Deep prior networks for inverse problems with applications to Computed Tomography and Magnetic Particle Imaging

Daniel Otero Baguer, Sören Dittmer, Tobias Kluth, Peter Maaß

Center for Industrial Mathematics (ZeTeM)
University of Bremen

12.07.2019
Applied Inverse Problems Conference
Grenoble, France

Universität Bremen

Some advertising...

Autumn School: Deep Learning and Inverse Problems

- When?
  $\Longrightarrow$ November, 4th-8th
- Where?
  $\Longrightarrow$ In Bremen, Germany
- Registration deadline?
  $\Longrightarrow$ August, 15^{th}

Outline

Section 1

Introduction

Preliminaries

Consider an operator $A : X \to Y$ between Hilbert spaces X and Y

Inverse Problem (General task)

Given measured noisy data

$$y^\delta = Ax^\dagger + \tau, \tag{1}$$

obtain an approximation $\hat{x}$ for $x^\dagger$, where τ, with $\|\tau\| \leq \delta$, describes the noise in the measurement

Preliminaries

Classical approach: Variational regularization

$$\hat{x}_\alpha = \arg\min \frac{1}{2}\|Ax - y^\delta\|^2 + \alpha\mathcal{R}(x) \qquad (2)$$

Examples of hand-crafted priors:

- $\|x\|^2$
- $\|x\|_1$
- $TV(x)$

Remark: α selection

Preliminaries

Classical approach: Variational regularization

$$\hat{x}_\alpha = \arg\min \frac{1}{2}\|Ax - y^\delta\|^2 + \alpha\mathcal{R}(x) \quad (2)$$

Examples of hand-crafted priors:

- $\|x\|^2$
- $\|x\|_1$
- $TV(x)$

Remark: α selection

Deep learning for inverse problems

- Learned gradient descent[1]
- Learned post-processing: $\mathcal{F}_\Theta \circ A^\dagger$
- Learned regularizer[2,3]: $\mathcal{R}_\Theta$
- Learned primal-dual[4]
- Generative network: $\varphi_\Theta(z)$ (e.g. GAN, VAE, ...)

---

[1] Andreas Hauptmann, Felix Lucka, Marta Betcke, Nam Huynh, Jonas Adler, Ben Cox, Paul Beard, Sebastien Ourselin, and Simon Arridge. "Model-Based Learning for Accelerated, Limited-View 3-D Photoacoustic Tomography". In: *IEEE transactions on medical imaging* 37.6 (2018), pp. 1382–1393.

[2] Sebastian Lunz, Ozan Öktem, and Carola-Bibiane Schönlieb. "Adversarial Regularizers in Inverse Problems". In: *arXiv preprint arXiv:1805.11572* (2018).

[3] Housen Li, Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. "NETT: Solving Inverse Problems with Deep Neural Networks". In: *arXiv preprint arXiv:1803.00092* (Feb. 2018).

[4] Jonas Adler and Ozan Öktem. "Learned primal-dual reconstruction". In: *IEEE transactions on medical imaging* 37.6 (2018), pp. 1322–1332.

Closer look: Generative networks approach

Consider a generative network $\varphi_\Theta(z)$ previously trained

- Θ is fixed after the training phase
- We can obtain images by sampling $\mathbf{z}$

Usual approach (e.g.[5]):

$$\hat{z} = \arg\min_z \frac{1}{2}\|A\varphi_\Theta(z) - y^\delta\|^2 \tag{3}$$

$$\hat{x} = \varphi_\Theta(\hat{z}) \tag{4}$$

[5] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G. Dimakis. "Compressed Sensing using Generative Models". In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. 2017, pp. 537–546.

Drawbacks

- Need a lot of data
- How to get the ground-truths?
- Real data noise might be different from the one present on the training samples

Is it possible to solve inverse problems using neural networks without any training data?

Drawbacks

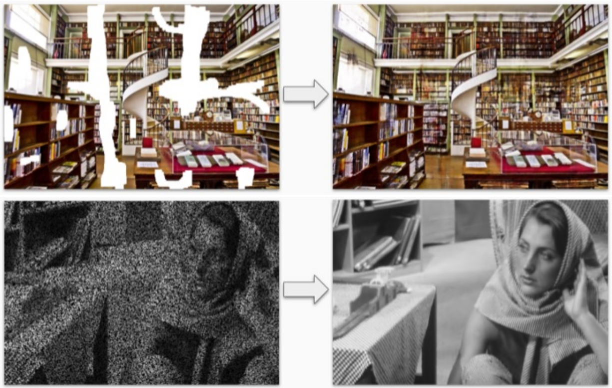
- Need a lot of data
- How to get the ground-truths?
- Real data noise might be different from the one present on the training samples

Is it possible to solve inverse problems using neural networks without any training data?

Section 2

Deep Image Prior

Examples [6]



[6] https://dmitryulyanov.github.io/deep_image_prior

Basic Idea[7]

Given measured noisy data

$$y^\delta = Ax^\dagger + \tau \tag{5}$$

1. Optimize a neural network $\varphi_\Theta(z_0)$ with a fixed input z_0

$$\hat{\Theta} = \arg\min_{\Theta} \frac{1}{2} \|A\varphi_\Theta(z_0) - y^\delta\|^2 \tag{6}$$

2. Set $\hat{x} = \varphi_{\hat{\Theta}}(z_0)$ as the reconstruction

[7] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. "Deep Image Prior". In: *CoRR* (2017). arXiv: 1711.10925.

Basic Idea[7]

Given measured noisy data

$$y^\delta = Ax^\dagger + \tau \tag{5}$$

1. Optimize a neural network $\varphi_\Theta(z_0)$ with a fixed input z_0

$$\hat{\Theta} = \arg\min_\Theta \frac{1}{2} \|A\varphi_\Theta(z_0) - y^\delta\|^2 \tag{6}$$

2. Set $\hat{x} = \varphi_{\hat{\Theta}}(z_0)$ as the reconstruction

[7] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. “Deep Image Prior”. In: *CoRR* (2017). arXiv: 1711.10925.

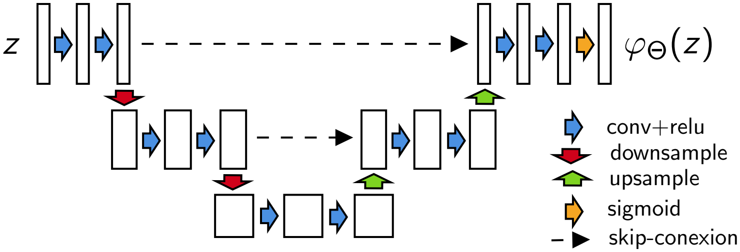
Some insights

- The network φ_Θ has a U-Net-like architecture
- It has enough expressive power to reproduce some noise
- Optimization method (ADAM[8]) with early stopping plays an important role
- Solving each instance requires training the network
- It takes a lot of time

[8] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

Task dependent hyper-parameters

- U-Net-like architecture
  - Number of scales (e.g. 2, 3, 4, 5, 6, . . .)
  - Filter size per scale (e.g. 3, 5, . . .)
  - Number of filters per scale (e.g. 8, 16, 32, 64, 128, . . .)
  - Number of filters per skip connection (e.g. 2, 4, . . .)



Task dependent hyper-parameters

- U-Net-like architecture
  - Number of scales (e.g. $2,\ 3,\ 4,\ 5,\ 6,\dots$)
  - Filter size per scale (e.g. $3,\ 5,\dots$)
  - Number of filters per scale (e.g. $8,\ 16,\ 32,\ 64,\ 128,\dots$)
  - Number of filters per skip connection (e.g. $2,\ 4,\dots$)
- Training
  - Number of iterations
  - Learning rate (e.g. $10^{-1}, 10^{-2}, 10^{-3},\dots$)
  - Variance of regularization noise (e.g. $0,\ 10^{-2},\ 2\cdot 10^{-2},\ 3\cdot 10^{-2}\dots$)

DIP with training data

Can we improve DIP with a *small*/*huge* data-set?

DIP with training data

Given K training images, compute optimal weights $\{\Theta_1, \Theta_2, \ldots, \Theta_K\}$, with $\Theta_i \in \mathbb{R}^d$

Compute:

- Co-variance matrix $\Sigma \in \mathbb{R}^{d \times d}$
- Mean vector $\mu \in \mathbb{R}^d$

Minimize:[9]

$$\min_{\Theta} \|A\varphi_{\Theta}(z_0) - y^{\delta}\|^2 + \alpha(\Theta - \mu)^T \Sigma^{-1}(\Theta - \mu) \tag{7}$$

[9] David Van Veen, Ajil Jalal, Eric Price, Sriram Vishwanath, and Alexandros G Dimakis. "Compressed Sensing with Deep Image Prior and Learned Regularization". In: *arXiv preprint arXiv:1806.06438* (2018).

Section 3

Analytic Deep Prior

Simple architecture

Can the DIP approach be used to solve ill-posed inverse problems?

Consider a trivial network $\varphi_\Theta(z) = \Theta$

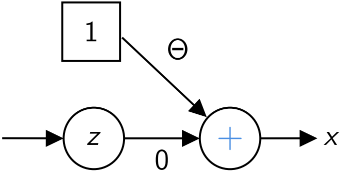
$\implies$ Minimizing $\|A\varphi_\Theta(z) - y^\delta\|^2 = \|A\Theta - y^\delta\|^2$ by gradient descent with respect to Θ is equivalent to the classical Landweber iteration

$$\alpha \sim \frac{1}{n} \tag{8}$$

Simple architecture

Can the DIP approach be used to solve ill-posed inverse problems?

Consider a trivial network $\varphi_\Theta(z) = \Theta$



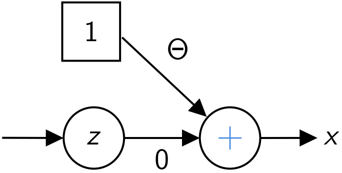
$\implies$ Minimizing $\|A\varphi_\Theta(z) - y^\delta\|^2 = \|A\Theta - y^\delta\|^2$ by gradient descent with respect to Θ is equivalent to the classical Landweber iteration

$$\alpha \sim \frac{1}{n} \tag{8}$$

Simple architecture

Can the DIP approach be used to solve ill-posed inverse problems?

Consider a trivial network $\varphi_\Theta(z) = \Theta$



$\Longrightarrow$ Minimizing $\|A\varphi_\Theta(z) - y^\delta\|^2 = \|A\Theta - y^\delta\|^2$ by gradient descent with respect to Θ is equivalent to the classical Landweber iteration

$$\alpha \sim \frac{1}{n} \tag{8}$$

Unrolled proximal gradient architecture

Consider a fully connected feed-forward network with L layers

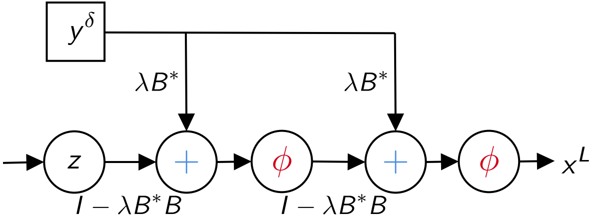
$$\varphi_\Theta(z) = x^L, \tag{9}$$

where

$$x^{k+1} = \phi\left(Wx^k + b\right) \tag{10}$$

- The affine linear map $\Theta = (W, b)$ is the same for all layers
- The matrix W is restricted to obey $I - W = \lambda B^* B$ for any B and the bias is determined via $b = \lambda B^* y^\delta$
- The activation function of the network is chosen as the proximal mapping of a regularizing functional $\lambda\alpha R : X \to \mathbb{R}$

Unrolled proximal gradient architecture



$\Longrightarrow$ $\varphi_\Theta(z) = x^L$ is identical to the L-th iterate of a proximal gradient descent method for minimizing

$$J_B(x) = \frac{1}{2}\|Bx - y^\delta\|^2 + \alpha R(x) \quad (11)$$

Deep priors and Tikhonov functionals

Given: measured data $y^\delta \in Y$, fixed $\alpha > 0$, convex penalty functional $R : X \to \mathbb{R}$ and the operator $A \in \mathcal{L}(X, Y)$

Solve:

$$\hat{B} = \underset{B \in \mathcal{L}(X,Y)}{\arg\min} \underbrace{\frac{1}{2}\|Ax(B) - y^\delta\|^2}_{F(B)} \quad (12)$$

subject to

$$x(B) = \underset{x \in X}{\arg\min} \frac{1}{2}\|Bx - y^\delta\|^2 + \alpha R(x) \quad (13)$$

Result: $x(\hat{B})$ as the solution to the inverse problem

$\Longrightarrow$ **Analytic Deep Prior**

Deep priors and Tikhonov functionals

Given: measured data $y^\delta \in Y$, fixed $\alpha > 0$, convex penalty functional $R : X \to \mathbb{R}$ and the operator $A \in \mathcal{L}(X, Y)$

Solve:

$$\hat{B} = \operatorname*{arg\,min}_{B \in \mathcal{L}(X,Y)} \underbrace{\frac{1}{2}\|Ax(B) - y^\delta\|^2}_{F(B)} \tag{12}$$

subject to

$$x(B) = \operatorname*{arg\,min}_{x \in X} \frac{1}{2}\|Bx - y^\delta\|^2 + \alpha R(x) \tag{13}$$

Result: $x(\hat{B})$ as the solution to the inverse problem

$\Longrightarrow$ **Analytic Deep Prior**

Deep priors and Tikhonov functionals

Given: measured data $y^\delta \in Y$, fixed $\alpha > 0$, convex penalty functional $R : X \to \mathbb{R}$ and the operator $A \in \mathcal{L}(X, Y)$

Solve:

$$\hat{B} = \operatorname*{arg\,min}_{B \in \mathcal{L}(X,Y)} \underbrace{\frac{1}{2} \|Ax(B) - y^\delta\|^2}_{F(B)} \tag{12}$$

subject to

$$x(B) = \operatorname*{arg\,min}_{x \in X} \frac{1}{2} \|Bx - y^\delta\|^2 + \alpha R(x) \tag{13}$$

Result: $x(\hat{B})$ as the solution to the inverse problem

$\Longrightarrow$ **Analytic Deep Prior**

Deep priors and Tikhonov functionals

Given: measured data $y^\delta \in Y$, fixed $\alpha > 0$, convex penalty functional $R : X \to \mathbb{R}$ and the operator $A \in \mathcal{L}(X, Y)$

Solve:

$$\hat{B} = \operatorname*{arg\,min}_{B \in \mathcal{L}(X,Y)} \underbrace{\frac{1}{2}\|Ax(B) - y^\delta\|^2}_{F(B)} \tag{12}$$

subject to

$$x(B) = \operatorname*{arg\,min}_{x \in X} \frac{1}{2}\|Bx - y^\delta\|^2 + \alpha R(x) \tag{13}$$

Result: $x(\hat{B})$ as the solution to the inverse problem

$\Longrightarrow$ **Analytic Deep Prior**

Deep priors and Tikhonov functionals

Theorem

Let

$$\psi(x, B) = \operatorname*{Prox}_{\lambda\alpha R}\left(x - \lambda B^*(Bx - y^\delta)\right) - x \tag{14}$$

then

$$\partial F(B) = \partial x(B)^* A^*(Ax(B) - y^\delta) \tag{15}$$

with

$$\partial x(B) = -\psi_x(x(B), B)^{-1}\psi_B(x(B), B) \tag{16}$$

This yields the gradient descent iteration

$$B^{\ell+1} = B^\ell - \eta\partial F(B^\ell). \tag{17}$$

Example i)

Assume $R(x) = \frac{1}{2}\|x\|^2$

Simple case: $x^\dagger = u$, where u is a singular function of A $(Au = \sigma v)$

$$y^\delta = Au + \delta v = (\sigma + \delta) v \tag{18}$$

A lengthy computation exploiting $B^0 = A$ and the iteration $B^{\ell+1} = B^\ell - \eta \partial F(B^\ell)$ yields

$$B^{\ell+1} = B^\ell - c_\ell v u^* \tag{19}$$

with

$$c_\ell = \eta\sigma(\sigma+\delta)^2(\alpha + \beta_\ell^2 - \sigma\beta_\ell)\frac{\beta_\ell^2 - \alpha}{(\beta_\ell^2 + \alpha)^3}$$

β_ℓ: singular value of B^ℓ $(B^\ell u = \beta^\ell v)$

Example i)

Assume $R(x) = \frac{1}{2}\|x\|^2$

Simple case: $x^\dagger = u$, where u is a singular function of A ($Au = \sigma v$)

$$y^\delta = Au + \delta v = (\sigma + \delta)v \tag{18}$$

A lengthy computation exploiting $B^0 = A$ and the iteration $B^{\ell+1} = B^\ell - \eta \partial F(B^\ell)$ yields

$$B^{\ell+1} = B^\ell - c_\ell v u^* \tag{19}$$

with

$$c_\ell = \eta\sigma(\sigma+\delta)^2(\alpha + \beta_\ell^2 - \sigma\beta_\ell)\frac{\beta_\ell^2 - \alpha}{(\beta_\ell^2 + \alpha)^3}$$

β_ℓ: singular value of B^ℓ ($B^\ell u = \beta^\ell v$)

Example i)

Assume $R(x) = \frac{1}{2}\|x\|^2$

Simple case: $x^\dagger = u$, where u is a singular function of A $(Au = \sigma v)$

$$y^\delta = Au + \delta v = (\sigma + \delta)v \tag{18}$$

A lengthy computation exploiting $B^0 = A$ and the iteration $B^{\ell+1} = B^\ell - \eta\partial F(B^\ell)$ yields

$$B^{\ell+1} = B^\ell - c_\ell v u^* \tag{19}$$

with

$$c_\ell = \eta\sigma(\sigma + \delta)^2(\alpha + \beta_\ell^2 - \sigma\beta_\ell)\frac{\beta_\ell^2 - \alpha}{(\beta_\ell^2 + \alpha)^3}$$

β_ℓ: singular value of B^ℓ $(B^\ell u = \beta^\ell v)$

Example i)

This results in the sequence β^ℓ converging to

$$\beta_\infty = \begin{cases} \frac{\sigma}{2} \pm \sqrt{\frac{\sigma^2}{4} - \alpha} & \sigma \geq 2\sqrt{\alpha} \\ \sqrt{\alpha} & \sigma < 2\sqrt{\alpha} \end{cases} \tag{20}$$

and the sequence $x(B^\ell)$ with the unique attractive stationary point[10]

$$x(B^\infty) = \begin{cases} \frac{1}{\sigma}(\sigma + \delta)u & \sigma \geq 2\sqrt{\alpha} \\ \frac{1}{2\sqrt{\alpha}}(\sigma + \delta)u & \sigma < 2\sqrt{\alpha} \end{cases} \tag{21}$$

[10] Sören Dittmer, Tobias Kluth, Peter Maass, and Daniel Otero Baguer. "Regularization by architecture: A deep prior approach for inverse problems". In: *CoRR* abs/1812.03889 (2018). arXiv: 1812.03889. URL: http://arxiv.org/abs/1812.03889.

Example ii)

Assume $R(x) = \frac{1}{2}\|x\|^2$ and we optimize over

$$B \in \left\{ \tilde{B} \in \mathcal{L}(X,Y) \;\middle|\; \tilde{B} = \sum_i \beta_i v_i u_i^*,\ \beta_i \in \mathbb{R}_+ \cup \{0\} \right\} \qquad (22)$$

where $\{u_i, \sigma_i, v_i\}$ is the singular value decomposition of A

Theorem

There exist a global minimizer given by $B_\alpha = \sum \beta_i^\alpha v_i u_i^*$ with

$$\beta_i^\alpha = \begin{cases} \frac{\sigma_i}{2} + \sqrt{\frac{\sigma_i^2}{4} - \alpha} & \sigma_i \geq 2\sqrt{\alpha} \\ \sqrt{\alpha} & \sigma_i < 2\sqrt{\alpha} \end{cases} \qquad (23)$$

Example ii)

Assume $R(x) = \frac{1}{2}\|x\|^2$ and we optimize over

$$B \in \left\{ \tilde{B} \in \mathcal{L}(X, Y) \;\middle|\; \tilde{B} = \sum_i \beta_i v_i u_i^*,\ \beta_i \in \mathbb{R}_+ \cup \{0\} \right\} \tag{22}$$

where $\{u_i, \sigma_i, v_i\}$ is the singular value decomposition of A

Theorem

There exist a global minimizer given by $B_\alpha = \sum \beta_i^\alpha v_i u_i^*$ *with*

$$\beta_i^\alpha = \begin{cases} \frac{\sigma_i}{2} + \sqrt{\frac{\sigma_i^2}{4} - \alpha} & \sigma_i \geq 2\sqrt{\alpha} \\ \sqrt{\alpha} & \sigma_i < 2\sqrt{\alpha} \end{cases} \tag{23}$$

Example ii)

Recap the regularized pseudo inverse in terms of filter functions:

$$T_\alpha(y^\delta) = \sum_{\sigma_i > 0} F_\alpha(\sigma_i) \frac{1}{\sigma_i} \langle y^\delta, v_i \rangle u_i \tag{24}$$

Theorem (Soft TSVD)

The regularized pseudo inverse $K_\alpha(y^\delta) = x(B_\alpha, y^\delta)$ is an order optimal regularization method[11] *given by the filter function*

$$F_\alpha(\sigma) = \begin{cases} 1 & \sigma \geq 2\sqrt{\alpha} \\ \frac{\sigma}{2\sqrt{\alpha}} & \sigma < 2\sqrt{\alpha} \end{cases} \tag{25}$$

[11] Alfred Karl Louis. *Inverse und schlecht gestellte Probleme.* Wiesbaden: Vieweg+Teubner Verlag, 1989.

Example ii)

Recap the regularized pseudo inverse in terms of filter functions:

$$T_\alpha(y^\delta) = \sum_{\sigma_i > 0} F_\alpha(\sigma_i) \frac{1}{\sigma_i} \langle y^\delta, v_i \rangle u_i \tag{24}$$

Theorem (Soft TSVD)

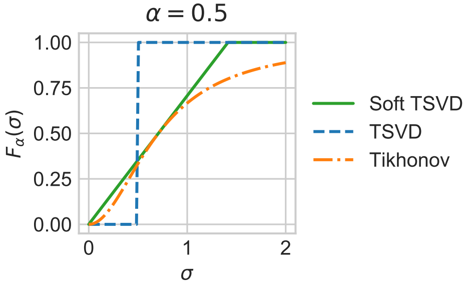
The regularized pseudo inverse $K_\alpha(y^\delta) = x(B_\alpha, y^\delta)$ is an order optimal regularization method[11] *given by the filter function*

$$F_\alpha(\sigma) = \begin{cases} 1 & \sigma \geq 2\sqrt{\alpha} \\ \frac{\sigma}{2\sqrt{\alpha}} & \sigma < 2\sqrt{\alpha} \end{cases} \tag{25}$$

[11] Alfred Karl Louis. *Inverse und schlecht gestellte Probleme*. Wiesbaden: Vieweg+Teubner Verlag, 1989.

Example ii)

Comparison with other regularization methods



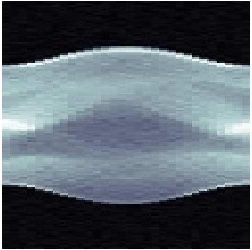
Section 4

Application I: Computed Tomography

Example a): Shepp-Logan phantom

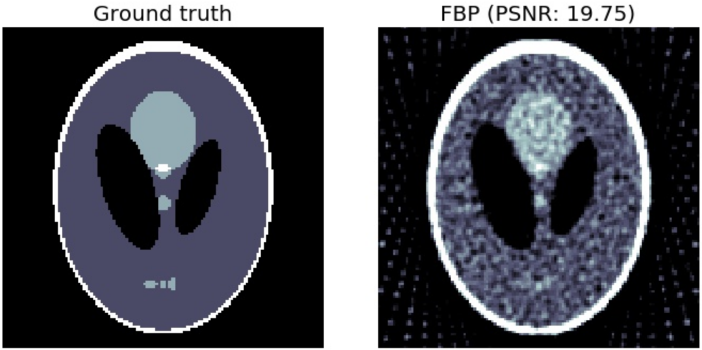
- Parallel beam geometry (30 angles, 183 detectors)
- 5% white noise
- Visualization window: $[0.1, 0.4]$

(a) Ground truth (128×128)

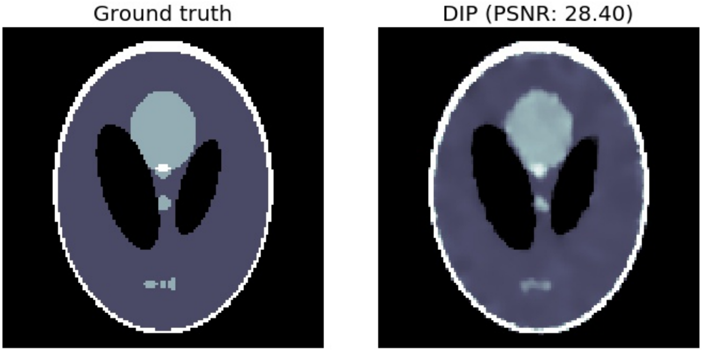


(b) Data (30×183)

Example a): Shepp-Logan phantom

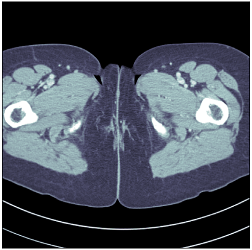


Example a): Shepp-Logan phantom

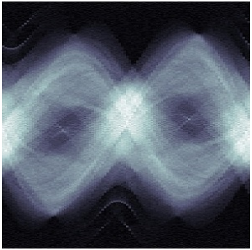


Example b): Human phantom[12]

- Case i: Fan-beam geometry (100 angles, 1000 detectors)
- Case ii: Fan-beam geometry (1000 angles, 1000 detectors)
- 5% white noise



(a) Ground truth (512 × 512)

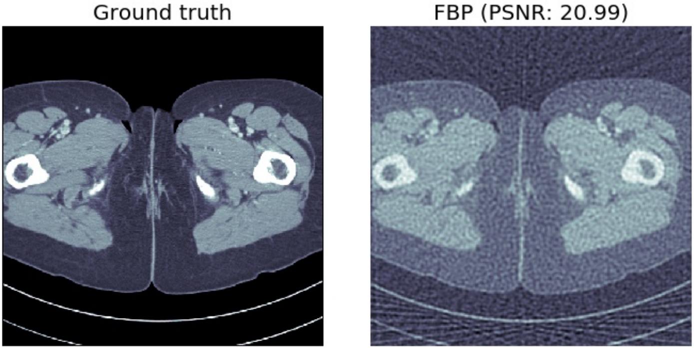


(b) Data (100 × 1000)

[12] Jonas Adler and Ozan Öktem. "Learned primal-dual reconstruction". In: *IEEE transactions on medical imaging* 37.6 (2018), pp. 1322–1332.

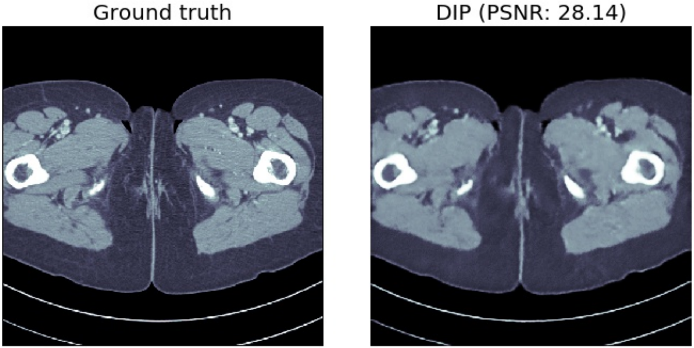
Example b): Human phantom

Case i: 100 angles



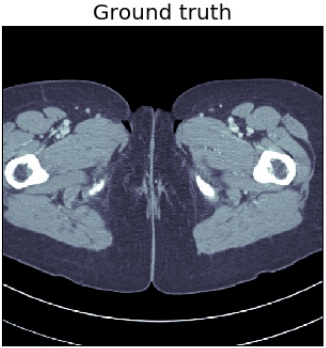
Example b): Human phantom

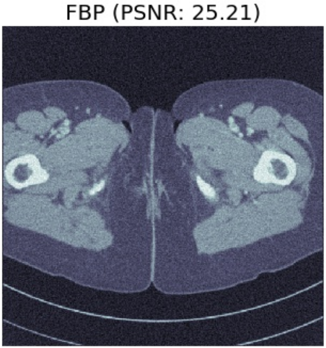
Case i: 100 angles



Example b): Human phantom

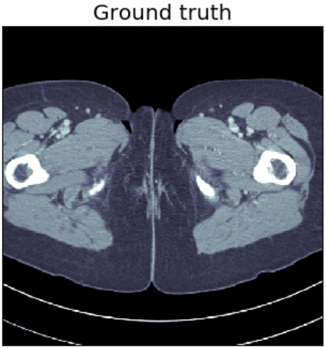
Case ii: 1000 angles

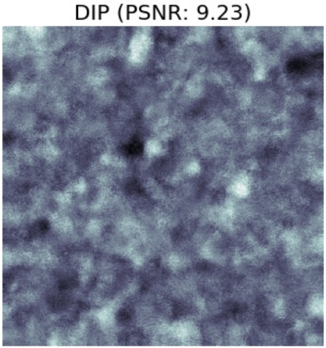




Example b): Human phantom

Case ii: 1000 angles





Example b): Human phantom

Case ii: 1000 angles

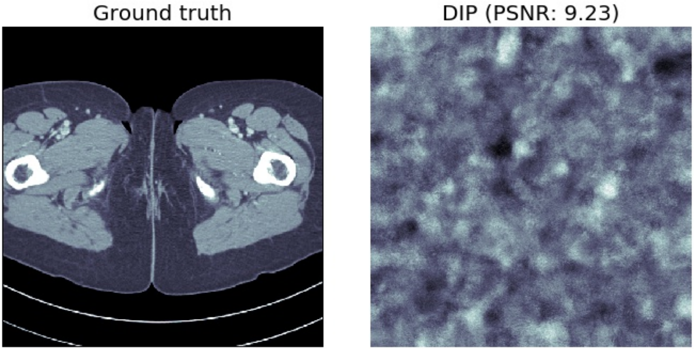


Figure: Iteration 0

Example b): Human phantom

Case ii: 1000 angles

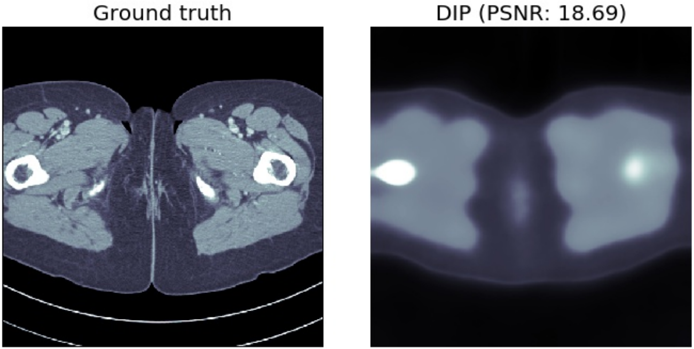


Figure: Iteration 100

Example b): Human phantom

Case ii: 1000 angles

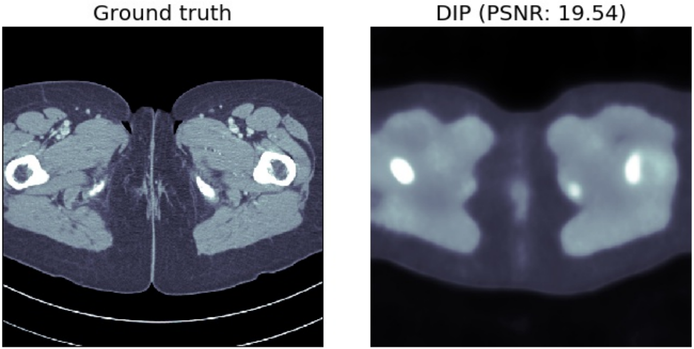


Figure: Iteration 200

Example b): Human phantom

Case ii: 1000 angles

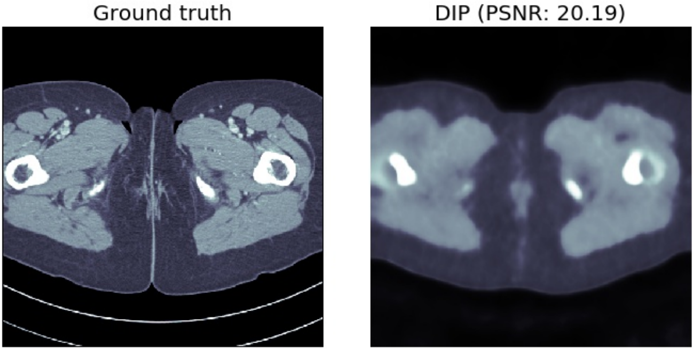


Figure: Iteration 300

Example b): Human phantom

Case ii: 1000 angles

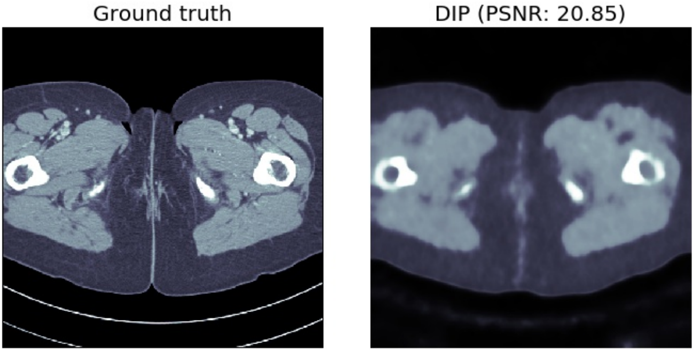


Figure: Iteration 400

Example b): Human phantom

Case ii: 1000 angles

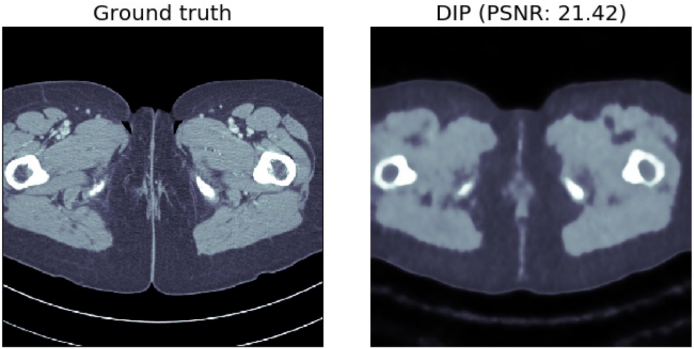


Figure: Iteration 500

Example b): Human phantom

Case ii: 1000 angles

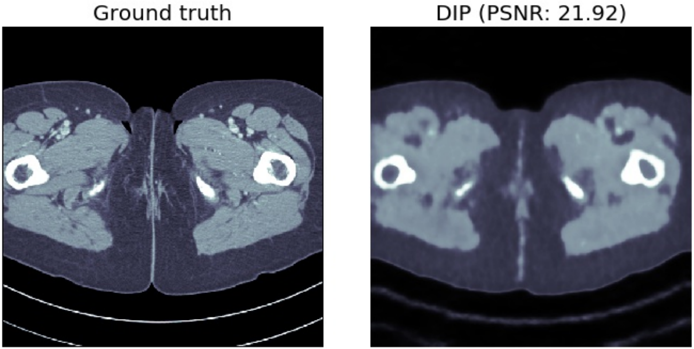


Figure: Iteration 600

Example b): Human phantom

Case ii: 1000 angles

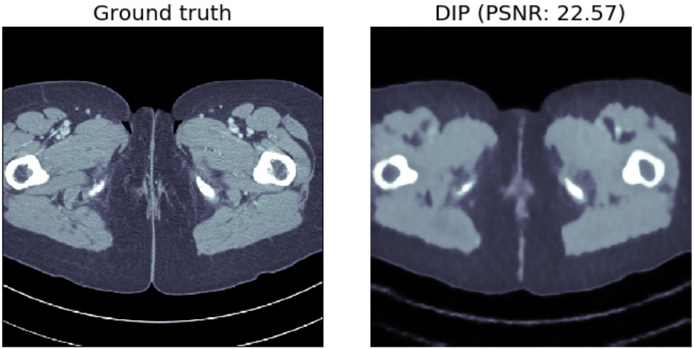


Figure: Iteration 700

Example b): Human phantom

Case ii: 1000 angles

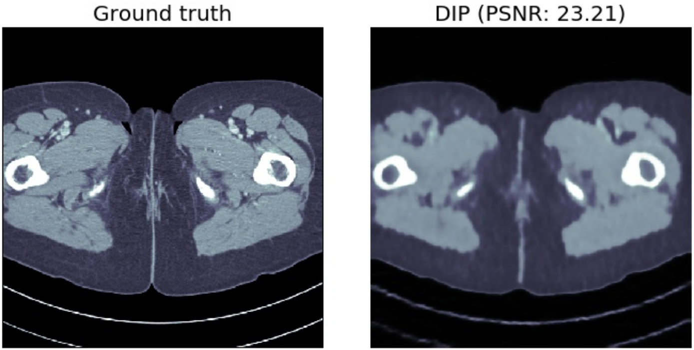


Figure: Iteration 800

Example b): Human phantom

Case ii: 1000 angles

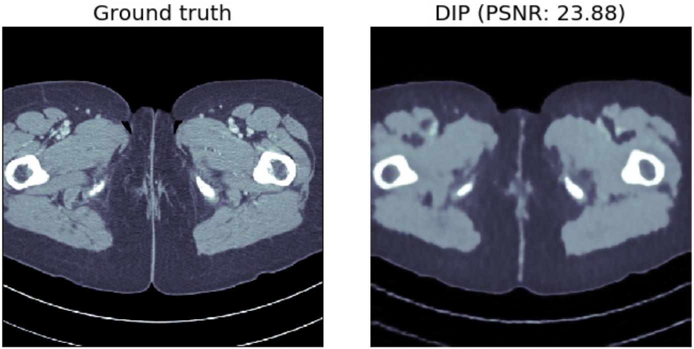


Figure: Iteration 900

Example b): Human phantom

Case ii: 1000 angles

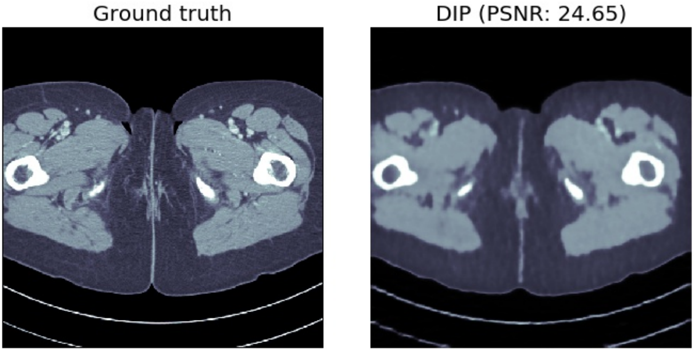


Figure: Iteration 1000

Example b): Human phantom

Case ii: 1000 angles

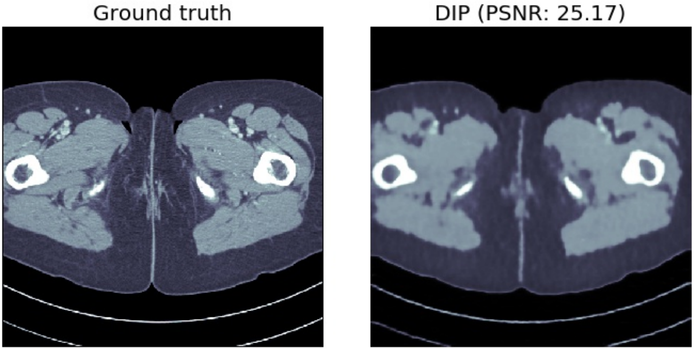


Figure: Iteration 1100

Example b): Human phantom

Case ii: 1000 angles

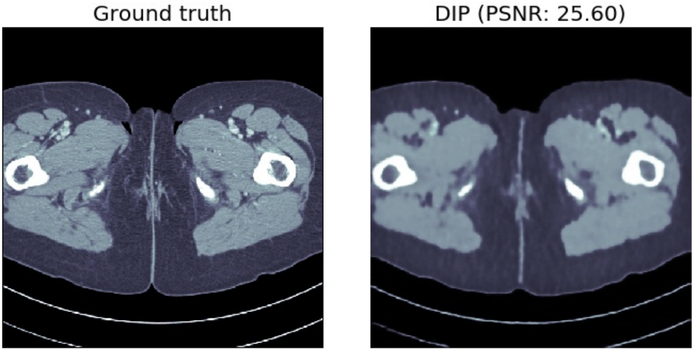


Figure: Iteration 1200

Example b): Human phantom

Case ii: 1000 angles

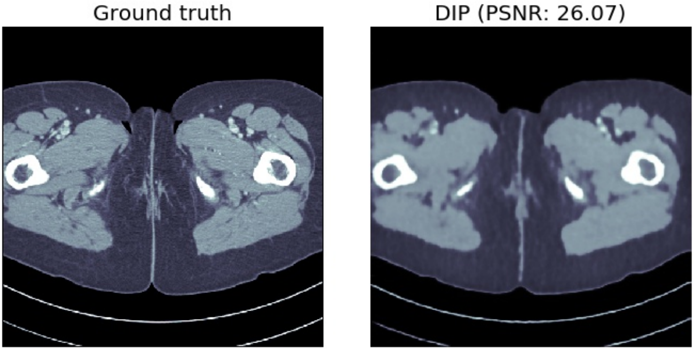


Figure: Iteration 1300

Example b): Human phantom

Case ii: 1000 angles

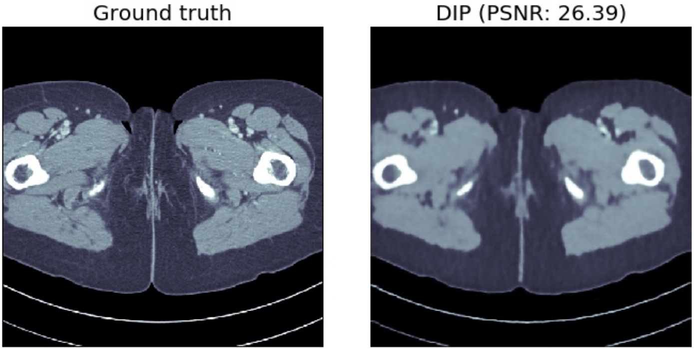


Figure: Iteration 1400

Example b): Human phantom

Case ii: 1000 angles

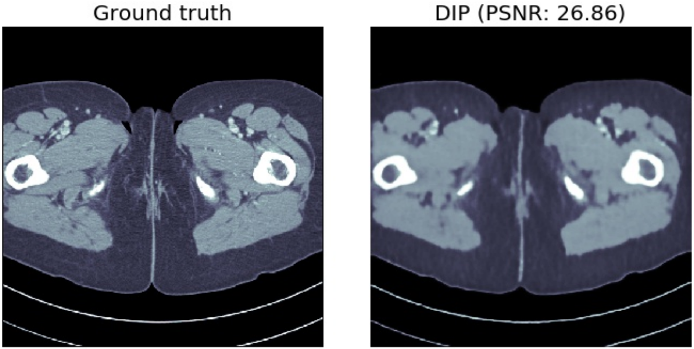


Figure: Iteration 1500

Example b): Human phantom

Case ii: 1000 angles

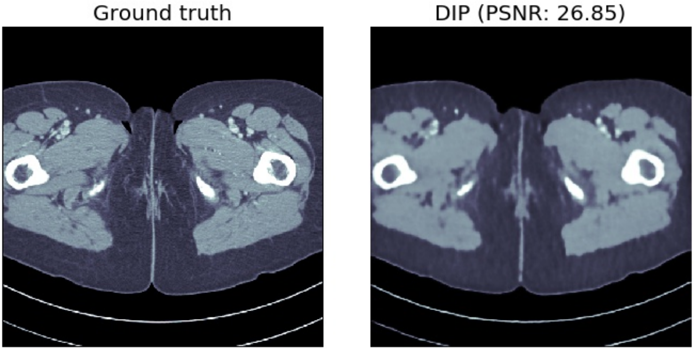


Figure: Iteration 1600

Example b): Human phantom

Case ii: 1000 angles

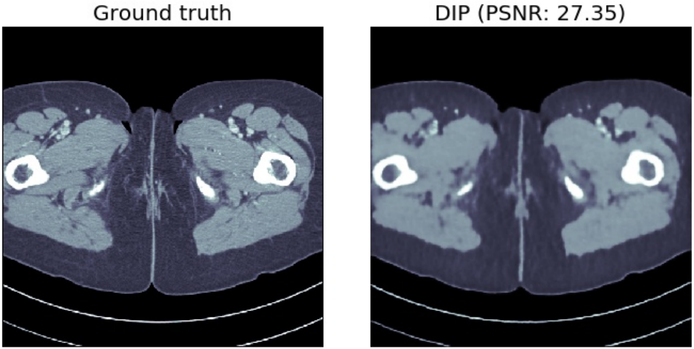


Figure: Iteration 1700

Example b): Human phantom

Case ii: 1000 angles

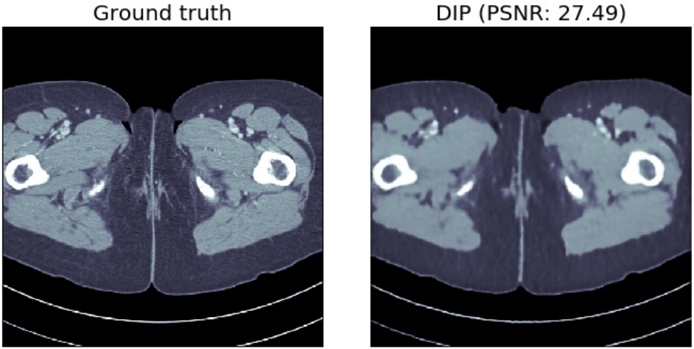


Figure: Iteration 1800

Example b): Human phantom

Case ii: 1000 angles

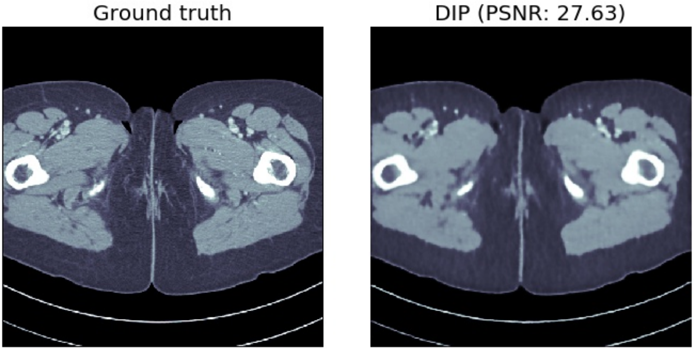


Figure: Iteration 1900

Example b): Human phantom

Case ii: 1000 angles

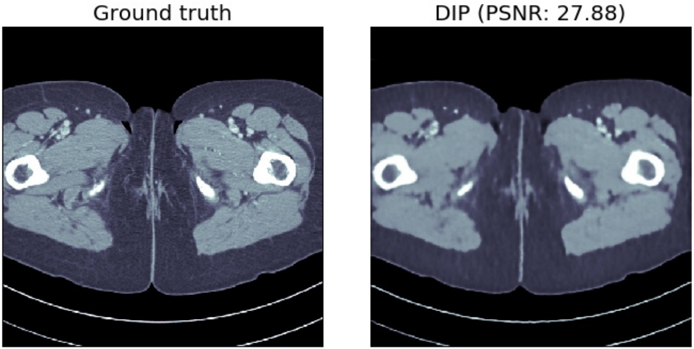


Figure: Iteration 2000

Example b): Human phantom

Case ii: 1000 angles

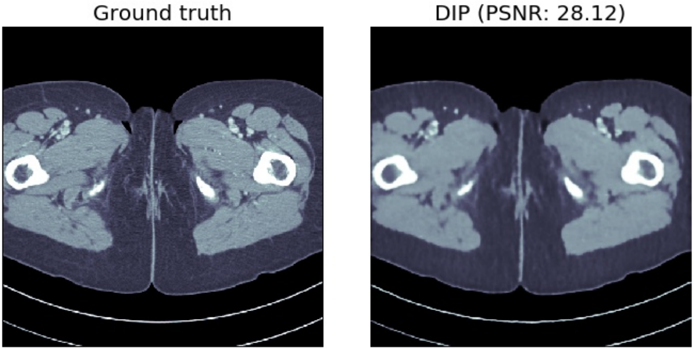


Figure: Iteration 2100

Example b): Human phantom

Case ii: 1000 angles

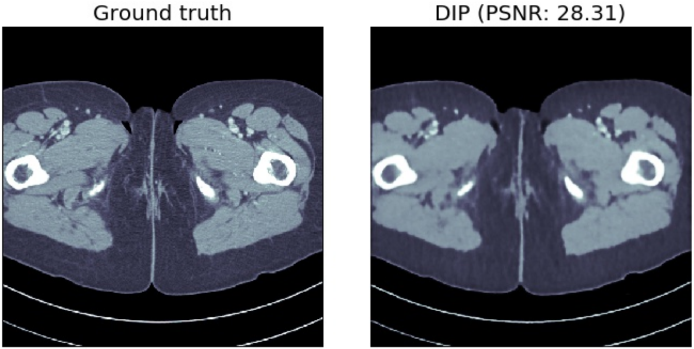


Figure: Iteration 2200

Example b): Human phantom

Case ii: 1000 angles

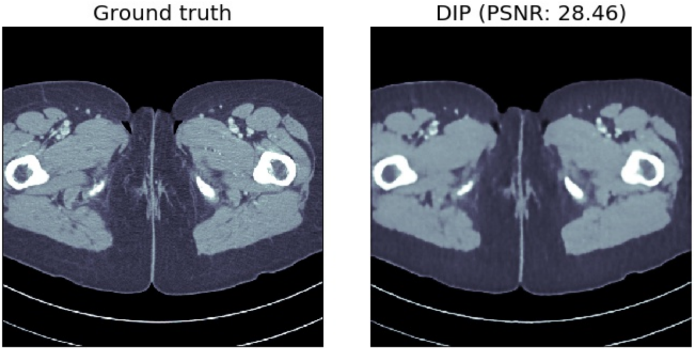


Figure: Iteration 2300

Example b): Human phantom

Case ii: 1000 angles

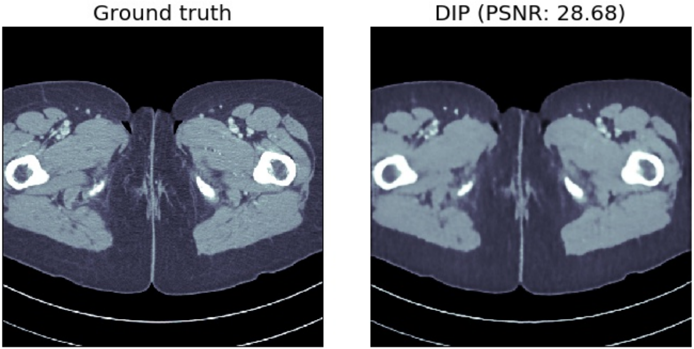


Figure: Iteration 2400

Example b): Human phantom

Case ii: 1000 angles

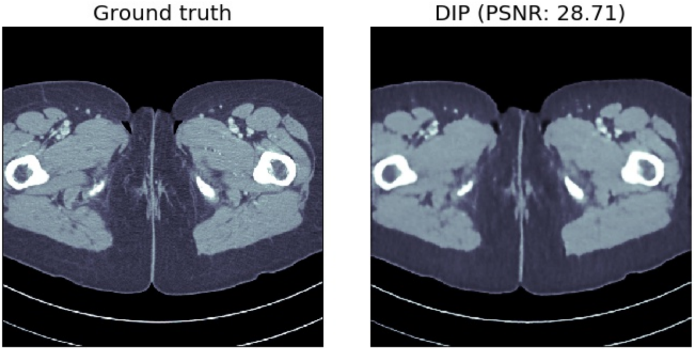


Figure: Iteration 2500

Example b): Human phantom

Case ii: 1000 angles

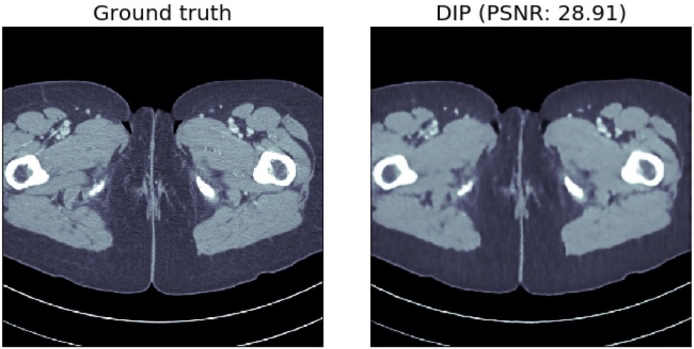


Figure: Iteration 2600

Example b): Human phantom

Case ii: 1000 angles

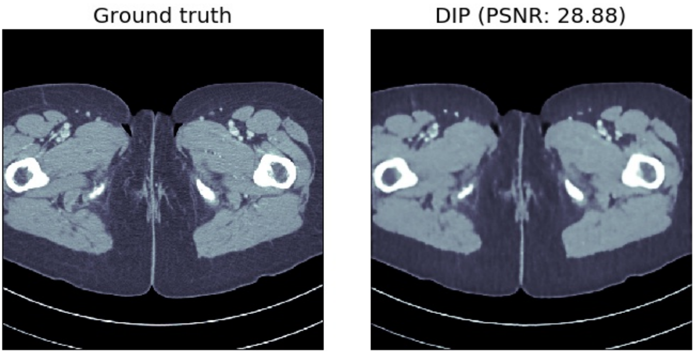


Figure: Iteration 2700

Example b): Human phantom

Case ii: 1000 angles

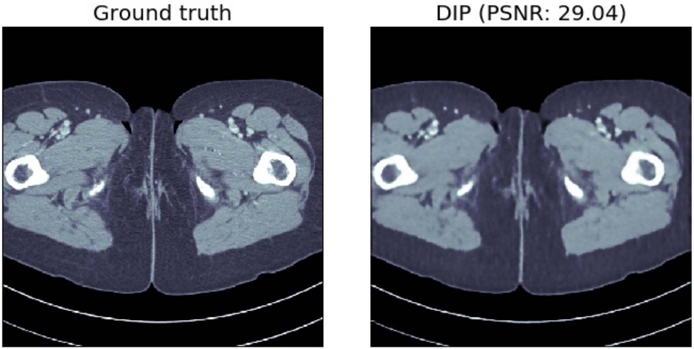


Figure: Iteration 2800

Example b): Human phantom

Case ii: 1000 angles

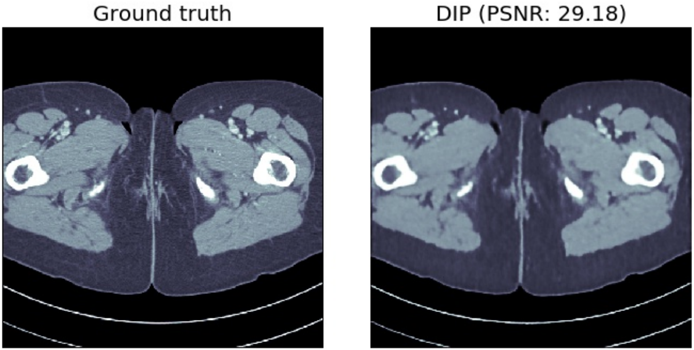


Figure: Iteration 2900

Example b): Human phantom

Case ii: 1000 angles

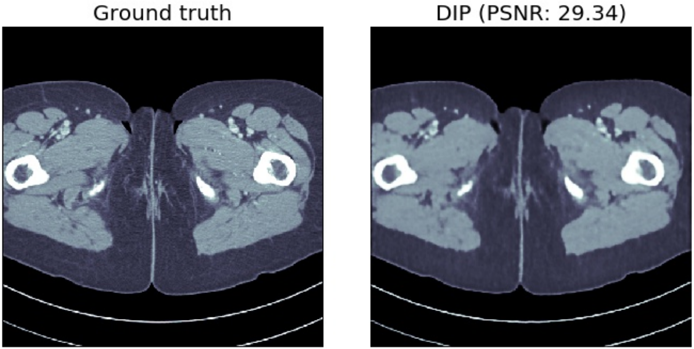


Figure: Iteration 3000

Example b): Human phantom

Case ii: 1000 angles

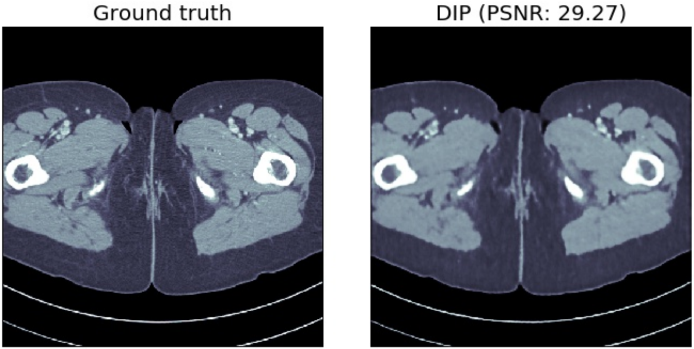


Figure: Iteration 3100

Example b): Human phantom

Case ii: 1000 angles

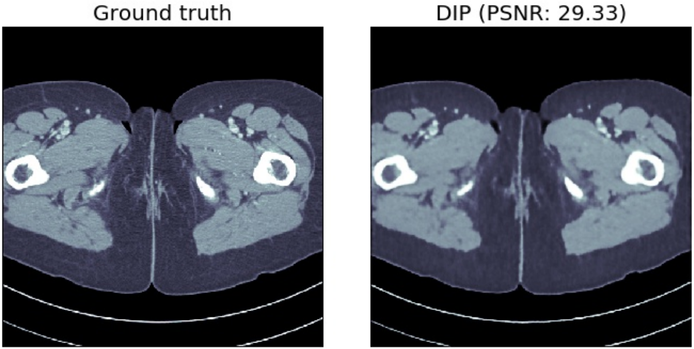


Figure: Iteration 3200

Example b): Human phantom

Case ii: 1000 angles

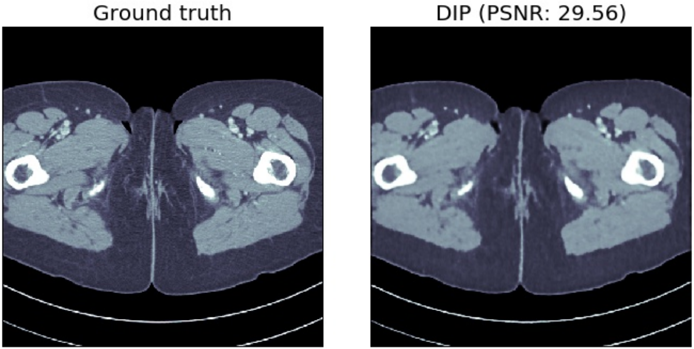


Figure: Iteration 3300

Example b): Human phantom

Case ii: 1000 angles

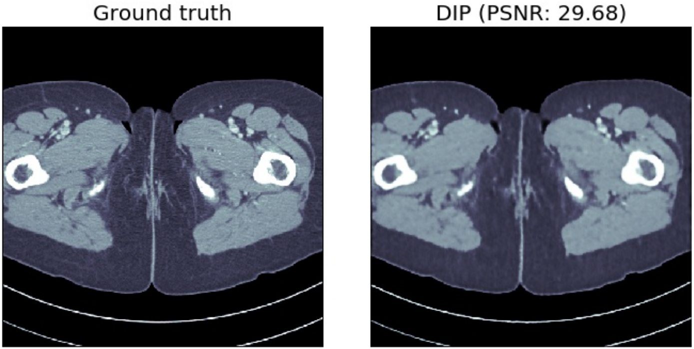


Figure: Iteration 3400

Example b): Human phantom

Case ii: 1000 angles

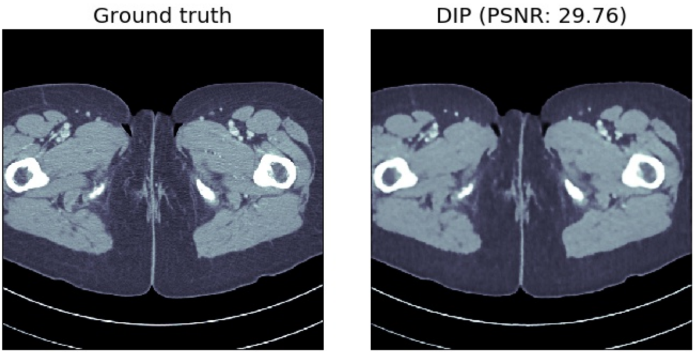


Figure: Iteration 3500

Example b): Human phantom

Case ii: 1000 angles

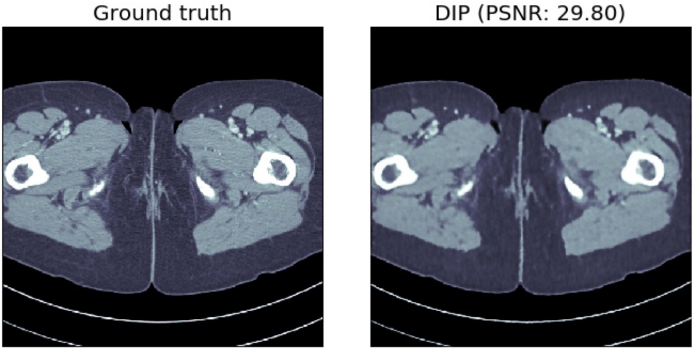


Figure: Iteration 3600

Example b): Human phantom

Case ii: 1000 angles

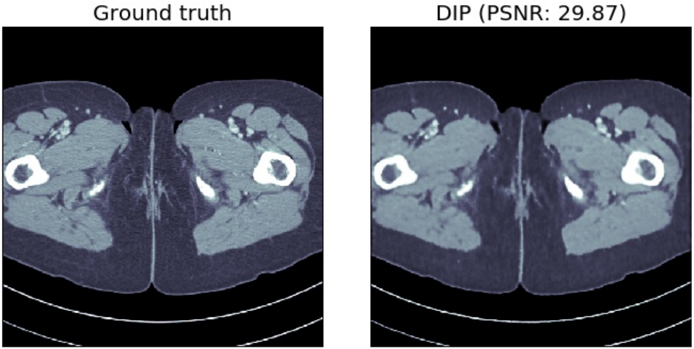


Figure: Iteration 3700

Example b): Human phantom

Case ii: 1000 angles

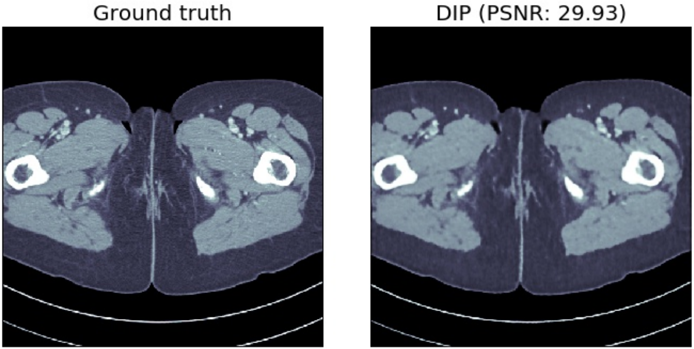


Figure: Iteration 3800

Example b): Human phantom

Case ii: 1000 angles

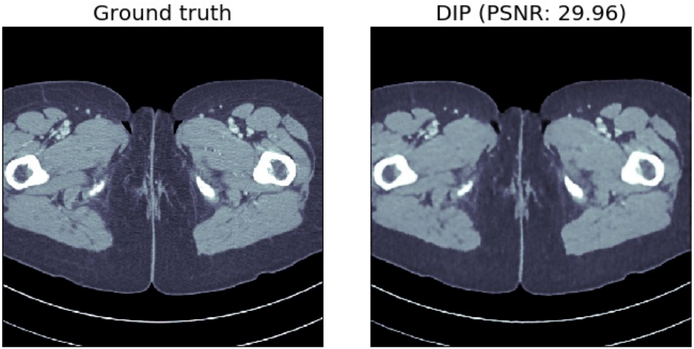


Figure: Iteration 3900

Example b): Human phantom

Case ii: 1000 angles

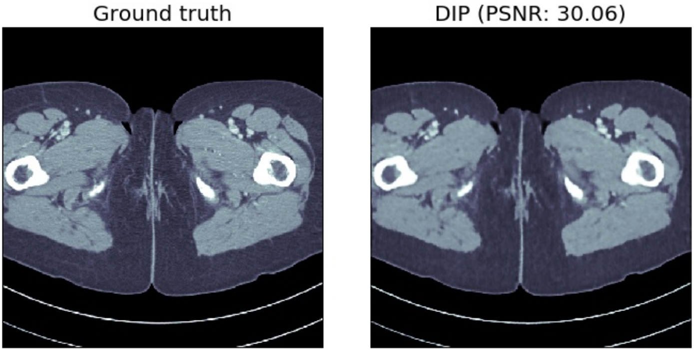


Figure: Iteration 4000

Example b): Human phantom

Case ii: 1000 angles

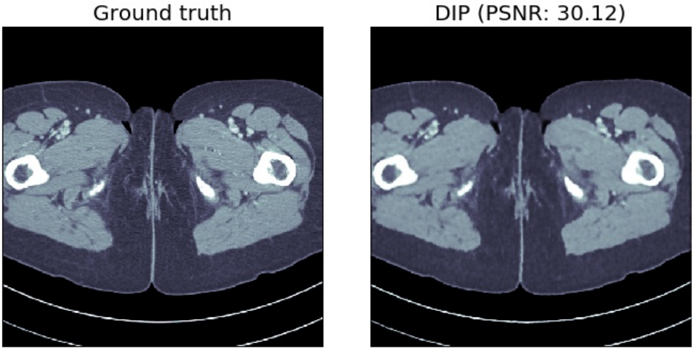


Figure: Iteration 4100

Example b): Human phantom

Case ii: 1000 angles

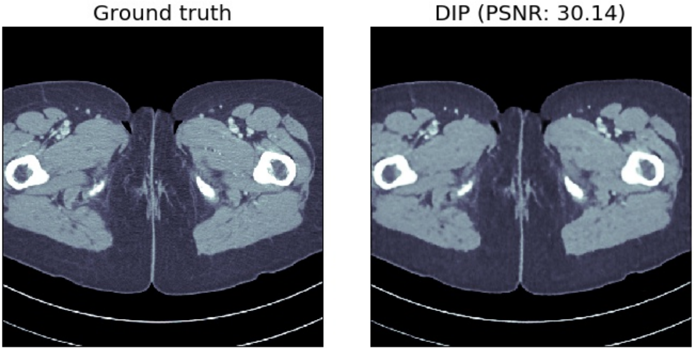


Figure: Iteration 4200

Example b): Human phantom

Case ii: 1000 angles

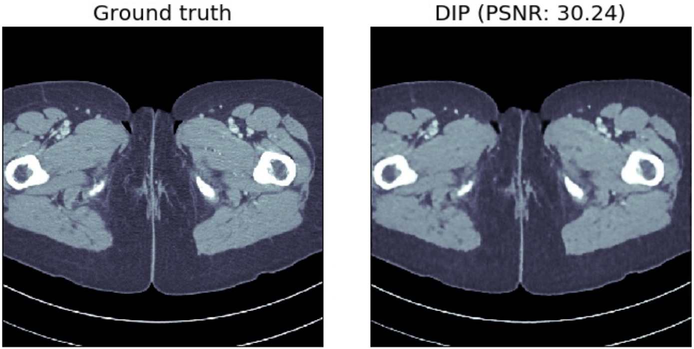


Figure: Iteration 4300

Example b): Human phantom

Case ii: 1000 angles

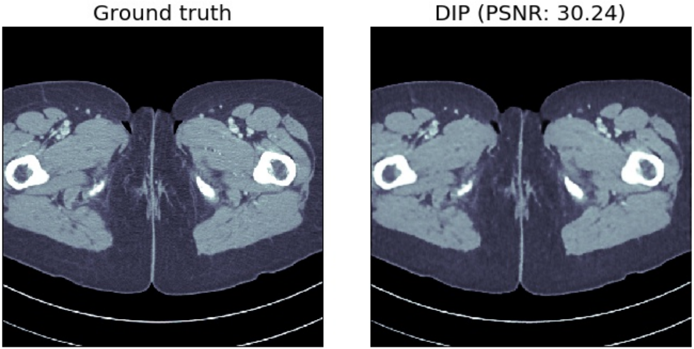


Figure: Iteration 4400

Example b): Human phantom

Case ii: 1000 angles

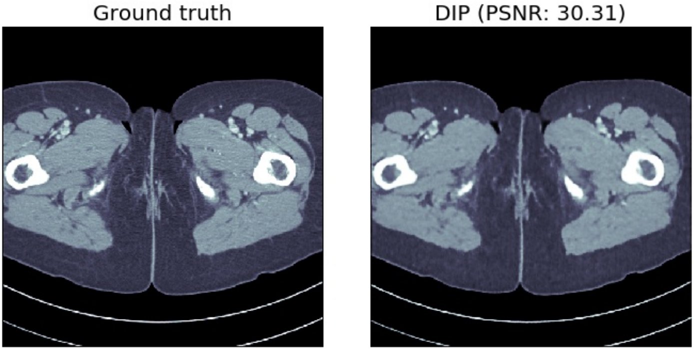


Figure: Iteration 4500

Example b): Human phantom

Case ii: 1000 angles

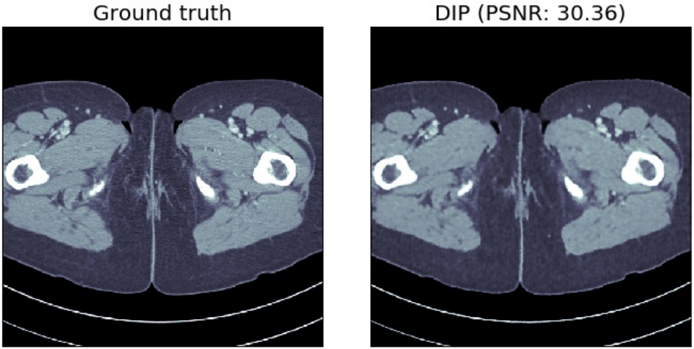


Figure: Iteration 4600

Example b): Human phantom

Case ii: 1000 angles

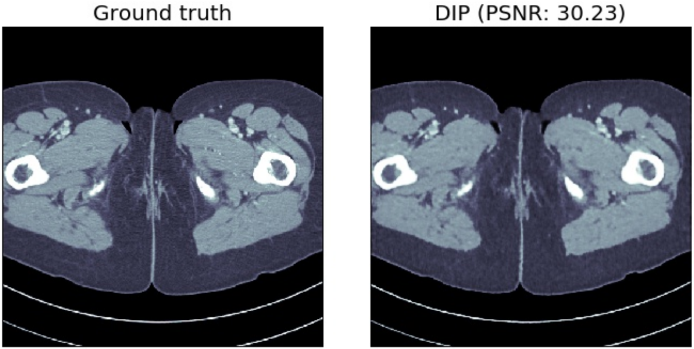


Figure: Iteration 4700

Example b): Human phantom

Case ii: 1000 angles

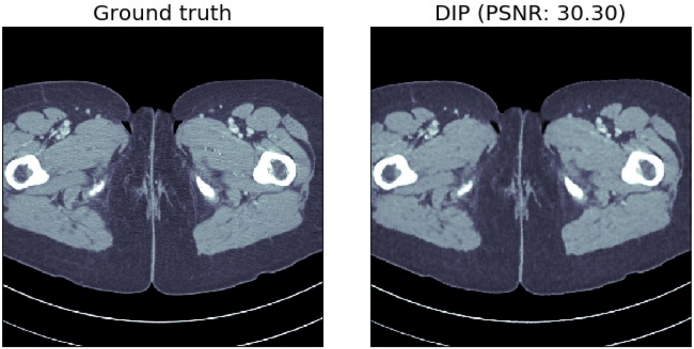


Figure: Iteration 4800

Example b): Human phantom

Case ii: 1000 angles

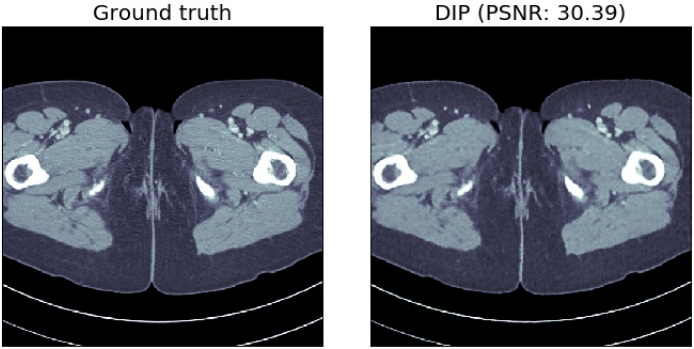


Figure: Iteration 4900

Example b): Human phantom

Case ii: 1000 angles

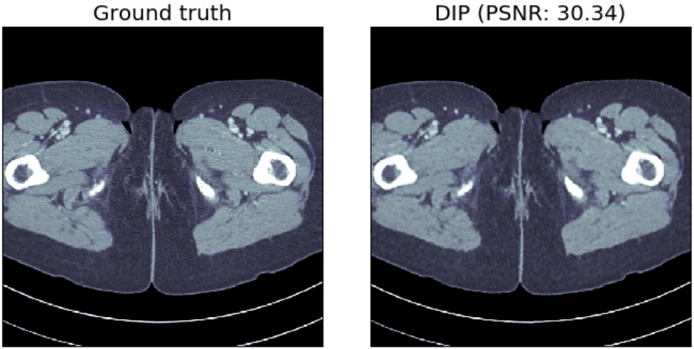


Figure: Iteration 5000

Example b): Human phantom

Case ii: 1000 angles

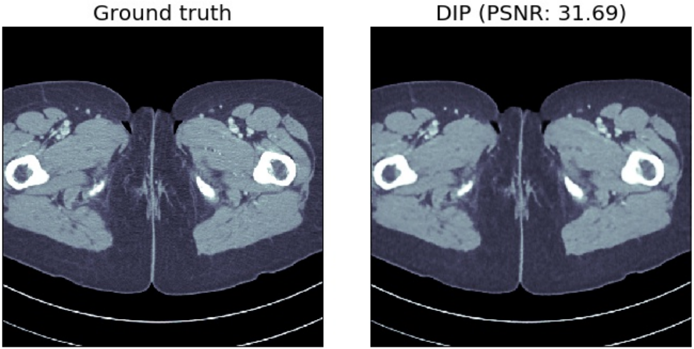


Figure: Final result

Example b): Human phantom

Case ii: 1000 angles (Running time ≈ 7 min)

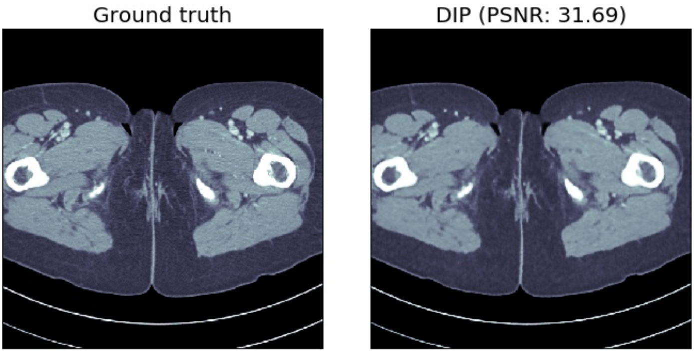


Figure: Final result

Implementation

Libraries:

- DIP source code[13]
- Operator Discretization Library (ODL)[14]

Training parameters:

- Iterations: 5000
- Learning rate: 10^{-3}
- Regularization noise: 10^{-2}

Architecture:

- Number of scales: 5
- Filter size per scale: 3
- Number of filters per scale: 128
- Number of filters per skip connection: 4

Hardware:

- Nvidia GeForce GTX 1080

[13] https://github.com/DmitryUlyanov/deep-image-prior

[14] https://github.com/odlgroup/odl

More advertising...

Deep Inversion Validation Library (DIV$\alpha\ell$)

- Library for testing and comparing deep learning based methods for inverse problems

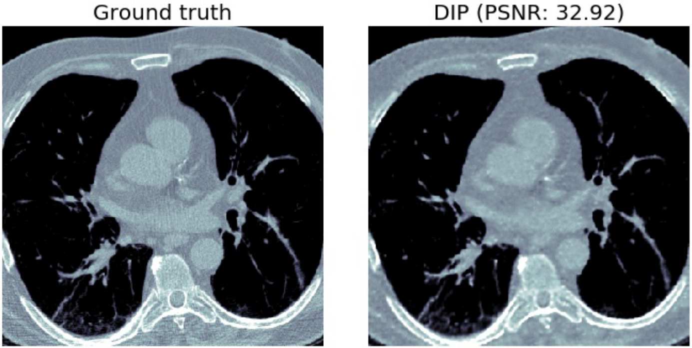
Main goal: Provide standard datasets

Link: `https://github.com/jleuschn/dival`

by Johannes Leuschner, Max Schmidt and Hannes Albers

Example c): Human phantom (DIV$\alpha\ell$)

Case ii: 1000 angles



Section 5

Application II: Magnetic Particle Imaging

What is MPI?

Imaging modality based on injecting ferromagnetic nano-particles which are consequently transported by the blood flow

Goal: Measure the 3-D location and concentration of the nanoparticles

Advantages:

- High spacial resolution (< 1mm)
- Measurement time (< 0.1 s)
- No harmful radiation

Figure: Magnetic particles developed in Lübeck

What is MPI?

Imaging modality based on injecting ferromagnetic nano-particles which are consequently transported by the blood flow

Goal: Measure the 3-D location and concentration of the nanoparticles

Advantages:

- High spacial resolution (< 1mm)
- Measurement time (< 0.1 s)
- No harmful radiation

Figure: Magnetic particles developed in Lübeck

What is MPI?

Imaging modality based on injecting ferromagnetic nano-particles which are consequently transported by the blood flow

Goal: Measure the 3-D location and concentration of the nanoparticles

Advantages:

- High spacial resolution (< 1mm)
- Measurement time (< 0.1 s)
- No harmful radiation

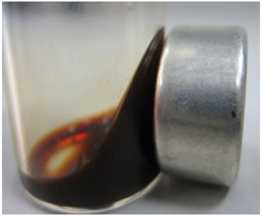


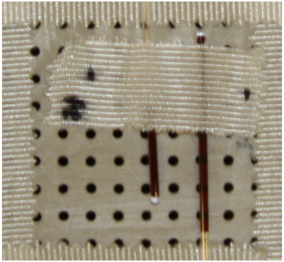
Figure: Magnetic particles developed in Lübeck

How it works?

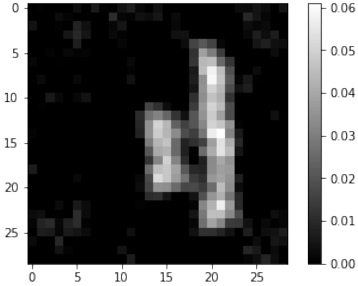
- A magnetic field is applied
- Mean magnetic moment of the nano-particles generates an electro-magnetic field
- Time-dependent voltages $v_\ell(t)$ depending on the concentration of the particles $c(x)$ at position $x \in \Omega$ are measured by so-called receive coils
- The forward problem is modeled[15] by an integral operator S
- c is reconstructed from measured noisy data $v^\delta = Sc + \tau$

[15] Tobias Kluth. "Mathematical models for magnetic particle imaging". In: *Inverse Problems* 34.8 (June 2018), p. 083001. DOI: 10.1088/1361-6420/aac535. URL: https://doi.org/10.1088%2F1361-6420%2Faac535.

Results

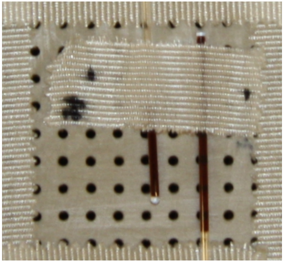


(a) Phantom (4mm)

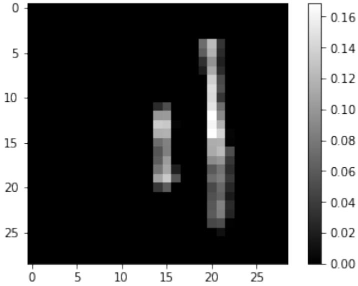


(b) Kacmarz reconstruction ($\alpha = 5 \cdot 10^{-4}$)

Results

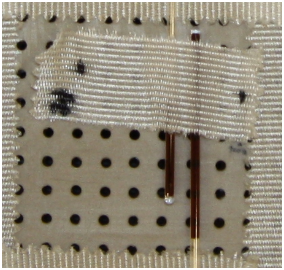


(a) Phantom (4mm)

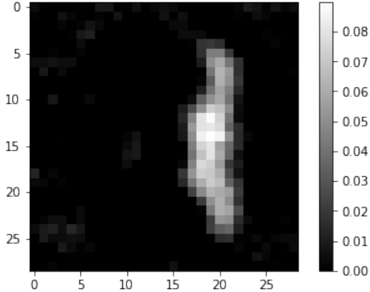


(b) DIP reconstruction ($lr = 5 \cdot 10^{-5}$)

Results

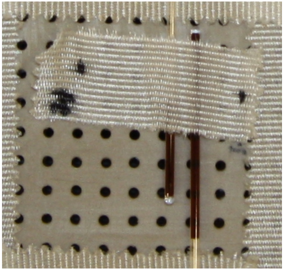


(a) Phantom (2mm)

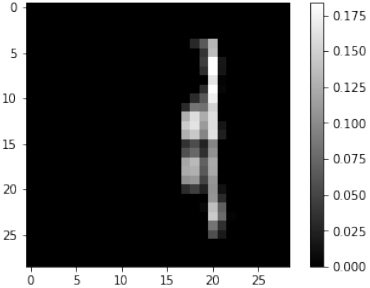


(b) Kacmarz reconstruction ($\alpha = 5 \cdot 10^{-4}$)

Results



(a) Phantom (2mm)



(b) DIP reconstruction ($lr = 5 \cdot 10^{-5}$)

Thanks!