Generative Tomography Reconstruction

Matteo Ronchetti [1] Davide Bacciu [2]

[1]mttronchetti@gmail.com, Università di Pisa [2]bacciu@di.unipi.it, Università di Pisa

arxiv.org/abs/2010.14933

Introduction

- This work tackles reconstruction of Computed Tomography (CT) images in the presence of large amount of noise.
- Our goal is to infer an image x, given a noisy sinogram $y_\delta \sim n(Ax)$, where A is the discrete Radon transform and $n(\cdot)$ is the noise distribution.

Contributions

- We design an **end-to-end differentiable architecture** that directly maps a noisy sinogram to a denoised reconstruction.
- By **combining reconstruction and denoising**, we allow the model to mitigate the reconstruction errors caused by the ill-conditioning of the Radon transform.
- We train a generative model to **sample from the posterior** over tomographic images given a noisy sinogram.

Noise Model

We simulate noisy readings taking into consideration shot and electronic noise, and quantization errors as follows

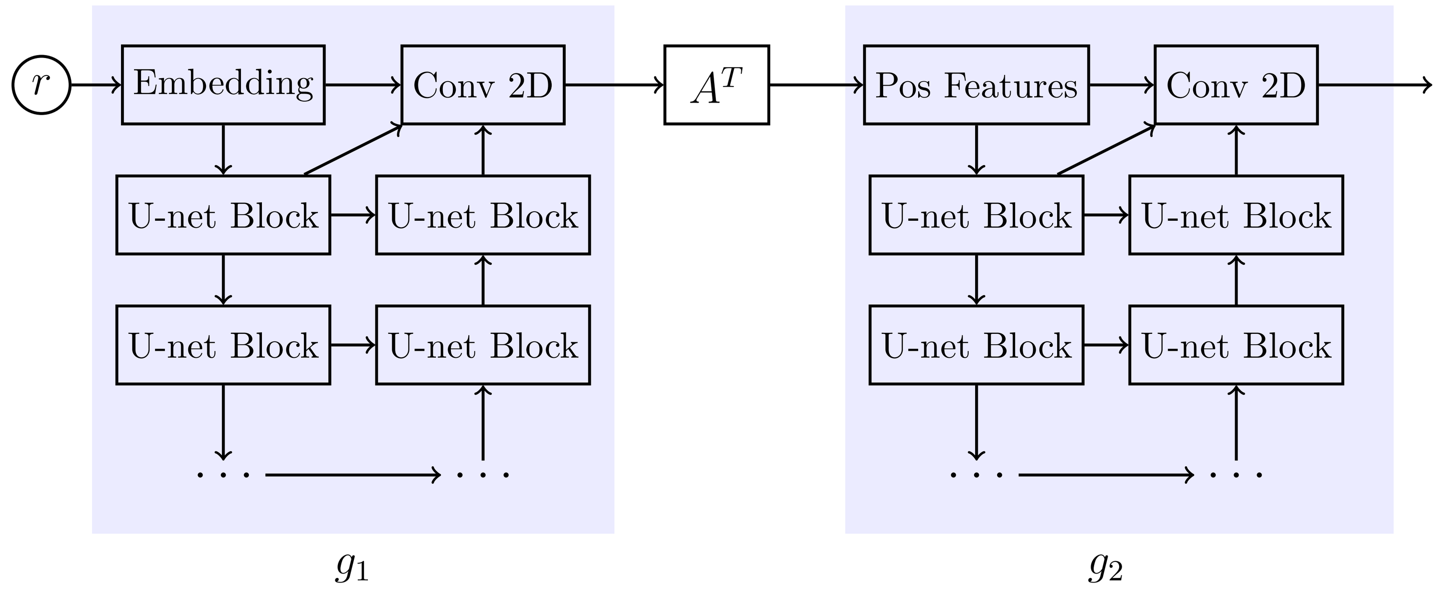
$$z \sim \text{Pois}(\exp(s-y)) + N(0,\epsilon)$$
$$r = \text{clamp}(\text{round}(z/k), 0, 2^b - 1)$$

where $\exp(s)$ is the X-ray intensity, b is the number of bits used by the detector and k is a scaling parameter.

We assume that, given readings r (that are integer values), sinograms are distributed as $y|r \sim \mathcal{N}(\mu(r), \text{diag}(\sigma^2(r)))$, where $\mu(\cdot)$ and $\sigma(\cdot)$ can be modelled by CNNs. We train $\mu(\cdot)$ and $\sigma(\cdot)$ networks to minimize the negative log-likelihood.

Reconstruction Model

Our model is composed of two U-nets[3] connected by Radon backprojection (A^T):



- The first U-net (g_1) denoises and filters the sinogram while the second (g_2) denoises the reconstruction.
- Sensor readings (r) are converted into floating point features using an embedding layer.
- The first layer of g_2 uses positional embeddings (which are concatenated to the input) to encode pixel coordinates.
- U-net blocks use strided convolutions for down-sampling and bilinear interpolation for up-sampling.

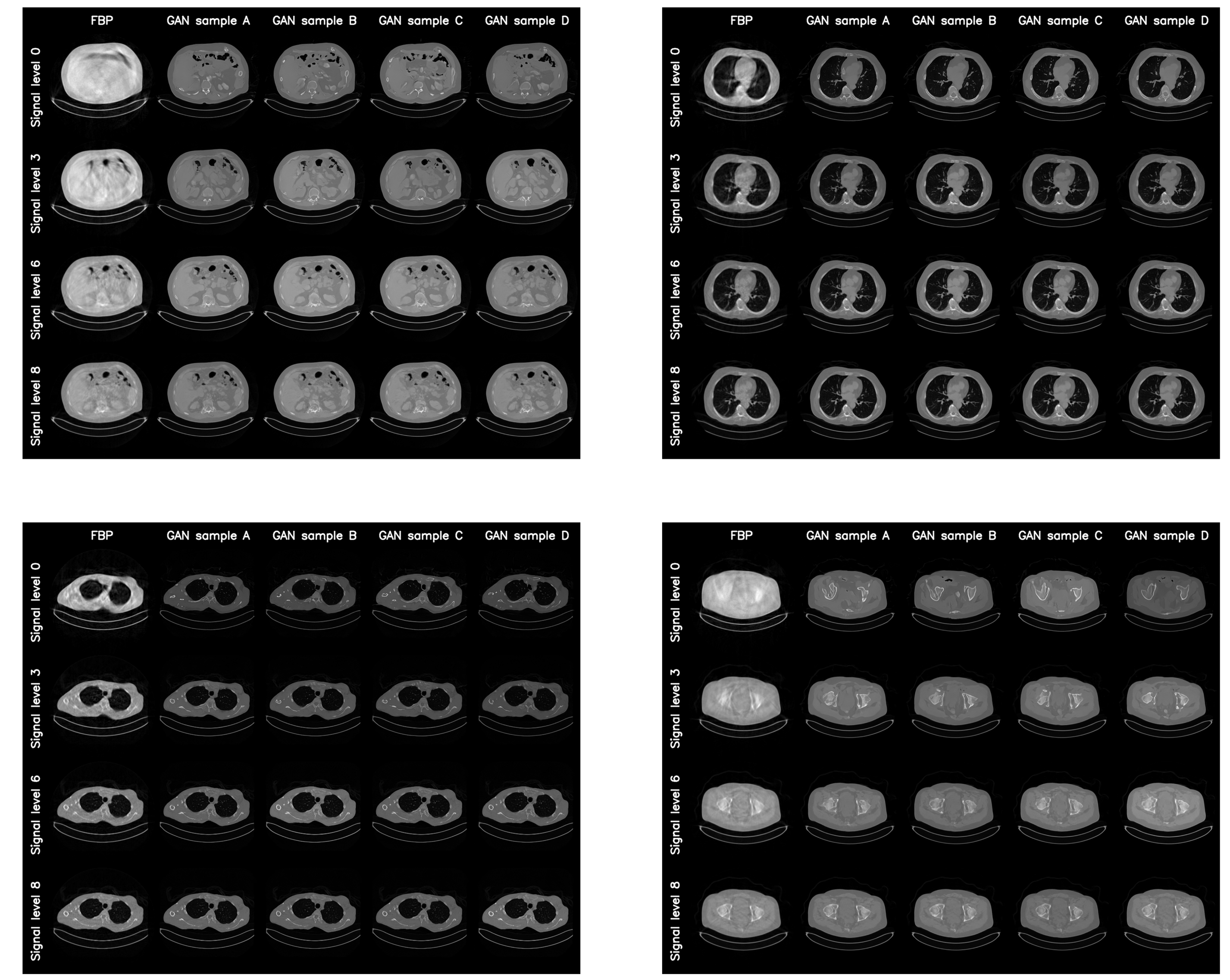


Figure 1:Reconstruction results using FBP compared with 4 samples generated by the proposed GAN. Notice that, as signal increases, GAN reconstructions become more similar to one another.

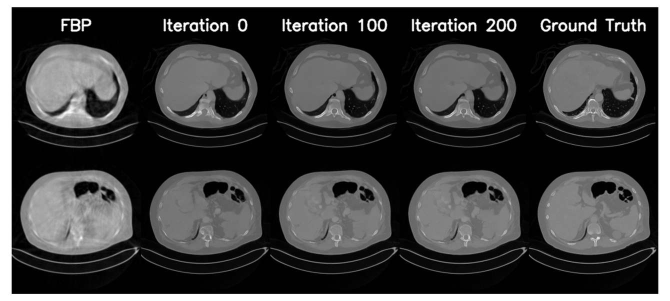


Figure 2:Progress of our iterative algorithm on two candidate reconstructions

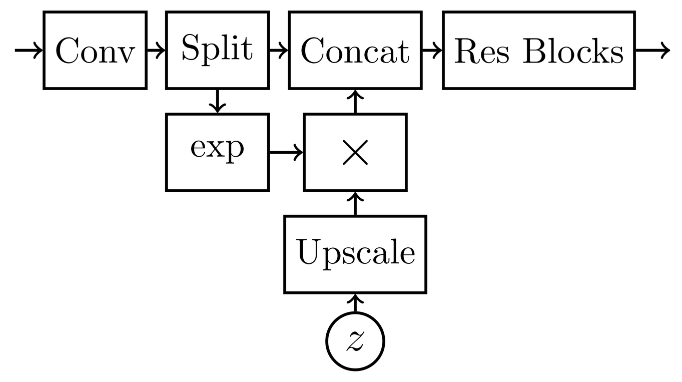


Figure 3:Structure of the modified U-net block used to insert z into the generator.

g_1	g_2	SSIM	Parameters	FPS
FBP	L	76.6%	$6.30 \cdot 10^6$	211
FBP	XL	76.7%	$9.37 \cdot 10^6$	178
FBP	XXL	76.9%	$1.56 \cdot 10^7$	121
XXS	S-64	76.7%	$2.63 \cdot 10^6$	239
XS	M	76.9%	$4.12 \cdot 10^6$	187
S	L	77.4%	$6.93 \cdot 10^6$	152

Figure 4:Comparison of denoising results using FBP based models (top) and end-to-end models (bottom).

References

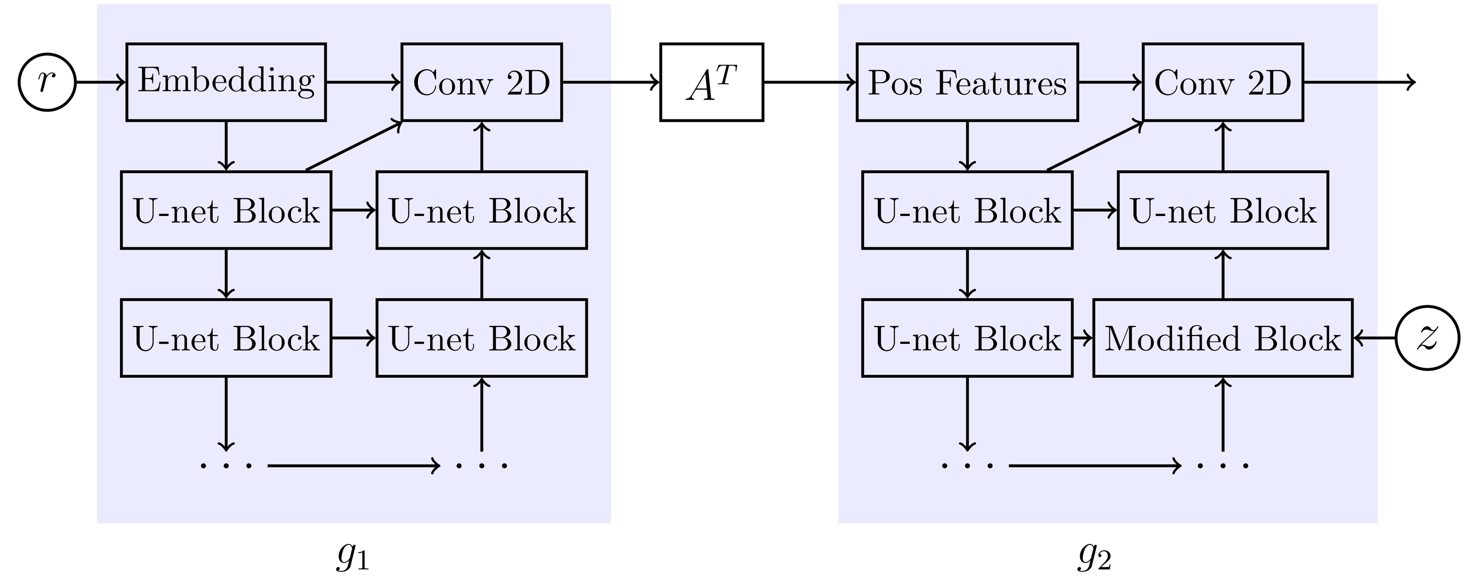
[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou.
Wasserstein generative adversarial networks.
In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 214–223. JMLR.org, 2017.

[2] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida.
Spectral normalization for generative adversarial networks.
In *International Conference on Learning Representations*, 2018.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox.
U-net: Convolutional networks for biomedical image segmentation.
In *International Conference on Medical image computing and computer-assisted intervention*, pages 234--241. Springer, 2015.

Generative Model

The generator $G(r,z)$, given sensor readings r, can sample different possible reconstructions by varying z. It is composed of two U-nets connected by Radon backprojection (A^T):



The input z is fed near the end of the model to make the computation of $\nabla_z G(r,z)$ efficient. The structure of the block that receives z is depicted in Figure 3.

Training

We build on the WGAN[1] framework and train G to minimize (z uniformly distributed over the sphere):

$$\mathbb{E}_{r,z}\left[\|AG(r,z) - \mu(r)\|^2_{\sigma(r)} + \lambda D(G(r,z))\right] \text{ where } \|v\|^2_{\sigma(r)} \overset{\text{def}}{=} \sum_i \left(\frac{v_i}{\sigma_i(r)}\right)^2 \quad (1)$$

Simultaneously, we train a discriminator D to minimize $\mathbb{E}_x[D(x)] - \mathbb{E}_{r,z}[D(G(r,z))]$ while enforcing $\|D\|_L \leq c$ using Spectral Normalization [2].

Avoiding mode collapse

To avoid mode collapse, we push the distribution of $AG(r,z)$ towards $\mathcal{N}(\mu(r), \text{diag}(\sigma^2(r)))$. We generate two different reconstructions for each noisy reading and project them into sinograms $y_1 = AG(r,z_1)$ and $y_2 = AG(r,z_2)$. Then, we fit a normal distribution with diagonal covariance and compare it against $\mathcal{N}(\mu(r), \text{diag}(\sigma^2(r)))$ using the Kullback–Leibler divergence:

$$\left\|\frac{y_1+y_2}{2} - \mu(r)\right\|^2_{\sigma(r)} + \sum_i \left[2\log\left(\frac{\sigma(r)}{\sigma_p}\right) + \left(\frac{\sigma_p}{\sigma(r)}\right)^2\right] \text{ where } \sigma_p^2 = \frac{(y_1-y_2)^2}{2}.$$

Using this in place of the first term of (1) solves mode collapse without reducing training stability.

Iterative improvement of reconstructions

The reconstruction produced by our generator can be iteratively improved using projected gradient to minimize $\|AG(r,z) - \mu(r)\|^2_{\sigma(r)} + \lambda D(G(r,z))$ subject to $\|z\| = 1$.

Results

Reconstruction quality of our model is compared against model based on FBP using structured similarity (SSIM) on the DeepLesion dataset. Table 4 reports the results of different model sizes together with inference speed measured in frames per second (FPS). From the table it can be noticed that end-to-end models achieve a more accurate reconstruction than FBP based models while being faster.