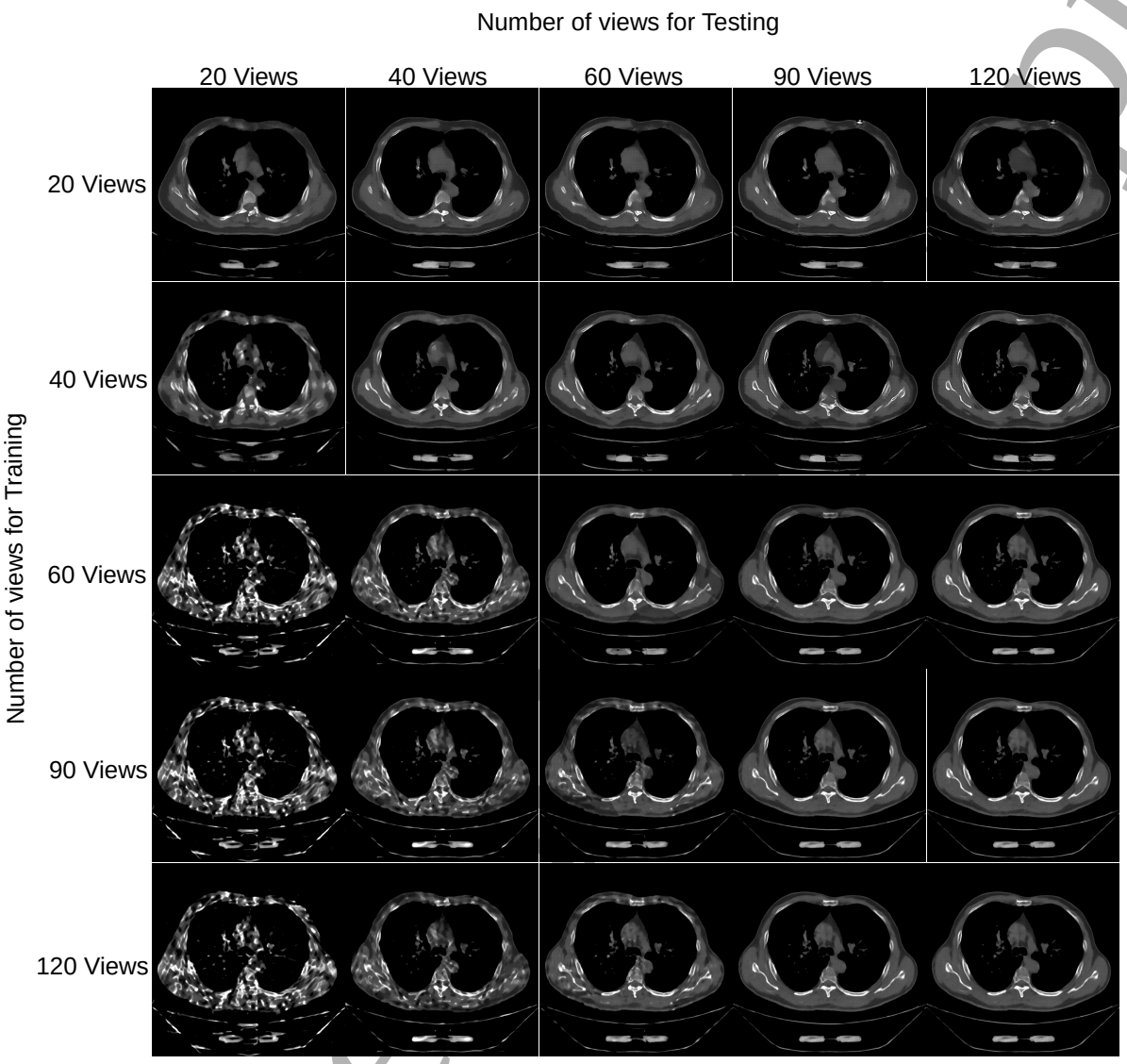


Figure 11: Stability study: Each row corresponds to the network trained on specific value of N_{a}, and tested with all the possible values of N_{a}.

examples. Although there was only a marginal difference in the performance of the network with 20,000 or 30,000 training examples, we observed that training with higher training samples reduced artifacts caused due to the perceptual loss in the reconstructed images. Hence, the choice of number of training examples was 30,000 in our experiments. The average SSIM values across the test patients tend to get similar as the number of training examples increases.

3.4.3. Epochs for Training Analysis Another important hyperparameter is the number of epochs for training the neural network. To ensure a balance between efficient utilization of computing resources and optimum performance of the network, it is necessary to select an appropriate number of training epochs. We trained our network with 10,000 training examples and the best double concatenation configuration from 3.4.1, varying

the number of training epochs. Using a similar evaluation methodology as above we compared the average SSIM for LRR-CED(D) trained for 10, 20, 50 and 100 epochs. As seen in Fig. 13(b) and Table 7, increased training time leads to better metrics. However, we observe that for neural network trained for 50 and 100 epochs, the average SSIM is almost identical. Hence, we choose to train our network for 50 epochs in our experiments.

3.5. Ablation Study

We performed an ablation study to understand the impact of the proposed concatenations on the neural network performance. DenseNet described earlier was trained for 50 epochs on 20,000 data samples in three different scenariosas shown in Figure 14, two of which used either a sinogram consisting of randomly distributed Gaussian noise and no low-resolution concatenations: (i) true sinogram and the reconstructed image only (no low-resolution concatenations), (ii) Gaussian noise sinogram, low-resolution concatenations and the reconstructed images, and (iii) true sinogram, low-resolution concatenations and the reconstructed images.

The image predictions by the three different neural networks are shown in Figure 15. DenseNet without the low-resolution concatenations does produce images with some structural information, but the other two configurations generate images of much better quality. We observe that the concatenations indeed help the network learn the structure of the image, while the sinograms contribute in artifact and noise removal. This is reflected upon closer inspection of the third and fourth images in Figure 15. The images predicted with LRR-CED(D) trained using the randomly distributed Gaussian noise sinogram instead of the true sinogram have artifacts and noise which is also seen quantitatively in Table 8. The best metrics and image quality are demonstrated by the neural network trained on the combination of sinograms and low-resolution estimates labeled as LRR-CED(D) in Figure 15.

Table 5: Average SSIM for different configurations of concatenations

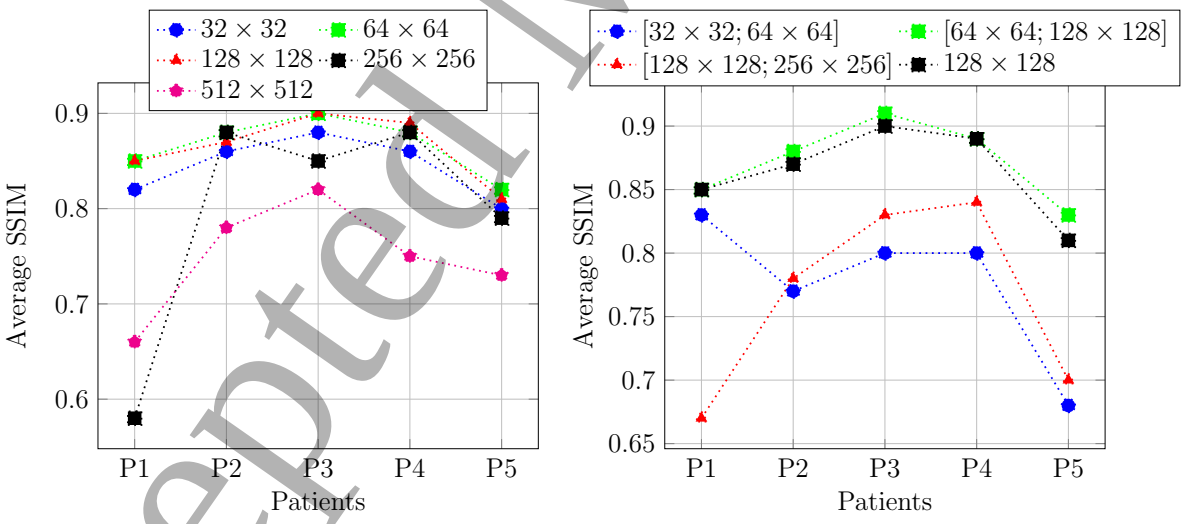
Concatenated FBP Resolution	Average SSIM				
	P1	P2	P3	P4	P5
(32×32)	0.82	0.86	0.88	0.86	0.80
(64×64)	0.85	0.88	0.90	0.88	0.82
(128×128)	0.85	0.87	0.90	0.89	0.81
(256×256)	0.58	0.88	0.85	0.88	0.79
(512×512)	0.66	0.78	0.82	0.75	0.73
$(32 \times 32, 64 \times 64)$	0.83	0.77	0.80	0.80	0.68
$(\mathbf{64 \times 64, 128 \times 128})$	**0.85**	**0.88**	**0.91**	**0.89**	**0.83**
$(128 \times 128, 256 \times 256)$	0.67	0.78	0.83	0.84	0.70

Table 6: Average SSIM for different number of training examples

Number of Training examples	Average SSIM				
	P1	P2	P3	P4	P5
1,000	0.82	0.79	0.86	0.85	0.72
5,000	0.84	0.77	0.86	0.84	0.69
10,000	0.85	0.88	0.91	0.89	0.83
20,000	**0.89**	**0.90**	**0.91**	**0.90**	**0.82**
30,000	0.89	0.89	0.90	0.90	0.82

Table 7: Average SSIM for different number of epochs for training

Number of Epochs for Training	Average SSIM				
	P1	P2	P3	P4	P5
10	0.85	0.89	0.91	0.91	0.85
20	0.86	0.89	0.91	0.90	0.86
50	**0.88**	**0.90**	**0.92**	**0.92**	**0.88**
100	0.88	0.90	0.92	0.92	0.86

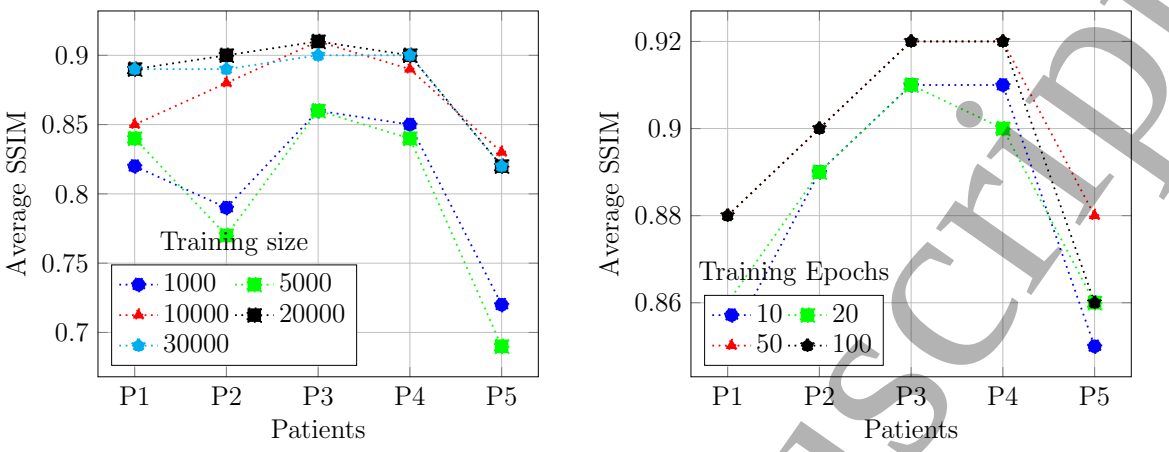


(a) Single concatenations (b) Double concatenations

Figure 12: Comparison of concatenations for the particular case of 90 views evaluated with SSIM on 5 different patients from the dataset. The best metrics were found with concatenations at 64 × 64 and 128 × 128 resolutions.

4. Discussion

The use of deep learning architectures in the framework of medical image reconstruction is propelled by potentially faster reconstruction without compromising on the quality



(a) Training Examples Analysis (b) Training epochs analysis

Figure 13: Comparison of Average SSIM for 5 different Patient data for 90 views with varying number of training samples. The configuration of the network is the one with best performance from the analysis in Fig. 12(a).(concatenations at 64×64 and 128×128).

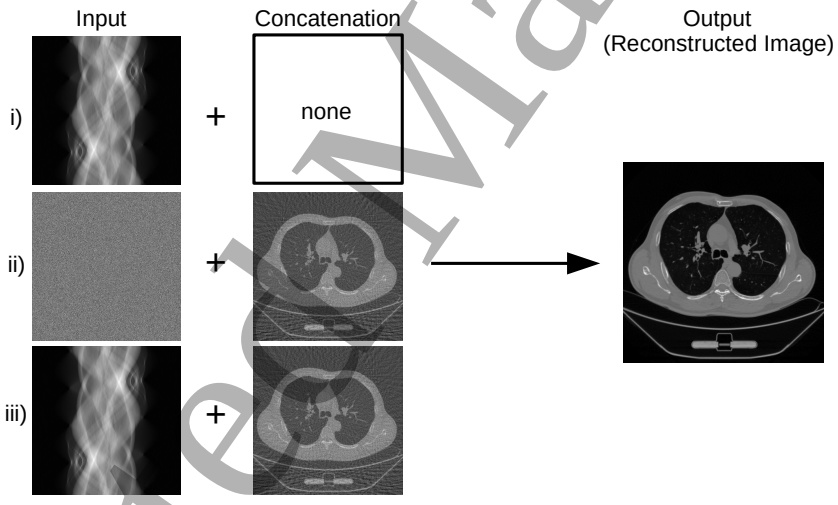


Figure 14: Schematic representation of configurations used in the ablation study: (i) true sinogram and the reconstructed image only (no low-resolution concatenations); (ii) randomly distributed Gaussian noise sinogram, low-resolution concatenations and the reconstructed images; (iii) true sinogram, low-resolution concatenations and the reconstructed images.

of the images. To this end, hybrid image reconstruction involving unrolled iterative algorithms with embedded deep learning architectures do not significantly reduce the reconstruction time. Hence, the use of deep learning architectures for either improving images from a fast analytic algorithm or direct reconstruction becomes more relevant for their incorporation into the image reconstruction pipeline. One significant problem for direct image reconstruction is the requirement of large and complex networks to learn the mapping from sinograms to images without the help of any reconstruction estimate.

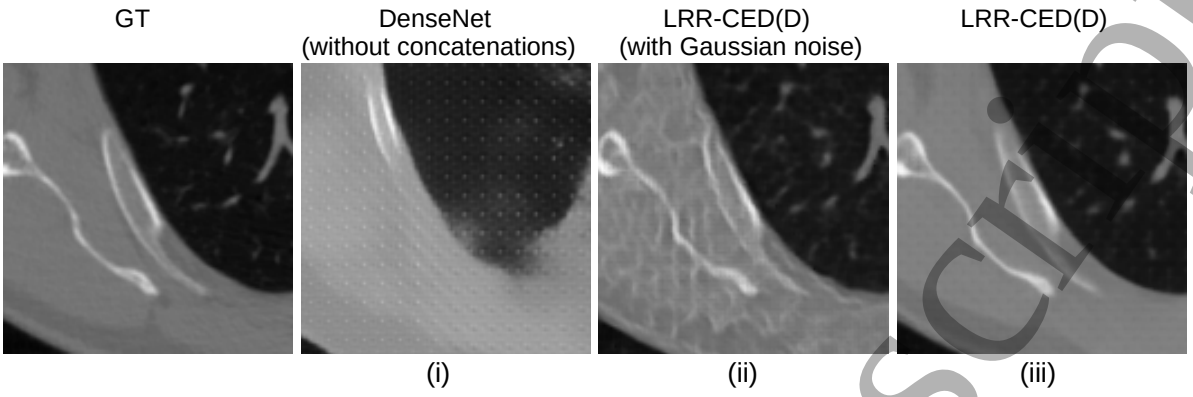


Figure 15: Ablation study: Predictions from different configurations of the network.

Table 8: Ablation Study: Quantitative comparison of different configurations of the DenseNet

Sl.No.	True sinograms	Concatenations	Gaussian noise sinograms	SSIM	PSNR
(i)	✓	✗	✗	0.29	12.05
(ii)	✗	✓	✓	0.70	28.89
(iii)	✓	✓	✗	**0.88**	**32.53**

The networks used for post-processing on the other hand are simpler and relatively easy to train. In this work we attempted to use these post-processing networks for the direct image reconstruction task along with low-resolution scout images from direct analytical method. We show that concatenating FBP estimates at lower resolutions is sufficient to allow the network to learn the mapping from sinogram to image space. Through the use of two different networks with the concatenation approach we demonstrate that this idea can be applied to CEDs in general.

In the sparse-view CT scenario artifact removal along with denoising increases the challenges of getting a clean well-resolved image. We observed that the use of traditional loss functions (L1 or L2) resulted in blurry images. To tackle this and to improve the sharpness of the images we used perceptual loss along with the standard L1 loss. The reconstructed images with our proposed LRR-CED(D) and LRR-CED(U) have higher SSIM and PSNR than images reconstructed with a traditional iterative algorithm and standard deep learning based post-processing methods DD-Net and FBP-ConvNet. The similarity in the images from the deep learning methods stems from the fact that the choice of networks used in our proposed work was inspired from post-processing CEDs. The contribution in this work is the use of these networks to learn the mapping from sparse sinograms to images with the same amount of training examples, which is possible only with the proposed addition of the concatenations. Through the ablation study from Section 3.5, we reiterate the contribution of both the sinogram and the low-resolution concatenations for image reconstruction. The CED without the concatenations could

learn the mapping but it would need much higher number of training examples for image quality comparable to other methods.

We are currently exploring the possibility of using image estimates from earlier iterations of standard iterative algorithms while ensuring that the trade-off between time and image quality is not compromised. The use of other alternative architectures is also being explored to arrive at reconstructed images which perform significantly better than existing post-processing approaches. We are working on experiments with low-dose CT and other tomographic reconstruction modalities to establish the adaptability of the proposed approach.

Recently, transformer based networks incorporating the self attention mechanism have been proposed in a variety of medical imaging tasks Pan et al. (2021). proposed Multi-domain Integrative Swin Transformer Network (MIST) for sparse-view CT image reconstruction that outperforms FBP-ConvNet. It is interesting to note that the details in various organs and soft tissue regions are well resolved compared to other popular reconstruction methods. We intend to work on transformer based networks for bringing about further improvements in our proposed method.

5. Conclusions

In this work we studied the use of fully convolutional encoder-decoder networks in direct sparse-CT image reconstruction. We introduced a new approach that uses lower dimension FBP estimates as concatenations to help the network learn the mapping from sinogram to image space. In the context of image reconstruction, we inject the information from the inverse of a CT physical system (FBP estimate) as a feature map in the decoder. We presented two variations of the proposed approach namely LRR-CED(D) using fully convolutional dense networks and LRR-CED(U) using U-Net. The proposed neural networks reconstruct images that are either better or are on par with traditional reconstruction algorithms and post-processing deep learning based approaches (DD-Net and FBP-ConvNet). A single pass of a sparse sinogram through the network results in reconstructed images without the artifacts and noise which are severely present in the concatenated FBP estimates. Finally, this idea of using task specific concatenations that enable one to have control over what the network learns, can be extended to various other problems in medical imaging.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. et al. (2016). Tensorflow: A system for large-scale machine learning, *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283.

Adler, J. & Öktem, O. (2018). Learned primal-dual reconstruction, *IEEE Transactions on Medical Imaging* **37**(6): 1322–1332.

Antun, V., Renna, F., Poon, C., Adcock, B. & Hansen, A. C. (2020). On instabilities of deep learning in image reconstruction and the potential costs of ai, *Proceedings of the National Academy of Sciences* **117**(48): 30088–30095.

Arjovsky, M., Chintala, S. & Bottou, L. (2017). Wasserstein generative adversarial networks, *International conference on machine learning*, PMLR, pp. 214–223.

Beck, A. & Teboulle, M. (2009). Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems, *IEEE transactions on image processing* **18**(11): 2419–2434.

Chollet, F. et al. (2015). Keras.
URL: *https://github.com/fchollet/keras*

Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M. et al. (2013). The cancer imaging archive (tcia): maintaining and operating a public information repository, *Journal of Digital Imaging* **26**(6): 1045–1057.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database, *2009 IEEE conference on computer vision and pattern recognition*, Ieee, pp. 248–255.

Elbakri, I. A. & Fessler, J. A. (2002). Statistical image reconstruction for polyenergetic X-ray computed tomography, *IEEE Transactions on Medical Imaging* **21**(2): 89–99.

Fu, L. & De Man, B. (2019). A hierarchical approach to deep learning and its application to tomographic reconstruction, *15th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, Vol. 11072, International Society for Optics and Photonics, p. 1107202.

Greenspan, H., Van Ginneken, B. & Summers, R. M. (2016). Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique, *IEEE Transactions on Medical Imaging* **35**(5): 1153–1159.

Gupta, H., Jin, K. H., Nguyen, H. Q., McCann, M. T. & Unser, M. (2018). Cnn-based projected gradient descent for consistent CT image reconstruction, *IEEE Transactions on Medical Imaging* **37**(6): 1440–1453.

Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. (2017). Densely connected convolutional networks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Jégou, S., Drozdzal, M., Vazquez, D., Romero, A. & Bengio, Y. (2017). The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation, *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 11–19.

Jin, K. H., McCann, M. T., Froustey, E. & Unser, M. (2017). Deep convolutional neural network for inverse problems in imaging, *IEEE Transactions on Image Processing* **26**(9): 4509–4522.

Johnson, J., Alahi, A. & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution, *European conference on computer vision*, Springer, pp. 694–711.

Kandarpa, V., Bousse, A., Benoit, D. & Visvikis, D. (2020). Dug-recon: A framework for direct image reconstruction using convolutional generative networks, *IEEE Transactions on Radiation and Plasma Medical Sciences* **5**(1): 44–53.

Kim, D., Ramani, S. & Fessler, J. A. (2014). Combining ordered subsets and momentum for accelerated X-ray CT image reconstruction, *IEEE Transactions on Medical Imaging* **34**(1): 167–178.

Kim, K., El Fakhri, G. & Li, Q. (2017). Low-dose ct reconstruction using spatially encoded nonlocal penalty, *Medical physics* **44**(10): e376–e390.

Lee, H., Lee, J., Kim, H., Cho, B. & Cho, S. (2018). Deep-neural-network-based sinogram synthesis for sparse-view CT image reconstruction, *IEEE Transactions on Radiation and Plasma Medical Sciences* **3**(2): 109–119.

Leuschner, J., Schmidt, M., Ganguly, P. S., Andriiashen, V., Coban, S. B., Denker, A., Bauer, D., Hadjifaradji, A., Batenburg, K. J., Maass, P. et al. (2021). Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications, *Journal of Imaging* **7**(3): 44.

Li, P., Wang, S., Li, T., Lu, J., HuangFu, Y. & Wang, D. (2020). A large-scale CT and PET/CT dataset for lung cancer diagnosis. Data retrieved from The Cancer Imaging Archive.
URL: *https://doi.org/10.7937/TCIA.2020.NNC2-0461*

Li, Y., Li, K., Zhang, C., Montoya, J. & Chen, G.-H. (2019). Learning to reconstruct computed tomography images directly from sinogram data under a variety of data acquisition conditions, *IEEE Transactions on Medical Imaging* **38**(10): 2469–2481.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B. & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis, *Medical image analysis* **42**: 60–88.

Liu, Y., Liang, Z., Ma, J., Lu, H., Wang, K., Zhang, H. & Moore, W. (2013). Total variation-Stokes strategy for sparse-view X-ray CT image reconstruction, *IEEE Transactions on Medical Imaging* **33**(3): 749–763.

McCollough, C. (2016). Tu-fg-207a-04: Overview of the low dose ct grand challenge, *Medical physics* **43**(6Part35): 3759–3760.

Moen, T. R., Chen, B., Holmes III, D. R., Duan, X., Yu, Z., Yu, L., Leng, S., Fletcher, J. G. & McCollough, C. H. (2021). Low-dose ct image and projection dataset, *Medical physics* **48**(2): 902–911.

Nuyts, J., De Man, B., Dupont, P., Defrise, M., Suetens, P. & Mortelmans, L. (1998). Iterative reconstruction for helical ct: a simulation study, *Physics in Medicine & Biology* **43**(4): 729.

Pan, J., Wu, W., Gao, Z. & Zhang, H. (2021). Multi-domain integrative swin transformer network for sparse-view tomographic reconstruction, *Available at SSRN 3991087* .

Reader, A. J., Corda, G., Mehranian, A., da Costa-Luis, C., Ellis, S. & Schnabel, J. A. (2020). Deep learning for PET image reconstruction, *IEEE Transactions on Radiation and Plasma Medical Sciences* **5**(1): 1–25.

Ronneberger, O., Fischer, P. & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation, *International Conference on Medical image computing and computer-assisted intervention*, Springer, pp. 234–241.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* .

Thaler, F., Hammernik, K., Payer, C., Urschler, M. & Štern, D. (2018). Sparse-view CT reconstruction using wasserstein gans, *International workshop on machine learning for medical image reconstruction*, Springer, pp. 75–82.

Van Aarle, W., Palenstijn, W. J., Cant, J., Janssens, E., Bleichrodt, F., Dabravolski, A., De Beenhouwer, J., Batenburg, K. J. & Sijbers, J. (2016). Fast and flexible x-ray tomography using the astra toolbox, *Optics express* **24**(22): 25129–25147.

Wang, G., Ye, J. C. & De Man, B. (2020). Deep learning for tomographic image reconstruction, *Nature Machine Intelligence* **2**(12): 737–748.

Wu, W., Hu, D., Niu, C., Yu, H., Vardhanabhuti, V. & Wang, G. (2021). Drone: Dual-domain residual-based optimization network for sparse-view ct reconstruction, *IEEE Transactions on Medical Imaging* .

Ye, D. H., Buzzard, G. T., Ruby, M. & Bouman, C. A. (2018). Deep back projection for sparse-view CT reconstruction, *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, IEEE, pp. 1–5.

Yedder, H. B., Cardoen, B. & Hamarneh, G. (2021). Deep learning for biomedical image reconstruction: A survey, *Artificial Intelligence Review* **54**(1): 215–251.

Zhang, Z., Liang, X., Dong, X., Xie, Y. & Cao, G. (2018). A sparse-view CT reconstruction method based on combination of densenet and deconvolution, *IEEE Transactions on Medical Imaging* **37**(6): 1407–1417.