Learning to Reconstruct Medical Images

Jonas Adler[1, 2] Ozan Öktem[1]

[1] Department of Mathematics
KTH - Royal Institute of Technology, Stockholm

[2] Research and Physics
Elekta, Stockholm

Why do we need machine learning?

Task: Identify a rabbit in an image

Why do we need machine learning?

Task: Identify a rabbit in an image

Proposed solution: If the animal is within this range of colors and has long ears and fur and has a slightly elliptical shape and has a nose like... then it is a rabbit

Why do we need machine learning?

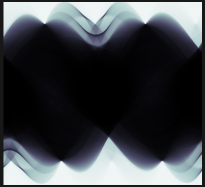
Task: Identify a rabbit in an image

Proposed solution: If the animal is within this range of colors and has long ears and fur and has a slightly elliptical shape and has a nose like... then it is a rabbit

Inverse Problems

$$g = \mathcal{T}(f_{\text{true}}) + \delta g.$$

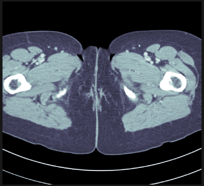
- $g \in Y$ *Data*
- $f_{\text{true}} \in X$ Image
- $\mathcal{T} : X \to Y$ Forward operator
- $\delta g \in Y$ Noise

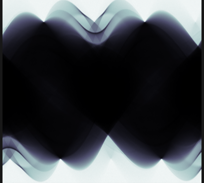


Inverse Problems

$$g = \mathcal{T}(f_{true}) + \delta g.$$

- $g \in Y$ Data
- $f_{true} \in X$ *Image*
- $\mathcal{T} : X \to Y$ Forward operator
- $\delta g \in Y$ Noise

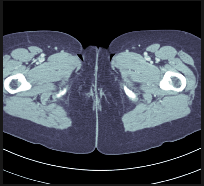




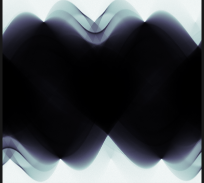
Inverse Problems

$$g = \mathcal{T}(f_{\text{true}}) + \delta g.$$

- $g \in Y$ Data
- $f_{\text{true}} \in X$ Image
- $\mathcal{T} : X \to Y$ *Forward operator*
- $\delta g \in Y$ Noise



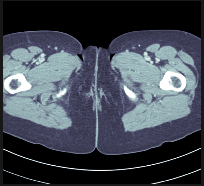
$$\xrightarrow{\mathcal{T}}$$



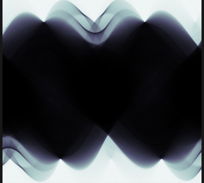
Inverse Problems

$$g = \mathcal{T}(f_{\text{true}}) + \delta g.$$

- $g \in Y$ — Data
- $f_{\text{true}} \in X$ — Image
- $\mathcal{T} : X \to Y$ — Forward operator
- $\delta g \in Y$ — *Noise*



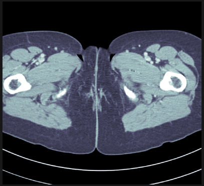
$\xrightarrow{\mathcal{T}}$



Inverse Problems

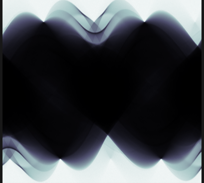
$$g = \mathcal{T}(f_{\text{true}}) + \delta g.$$

- $g \in Y$ Data
- $f_{\text{true}} \in X$ Image
- $\mathcal{T} : X \to Y$ Forward operator
- $\delta g \in Y$ Noise



$$\xrightarrow{\mathcal{T}}$$

$$\underset{\text{“}\mathcal{T}^{-1}\text{”}}{\longleftarrow}$$



Solution methods

- Analytic pseudoinverse (FBP, FDK)
  $f = \mathcal{T}^\dagger(g)$

Solution methods

- Analytic pseudoinverse (FBP, FDK)
  $f = \mathcal{T}^{\dagger}(g)$
- *Variational methods (TV, TGV, Huber)*
  $$f = \arg\min_{f} ||\mathcal{T}(f) - g||_Y^2 + \lambda||\nabla f||_1$$

Variational methods

- Strategy, solve an optimization problem:

$$f = \arg\min_f ||\mathcal{T}(f) - g||_Y^2 + \lambda ||\nabla f||_1$$

Several issues:

Variational methods

- Strategy, solve an optimization problem:

$$f = \arg\min_f ||\mathcal{T}(f) - g||_Y^2 + \lambda||\nabla f||_1$$

Several issues:

- Prior is typically unknown - have to "guess"

Variational methods

- Strategy, solve an optimization problem:

$$f = \arg\min_{f} ||\mathcal{T}(f) - g||_Y^2 + \lambda ||\nabla f||_1$$

Several issues:

- Prior is typically unknown - have to "guess"
- Parameters (λ) need to be selected

Variational methods

- Strategy, solve an optimization problem:

$$f = \arg\min_{f} ||\mathcal{T}(f) - g||_Y^2 + \lambda||\nabla f||_1$$

Several issues:

- Prior is typically unknown - have to "guess"
- Parameters (λ) need to be selected
- Large computational burden

Solution methods

- Analytic pseudoinverse (FBP, FDK)
  $f = \mathcal{T}^{\dagger}(g)$
- Variational methods (TV, TGV, Huber)
  $f = \arg\min_{f} ||\mathcal{T}(f) - g||_Y^2 + \lambda||\nabla f||_1$
- *Machine learning*
  $f = \mathcal{T}_{\theta}^{\dagger}(g)$

Supervised learning

- We are given training data (f, g) which is a $X \times Y$ valued random variable such that $\mathcal{T}(\mathrm{f}) \approx \mathrm{g}$.

Supervised learning

- We are given training data (f, g) which is a $X \times Y$ valued random variable such that $\mathcal{T}(\mathrm{f}) \approx \mathrm{g}$.
- We give a class of operators $\mathcal{T}^\dagger_\theta \colon Y \to X$

Supervised learning

- We are given training data (f, g) which is a $X \times Y$ valued random variable such that $\mathcal{T}(\mathrm{f}) \approx \mathrm{g}$.
- We give a class of operators $\mathcal{T}_\theta^\dagger \colon Y \to X$
- Parametrized by θ which we *learn*

Supervised learning

- We are given training data (f, g) which is a $X \times Y$ valued random variable such that $\mathcal{T}(\mathrm{f}) \approx \mathrm{g}$.
- We give a class of operators $\mathcal{T}_\theta^\dagger \colon Y \to X$
- Parametrized by θ which we learn
- Selected by optimization of a *loss* function $\mathrm{L}(\theta)$

$$\theta^* = \arg\min_\theta \mathrm{L}(\theta)$$

Supervised learning

- We are given training data (f, g) which is a $X \times Y$ valued random variable such that $\mathcal{T}(\mathrm{f}) \approx \mathrm{g}$.
- We give a class of operators $\mathcal{T}_\theta^\dagger \colon Y \to X$
- Parametrized by θ which we learn
- Selected by optimization of a *loss* function $\mathrm{L}(\theta)$

$$\theta^* = \arg\min_\theta \mathrm{L}(\theta)$$

- *Different from classification ($X \to \mathbb{R}^n$) and image processing ($X \to X$)*

Learned inversion methods

- *Fully learned*
- Learned post-processing
- Learned iterative schemes

Fully learned reconstruction

Goal: Learn ”the whole” mapping from data to signal

Tomographic image reconstruction based on artificial neural network (ANN) techniques
Argyrou et. al. NSS/MIC 2012

Tomographic image reconstruction using artificial neural networks.
Paschalis et. al. Nucl Instrum Methods Phys Res A 2004

Image reconstruction by domain-transform manifold learning.
Zhu et. al. Nature 2018

Fully learned reconstruction

Goal: Learn ”the whole” mapping from data to signal

- *Tomographic image reconstruction based on artificial neural network (ANN) techniques*
  Argyrou et. al. NSS/MIC 2012
- *Tomographic image reconstruction using artificial neural networks.*
  Paschalis et. al. Nucl Instrum Methods Phys Res A 2004
- *Image reconstruction by domain-transform manifold learning.*
  Zhu et. al. Nature 2018

Problem: $\mathcal{T}$ typically has symmetries, but the network has to learn them.
Example: 3D CBCT, data: 10^8 pixels and 10^8 voxels $\implies$ 10^{16} connections!

Learned inversion methods

- Fully learned
- *Learned post-processing*
- Learned iterative schemes

Learned post-processing

Use deep learning to improve the result of another reconstruction

$$\mathcal{T}_\theta^\dagger = \Lambda_\theta \circ \mathcal{T}^\dagger$$

where $\mathcal{T}^\dagger$ is some reconstruction (FBP, TV, ...) and Λ_θ is a learned post-processing operator.

Learned post-processing

Use deep learning to improve the result of another reconstruction

$$\mathcal{T}_\theta^\dagger = \Lambda_\theta \circ \mathcal{T}^\dagger$$

where $\mathcal{T}^\dagger$ is some reconstruction (FBP, TV, ...) and Λ_θ is a learned post-processing operator.

Allows *separation of inversion and learning*, data can be seen as $(\underbrace{\mathcal{T}^\dagger(\mathrm{g})}_{\in X}, \underbrace{\ \mathrm{f}\ }_{\in X})$.

The problem becomes an image processing problem $\implies$ easy to solve.

Learned post-processing

Denoise in transform domain (Fourier, Wavelet, Shearlet, etc)

Won AAPM Low-Dose CT Grand Challenge:

A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction
Kang et. al. 2016

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g
- $I_{total} = I_{prior} + I(g)$

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g
- $I_{total} = I_{prior} + I(g)$
- For post processing, we only have $I(\mathcal{T}^{\dagger}(g))$

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g
- $I_{total} = I_{prior} + I(g)$
- For post processing, we only have $I(\mathcal{T}^\dagger(g))$
- $I(\mathcal{T}^\dagger(g)) \leq I(g)$

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g
- $I_{total} = I_{prior} + I(g)$
- For post processing, we only have $I(\mathcal{T}^\dagger(g))$
- $I(\mathcal{T}^\dagger(g)) \leq I(g)$
- Post processing beats traditional methods because it utilizes the prior information better

Information theoretic bottleneck

- $I(g) \leftarrow$ information in g
- $I_{total} = I_{prior} + I(g)$
- For post processing, we only have $I(\mathcal{T}^\dagger(g))$
- $I(\mathcal{T}^\dagger(g)) \leq I(g)$
- Post processing beats traditional methods because it utilizes the prior information better

We could do better by using the raw data!

Learned inversion methods

- Fully learned
- Learned post-processing
- *Learned iterative schemes*

Learned iterative reconstruction

- Problem: Data $g \in Y$, reconstruction $f \in X$
  How to include data in each iteration?

Learned iterative reconstruction

- Problem: Data $g \in Y$, reconstruction $f \in X$
  How to include data in each iteration?
- Inspiration from iterative optimization methods

$$f = \arg\min \frac{1}{2} || \mathcal{T}(f) - g ||_Y^2$$

Algorithm 1 Generic gradient based optimization algorithm

1: **for** $i = 1, \ldots$ **do**
2: $\quad f_{i+1} \leftarrow \text{Update}\big(f_i, \mathcal{T}^*(\mathcal{T}(f_i) - g)\big)$

Gradient descent:

$$\text{Update}\big(f_i, \mathcal{T}^*(\mathcal{T}(f_i) - g)\big) = f_i - \alpha \mathcal{T}^*(\mathcal{T}(f_i) - g)$$

Learned gradient descent

- Set a stopping criteria (fixed number of steps)
- Learn the function Update $= \Lambda_\theta$

Learned gradient descent

- Set a stopping criteria (fixed number of steps)
- Learn the function Update $= \Lambda_\theta$

Algorithm 2 Learned gradient descent

1: **for** $i = 1, \dots, I$ **do**
2: $\quad f_{i+1} \leftarrow \Lambda_\theta\big(f_i, \mathcal{T}^*(\mathcal{T}(f_i) - g)\big)$
3: $\mathcal{T}_\theta^\dagger(g) \leftarrow f_I$

Learned gradient descent

- Set a stopping criteria (fixed number of steps)
- Learn the function Update $= \Lambda_\theta$

Algorithm 2 Learned gradient descent

1: **for** $i = 1, \dots, I$ **do**
2: $\quad f_{i+1} \leftarrow \Lambda_\theta\big(f_i, \mathcal{T}^*(\mathcal{T}(f_i) - g)\big)$
3: $\mathcal{T}_\theta^\dagger(g) \leftarrow f_I$

We separate problem dependent (and possibly global) components into $\mathcal{T}^*(\mathcal{T}(f_i) - g)$, and local into Λ_θ!

Learned Primal-Dual

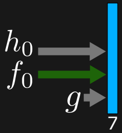
Algorithm 3 Learned Primal-Dual (conceptual)

1: **for** $i = 1, \dots, I$ **do**
2: $\quad h_i \leftarrow \Gamma_{\theta_i^d}\big(h_{i-1}, \mathcal{T}(f_{i-1}), g\big)$
3: $\quad f_i \leftarrow \Lambda_{\theta_i^p}\big(f_{i-1}, \mathcal{T}^*(h_i)\big)$
4: $\mathcal{T}_\theta^\dagger(g) \leftarrow f_I$

Learned Primal-Dual

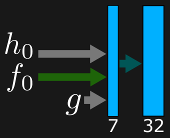
- apply $\mathcal{T}$
- apply $\mathcal{T}^*$
- copy
- 3x3 conv + PReLU
- 3x3 conv

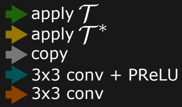
Learned Primal-Dual



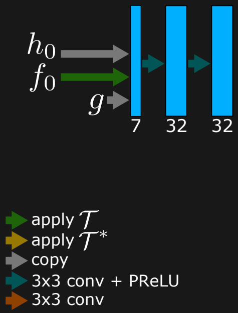
apply $\mathcal{T}$
apply $\mathcal{T}^*$
copy
3x3 conv + PReLU
3x3 conv

Learned Primal-Dual

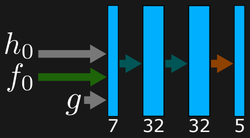


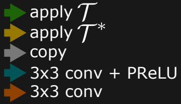


Learned Primal-Dual

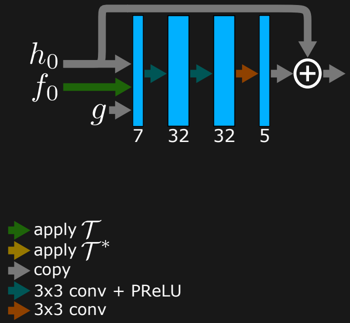


Learned Primal-Dual

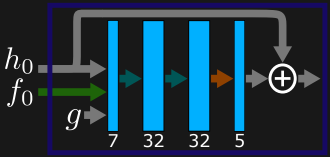


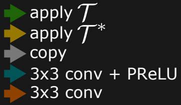


Learned Primal-Dual

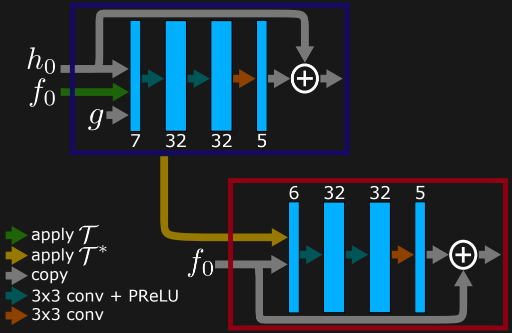


Learned Primal-Dual

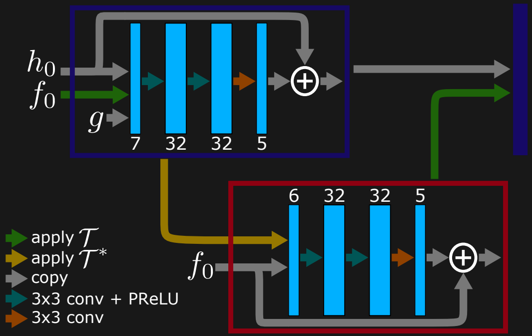




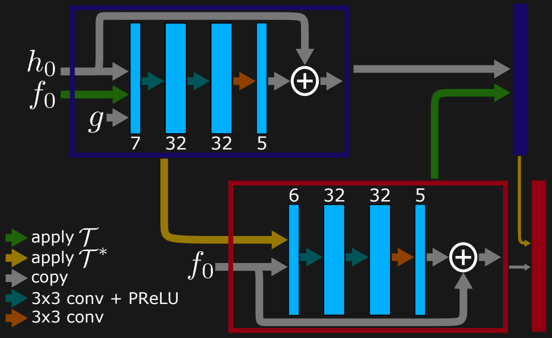
Learned Primal-Dual



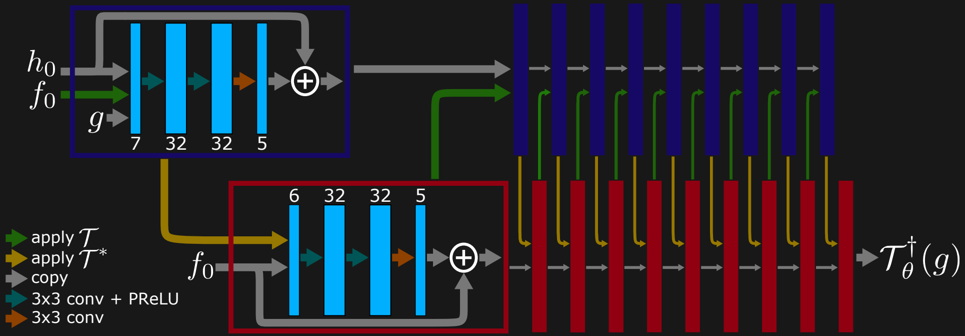
Learned Primal-Dual



Learned Primal-Dual



Learned Primal-Dual



References

- *ADMM-Net: A Deep Learning Approach for Compressive Sensing MRI*
  Yang et. al. NIPS 2016
- *Recurrent inference machines for solving inverse problems*
  Putzky and Welling, arXiv 2017
- *Learning a Variational Network for Reconstruction of Accelerated MRI Data*
  Hammernick et. al., arXiv 2017
- *Solving ill-posed inverse problems using iterative deep neural networks*
  Adler and Öktem, Inverse Problems 2017
- *Learned Primal-Dual Reconstruction*
  Adler and Öktem, IEEE TMI 2018

References

- *ADMM-Net: A Deep Learning Approach for Compressive Sensing MRI*
  Yang et. al. NIPS 2016
- *Recurrent inference machines for solving inverse problems*
  Putzky and Welling, arXiv 2017
- *Learning a Variational Network for Reconstruction of Accelerated MRI Data*
  Hammernick et. al., arXiv 2017
- *Solving ill-posed inverse problems using iterative deep neural networks*
  Adler and Öktem, Inverse Problems 2017
- *Learned Primal-Dual Reconstruction*
  Adler and Öktem, IEEE TMI 2018

References

- *ADMM-Net: A Deep Learning Approach for Compressive Sensing MRI*
  Yang et. al. NIPS 2016
- *Recurrent inference machines for solving inverse problems*
  Putzky and Welling, arXiv 2017
- *Learning a Variational Network for Reconstruction of Accelerated MRI Data*
  Hammernick et. al., arXiv 2017
- *Solving ill-posed inverse problems using iterative deep neural networks*
  Adler and Öktem, Inverse Problems 2017
- *Learned Primal-Dual Reconstruction*
  Adler and Öktem, IEEE TMI 2018

Results

Results for CT with *Human data*

- Inverse problem:
$$g = \mathcal{P}(f) + \delta g$$
- Geometry: fan beam 1000 angles
- Noise: Poisson noise (low dose CT)
- Training data: 2000 512×512 pixel slices

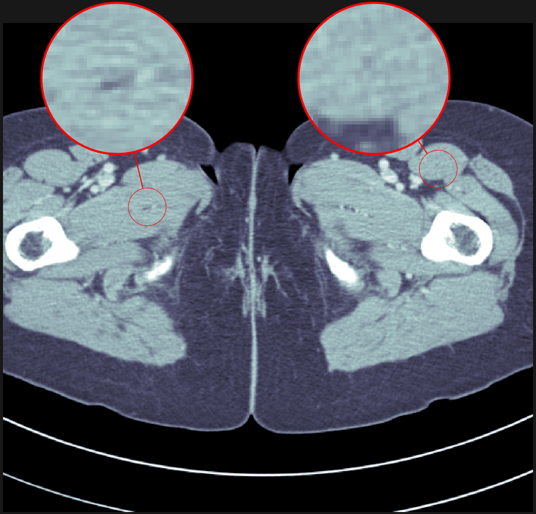
Results

Results for CT with *Human data*

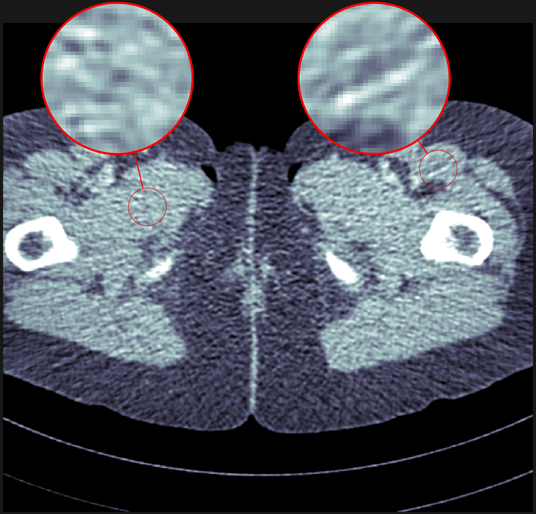
- Inverse problem:
$$g = \mathcal{P}(f) + \delta g$$
- Geometry: fan beam 1000 angles
- Noise: Poisson noise (low dose CT)
- Training data: 2000 512×512 pixel slices

Compare to:

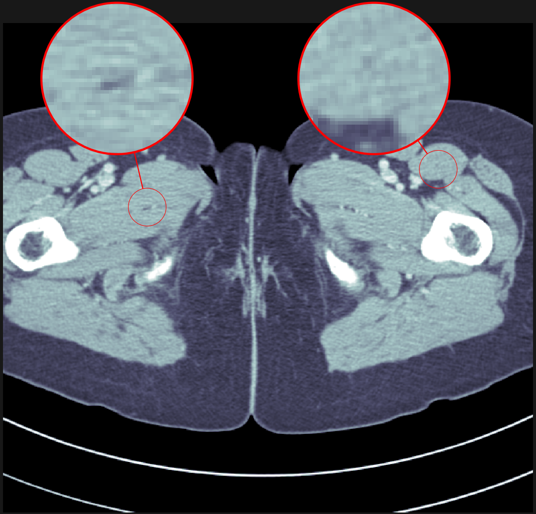
- Analytic Pseudo-Inverse (FBP)
- Variational methods (TV-regularization)
- Post-processing deep learning by U-Net



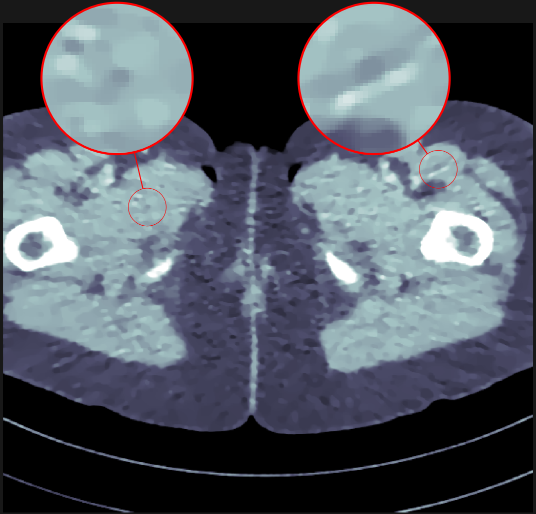
Phantom



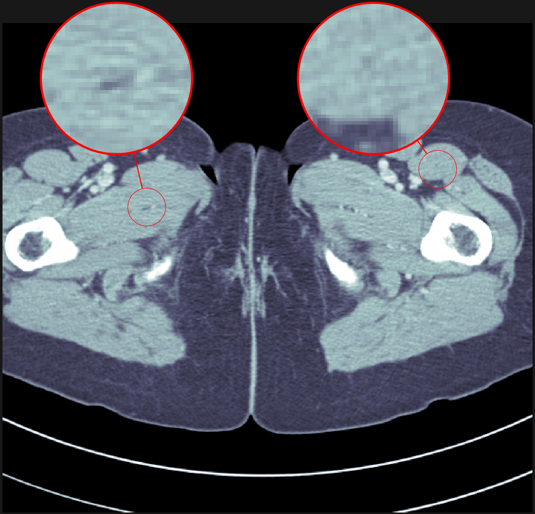
FBP
PSNR 33.65 dB, SSIM 0.830, 423 ms



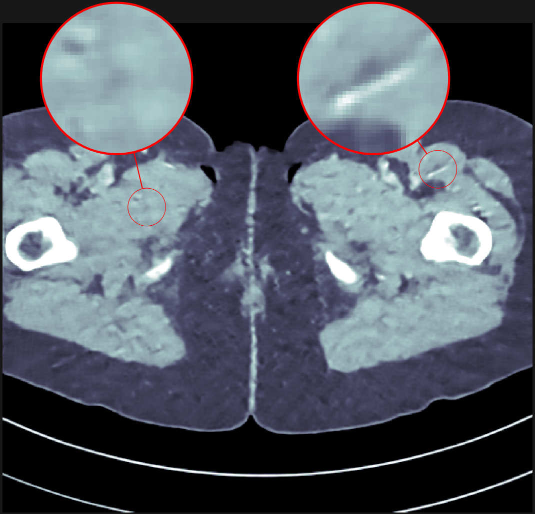
Phantom



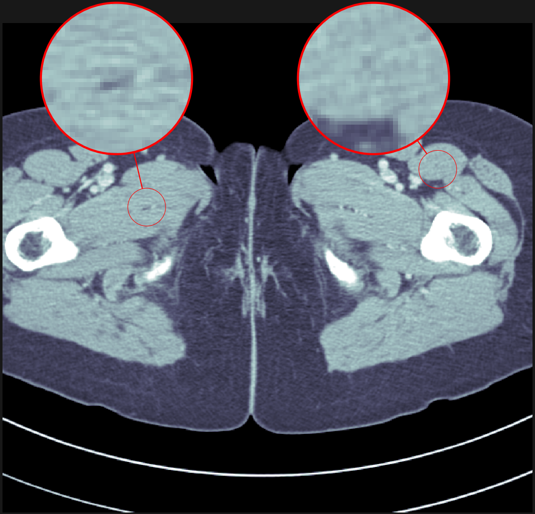
TV
PSNR 37.48 dB, SSIM 0.946, $64\,371$ ms



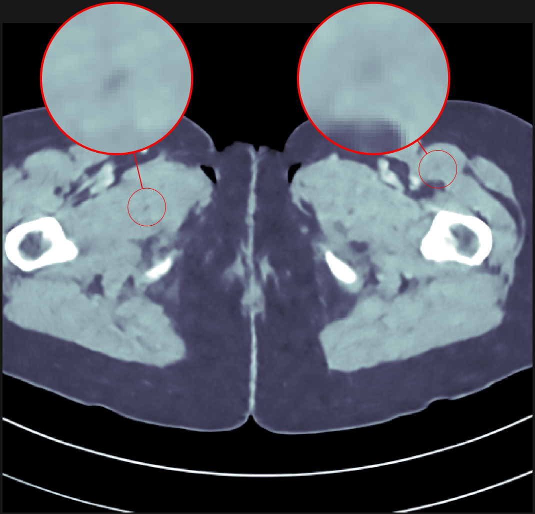
Phantom



FBP + U-Net denoising
PSNR 41.92 dB, SSIM 0.941, 463 ms



Phantom



Learned Primal-Dual
PSNR 44.11 dB, SSIM 0.969, 620 ms

Conclusions

- Machine learning allows us to handle complicated priors

Source:
`github.com/adler-j`
Contact:
`jonasadl@kth.se`

Conclusions

- Machine learning allows us to handle complicated priors
- Fully learned reconstruction is in-feasible

Source:
`github.com/adler-j`
Contact:
`jonasadl@kth.se`

Conclusions

- Machine learning allows us to handle complicated priors
- Fully learned reconstruction is in-feasible
- Learned post-processing gives good results

Source:
`github.com/adler-j`

Contact:
`jonasadl@kth.se`

Conclusions

- Machine learning allows us to handle complicated priors
- Fully learned reconstruction is in-feasible
- Learned post-processing gives good results
- Learned iterative reconstruction gives better results

Source:
`github.com/adler-j`
Contact:
`jonasadl@kth.se`

Conclusions

- Machine learning allows us to handle complicated priors
- Fully learned reconstruction is in-feasible
- Learned post-processing gives good results
- Learned iterative reconstruction gives better results

Questions!

Source:
`github.com/adler-j`
Contact:
`jonasadl@kth.se`