# 2-Step Sparse-View CT Reconstruction with a Domain-Specific Perceptual Network (Supplementary)

Haoyu Wei[1*] Florian Schiffers[1*] Tobias Würfl[2] Daming Shen[3]
Daniel Kim[3] Aggelos Katsaggelos[1] Oliver Cossairt[1]
[1]Northwestern University, Evanston, USA [2]University of Erlangen-Nuremberg, Germany
[3]Feinberg School of Medicine, Northwestern University, Chicago, USA
florian.schiffers@northwestern.edu

## 1. Introduction

This document contains the supplementary materials that were left our in our main submission since they are too detailed for and do not significantly enhance the message we want to convey. Section 2 presents an incremental strategy for training GAN with a discriminator perceptual losses. Section 3 explains in detail the adversarial loss function mentioned in the main paper. The model architecture and hyperparameters are shown in Sec. 4. Finally, more qualitative and quantitative results for ablation study are presented in Sec. 5.

We provide all code that is necessary to reproduce the experiments reported in the paper and supplementary at https://github.com/anonyr7/Sinogram-Inpainting. This git-repository contains PyTorch implementations for our proposed networks, state-of-the-art methods, our CT dataset and a sub-set of around 30 images as examples for evaluation of each model.

## 2. Discriminator Perceptual Loss

We present an alternative training strategy using the DP loss in Alg. 1. We show the example procedure with a super-resolution task that learns from a low-resolution (LR) image to a high-resolution (HR) image. We start by training the discriminator once in every generator iteration, then increment the discriminator iteration by one in every $k$ generator iterations, where k is an arbitrary constant, for which we chose $k = 10$ in our paper. The intuition is that with more training performed, the discriminator learns better high-level features, thus benefits the generator using the DP loss.

*The first two authors have equal contribution.

**Algorithm 1:** An Incremental Training Procedure of a naive Super-Resolution GAN with DP loss.

**Data:** LR and HR images; Generator training iteration $n$; Discriminator increment period $k$.
**Result:** Super-resolved (SR) images.
**while** *not converge or iteration* $i \leq n$ **do**
  Sample a batch of LR and HR image pairs $x$ and $y$ from the training data;
  Generate a SR image $G(x)$ from generator $G$;
  **while** *discriminator iteration* $j \leq \lceil i/k \rceil$ **do**
    Update discriminator $D$ with $\mathcal{L}_{adv}(D)$;
    j = j+1;
  **end**
  Calculate $\mathcal{L}_{DP}$ from $D$ using $G(x)$ and $y$;
  Update G with $\mathcal{L}_{DP}$ and $\mathcal{L}_{adv}(G)$;
  i = i+1;
**end**

## 3. Objective Function Details

### 3.1. Adversarial Loss

For a typical GAN network that generates images from corrupted input $x$ to target distribution $y$, the adversarial losses for the generator and discriminator are defined as:

$$\begin{aligned}\mathcal{L}_{adv}(G) =& \mathbb{E}_{x\sim corrupt}[\log(1 - D(G(x)))],\\ \mathcal{L}_{adv}(D) =& \frac{1}{2}(\mathbb{E}_{y\sim target}[\log D(y)]+\\ & \mathbb{E}_{x\sim corrupt}[\log(1 - D(G(x)))],\end{aligned} \tag{1}$$

where $G$ and $D$ represent the generator and discriminator networks, respectively.

Equation 1 applies to the SIN with global discriminator and PRN model. For the local discriminator of SIN, a pair of image patches $G(x)_{patch}$ and $y_{patch}$ randomly selected from the generated image $G(x)$ and target image $y$ pair are

used for calculating an additional local adversarial cost for SIN:

$$\begin{aligned}\mathcal{L}_{local\_adv}(G) =& \mathbb{E}_{x\sim corrupt}[\log(1 - D(G(x)_{patch}))],\\ \mathcal{L}_{local\_adv}(D) =& \frac{1}{2}(\mathbb{E}_{y\sim target}[\log D(y_{patch})]+\\ & \mathbb{E}_{x\sim corrupt}[\log(1 - D(G(x)_{patch}))]),\end{aligned} \tag{2}$$

Therefore, the total adversarial loss for SIN is the sum of Eq. 1 and Eq. 2 with equal weights.

# 4. More Training Details

## 4.1. Detailed Architecture

As discussed in the main paper, our generator uses an adapted U-Net architecture, which consists of four average pooling and four bilinear upsampling layers, each scales by a factor of two in both dimensions. Each of the scaling layers is followed by a double-conv block, which consists of two sets of convolution, ReLU activation and batch normalization layers. All convolution layers in the U-Net have kernel size of three and stride equal to one.

We also use a patch discriminator [2], where the first and last convolution layers have kernel size of three and stride of one, and the middle two convolutions have kernel size of four and stride of two. It also includes LeakyReLU activation and batch normalization layers. In total, the patch discriminator outputs a patch of size $1/4 \times 1/4$ than before.

Please find the implemention of this architecture here: https://github.com/anonyr7/Sinogram-Inpainting/blob/master/SIN/model.py

## 4.2. Data Augmentation

We augment our dataset by applying random affine transformation to each image. Specifically, we randomly perform the following operations: rotate between $\pm 30°$, translate in two dimensions between $\pm 0.1$, scale by $0.5$ to $1.1$, and shear between $\pm 20°$. Such augmented images are only used for training phase, in order to bring more diversity to both sinogram and reconstruction training sets.

The code for our data augmentation is found in this Jupyter notebook: https://github.com/anonyr7/Sinogram-Inpainting/blob/master/TCIA_data_preprocessing.ipynb

# 5. More Results from Ablation Study and State-of-the-Art

In this section, we show complement of the ablation study discussed in the main paper. Specifically, we present more visual results of different composition of our models, analysis of the cascaded inputs and more sinogram-domain results for our discriminator perceptual loss.

## 5.1. Checkerboard Artifacts in Sinograms

We present more sinogram results from the cGAN model introduced by Ghani *et al* [1] in Fig. 3. The sinograms exhibit streak artifacts especially on the boundary areas, causing the reconstructions to have skewed streak artifacts. More examples can be found at https://github.com/anonyr7/Sinogram-Inpainting/tree/master/Toy-Dataset/CGAN_sinogram.

## 5.2. Qualitative comparison among our models

We present the visual results of different compositions of our models in Fig. 1, corresponding to the quantitative metrics among our models in the main paper. We show that although a single PRN model generates smooth and clean images, they still either suffer from unremoved streak artifacts or hallucinate wrong details that are critical in analysis of the reconstructions. Meanwhile, we also observe secondary artifacts from single SIN model. This is in correspondence with our argument that single-domain models are not enough for SV-CT reconstruction tasks, and our SIN-4c-PRN model benefits from both domain representations.

## 5.3. Effectiveness of cascade inputs of PRN

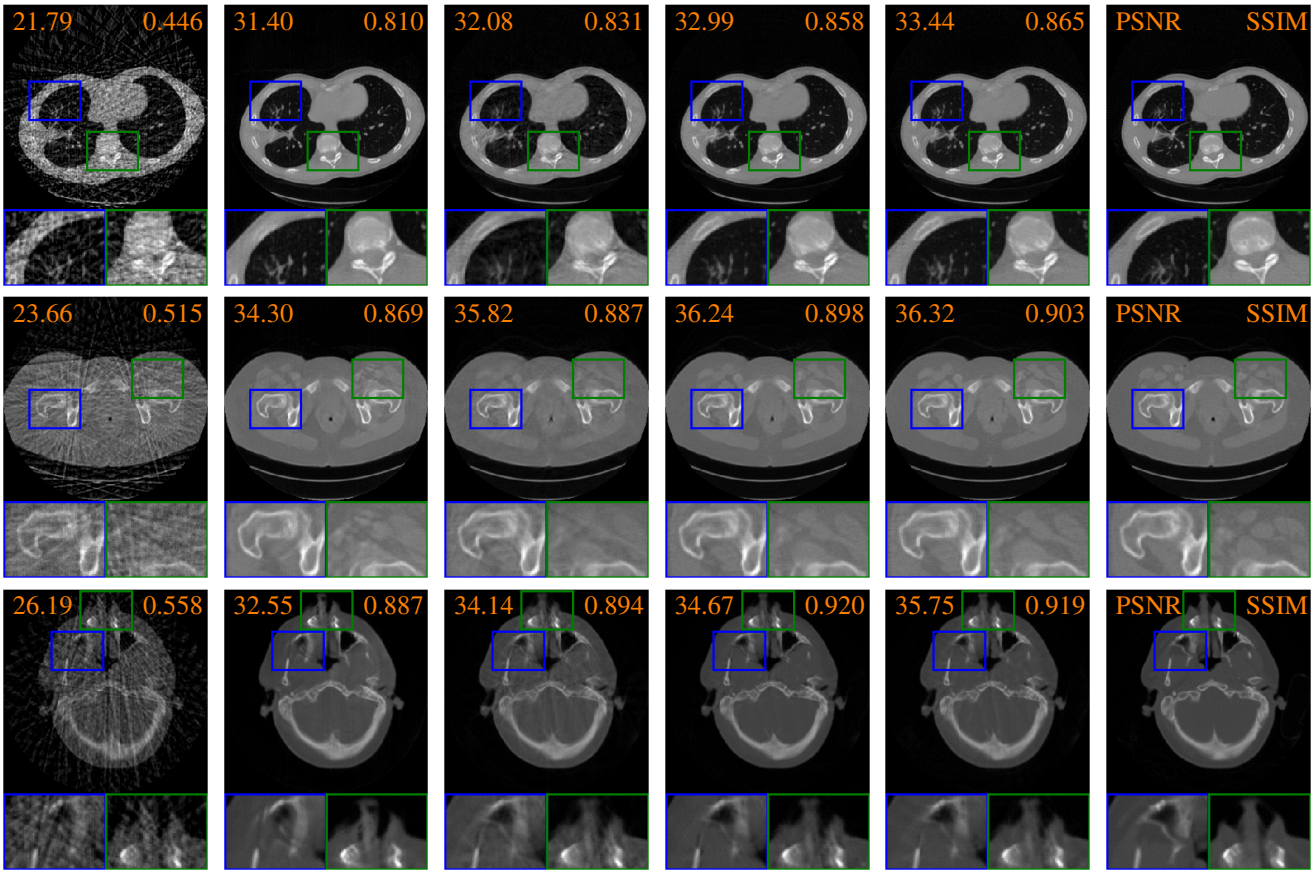
Our proposed model SIN-4c-PRN is a concatenation of SIN and PRN models trained in two steps. In particular, we propose a four-channel input for PRN that consists of reconstructions from sinograms with different undersampling factors.

Specifically, we create an input with each channel being a FBP reconstruction from a sinogram with 23, 45, 90 and 180 angles, respectively. The 23-angle sinogram is the original sparse-view sinogram, and the intermediate angles are downsampled from the 180-angle SIN-inpainted sinogram.

In order to make fair comparisons of the proposed cascade inputs, we train two networks with the same capacity, i.e., number of learnable parameters. SIN-4x-PRN is a model with a repeated four-channel input, each channel being the same FBP reconstruction from the 180-angle SIN-inpainted sinogram. We compare the performance of SIN-4x-PRN with SIN-4c-PRN in Tb. 1. Both PSNR and SSIM metrics show that SIN-4c-PRN performs better than SIN-4x-PRN, indicating that the model learns from the cascaded inputs with different level of representations.

## 5.4. Sinogram results for different perceptual losses

We qualitatively and quantitatively compare the sinogram results of our DP loss with the original VGG16 perceptual loss [3] trained with a SIN model. The SIN model learns 1D super-resolution from a 23-angle sparse-view sinogram to a 180-angle full-view sinogram. The models used for comparison share the same architecture, data and




(a) Sparse-View (b) PRN (c) SIN (d) SIN-PRN (e) SIN-4c-PRN (f) Ground Truth

Figure 1: Ablation comparison with zoom-in details. From top to bottom row: **Chest:** While PRN generates a visually clean image, it losses important details such as the bones in the blue box. In contrast, SIN-4c-PRN provides both smooth and detail-preserved reconstructions. **Abdomen:** PRN fails to remove the heavy streak artifacts in the green box area. **Head:** PRN hallucinates non-exist features on the nose and eye regions, while SIN-4c-PRN produces results close to the ground truth.

| Model | PSNR ($\sigma$) | SSIM ($\sigma$) |
|---|---|---|
| SIN-PRN | 34.61 (2.14) | 0.873 (0.035) |
| SIN-4x-PRN | 34.62 (1.97) | 0.874 (0.029) |
| SIN-4c-PRN | **34.90 (2.15)** | **0.877 (0.029)** |

Table 1: Reconstruction performance comparison. SIN-4x-PRN: Same model with SIN-4c-RPN, but duplicating input reconstruction by four times. $\sigma$ denotes standard deviation.

| | Model | PSNR ($\sigma$) | SSIM ($\sigma$) |
|---|---|---|---|
| S | Non-Perceptual | 42.50 (2.54) | 0.979 (0.005) |
| | VGG16 [3] | 39.70 (2.25) | 0.984 (0.004) |
| | Ours (DP) | 41.23 (2.83) | 0.988 (0.005) |
| R | Non-Perceptual | 33.39 (2.09) | 0.829 (0.037) |
| | VGG16 [3] | 33.51 (2.02) | 0.852 (0.036) |
| | Ours (DP) | 34.19 (2.35) | 0.859 (0.036) |

Table 2: Different perceptual loss performance trained with a single SIN. 'S' and 'R' denote Sinogram and Reconstruction domain, respectively. $\sigma$ denotes standard deviation.

training procedure, thus all the differences are contributed by the different perceptual losses.

In the main paper we compared the PSNR and SSIM metrics of the FBP reconstructions of result sinograms. Here, we further provide metrics measured directly on the result sinograms in Tb. 2. We observe better metrics of DP than VGG16, despite both of them have slightly lower PSNR than without perceptual loss. This is consistent with the observation in the original work by Johnson *et al.* [3], because perceptual losses are optimized with a different criterion, i.e., the $l_2$-norm is minimal when one optimizes for

the $l_2$ loss.

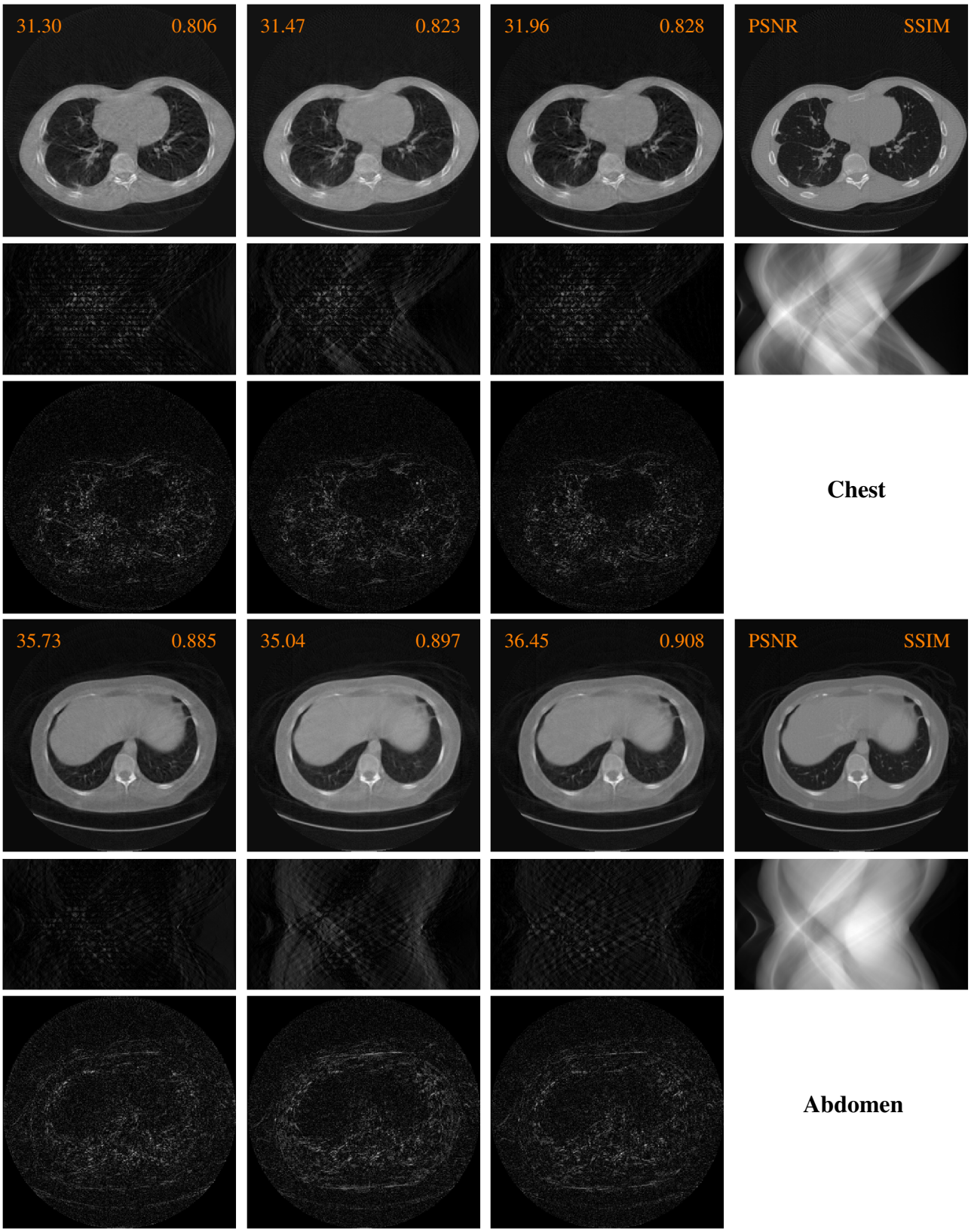
Then we show the visual results of the sinograms and their FBP reconstructions in Fig. 2. Specifically, we provide the sinogram residuals compared to the ground truth for ease of comparison. From these residuals, we find that VGG16 produce slightly more errors than the others in the sinogram domain, i.e., the training domain, possibly because sinograms are drastically different from natural RGB images where VGG16 was trained with. However, we also observe that after the FBP reconstruction, VGG16 and DP produce visually less noisy images than without perceptual losses. The reconstruction domain metrics and residuals indicate the same observation.

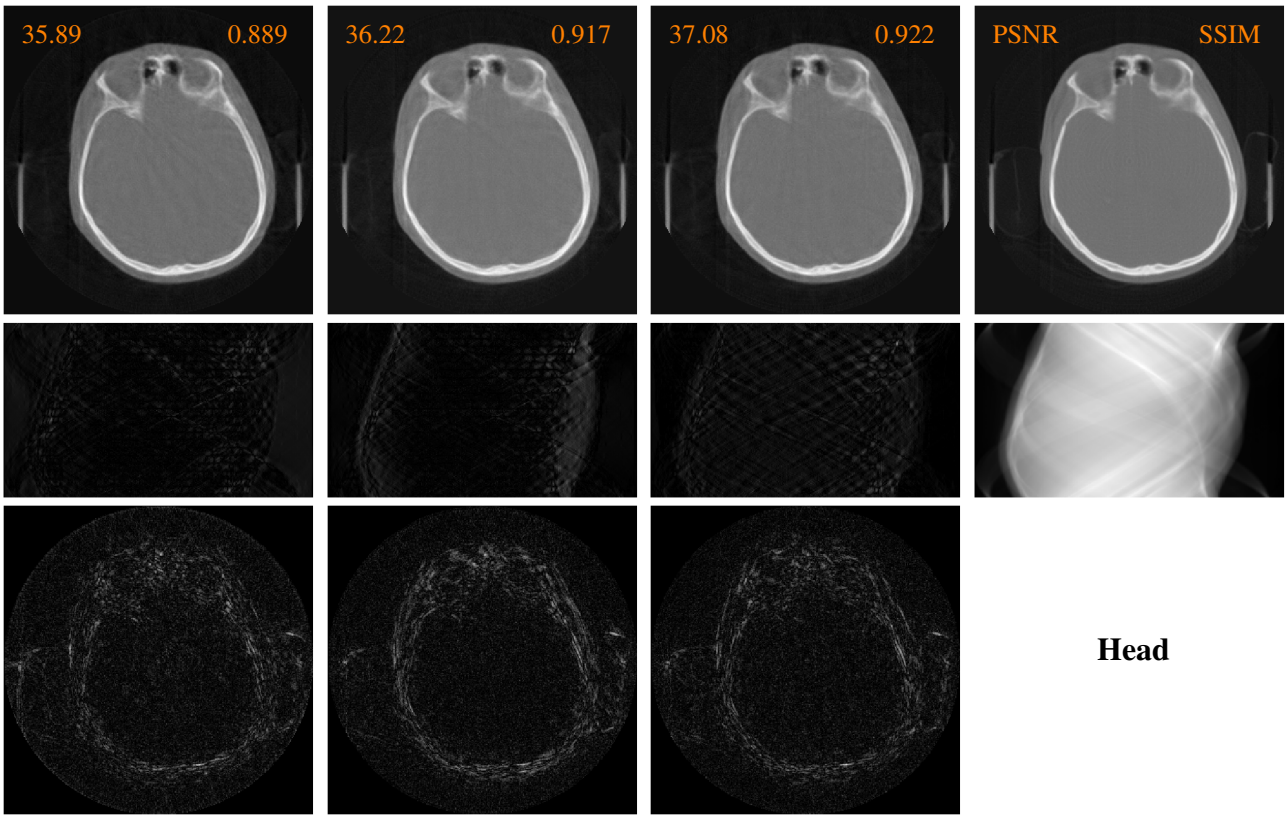
## References


[1] Muhammad Usman Ghani and W Clem Karl. Deep learning-based sinogram completion for low-dose ct. In *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, pages 1–5. IEEE, 2018. 2, 7

[2] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2

[3] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 2, 3, 6


31.30
0.806
31.47
0.823
31.96
0.828
PSNR
SSIM
Chest
35.73
0.885
35.04
0.897
36.45
0.908
PSNR
SSIM
Abdomen




(a) Non-Perceptual (b) VGG16 [3] (c) Ours (DP) (d) Ground Truth

Figure 2: (Continued) Sinogram residuals and corresponding FBP reconstructions and residuals from a single SIN using no perceptual loss, VGG16 perceptual loss or our discriminator perceptual loss. The residual (error) maps are displayed in the same range, hence we omit the scale-bar for visualization reasons. Both the non-perceptual and our DP generated sinograms have fewer errors than VGG16. However, the reconstructions of Non-perceptual sinograms are slightly noiser and have more remaining streak artifacts than the others.

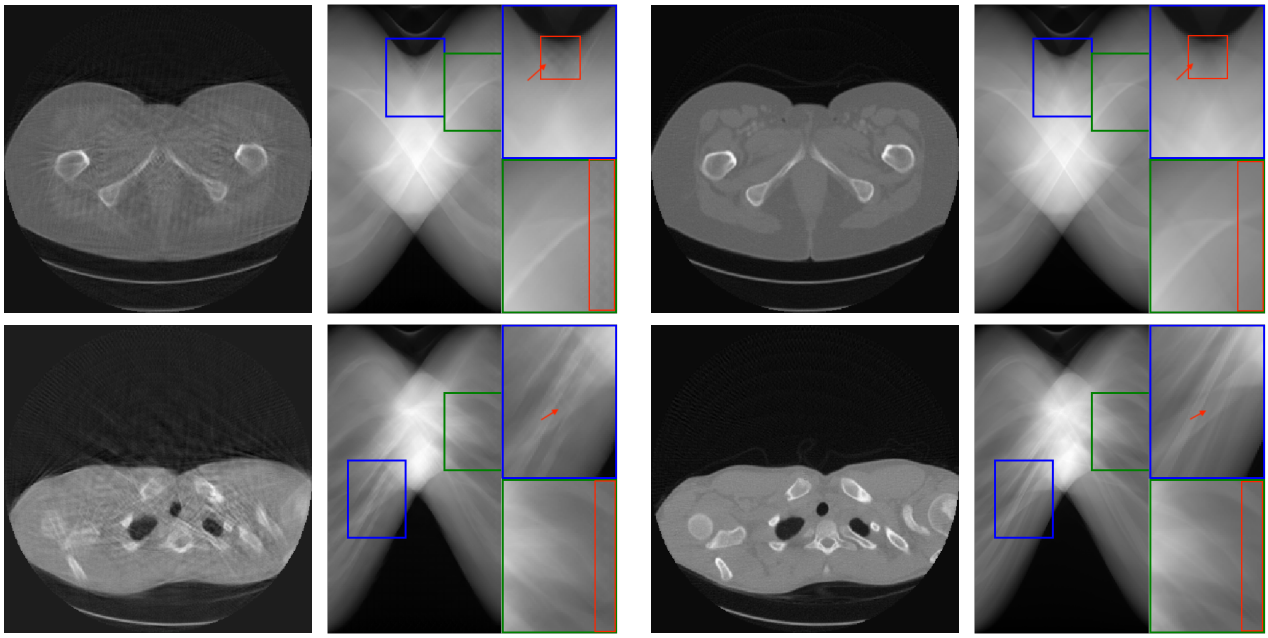


Figure 3: Checkerboard artifacts in the generated sinograms of the state-of-the-art learning-based approach [1] and consequent artifacts in their FBP reconstructions. **Left:** result sinograms and reconstructions. Checkerboard artifacts are especially severe in the boundary of sinograms. Please zoom in the sinogram to see more closely; **Right:** Ground Truth sinograms and reconstructions.