Finnish Centre of Excellence in Inverse Modelling and Imaging 2018-2025

UNIVERSITY OF OULU

UCL

DATA-DRIVEN MODEL CORRECTIONS AND LEARNED ITERATIVE RECONSTRUCTION

Andreas Hauptmann

University of Oulu

Research Unit of Mathematical Sciences

&

University College London

Department of Computer Science

Special Semester on Tomography Across the Scales

Inverse Problems on Large Scales

30 November 2022

THE JOURNEY TO SCALABLE LEARNED RECONSTRUCTIONS IN 3D PAT

Andreas Hauptmann

University of Oulu

Research Unit of Mathematical Sciences

&

University College London

Department of Computer Science

Special Semester on Tomography Across the Scales

Inverse Problems on Large Scales

30 November 2022

UNIVERSITY OF CAMBRIDGE

Sebastian Lunz
Carola-Bibiane Schönlieb

UNIVERSITY OF EASTERN FINLAND

Tanja Tarvainen

UCL

Simon Arridge

UNIVERSITY OF OULU

Jenni Poimala

Finnish Centre of Excellence in Inverse Modelling and Imaging

2018-2025

CENTRES OF EXCELLENCE
IN RESEARCH

UNIVERSITY OF OULU

UNIVERSITY OF JYVÄSKYLÄ

TAMPERE UNIVERSITY OF TECHNOLOGY

FINNISH METEOROLOGICAL INSTITUTE

UNIVERSITY OF EASTERN FINLAND

LUT
Lappeenranta
University of Technology

UNIVERSITY OF HELSINKI

Aalto University

LEARNED ITERATIVE RECONSTRUCTIONS

Classic variational approach: find x from measurement y as a minimiser of

$$x \in \arg\min_{x'} \left\{ J(x') \right\} = \arg\min_{x'} \left\{ \mathcal{D}(x'; y) + \lambda \mathcal{R}(x') \right\}.$$

$$\mathcal{D}(x; y) = \frac{1}{2} \| \mathcal{A}x - y \|_2^2$$

and

$$\nabla \mathcal{D}(x; y) := \mathcal{A}^*(\mathcal{A}x - y)$$

A simple learned gradient-like scheme would be given by

$$x_{i+1} = \mathcal{G}_{\theta_i}\left(x_i, \mathcal{A}^*(\mathcal{A}x_i - y)\right), \quad i = 0, \ldots, N-1.$$

Defines a reconstruction operator when stopped after N iterates:

$$\mathcal{A}_\theta^\dagger(y) := x_N \quad \text{where } \theta = (\theta_0, \ldots, \theta_{N-1})$$

and initialisation $x_0 = \mathcal{A}^\dagger(g)$.

[Adler & Öktem, 2018], [Putzky & Welling, 2017]

TRAINING PROCEDURE: END-TO-END

Given supervised training data $(x^{(j)}, y^{(j)}) \in X \times Y$.

Then an optimal parameter is found by

$$\min_{\theta} \frac{1}{m} \sum_{j=1}^{m} \mathrm{L}_{\theta}(x^{(j)}, y^{(j)})$$

where the loss function is given as

$$\mathrm{L}_{\theta}(x, y) := \left\| \mathcal{A}_{\theta}^{\dagger}(y) - x \right\|_X^2 \quad \text{for } (x, y) \in X \times Y.$$

Note: Computing the gradient of the loss function w.r.t. θ requires performing back-propagation through all of the unrolled iterates $i = 0, \ldots, N-1$.

PROBLEM WITH END-TO-END TRAINING?

- End-to-end training is not scalable depending on two factors:

  - ➢ Memory limitations: Standard CNN creates "copies" of image → $O(n^d)$
    Gradient check-pointing or invertible networks
    [Putzky&Welling, 2019], [Etmann, Ke, Schönlieb, 2020]

  - ➢ Operator evaluation: Repeated application of forward/adjoint operator
    - ➢ No direct work-around for "non-trivial" operators

Possible solution: Greedy (sequential) training of each iterate

- ➢ Separate evaluation of forward operator from the training task.

TRAINING PROCEDURE: GREEDY APPROACH

Given supervised training data $(x^{(j)}, y^{(j)}) \in X \times Y$.

Then an optimal parameter is found by

$$\min_{\theta} \frac{1}{m} \sum_{j=1}^{m} \mathrm{L}_{\theta}(x^{(j)}, y^{(j)})$$

where the loss function is given as

$$\mathrm{L}_{\theta}(x, y) := \left\| \mathcal{A}_{\theta}^{\dagger}(y) - x \right\|_X^2 \quad \text{for } (x, y) \in X \times Y.$$

Greedy training: Require iterate-wise optimality.

Given only a loss function for the i:th unrolled iterate:

$$\mathrm{L}_{\theta_i}(x_i, y) = \left\| \mathcal{G}_{\theta_i}\big(x_i, \mathcal{A}^*(\mathcal{A}(x_i) - y)\big) - x \right\|_X^2$$

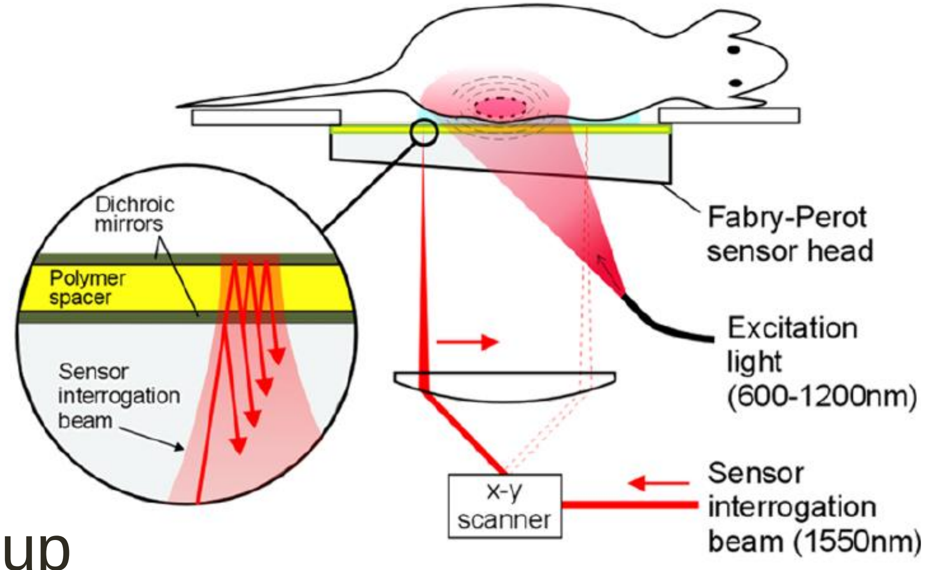
where $x_i := \mathcal{G}_{\theta_{i-1}}\big(x_{i-1}, \mathcal{A}^*(\mathcal{A}(x_{i-1}) - y)\big)$.

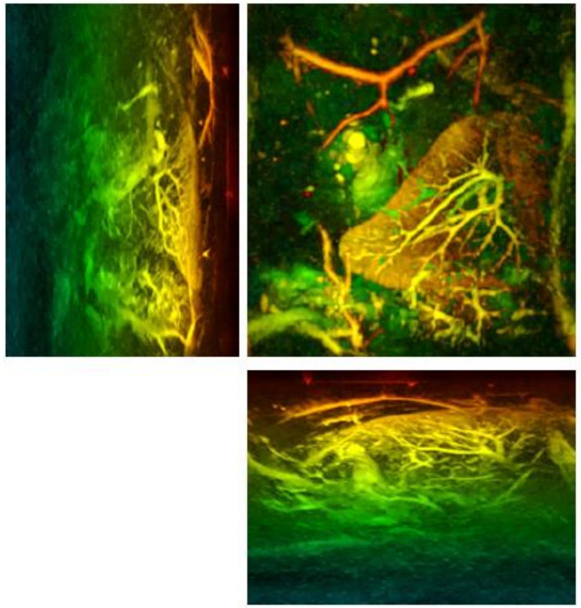
This constitutes an upper bound to end-to-end networks.

Note: Computing the gradient of the loss function w.r.t. θ requires performing back-propagation through all of the unrolled iterates $i = 0, \ldots, N-1$.

LIMITED-VIEW PHOTOACOUSTIC TOMOGRAPHY

- Fabry Perot polymer film ultrasound sensor is a planar interferometer
  - ➔ Limited-view setting
  - ➔ Sparse-sampling for speed-up

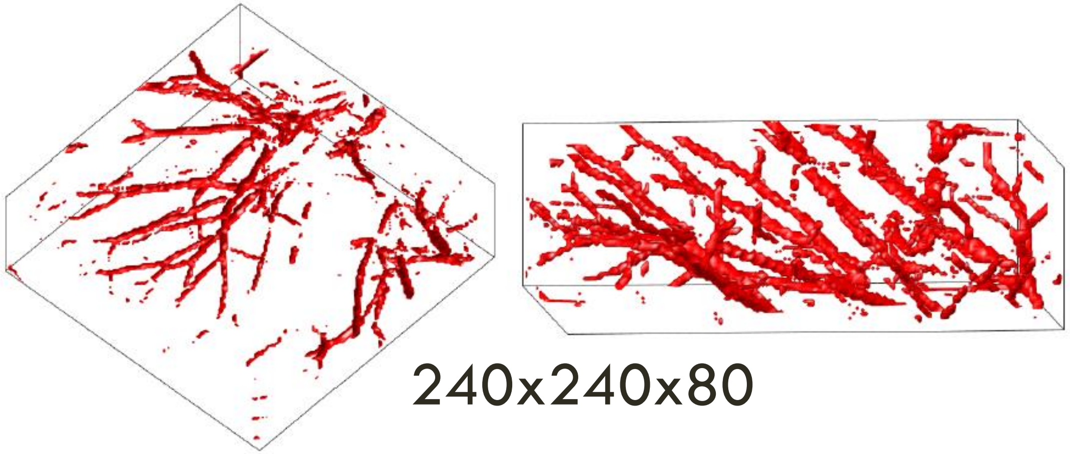




[Jathoul et al., *Nature Photonics*, 2015]

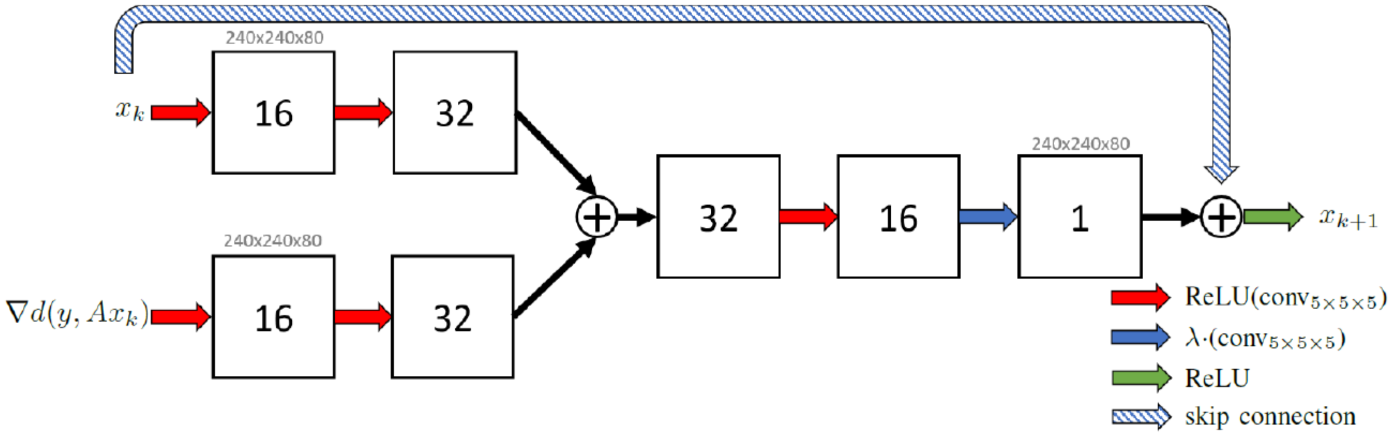
TRAINING ON VESSEL PHANTOMS

- With the computation of the gradient, total training time for 5 iterations takes 7 days
- Compare: End-to-end training would take about ∼140 days



240x240x80

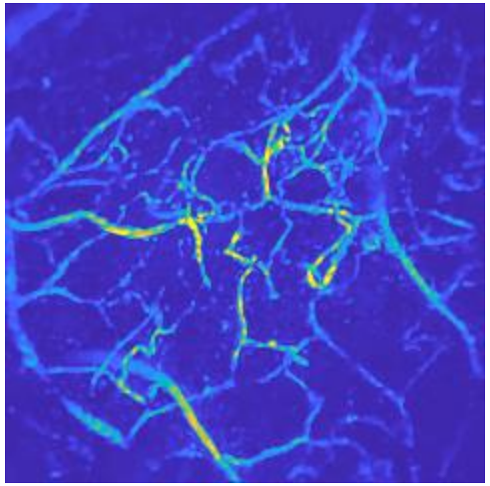
$$x_{i+1} = \mathcal{G}_{\theta_i}(x_i, \mathcal{A}^*(\mathcal{A}x_i - y))$$



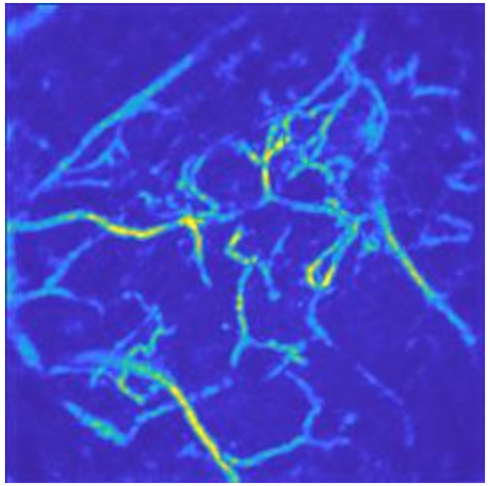
APPLICATION TO HUMAN IN-VIVO MEASUREMENTS

- Reduces reconstruction time by a factor 4 (by reduction of iterations), but reconstruction time still limited by operator evaluation.
- Considerably improves reconstruction quality

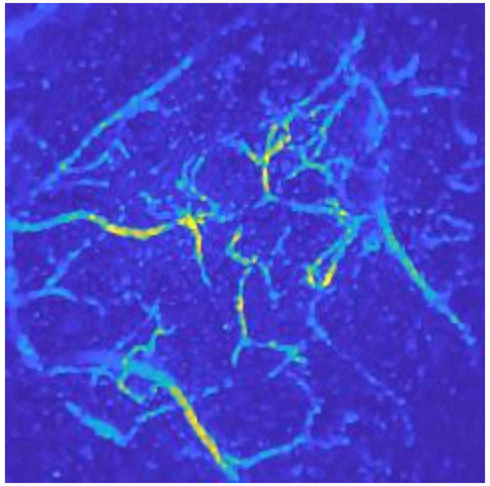
Reference
Fully-sampled data



Learned Reconstruction
4x sub-sampled, 5 Iterations,
Time: 2.5 min., PSNR: 41.40



Total Variation Reconstruction
4x sub-sampled, 20 Iterations,
Time: 10 min., PSNR: 38.05



[Hauptmann et al., *IEEE Transactions on Medical Imaging*, 2018]

UTILISING REDUCED MODELS

Can we formulate a principled way to achieve scalability and computational speed-up, using model reduction techniques?

Here we understand reduced models in a broad sense:

- To achieve a reduction in computational complexity by coarser discretisations, analytic approximations or computationally more efficient formulations.

When using a reduced/approximate model, we typically suffer a loss of accuracy. This needs to be compensated for.

- In the following we will discuss two different paradigms to compensate for the introduced approximation errors: implicit or explicit

UTILISING AN APPROXIMATE MODEL

If the measurement points lie on a plane ($x_3 = 0$), then the measurement $y = p(\mathbf{x}, t)$ there can be related to x by

$$p(x_1, x_2, t) = \frac{1}{c^2}\mathcal{F}_{k_1,k_2}\{\{\mathcal{C}_\omega\{B(k_1, k_2, \omega)\tilde{x}(k_1, k_2, \omega)\}\}\},$$

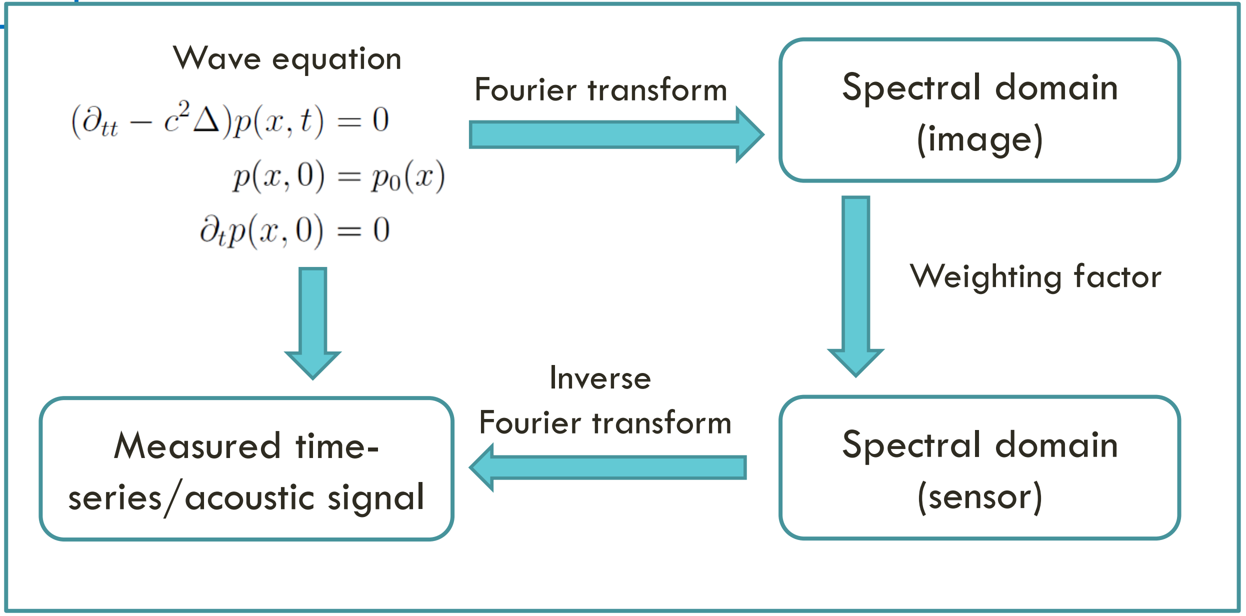
where $\tilde{x}(k_1, k_2, \omega)$ is obtained via the dispersion relation from the 3D Fourier transform of x.

The weighting factor,

$$B(k_1, k_1, \omega) = \omega / \left(\mathrm{sgn}(\omega)\sqrt{(\omega/c)^2 - k_1^2 - k_1^2}\right),$$

contains an integrable singularity.

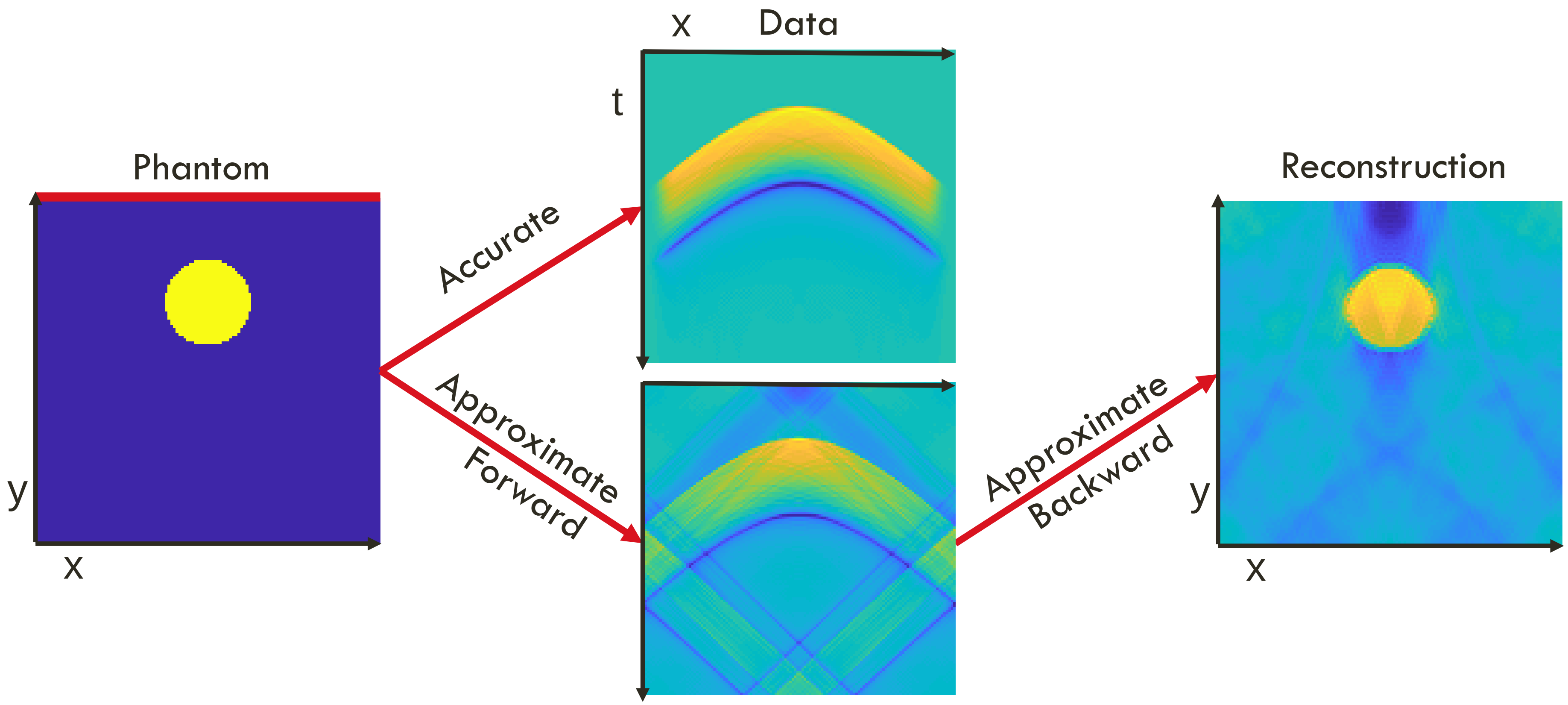
$\Rightarrow$ On a discrete rectangular grid aliasing in $p(x_1, x_2, t)$ results.



[Köstli et al., 2001], [Cox and Beard, 2005]

UTILISING A REDUCED MODEL

- Bottleneck of iterative reconstruction time is the application of the forward model
  - Use a fast approximate model in the iterative reconstruction instead (8x faster)
  - But approximate model introduces additional artefacts



UTILISING A REDUCED MODEL: IMPLICIT CORRECTION

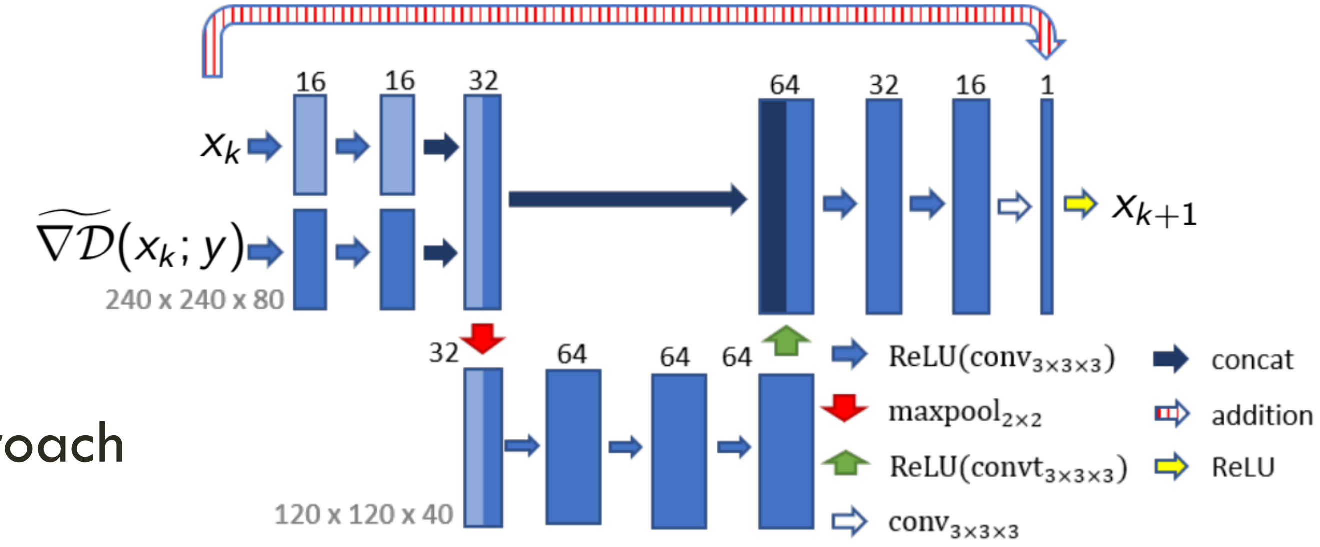
We formulate the updates now using an approximate gradient

$$x_{k+1} = \mathcal{G}_{\theta_k}(\widetilde{\nabla\mathcal{D}}(x_k; y), x_k)$$

with

$$\widetilde{\nabla\mathcal{D}}(x_k; y) := \widetilde{A}^*(\widetilde{A}x_k - y).$$

- Trained supervised on reference reconstruction from fully sampled data
- 5 iterates are trained in a greedy approach



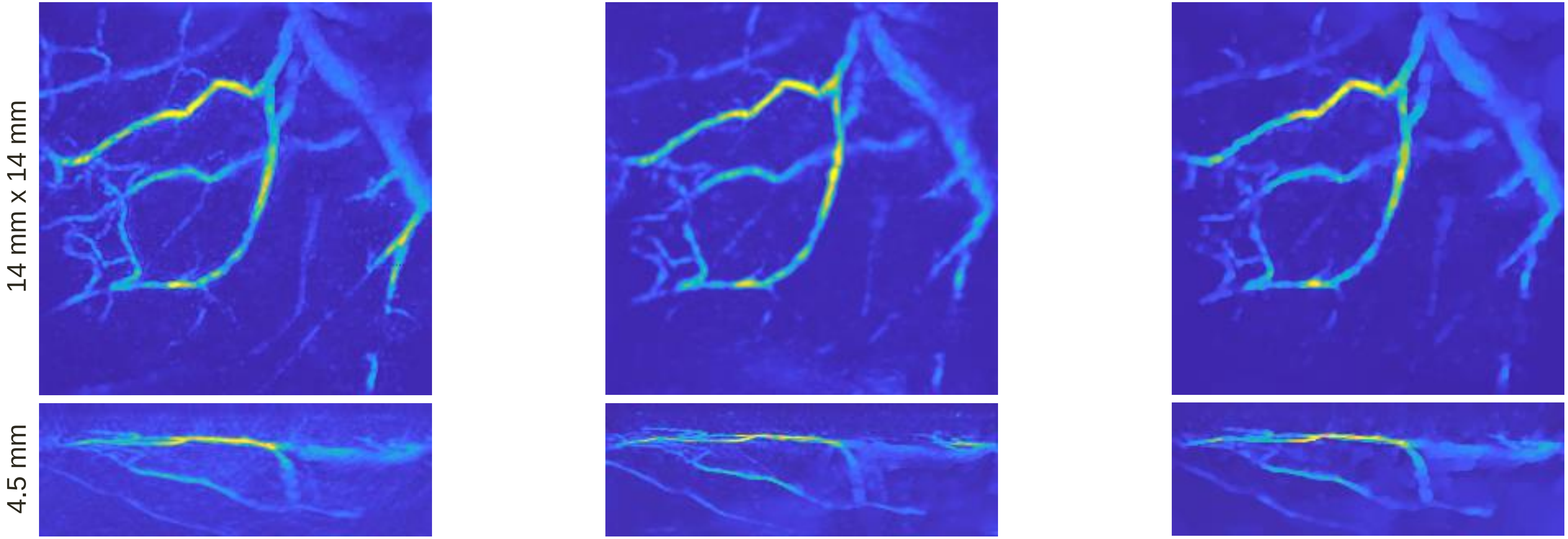
ACCELERATION BY USING AN APPROXIMATE MODEL

- Reduces reconstruction time by another factor of ~8 (→ 32x compared to TV)

Reference
Fully-sampled data

Learned Reconstruction
4x sub-sampled, 5 Iterations,
Time: 20 sec., PSNR: 42.18

Total Variation Reconstruction
4x sub-sampled, 20 Iterations,
Time: 10 min., PSNR: 41.16



[Hauptmann et al., *Machine Learning for Medical Image Reconstruction*, 2018]

LEARNING AN EXPLICIT MODEL CORRECTION

- The previous approach can be understood as an implicit model correction
  - ➔Works well, but provides limited insight

- In the following we investigate the question: Can we learn an explicit (nonlinear) model correction?
  - ➔Can we then solve a variational problem and establish convergence guarantees?

LEARNING AN EXPLICIT MODEL CORRECTION

Consider $F_\Theta : Y \to Y$, applied as a correction to $\widetilde{A}$.
Then the corrected operator is a composition

$$A_\Theta = F_\Theta \circ \widetilde{A}.$$

Ideally, we would like $A_\Theta(x) \approx Ax$ for some $x \in X$ of interest.

The primary question is: can A_Θ be (subsequently) used in a variational setting

$$x^* = \arg\min_{x \in X} \frac{1}{2} \|A_\Theta(x) - y\|_Y^2 + \lambda R(x).$$

A: Accurate model
$\widetilde{A}$: Approximate model
F_Θ: Forward correction
A_Θ: Corrected model

INCORPORATION INTO VARIATIONAL APPROACHES

We require that the solutions of the two minimisation problems, involving the operator correction A_Θ and A, are close

$$\arg\min_{x \in X} \frac{1}{2}\|A_\Theta(x) - y\|_Y^2 + \lambda R(x) \approx \arg\min_{x \in X} \frac{1}{2}\|Ax - y\|_Y^2 + \lambda R(x).$$

We consider first order methods to draw connections to learned iterative schemes.

Using a classic gradient descent scheme:

$$x_{k+1} = x_k - \gamma_k \nabla_x \left(\frac{1}{2}\|Ax_k - y\|_X^2 + \lambda R(x_k) \right).$$

Thus, we need a *gradient consistency* of the approximate gradient

$$\nabla_x \|A_\Theta(x) - y\|_X^2 \approx \nabla_x \|Ax - y\|_X^2.$$

A: Accurate model
$\widetilde{A}$: Approximate model
F_Θ: Forward correction
A_Θ: Corrected model

GRADIENT CONSISTENCY AND THE ADJOINT PROBLEM

Given a nonlinear correction operator F_Θ and the corrected operator $A_\Theta = F_\Theta \circ \widetilde{A}$ we obtain the following gradient

$$\frac{1}{2}\nabla_x \|A_\Theta(x) - y\|_2^2 = \widetilde{A}^* \left[DF_\Theta(\widetilde{A}x)\right]^* \left(F_\Theta(\widetilde{A}x) - y\right).$$

$DF_\Theta(y)$ is the Fréchet derivative of F_Θ at y, which is a linear operator $Y \to Y$.

That means, to satisfy the gradient consistency condition, we would need

$$\widetilde{A}^* \left[DF_\Theta(\widetilde{A}x)\right]^* \left(F_\Theta(\widetilde{A}x) - y\right) \approx A^*(Ax - y).$$

A: Accurate model
$\widetilde{A}$: Approximate model
F_Θ: Forward correction
A_Θ: Corrected model

However this solution comes with its own drawback:
the range of the corrected fidelity term's gradient is limited by the range of the approximate adjoint, **rng**$(\widetilde{A}^*)$. Thus, the key difficulty lies in the differences of the range of the accurate and the approximate adjoints (rather than the differences in the forward operators themselves).

A FORWARD-ADJOINT CORRECTION

To achieve a gradient consistent model correction, we need two networks instead:

$$A_\Theta := F_\Theta \circ \widetilde{A}, \quad A^*_\Phi := G_\Phi \circ \widetilde{A}^*.$$

The corrected operators can then be used to compute approximate gradients:

$$A^*(Ax - y) \approx \left(G_\Phi \circ \widetilde{A}^*\right) \left(F_\Theta(\widetilde{A}x) - y\right).$$

ESSENTIAL TOOL: GRADIENT ALIGNMENT

We can consider now the two functionals

$$\mathcal{L}(x) := \frac{1}{2}\|Ax - y\|_Y^2 + \lambda R(x),\ \mathcal{L}_\Theta(x) := \frac{1}{2}\|A_\Theta(x) - y\|_Y^2 + \lambda R(x)$$

and aim to establish a convergence result using the forward-adjoint correction.

For that purpose, we need the alignment of the gradients

$$\cos\Phi_v(x) := \frac{\langle \nabla\mathcal{L}(x), \nabla^\dagger\mathcal{L}_\Theta(x)\rangle}{\|\nabla\mathcal{L}(x)\|^2}.$$

With a slight abuse of notation, we denote the corrected gradient $\nabla^\dagger\mathcal{L}_\Theta(x) := A_\Phi^*(A_\Theta(x) - y) + \lambda\nabla R(x)$.

CONVERGENCE RESULT

$\mathcal{L}$: "Accurate" functional
$\mathcal{L}_\Theta$: "Corrected" functional
$\nabla^\dagger \mathcal{L}_\Theta$: Corrected gradient
$\hat{x}$: Minimiser of $\mathcal{L}$

A: Accurate model
$\widetilde{A}$: Approximate model
F_Θ: Forward correction
A_Θ: Corrected model
G_Φ: Adjoint correction
A^*_Φ: Corrected Adjoint

Theorem (Convergence to a neighbourhood of $\hat{x}$)

Let $\epsilon > 0$ and suitable δ (controlling the subdifferential of $\mathcal{L}_\Theta$). Assume adjoint and forward operator are fit up to a $\delta/4$-margin, i.e.

$$\|A\|_{X\to Y}\|(A - A_\Theta)(x_n)\|_Y < \delta/4, \quad \|(A^* - A^*_\Phi)(A_\Theta(x_n) - y)\|_X < \delta/4$$

for all y and x_n obtained during gradient descent over $\mathcal{L}_\Theta$. Then eventually the gradient descent dynamics over $\mathcal{L}_\Theta$ will reach an ϵ neighbourhood of the accurate solution $\hat{x}$.

[Lunz, Hauptmann, Tarvainen, Schönlieb, Arridge, *SIAM J. Imaging Sciences*, 2021]

TRAINING REGIME

A: Accurate model
$\widetilde{A}$: Approximate model
F_Θ: Forward correction
A_Θ: Corrected model
G_Φ: Adjoint correction
A^*_Φ: Corrected Adjoint

Given the forward and adjoint corrections:

$$A_\Theta := F_\Theta \circ \widetilde{A}, \quad A^*_\Phi := G_\Phi \circ \widetilde{A}^*.$$

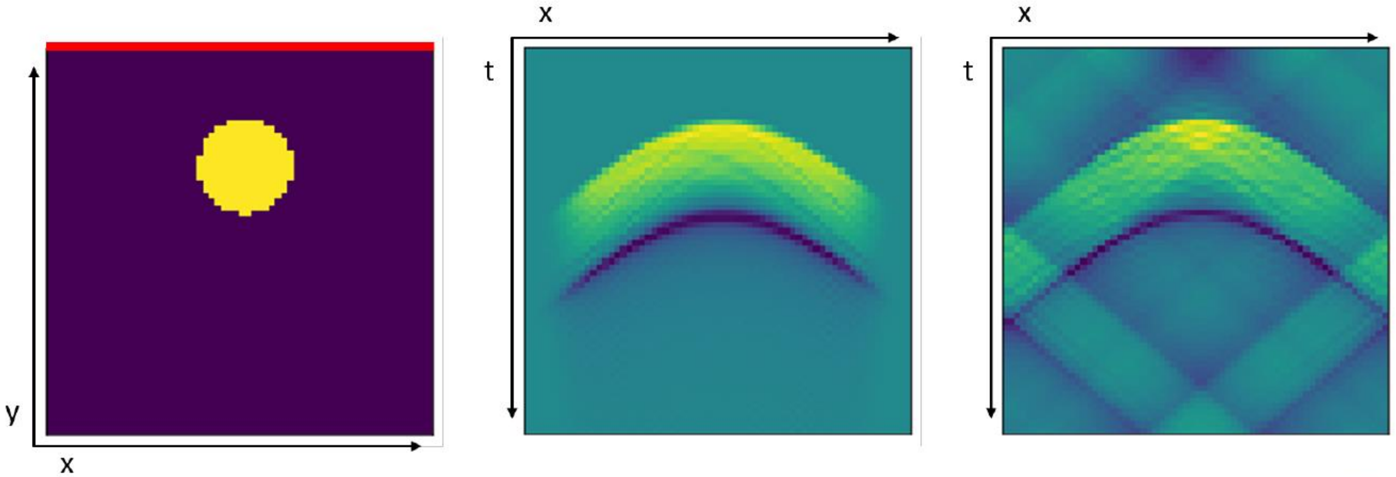
And training samples (x^i, Ax^i), we can then train the corrections:

$$\min_\Theta \sum_i \|F_\Theta(\widetilde{A}x^i) - Ax^i\|_Y \text{ and } \min_\phi \sum_i \|G_\Phi(\widetilde{A}^* r^i) - A^* r^i\|_X.$$

Note, for the adjoint correction, we choose the direction $r^i = F_\Theta(\widetilde{A}x^i) - y^i$. This ensures that the adjoint correction is trained in relevant directions for the variational problem.

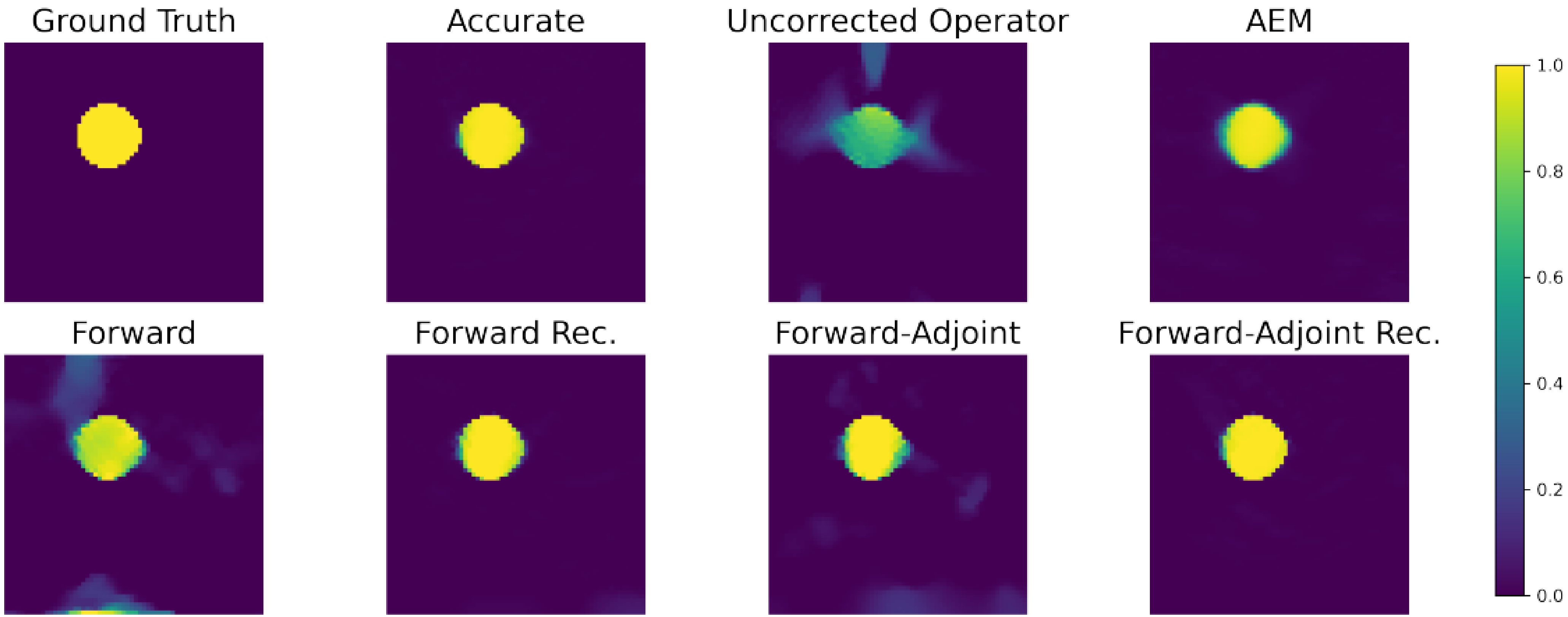
TRAINING REGIME

- Training in 2D limited-view scenario (PAT)
- Use of accurate and approximate model (FFT based)
- Train corrections on 2 simulated datasets (ball and vessel phantoms)
- Solve variational problem with total variation as regulariser



Phantom: x Accurate data: Ax Approximate data: $\widetilde{A}x$

NUMERICAL EVALUATION ON SIMPLE DATA



SOME THOUGHTS ON THE OPERATOR CORRECTION

- Approximate models can be used to speed up reconstruction time

- Implicit corrections work well within learned iterative reconstructions, but are difficult to analyse

- Explicit corrections can be incorporated into classical variational framework to obtain convergence results
  - ➔Primary limitation: Accurate operator needs to be known

- Theoretical analysis reveals problems as well as solutions: Approximate operators need correction for forward and adjoint
  ➔ Primal-dual methods

COMBINING THE GAINED KNOWLEDGE

We now aim to formulate a model-corrected learned primal-dual approach:
→ Require end-to-end training to work well (by empirical evidence)
→ We run in the aforementioned scalability issues

The originally proposed Learned Primal Dual is given by:

$$\begin{cases} q^0 = y \text{ and } x^0 \in X \text{ given} \\ q^{k+1} = \Lambda_{\phi_k}\left(q^k, Ax^k, y\right) \\ x^{k+1} = \Gamma_{\theta_k}\left(x^k, A^* q^{k+1}\right) \end{cases} \qquad \text{for } k = 0, \dots, N-1.$$

Here, $\Gamma_{\theta_k} : X \times X \to X$ and $\Lambda_{\phi_k} : Y \times Y \times Y \to Y$ are update operators (neural networks) in image (primal) and measurement (dual) space, respectively.

[Adler, Öktem, *IEEE Transactions on Medical Imaging*, 2018]

TOWARDS AN END-TO-END METHOD

We consider the variational problem

$$\widehat{x} = \arg\min_{x \in X} \|Ax - y\|_2^2 + \lambda R(x).$$

The primal dual hybrid gradient method then computes:

$$q^0 = y \text{ and } x^0 \in X \text{ given}$$

$$q_{k+1} = \frac{q_k + \sigma(A\widetilde{x}_k - y)}{1 + \sigma}$$

$$x_{k+1} = \mathrm{prox}_{R,\lambda\tau}\left(x_k - \tau A^* q_{k+1}\right),$$

$$\widetilde{x}_{k+1} = x_{k+1} + \theta(x_{k+1} - x_k).$$

TOWARDS AN END-TO-END METHOD

- Replace the accurate model A with the approximate $\widetilde{A}$
- Replace the accurate adjoint A^* with the fast inverse $A^\dagger$
- Include the model correction $F_\theta(\widetilde{A})$
- Replace the proximal operator with a network G_ϕ
- Use weight sharing (also reduces memory foot print)

$$q^0 = y \text{ and } x^0 \in X \text{ given}$$

$$q_{k+1} = \frac{q_k + \sigma(A\widetilde{x}_k - y)}{1+\sigma}$$

$$x_{k+1} = \text{prox}_{R,\lambda\tau}\left(x_k - \tau A^* q_{k+1}\right),$$

$$\widetilde{x}_{k+1} = x_{k+1} + \theta(x_{k+1} - x_k).$$

TOWARDS AN END-TO-END METHOD

- Replace the accurate model A with the approximate $\widetilde{A}$
- Replace the accurate adjoint A^* with the fast inverse $A^\dagger$
- Include the model correction $F_\theta(\widetilde{A})$
- Replace the proximal operator with a network G_ϕ
- Use weight sharing (also reduces memory foot print)

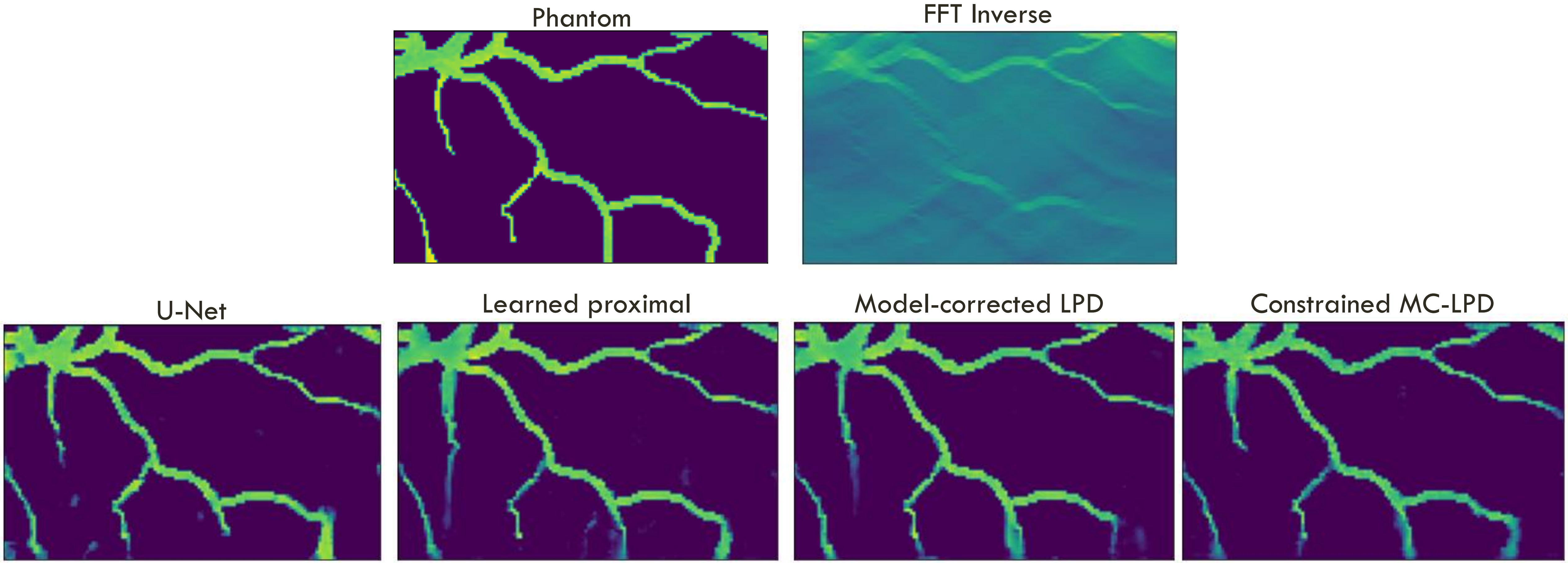
We then obtain a model-corrected learned primal dual as:

$$q_{k+1} = \frac{q_k + \sigma(F_\theta(\widetilde{A}x_k) - y)}{1+\sigma}$$

$$x_{k+1} = G_\phi\left(x_k - \tau A^\dagger q_{k+1}\right).$$

PRELIMINARY RESULTS IN 2D

- We trained the model in 2D for a resolution of 120x80 in only 1 hour
- Models are implemented using pytorch with full support of automatic differentiation



FINAL REMARKS

Extension and training for 3D and in-vivo measurements is ongoing (promising!)
→Full approach with constrained training soon on arXiv

Convergence and stability guarantees depend on:

- Choice of loss function for the model correction
- Choices for the "proximal network"

- See also the survey paper on *convergent learned reconstructions*:
  [Mukherjee, Hauptmann, Öktem, Pereyra, Schönlieb, *IEEE Signal Processing Magazine* (to appear)]

ACADEMY
OF FINLAND

EPSRC
Engineering and Physical Sciences
Research Council

Thank you for your attention