Task Oriented Reconstruction for Inverse Problems

Jonas Adler[1, 2] Sebastian Lunz[3] Olivier Verdier[1] Carola-Bibane Schönlieb[3] Ozan Öktem[1]

[1]KTH - Royal Institute of Technology

[2]Elekta

[3]University of Cambridge

Medical Image Reconstruction

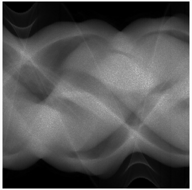
$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise

Medical Image Reconstruction

$$y = \mathcal{A}(x^*) + e.$$

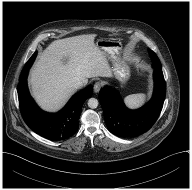
$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise

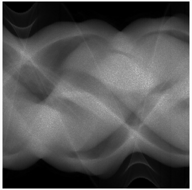


Medical Image Reconstruction

$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise

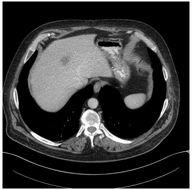




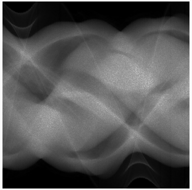
Medical Image Reconstruction

$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise



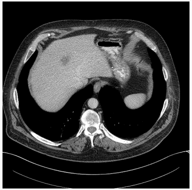
$\xrightarrow{\mathcal{A}}$



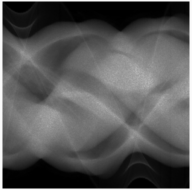
Medical Image Reconstruction

$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise



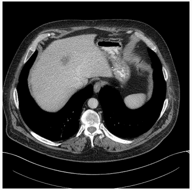
$\xrightarrow{\mathcal{A}}$



Medical Image Reconstruction

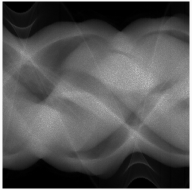
$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise



$\xrightarrow{\mathcal{A}}$

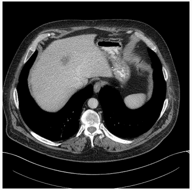
$\xleftarrow[\text{"}\mathcal{A}^{-1}\text{"}]{}$



Medical Image Reconstruction

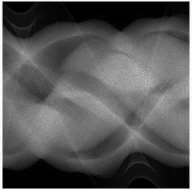
$$y = \mathcal{A}(x^*) + e.$$

$y \in Y$	Data
$x^* \in X$	Image
$\mathcal{A} : X \to Y$	Forward operator
$e \in Y$	Noise



$$\xrightarrow{\mathcal{A}}$$

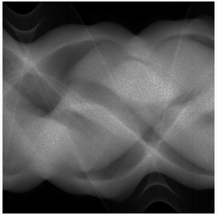
$$\xleftarrow[\text{"}\,\mathcal{A}^{-1}\,\text{"}]{}$$



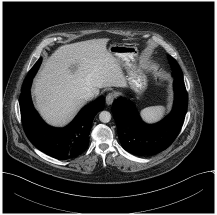
The problem is ill-posed: non-uniqueness, instability

Learned inversion methods

Main complication: $\mathcal{A}^\dagger : Y \to X$.

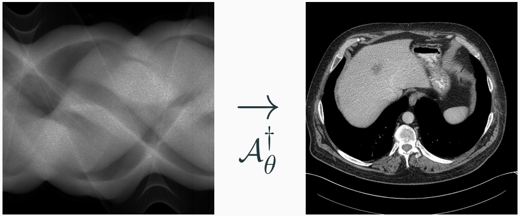


$\xrightarrow[\mathcal{A}_\theta^\dagger]{}$



Learned inversion methods

Main complication: $\mathcal{A}^\dagger : Y \to X$.



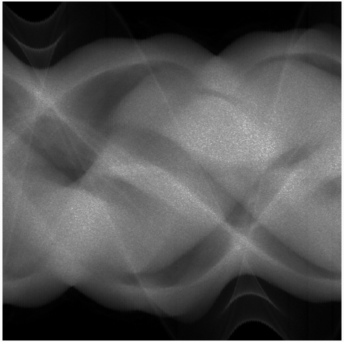
- Fully learned
- Learned post-processing
- Learned iterative schemes

Fully learned reconstruction

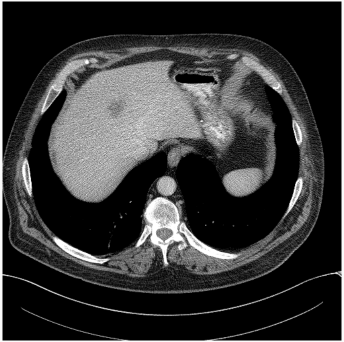
Goal: Learn "the whole" mapping from data to signal

Fully learned reconstruction

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

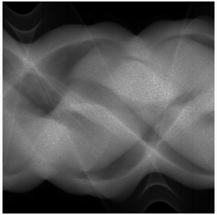


$\xrightarrow{\mathcal{A}_\theta^\dagger}$

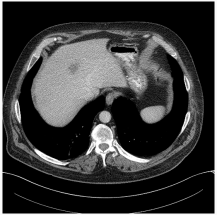


Learned inversion methods

Main complication: $\mathcal{A}_\theta^\dagger : Y \to X$.



$$\xrightarrow[\mathcal{A}_\theta^\dagger]{}$$



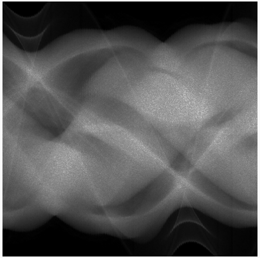
- Fully learned
- Learned post-processing
- Learned iterative schemes

Learned post-processing

Use deep learning to improve the result of another reconstruction

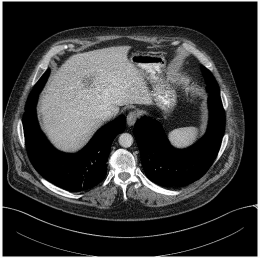
$$\mathcal{A}_\theta^\dagger = \Lambda_\theta \circ \mathcal{A}^\dagger$$

where $\mathcal{A}^\dagger$ is some reconstruction and Λ_θ is a learned post-processing.



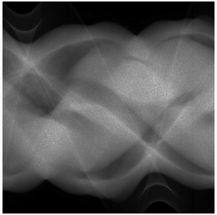
$\xrightarrow[\mathcal{A}^\dagger]{}$

$\xrightarrow[\Lambda_\theta]{}$

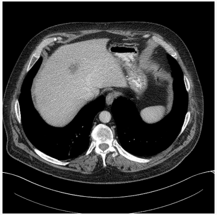


Learned inversion methods

Main complication: $\mathcal{A}_\theta^\dagger : Y \to X$.



$\xrightarrow{\mathcal{A}_\theta^\dagger}$



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

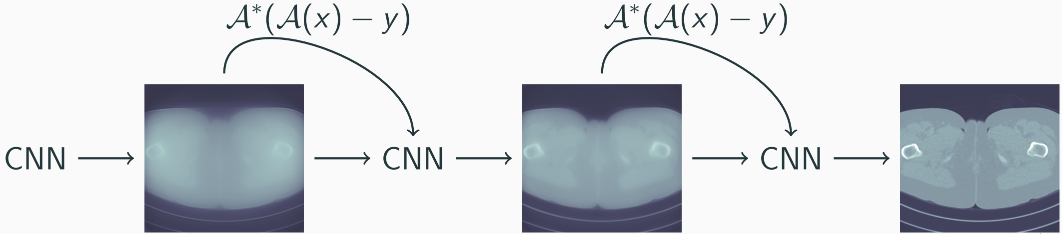
Learned Iterative Reconstruction

Idea: Include physics knowledge $(\mathcal{A}, \mathcal{A}^*)$ *inside* CNN.

Learned Iterative Reconstruction

Idea: Include physics knowledge $(\mathcal{A}, \mathcal{A}^*)$ *inside* CNN.

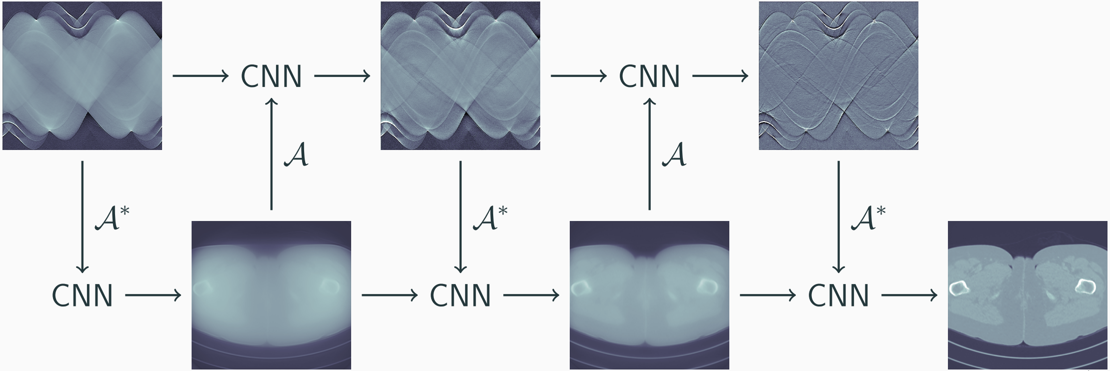
- Gradient (Learned/Deep Gradient Descent, Variational Networks)



Learned Iterative Reconstruction

Idea: Include physics knowledge $(\mathcal{A}, \mathcal{A}^*)$ *inside* CNN.

- Gradient (Learned/Deep Gradient Descent, Variational Networks)
- Primal-Dual



Learned Iterative Reconstruction

- Several learned methods for image reconstruction

Learned Iterative Reconstruction

- Several learned methods for image reconstruction
- These are all end-to-end differentiable

Learned Iterative Reconstruction

- Several learned methods for image reconstruction
- These are all end-to-end differentiable
- Trains like any neural network

Loss functions

- Train by minimizing Bayes risk

$$L(\theta) = \mathbb{E}\, \ell_X(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x})$$

Loss functions

- Train by minimizing Bayes risk

$$L(\theta) = \mathbb{E}\ \ell_X(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x})$$

- Traditional losses
  - $\|x_1 - x_2\|_2^2$
  - $\|x_1 - x_2\|_1$

Loss functions

- Train by minimizing Bayes risk

$$L(\theta) = \mathbb{E}\ \ell_X(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x})$$

- Traditional losses
  - $\|x_1 - x_2\|_2^2$
  - $\|x_1 - x_2\|_1$
- Fancy losses
  - Adversarial
  - Perceptual

Loss functions

- Train by minimizing Bayes risk

$$L(\theta) = \mathbb{E}\ \ell_X(\mathcal{A}_\theta^\dagger(y), x)$$

- Traditional losses
  - $\|x_1 - x_2\|_2^2$
  - $\|x_1 - x_2\|_1$
- Fancy losses
  - Adversarial
  - Perceptual
- Medical imaging is not done for fun, we want to solve a task!

Loss functions

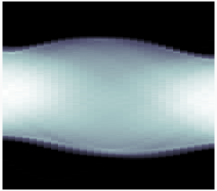
- Train by minimizing Bayes risk

$$L(\theta) = \mathbb{E}\ \ell_X(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x})$$

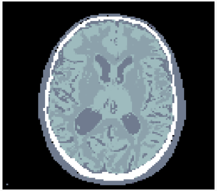
- Traditional losses
  - $\|x_1 - x_2\|_2^2$
  - $\|x_1 - x_2\|_1$
- Fancy losses
  - Adversarial
  - Perceptual
- Medical imaging is not done for fun, we want to solve a task!
- How do we define "good for segmentation/classification/radiomics"?

Task Adapted Reconstruction: Introduction

- We can learn to go from data to reconstruction

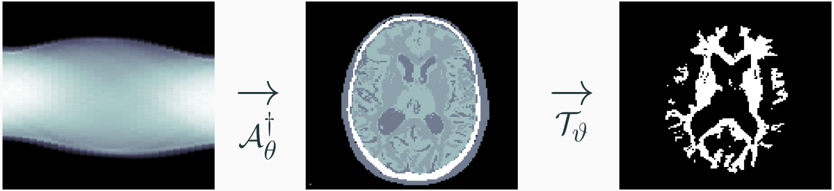


$$\xrightarrow[\mathcal{A}_\theta^\dagger]{}$$



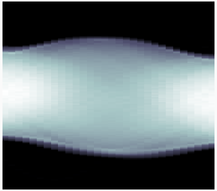
Task Adapted Reconstruction: Introduction

- We can learn to go from data to reconstruction
- Combine with learned task operator

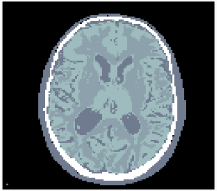


Task Adapted Reconstruction: Introduction

- We can learn to go from data to reconstruction
- Combine with learned task operator
- End-to-end differentiable training!



$\xrightarrow[\mathcal{A}_\theta^\dagger]{}$



$\xrightarrow[\mathcal{T}_\vartheta]{}$

Task Adapted Reconstruction: Approaches

- Sequential training: First train a reconstruction, then train the task

$$L(\theta) = \mathbb{E}_{\mathsf{x},\mathsf{y}}\Big[\ell_X\big(\mathcal{A}_\theta^\dagger(\mathsf{y}), \mathsf{x}\big)\Big].$$

$$L(\vartheta) = \mathbb{E}_{\mathsf{y},\mathsf{d}}\Big[\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}_{\theta^*}^\dagger(\mathsf{y}), \mathsf{d}\big)\Big].$$

Task Adapted Reconstruction: Approaches

- Sequential training: First train a reconstruction, then train the task

$$L(\theta) = \mathbb{E}_{\mathrm{x},\mathrm{y}}\Big[\ell_X\big(\mathcal{A}^\dagger_\theta(\mathrm{y}), \mathrm{x}\big)\Big].$$

$$L(\vartheta) = \mathbb{E}_{\mathrm{y},\mathrm{d}}\Big[\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}^\dagger_{\theta^*}(\mathrm{y}), \mathrm{d}\big)\Big].$$

- End-to-end training: Straight from data to task

$$L(\theta, \vartheta) = \mathbb{E}_{\mathrm{y},\mathrm{d}}\Big[\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}^\dagger_{\theta}(\mathrm{y}), \mathrm{d}\big)\Big].$$

Task Adapted Reconstruction: Approaches

- Sequential training: First train a reconstruction, then train the task

$$L(\theta) = \mathbb{E}_{\mathrm{x},\mathrm{y}}\Big[\ell_X\big(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x}\big)\Big].$$

$$L(\vartheta) = \mathbb{E}_{\mathrm{y},\mathrm{d}}\Big[\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}_{\theta*}^\dagger(\mathrm{y}), \mathrm{d}\big)\Big].$$

- End-to-end training: Straight from data to task

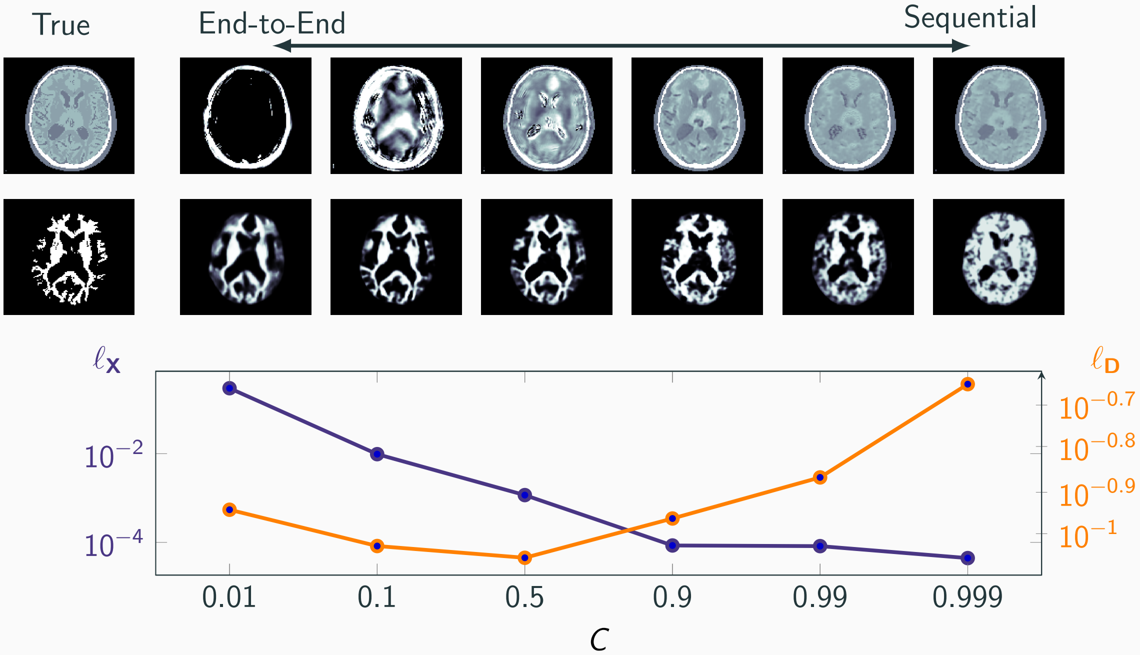
$$L(\theta, \vartheta) = \mathbb{E}_{\mathrm{y},\mathrm{d}}\Big[\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{d}\big)\Big].$$

- Task Adapted Reconstruction: Anything in between

$$L(\theta, \vartheta) = \mathbb{E}_{\mathrm{x},\mathrm{y},\mathrm{d}}\Big[C\ell_X\big(\mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{x}\big) + (1-C)\ell_D\big(\mathcal{T}_\vartheta \circ \mathcal{A}_\theta^\dagger(\mathrm{y}), \mathrm{d}\big)\Big].$$

Results

- 7 CT brain scans
  - Segmented semi-manually
  - Simulated low-dose data
- Task: Segment white matter
- Reconstruction $\mathcal{A}_\theta^\dagger$: Learned Primal-Dual
- Task $\mathcal{T}_\vartheta$: U-Net


True
End-to-End
Sequential
$\ell_{\mathbf{X}}$
$\ell_{\mathbf{D}}$
10^{-2}
10^{-4}
$10^{-0.7}$
$10^{-0.8}$
$10^{-0.9}$
10^{-1}
0.01
0.1
0.5
0.9
0.99
0.999
C

Conclusions

- Reconstruction is often seen as a "pre-processing" step in image analysis

Conclusions

- Reconstruction is often seen as a "pre-processing" step in image analysis
- Deep learning for image reconstruction $\implies$ end-to-end learning

Conclusions

- Reconstruction is often seen as a "pre-processing" step in image analysis
- Deep learning for image reconstruction $\implies$ end-to-end learning
- Combining reconstruction with segmentation seems to give the best results

Conclusions

- Reconstruction is often seen as a "pre-processing" step in image analysis
- Deep learning for image reconstruction $\implies$ end-to-end learning
- Combining reconstruction with segmentation seems to give the best results

Deep Learning and Inverse Problems

21-25 Jan 2019.