Computed tomography: Advanced reconstruction, software and applications

Jakob Sauer Jørgensen

Senior Researcher, DTU Compute

Guest lecture – DTU course 22485
6 November 2023

Joint work with

Evelina Ametova, Genoveva Burca, Sarah Fisher, Gemma Fardell, Parmesh Gajjar, Danny Holmes, Daniil Kazantsev, Bill Lionheart, Evangelos Papoutsellis, Edoardo Pasca, Martin Turner, Ryan Warr & Phil Withers

Main take-away messages

Filtered back-projection works really well!
If data is good, look no further.

If data is bad, iterative reconstruction methods *may* help...
Different kinds of data need different methods.

Core Imaging Library (CIL) is an open-source Python package with a wide range of iterative tomographic reconstruction methods.

My background

Computational algorithms and software for Computed Tomography, inverse problems and Uncertainty Quantification

2009	PhD Student, DTU
2013	Postdoc, DTU – HD-Tomo project
2015	Postdoc, Manchester X-ray Imaging Facility, Dept. Materials, UK
2018	Presidential Fellow (tenure track), Dept. Maths, University of Manchester, UK
2020	Senior Researcher, DTU Compute

Leading development of two software packages

Imaging Inverse problems including tomography

- Examples shown today
- https://ccpi.ac.uk/cil

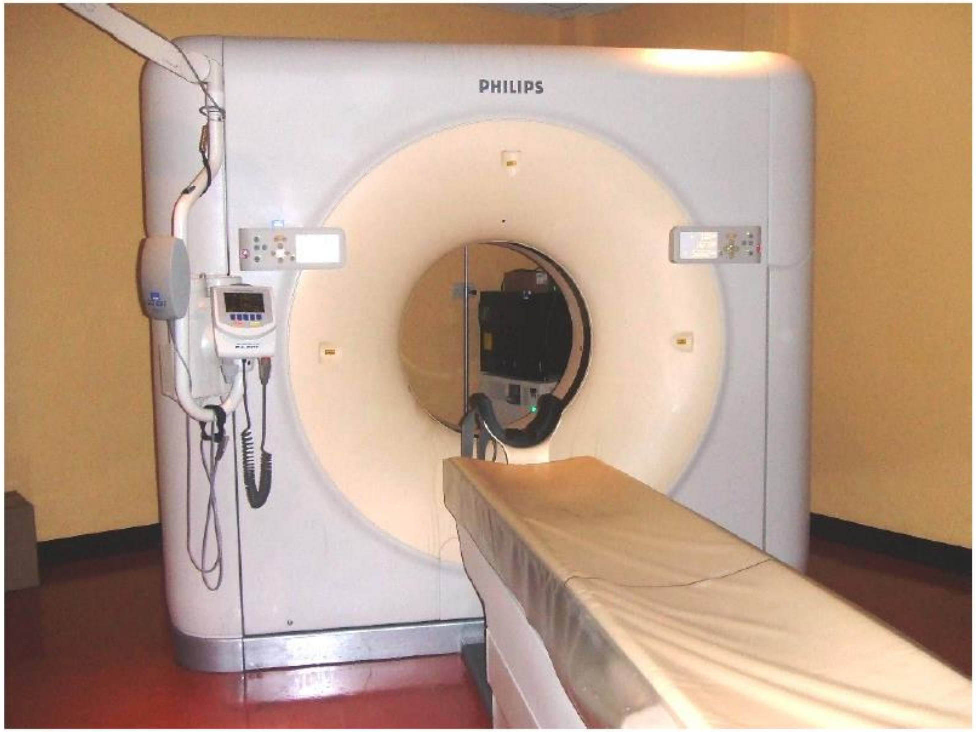
CIL
CORE IMAGING LIBRARY

Uncertainty Quantification for Inverse Problems

- Very briefly mentioned today
- https://cuqi-dtu.github.io/CUQIpy/

CUQIpy

CT applications - medical

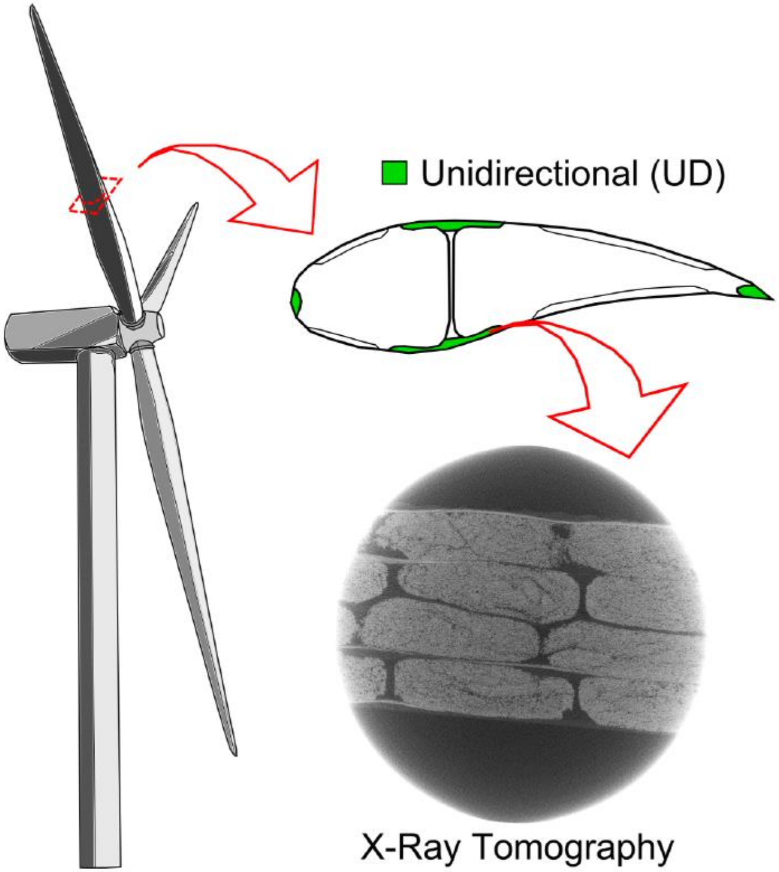


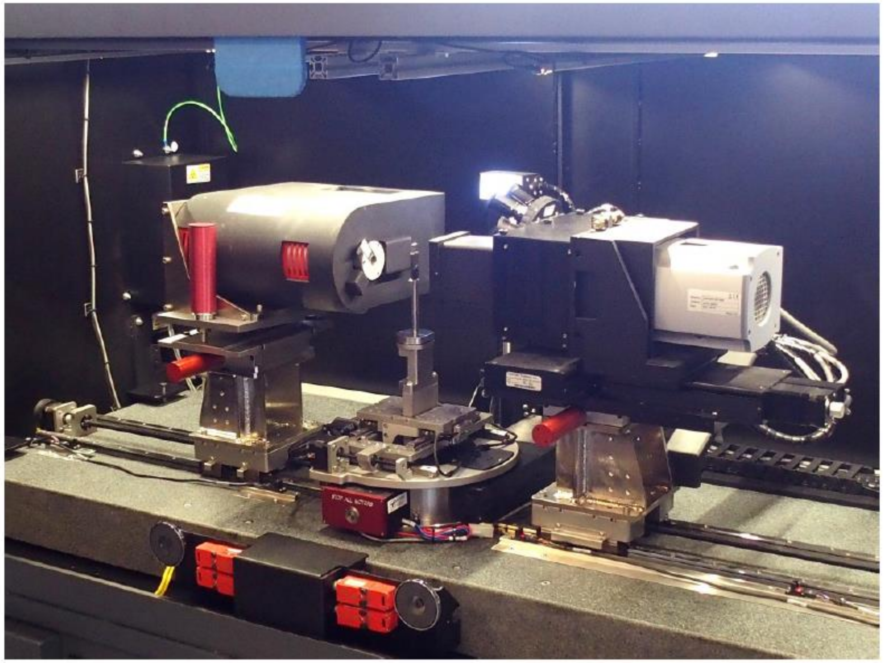
CT applications – security screening e.g. Luggage in airports

Jakob Sauer Jørgensen | jakj@dtu.dk

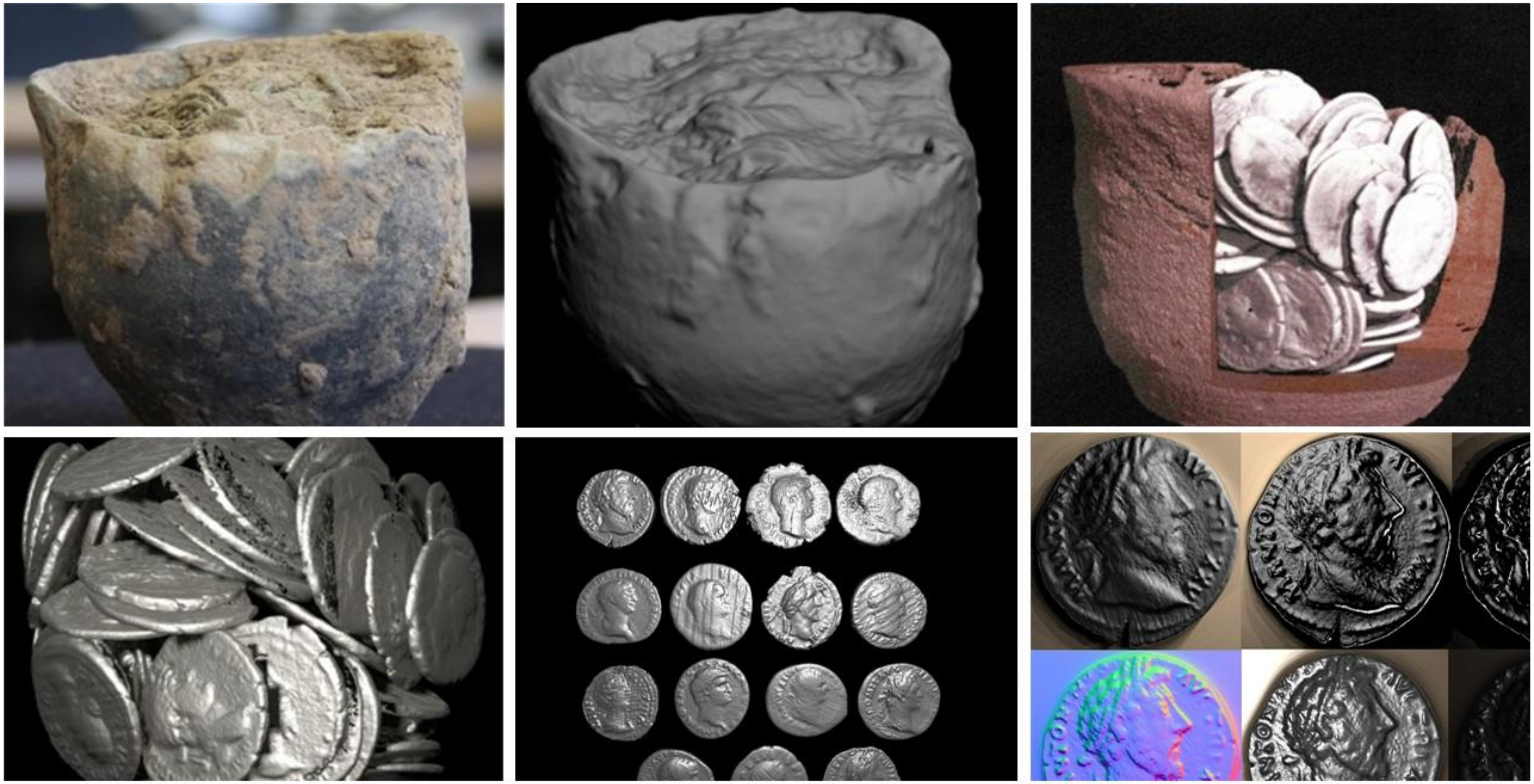
CT applications – material micro-structure

Macroscopic properties (strength, elasticity, …) depend on micro-structure
Wind turbine blades, batteries, …



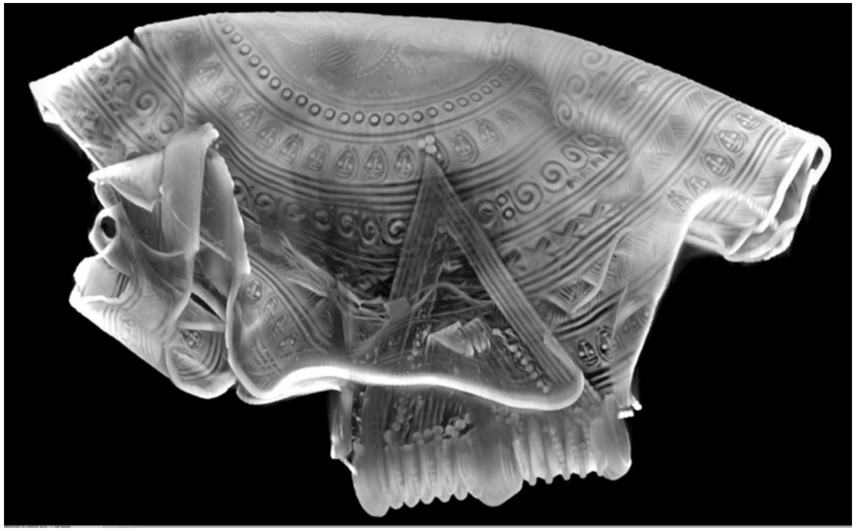


CT applications – cultural heritage



Miles, J., Mavrogordato, M., Sinclair, I., Hinton, D., Boardman, R., and Earl, G. (2016). The use of computed tomography for the study of archaeological coins, *Journal of Archaeological Science: Reports* 6: 35-41

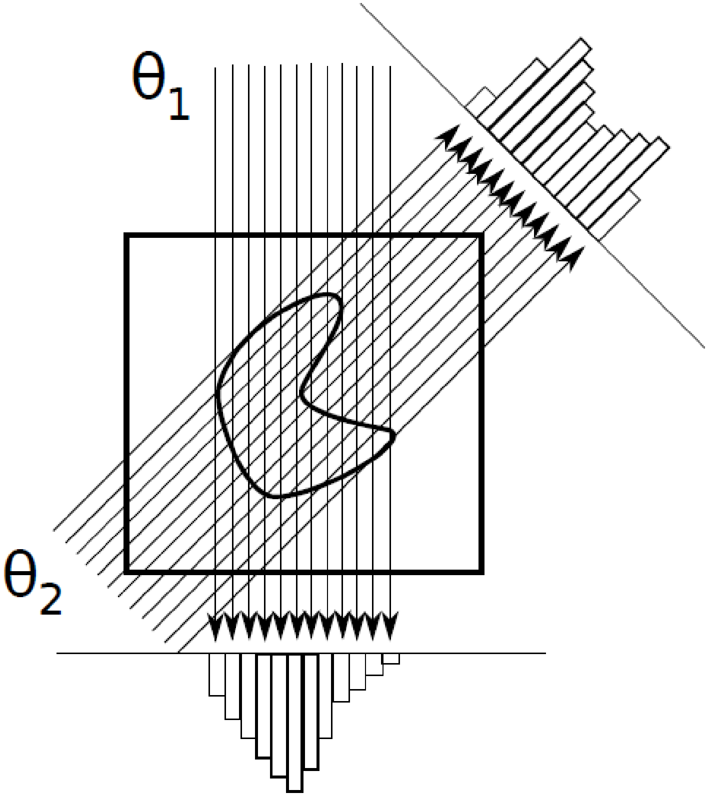
Vindelev gold
DTU 3D Imaging Centre
Virtual unfolding
DTU Dynamo vol. 74



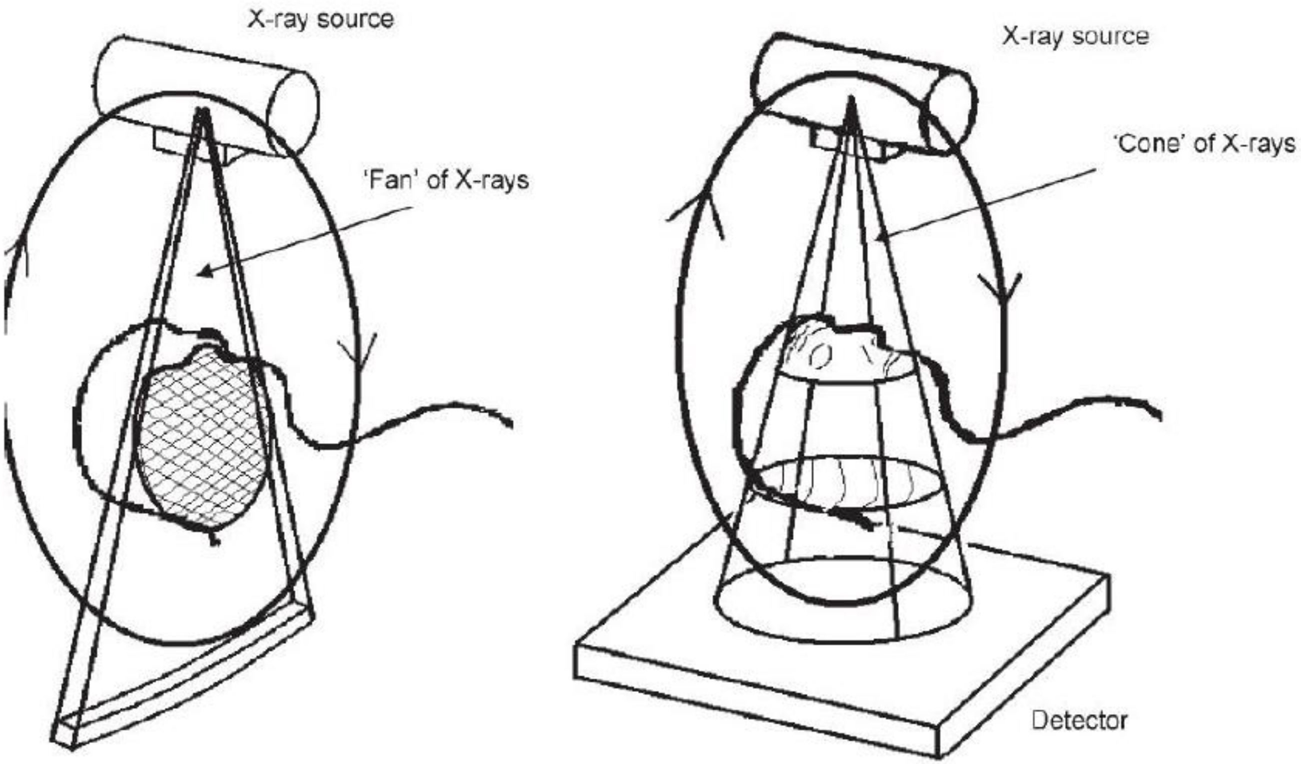
One of the researchers' first results, the unfolding of X17, shows that parts of the bracteate can be partially unfolded. However, due to artifacts on the scan, it is difficult to recreate all the edges and lines of the medallion. Photo: DTU

CT: Imaging from projections

- Projections are measured around an object using X-rays.
- Goal is to reconstruct the object from the projections.
- Simplest is 2D *parallel beam geometry*, which we focus on.
- Used in early scanners and in large-scale synchrotron facilities.



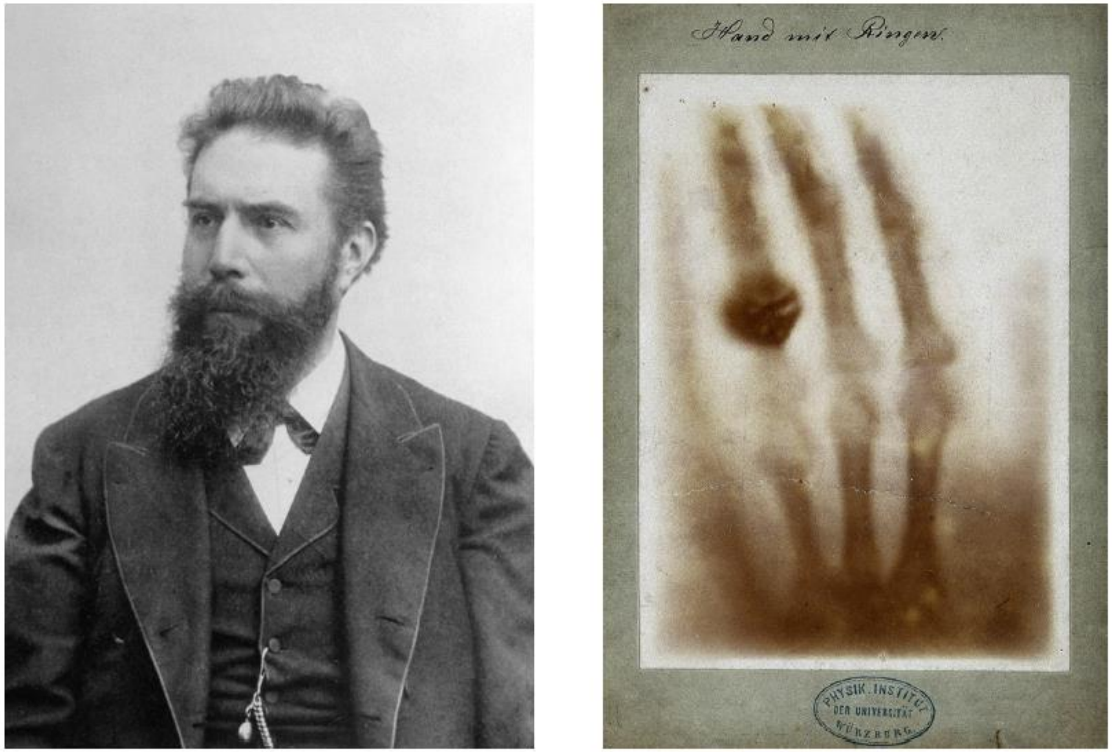
Cone-beam geometry often used



- Cone-beam (medical CT scanners, lab-based micro-CT, etc.)
- Cone-beam restricted to central slice → fan-beam
- Move source far away *approx* parallel-beam (synchrotron)

Contrast mechanism: X-ray attenuation

“Heavier” matter attenuate X-rays more: air – tissue – bone – metal.
Quantified by so-called linear attenuation coefficient μ.

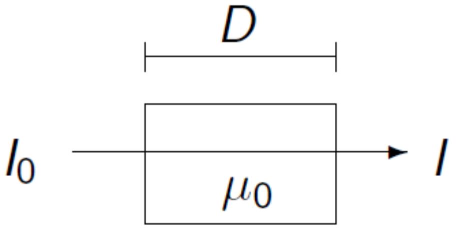


Wilhelm Conrad Röntgen and the first X-ray image ever taken showing his wife’s hand (1895).

Lambert-Beer law of attenuation

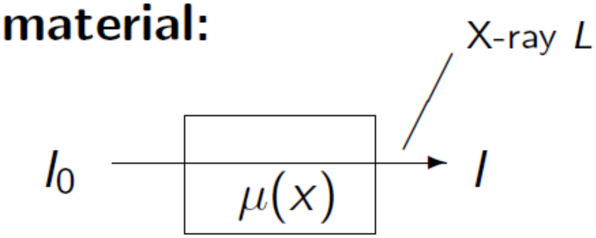
Homogeneous material:

$$I = I_0 \exp\left\{ -\mu_0 D \right\}$$



Non-homogeneous (more interesting) material:

$$I = I_0 \exp\left\{ -\int_L \mu(x) dx \right\}$$



Rearrange to line integral form:

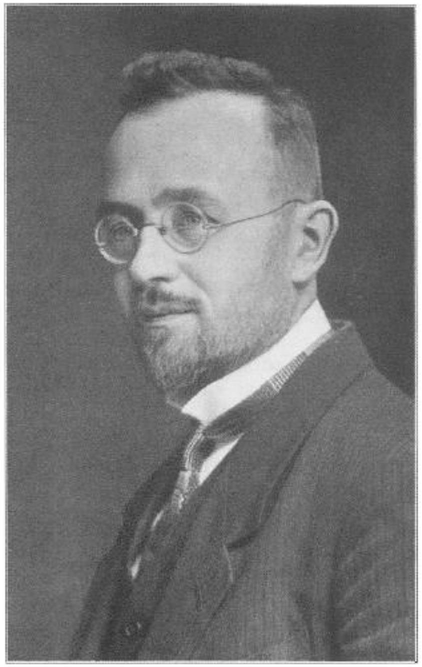
$$-\log \frac{I}{I_0} = \int_L \mu(x) dx$$

I_0 is the incident flux and $\mu(x)$ is the absorption coefficient..
The intensity I is called the *transmission*, while the corresponding $-\log(I/I_0)$ is called the *absorption* or *projection*.

Origin of tomography

Original reference:

Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. (Johann Radon, 1917):



Dr J. Radon

An object can be reconstructed **perfectly** from a **full** set of line integrals.

Radon transform

Object $f(x, y)$:

- contained in a unit disk of radius 1

Line of integration $L_{\theta,s}$ given by:

- θ: angle of line to be projected onto
- s: position on line

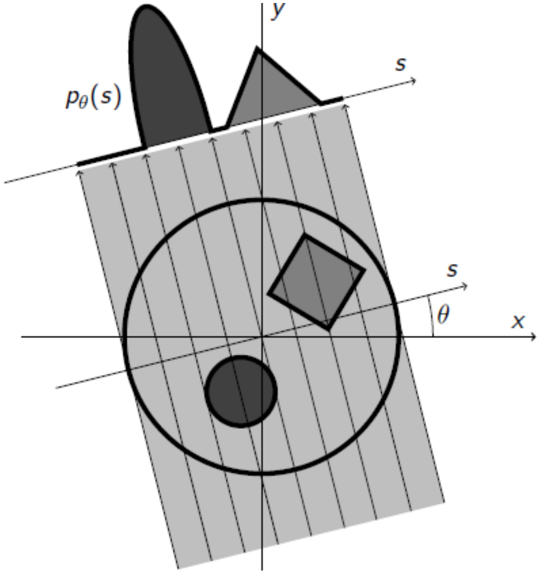
Projection: all line integrals at angle θ:

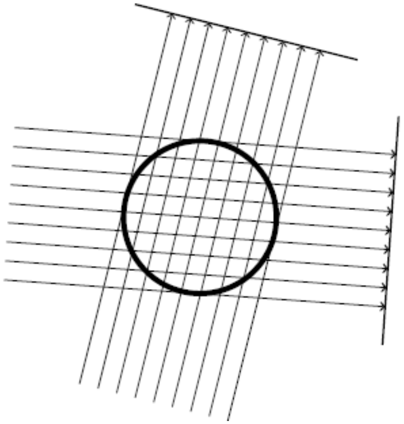
$$p_\theta(s) = \int_{L_{\theta,s}} f(x,y)\, d\ell \quad \text{for} \quad s \in [-1, 1].$$

The Radon transform is:

$$[\mathcal{R}f](\theta, s) = p_\theta(s) = \int_{L_{\theta,s}} f(x,y)\, d\ell$$

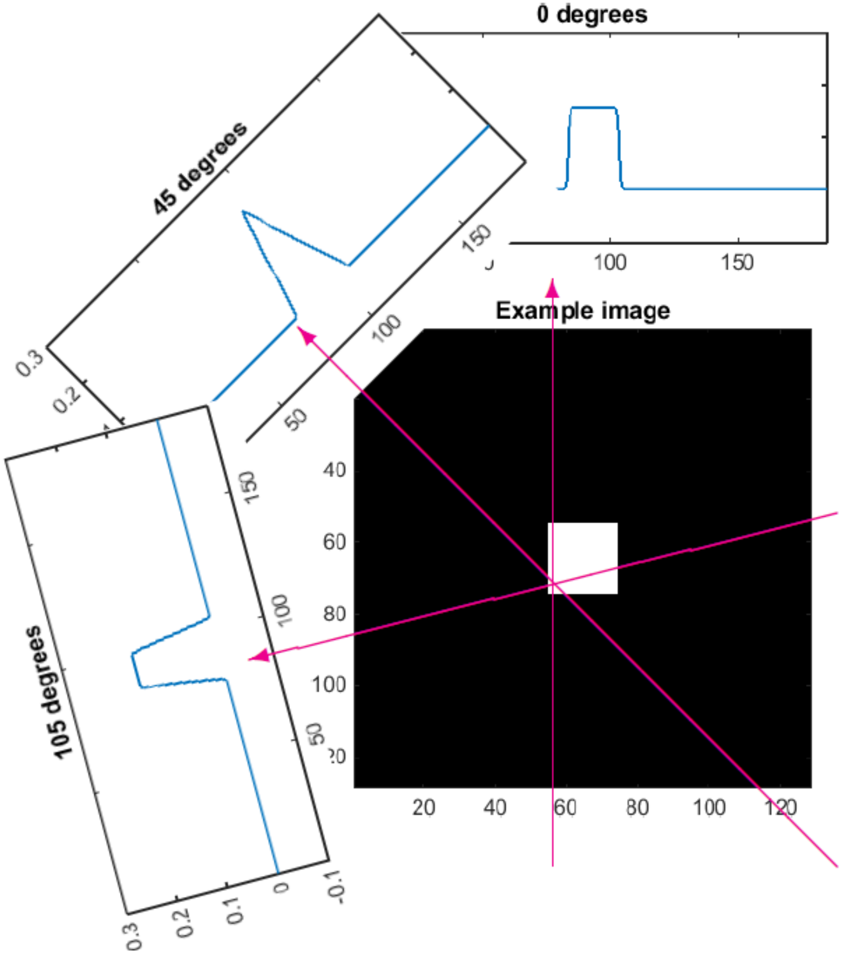
for $\theta \in [0°, 360°[$ and $s \in [-1, 1]$.





Projections all around the object

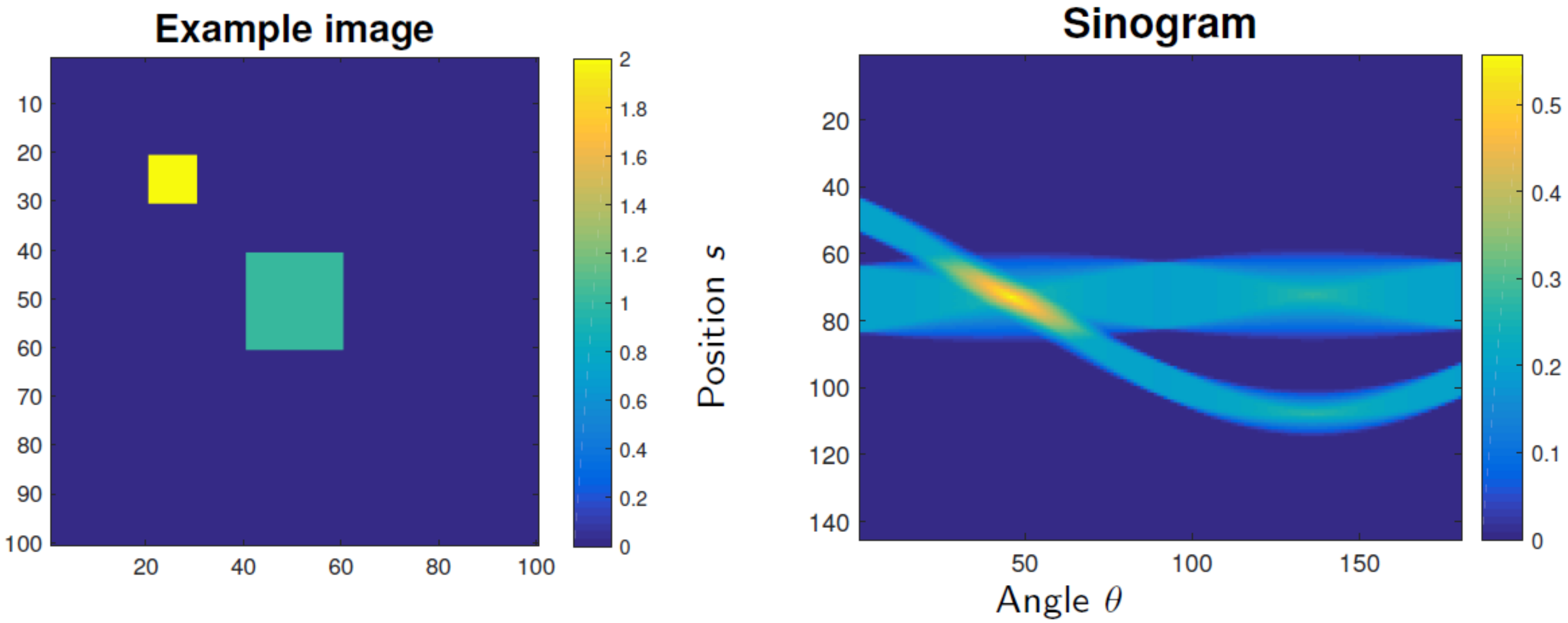
The Radon transform describes the forward problem of how (ideal) X-ray projection data arises in a parallel-beam scan geometry.



Jakob Sauer Jørgensen | jakj@dtu.dk

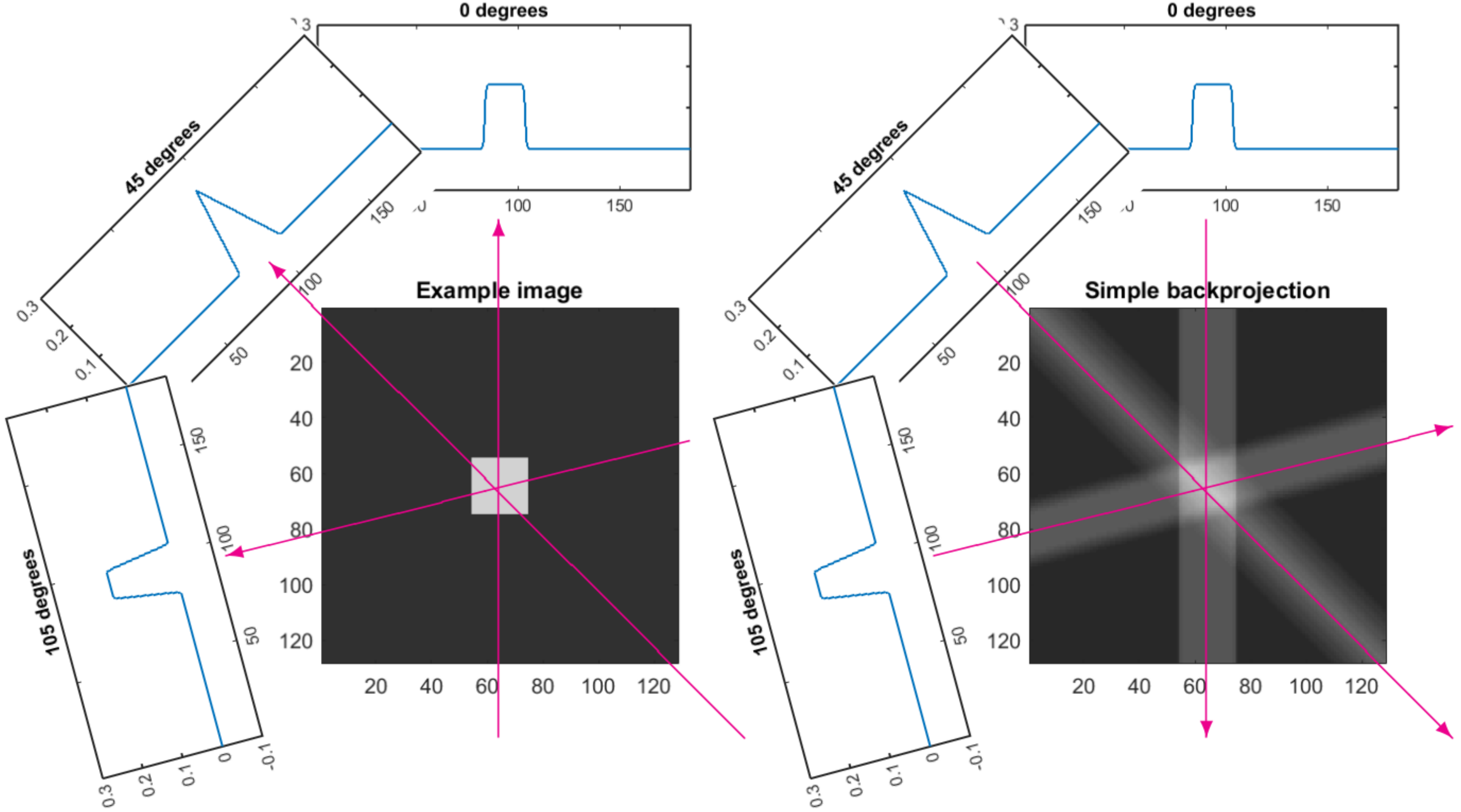
Image and sinogram

The output of the Radon transform is called a sinogram:

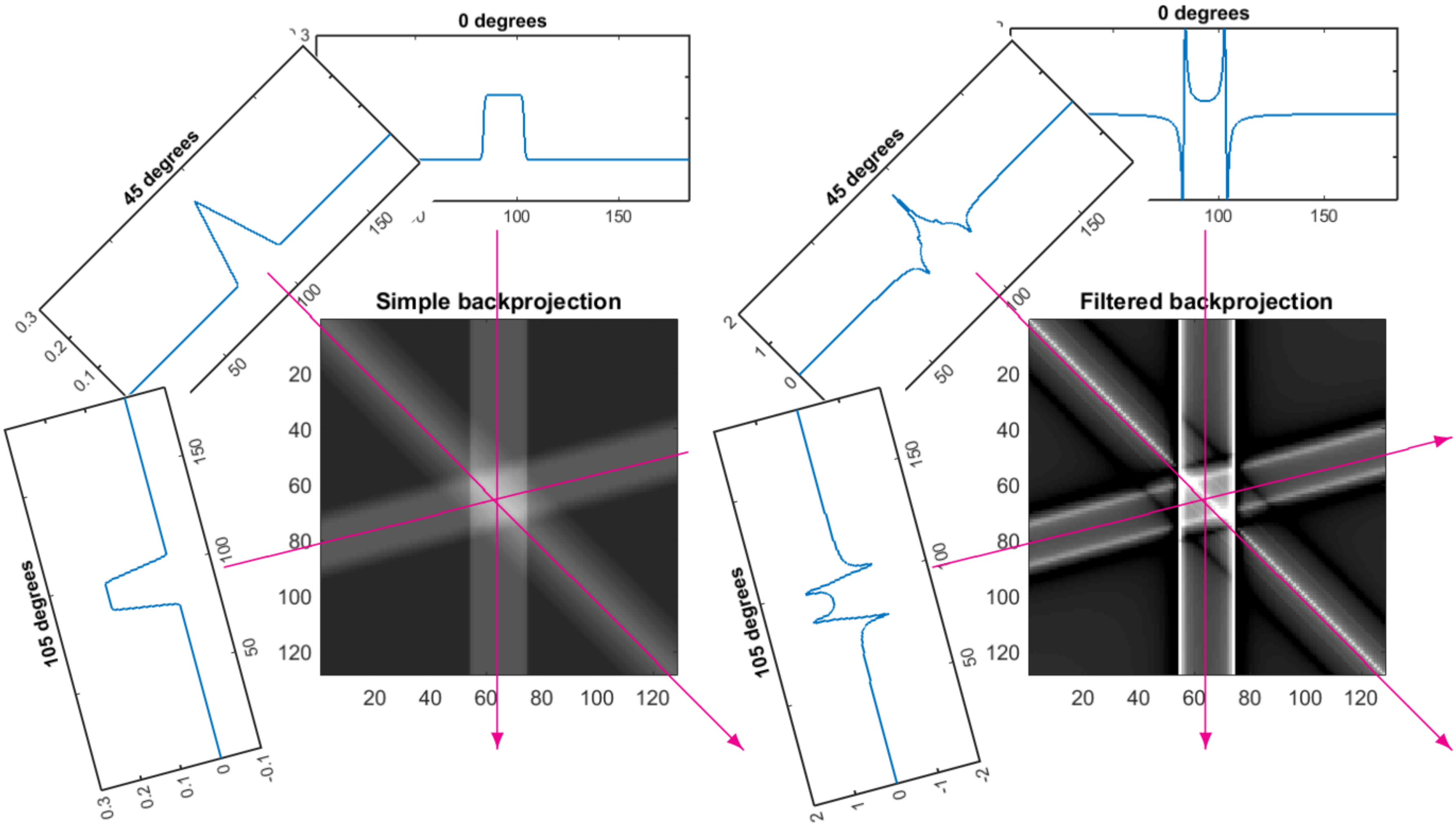


Note that [0°, 180°] captures all necessary projections of the object. The angular range [180°, 360°] gives a "mirror image."

The classical reconstruction method: Filtered back-projection (FBP)



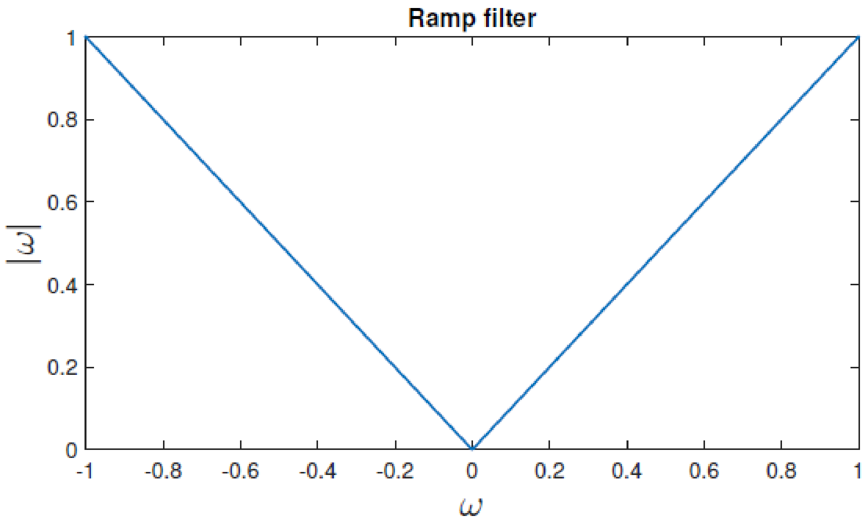
The classical reconstruction method: Filtered back-projection (FBP)

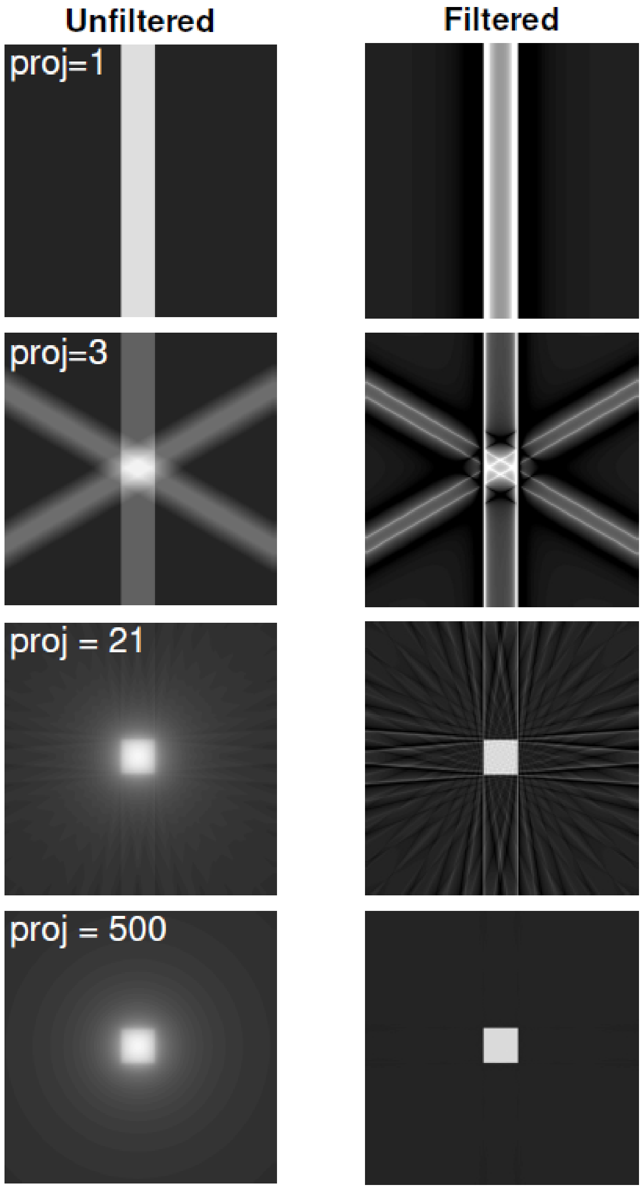


Recap Filtered back-projection (FBP)

Projections must be filtered with a "ramp" filter before back-projection.

In Fourier domain: $|\omega|$





Pros and cons of FBP

Strengths:

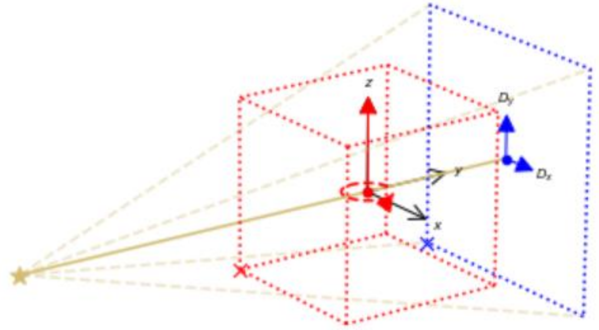
- Fast: Based on FFT and a single back-projection
- Few parameters to adjust
- Conceptually easy to understand and implement
- Reconstruction behavior well understood
- Typically works very well (for complete/good data)

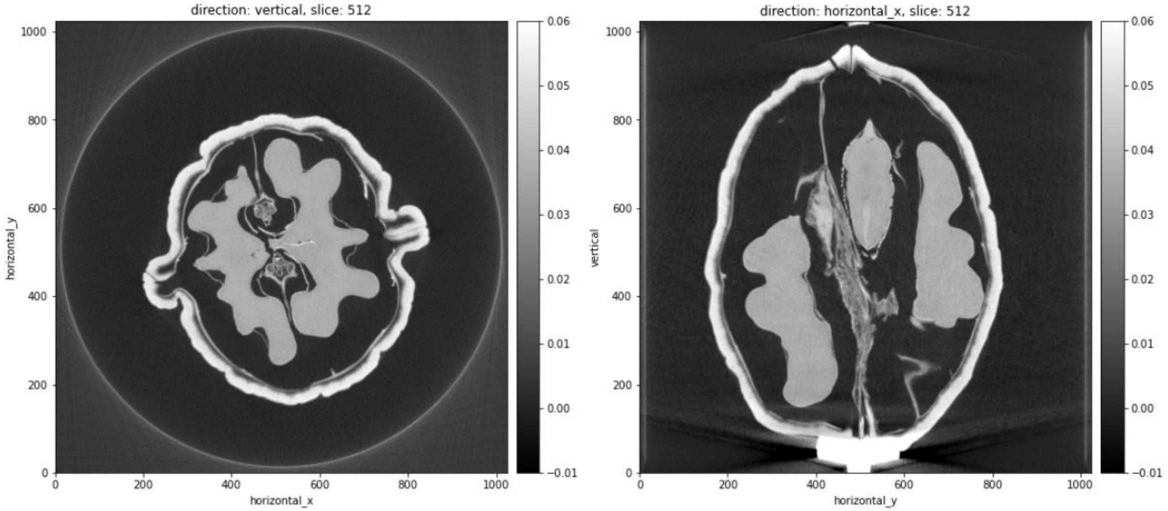
Weaknesses:

- Large number of projections required.
- Full angular range required (limited angle problem)
- Only modest amount of noise in data can be tolerated.
- Fixed scan geometries – others require own inversion formulas
- Cannot make use of prior knowledge such as non-negativity.

Core Imaging Library (CIL) for CT and other inverse problems

```
data = ZEISSDataReader(filename).read()
data = TransmissionAbsorptionConverter()(data)
show_geometry(data.geometry)
recon = FDK(data).run()
show2D(recon)
```



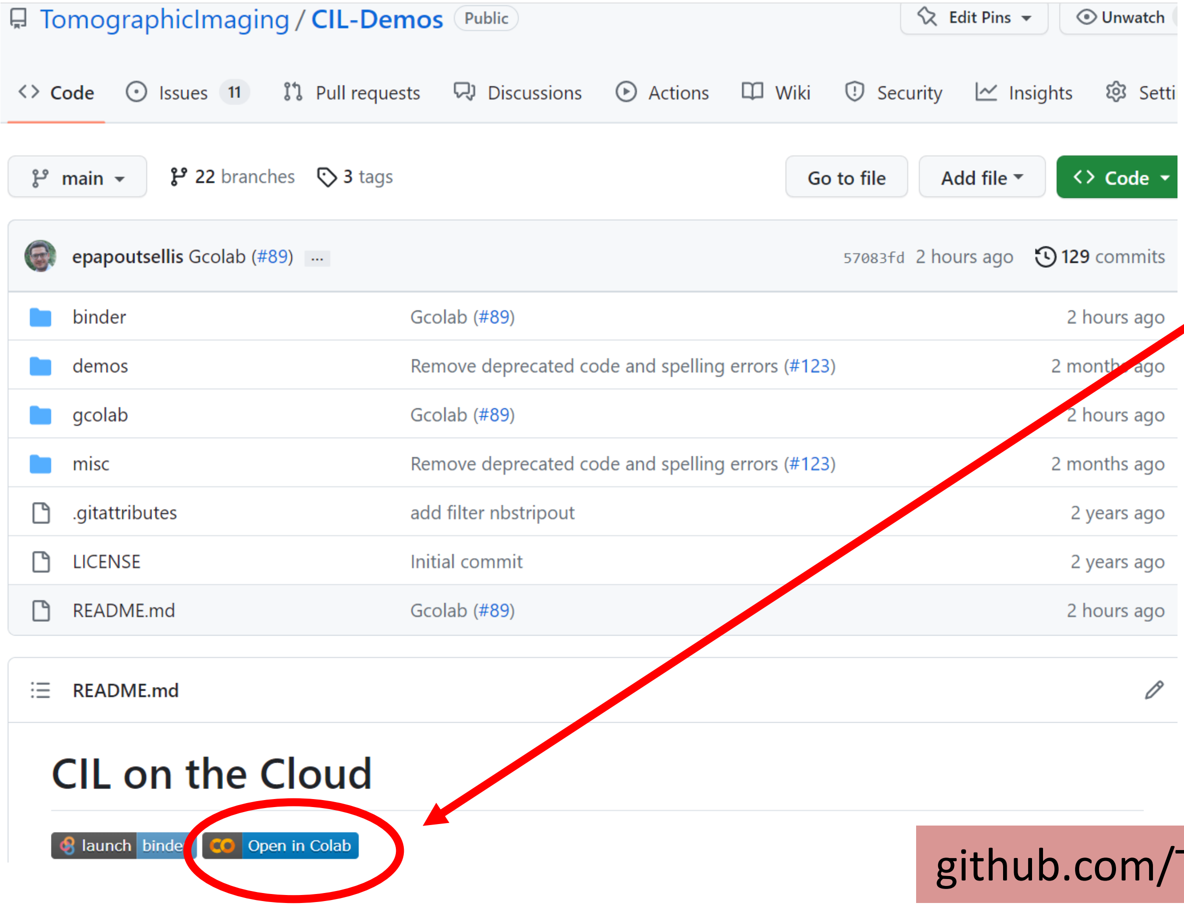


- Data readers/writers
- Pre-processing tools
- TIGRE and ASTRA backend
- 2D, 3D and 4D data
- *Near math* optimisation syntax
- Visualisation

ccpi.ac.uk/CIL

Jørgensen et al. 2021: *Core Imaging Library - Part I: a versatile Python framework for tomographic imaging*, Phil. Trans. R. Soc. A, **379**, 20200193: https://doi.org/10.1098/rsta.2020.0192

Try CIL online



TomographicImaging / CIL-Demos Public

epapoutsellis Gcolab (#89) … 57083fd 2 hours ago 129 commits

binder	Gcolab (#89)	2 hours ago
demos	Remove deprecated code and spelling errors (#123)	2 months ago
gcolab	Gcolab (#89)	2 hours ago
misc	Remove deprecated code and spelling errors (#123)	2 months ago
.gitattributes	add filter nbstripout	2 years ago
LICENSE	Initial commit	2 years ago
README.md	Gcolab (#89)	2 hours ago

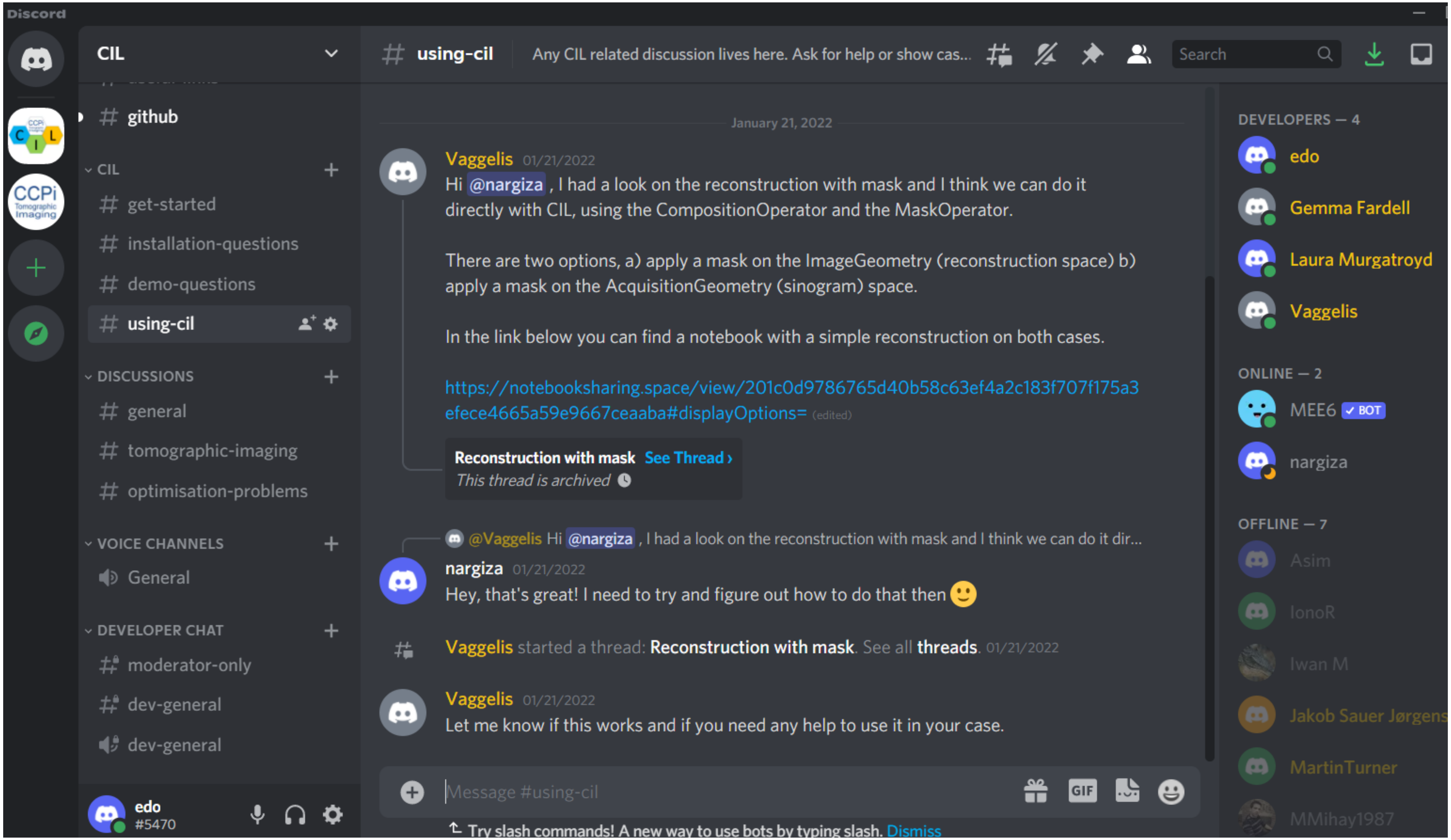
README.md

CIL on the Cloud

CIL on Google Colab – with GPU!

github.com/TomographicImaging/CIL-Demos

CIL Discord community



January 21, 2022

Vaggelis 01/21/2022
Hi @nargiza , I had a look on the reconstruction with mask and I think we can do it directly with CIL, using the CompositionOperator and the MaskOperator.

There are two options, a) apply a mask on the ImageGeometry (reconstruction space) b) apply a mask on the AcquisitionGeometry (sinogram) space.

In the link below you can find a notebook with a simple reconstruction on both cases.

https://notebooksharing.space/view/201c0d9786765d40b58c63ef4a2c183f707f175a3efece4665a59e9667ceaaba#displayOptions= (edited)

Reconstruction with mask See Thread ›
This thread is archived

@Vaggelis Hi @nargiza , I had a look on the reconstruction with mask and I think we can do it dir...

nargiza 01/21/2022
Hey, that's great! I need to try and figure out how to do that then 🙂

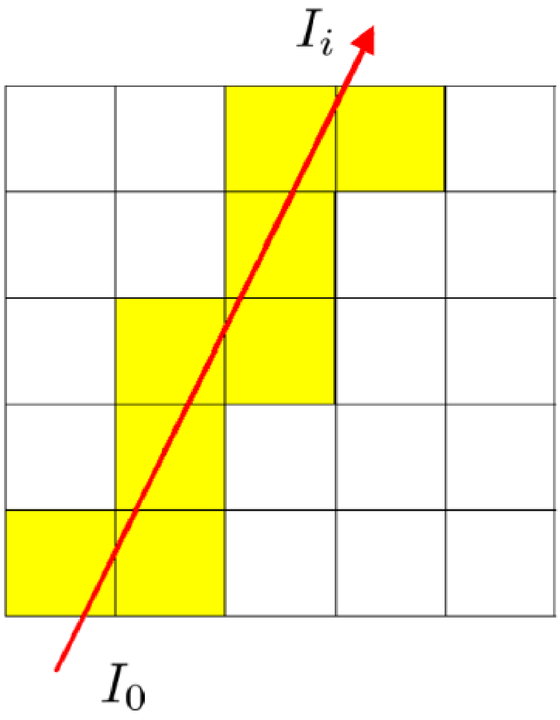
Vaggelis started a thread: **Reconstruction with mask**. See all **threads**. 01/21/2022

Vaggelis 01/21/2022
Let me know if this works and if you need any help to use it in your case.

discord.gg/9NTWu9MEGq

- Demo of FBP using CIL

Imaging model for iterative reconstruction



Beer-Lambert for ith ray (along line L_i):

$$\int_{L_i} \mu \, ds = -\log \frac{I_i}{I_0} = b_i$$

Assume object constant in each pixel:

- x_j is the jth pixel value.
- a_{ij} is path length through jth pixel.

Approximate line integral by sum:

$$\sum_j a_{ij} x_j = b_i$$

Extremely large set of linear equations:

$$Ax = b$$

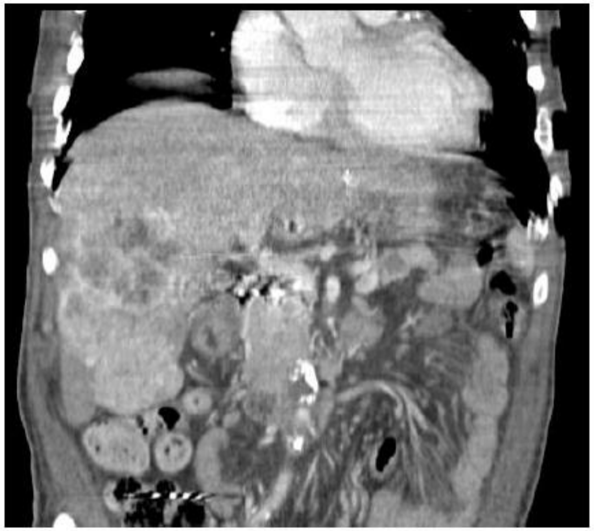
Operator *A*:

- Direct *Ax*: Projection
- Adjoint $A^T b$: Backprojection

Iterative reconstruction is based on optimisation problems and algorithms

Discrete imaging model:

$$Ax = b$$



Typical CT images:

- Regions of homogeneous tissue.
- Separated by sharp boundaries.

Reconstruction by regularization:

$$x^\star \quad = \quad \underset{x}{\operatorname{argmin}} \quad \underset{\searrow \text{ data fidelity}}{\mathcal{D}(Ax, b)} \quad + \quad \lambda \cdot \underset{\searrow \text{ regularizer}}{\mathcal{R}(x)}$$

Most basic optimization problem (no regularization):

$$u^\star = \arg\min_{u} \| Au - b \|_2^2 \quad = \sum_i ((Au)_i - b_i)^2$$

Solve optimization problem iteratively

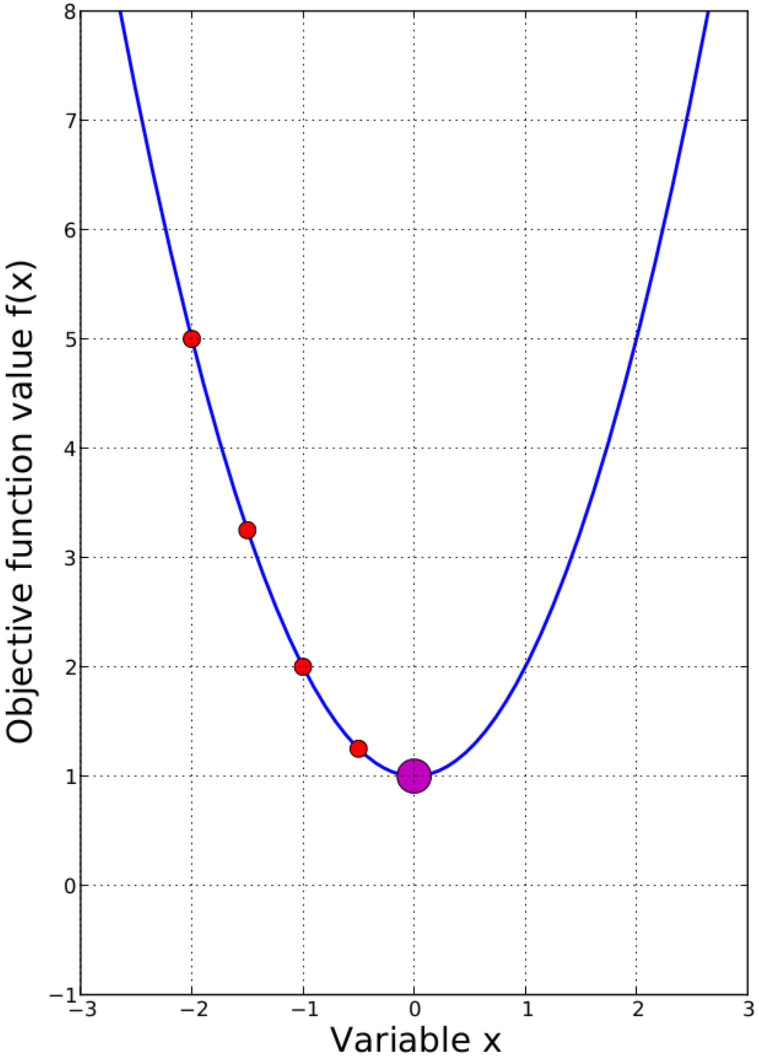
- Now how do we find this best u, the minimizer:

$$u_{\mathrm{LS}}: \quad \min_{u} \|Au - b\|_2^2$$

- We use an *iterative method*:

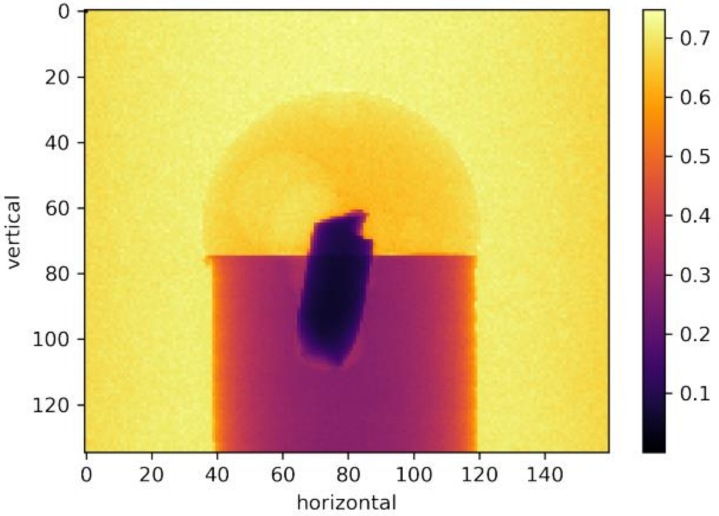
  1. Start from initial estimate $u^{(0)}$.
  2. For $k = 1, 2, \ldots$ Update $u^{(k)}$ to obtain $u^{(k+1)}$ with smaller misfit $\|Au - b\|_2^2$.
  3. Abort when minimizer is found.

- Many different algorithms.

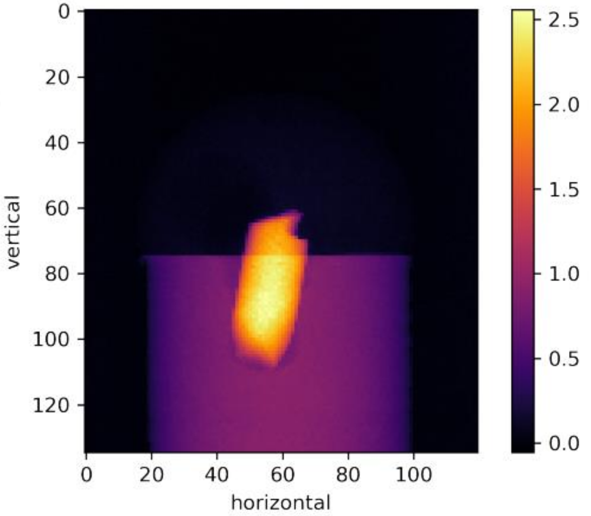


Demonstration data set

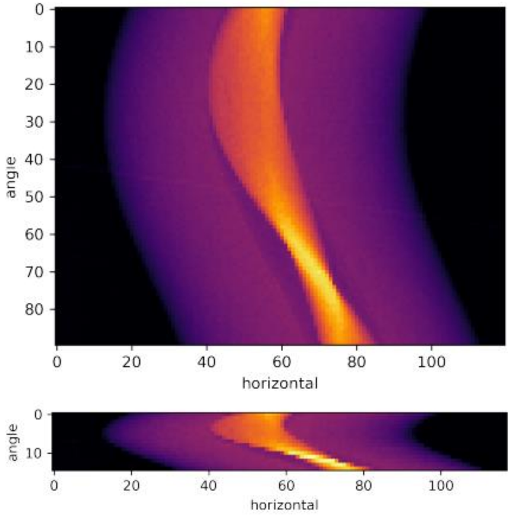
Raw projection



Negative log, cropped, centered



Sinogram

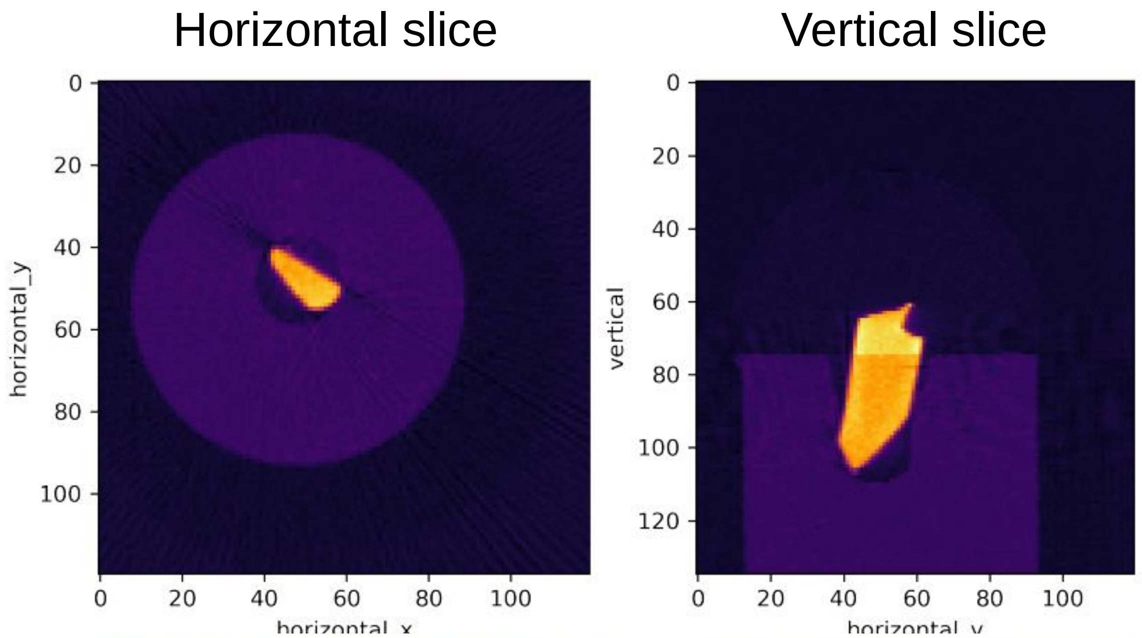


- 3D parallel-beam data set from Diamond Light Source, UK
- 0.5mm aluminium cylinder with piece of steel wire
- Droplet salt water causing corrosion + hydrogen bubbles
- Part of a fast time-lapse experiment
- 90 projections over 180 degrees, and **15 projections**
- Downsampled to 160-by-135 pixels for quick demonstration

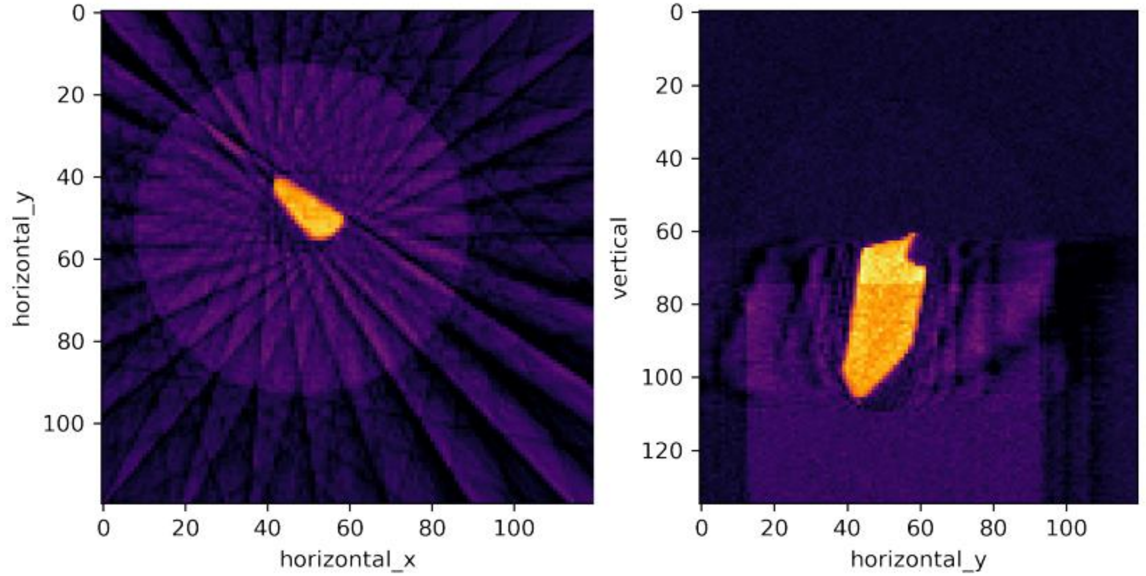
J. et al. 2021: *Core Imaging Library - Part I: a versatile Python framework for tomographic imaging,* Phil Trans A. https://doi.org/10.1098/rsta.2020.0192

Filtered backprojection

90
projections



15
projections



Algebraic iterative methods (regularizing by number of iterations)

CGLS

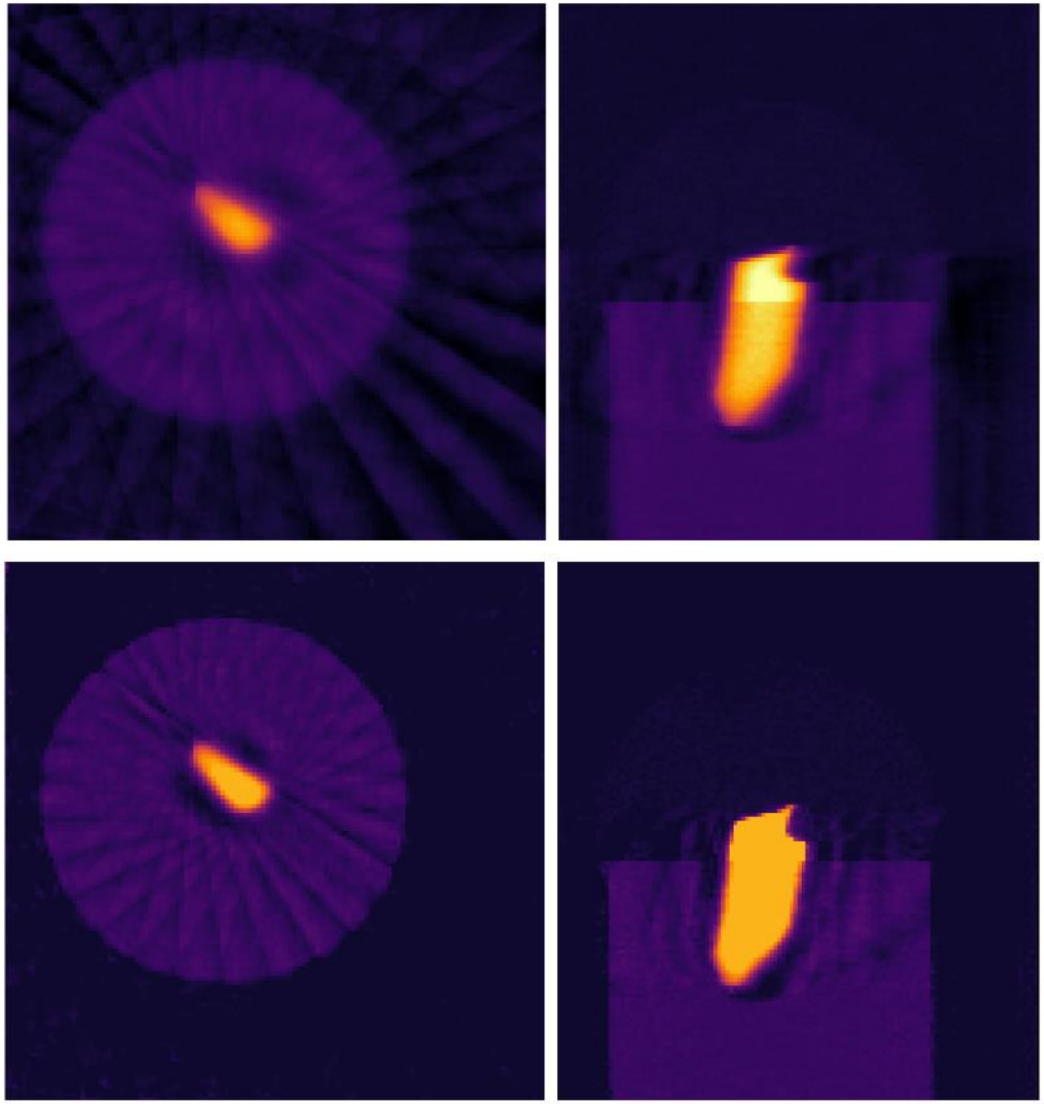
$$u^\star = \arg\min_u \|Au - b\|_2^2$$

Typically 10s of iterations

SIRT

As above and allowing lower and upper bounds on pixel values, here Non-negative and <= 0.9

Typically 100s of iterations



Regularised reconstruction – example Tikhonov

Data fidelity

Regulariser

$$u^\star = \arg\min_u \left\{ \|Au - b\|_2^2 + \alpha^2 \|Lu\|_2^2 \right\}$$

Minimiser:
Solution image

Unknown
image TBD

Regularisation
parameter

- Balance between fitting data and penalizing ”large values of *Lu*”
- Different choices of *L*:
  - Identity operator – make pixel values small
  - Finite difference gradient operator – make neighbor pixels similar
- No one best regulariser - different image types need different regularisers!

Regularised reconstruction – example Tikhonov

Data fidelity

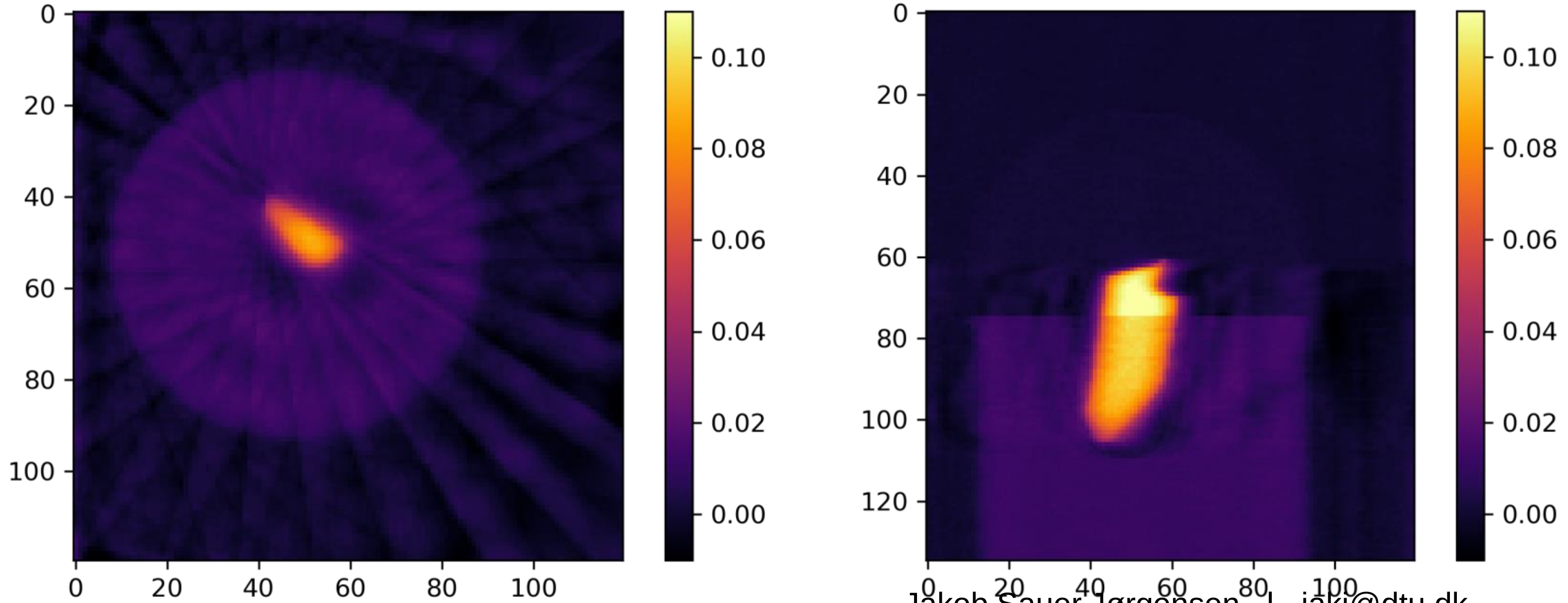
Regulariser

$$u^\star = \arg\min_u \left\{ \|Au - b\|_2^2 + \alpha^2 \|Lu\|_2^2 \right\}$$

Minimiser:
Solution image

Unknown
image TBD

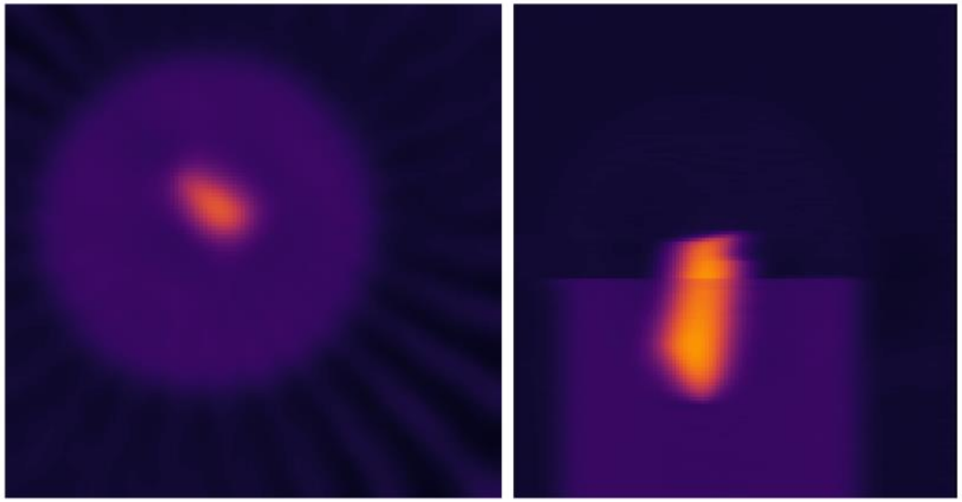
Regularisation
parameter



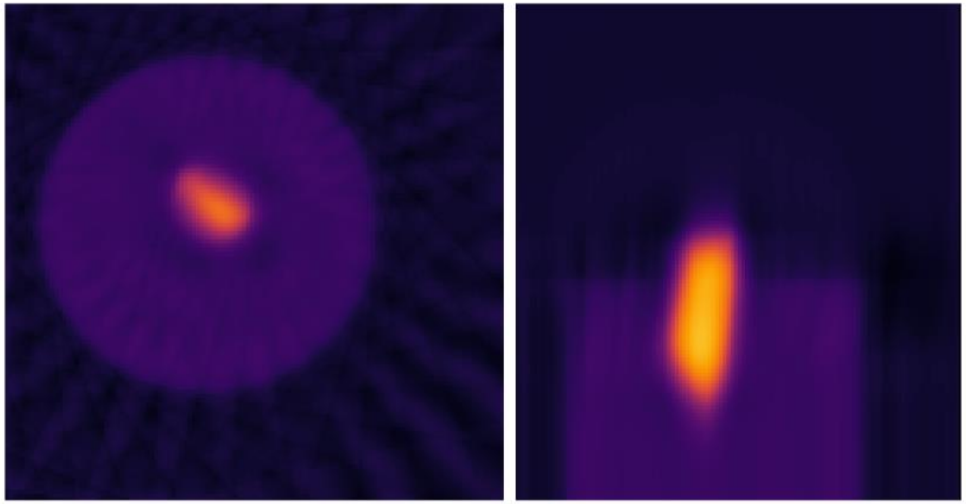
Anisotropic Tikhonov regularization

$$u^\star = \arg\min_u \left\{ \|Au-b\|_2^2 + \alpha_x^2\|L_x u\|_2^2 + \alpha_y^2\|L_y u\|_2^2 + \alpha_z^2\|L_z u\|_2^2 \right\}$$

Large horizontal,
small vertical smoothing



Small horizontal,
large vertical smoothing



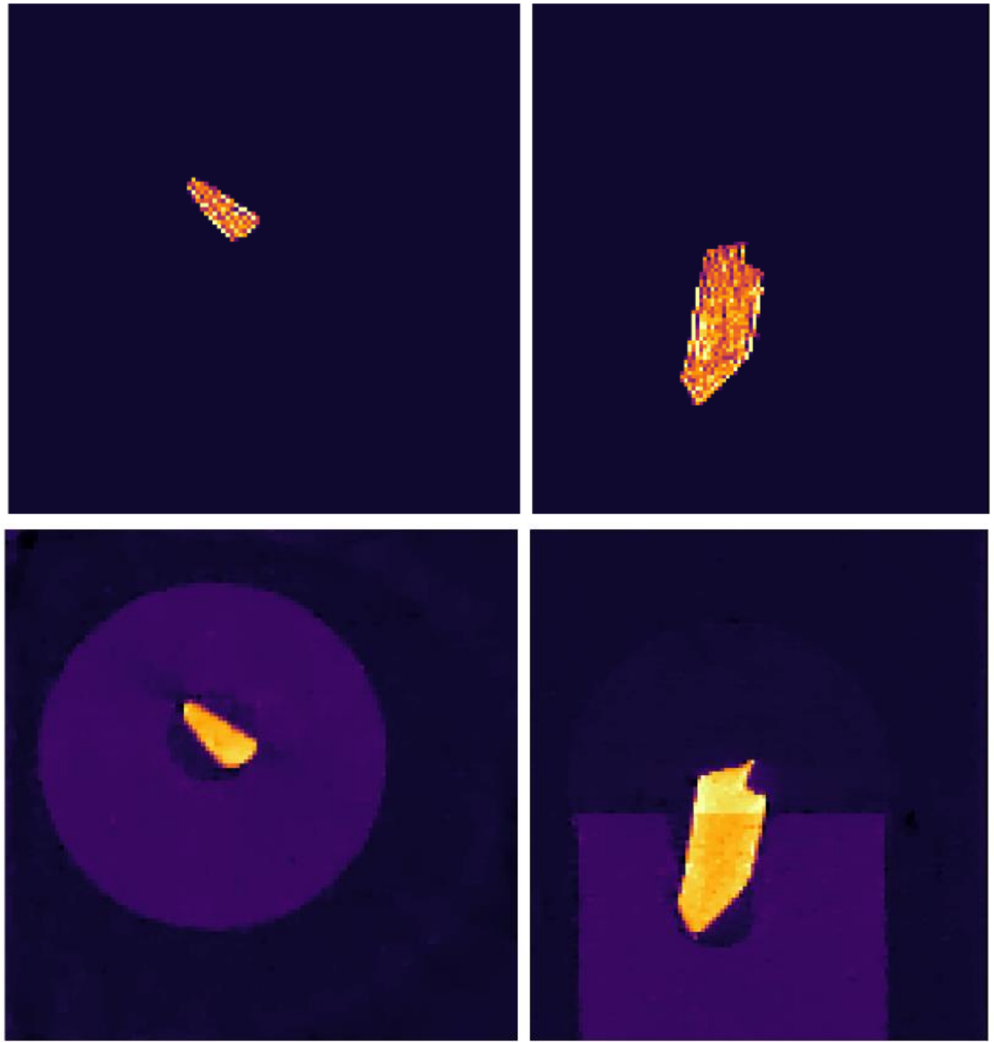
Sparsity and total variation regularization

L1-norm regularization:

$$\|u\|_1 = \sum_j |u_j|$$

Total variation regularization:

$$\sum_j \|D_j u\|_2$$



Jakob Sauer Jørgensen | jakj@dtu.dk

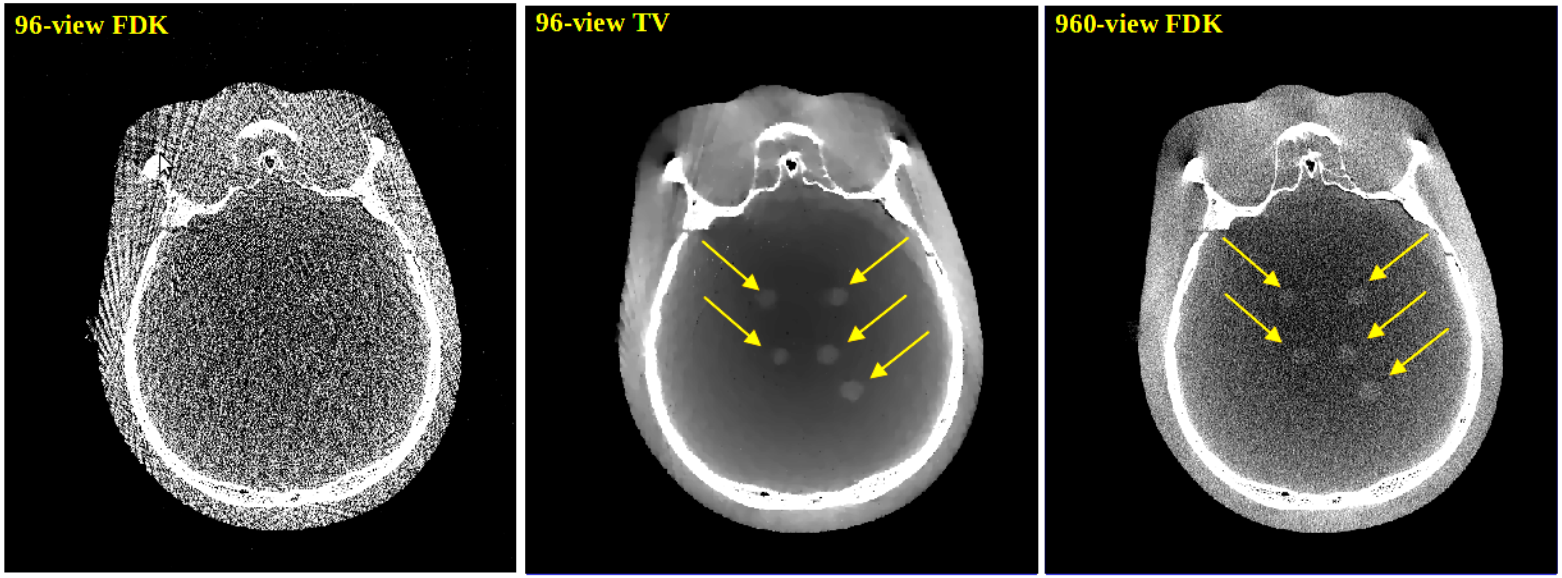
TV reconstruction example, physical head phantom, cone-beam X-ray CT

Total variation: Homogeneous regions with sharp boundaries.

$$x^\star = \underset{x}{\operatorname{argmin}} \left\{ \|Ax - b\|_2^2 + \alpha \|x\|_{\mathsf{TV}} \right\}$$

$$\|x\|_{\mathsf{TV}} = \sum_j \|D_j x\|_2, \quad D_j \text{ finite diff. gradient at voxel } j.$$

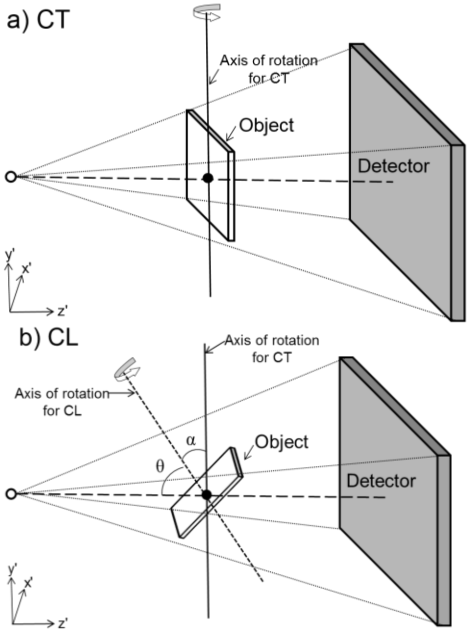
TV is an example of **sparsity-regularized reconstruction.**

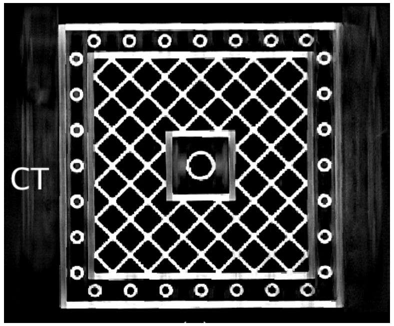


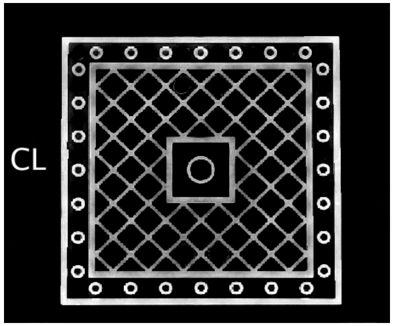
[Bian et al. 2010, Phys. Med. Biol. **55**, 6575–6599]. Courtesy: X. Pan, U. Chicago.

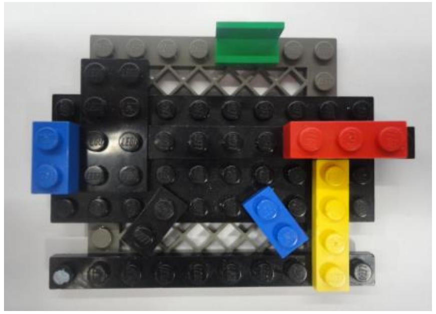
Tomographic imaging of planar samples with laminography

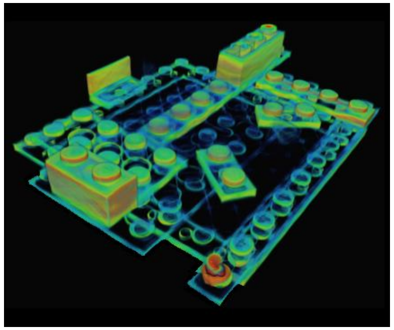
- Planar samples like composite panels and printed circuit boards difficult to scan due to different exposure along views.
- Conventional scan gives limited-angle artifacts, missing edges.
- Laminography allows uniform exposure.
- Non-standard geometry needs dedicated reconstruction – here used CGLS.

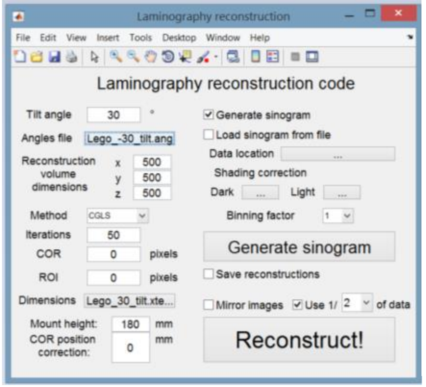










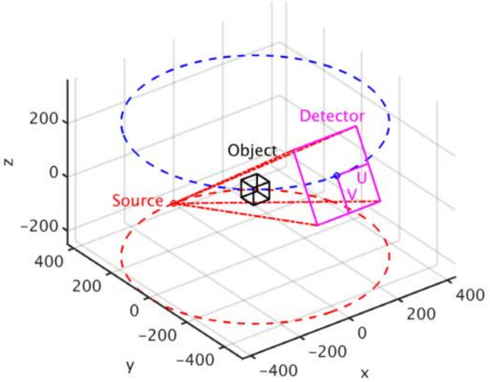


Laminography reconstruction code

Tilt angle: 30 °
Angles file: Lego_-30_tilt.ang
Reconstruction volume dimensions: x 500, y 500, z 500
Method: CGLS
Iterations: 50
COR: 0 pixels
ROI: 0 pixels
Dimensions: Lego_30_tilt.xte...
Mount height: 180 mm
COR position correction: 0 mm

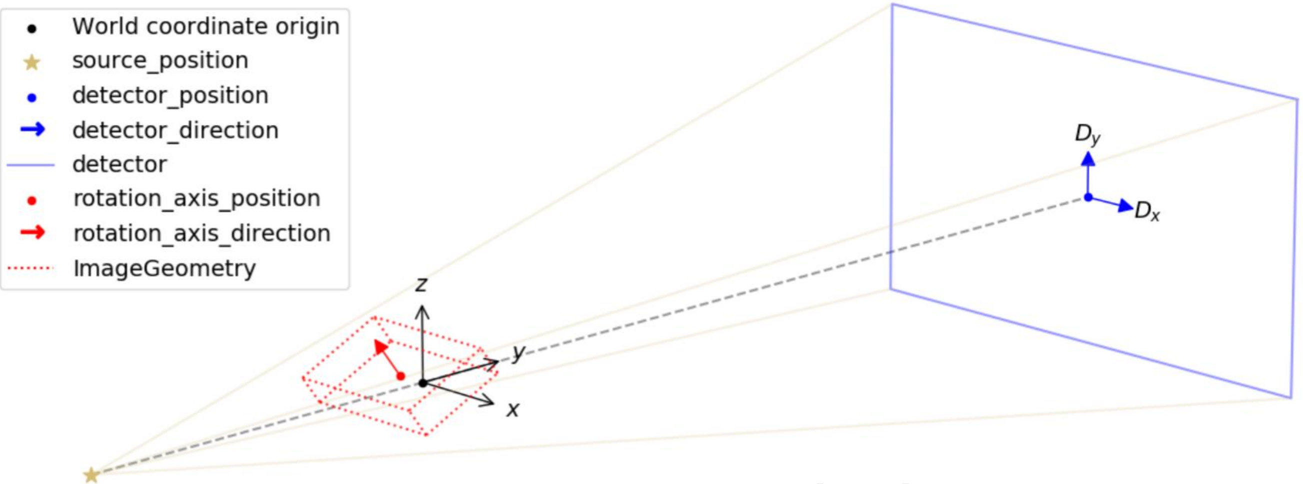
☑ Generate sinogram
☐ Load sinogram from file
Data location: ...
Shading correction
Dark: ... Light: ...
Binning factor: 1

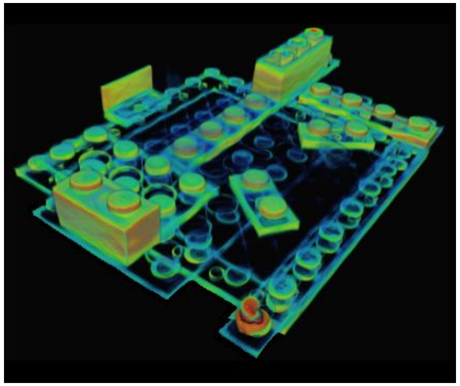
☐ Save reconstructions
☐ Mirror images ☑ Use 1/ 2 of data

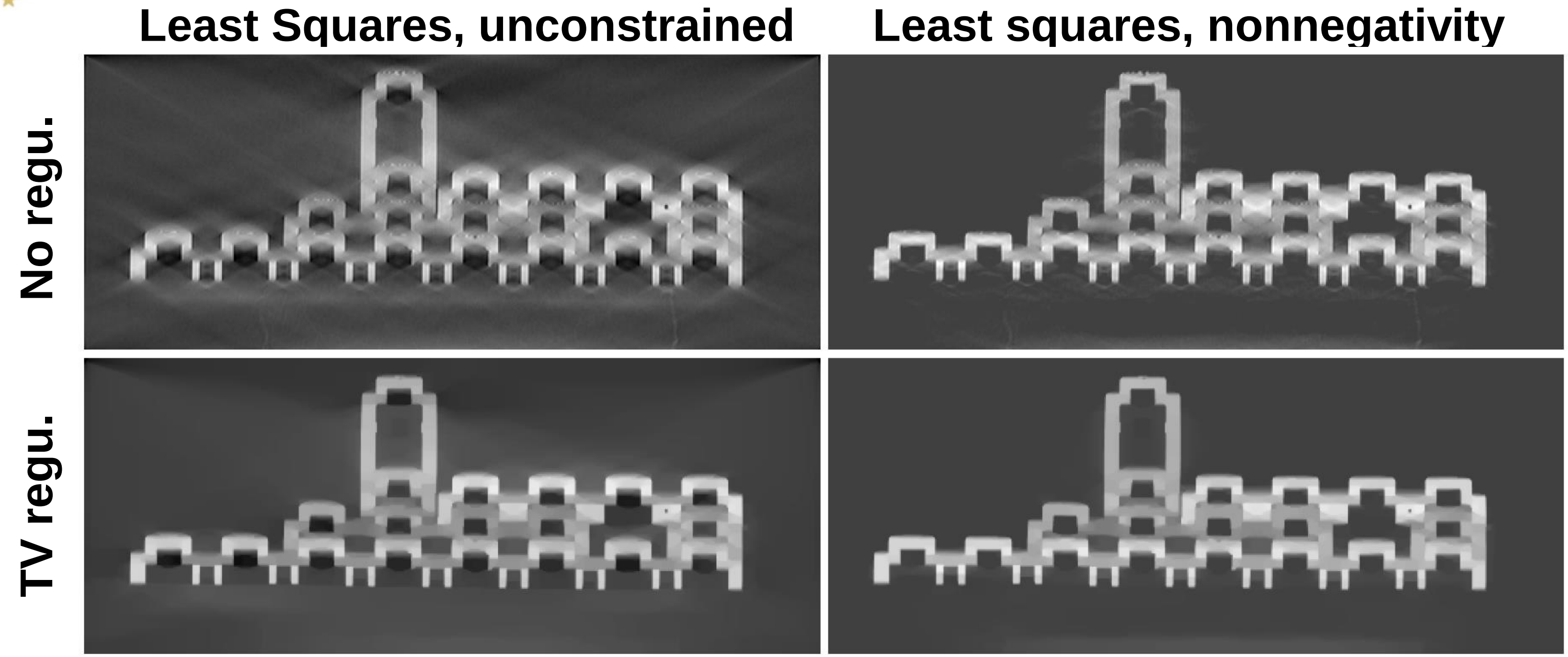


Fisher, Holmes, Jørgensen, Gajjar, Behnsen, Lionheart & Withers, Meas. Sci. Technol. 30 (2019), pp. 035401

Laminography artifacts reduced by TV

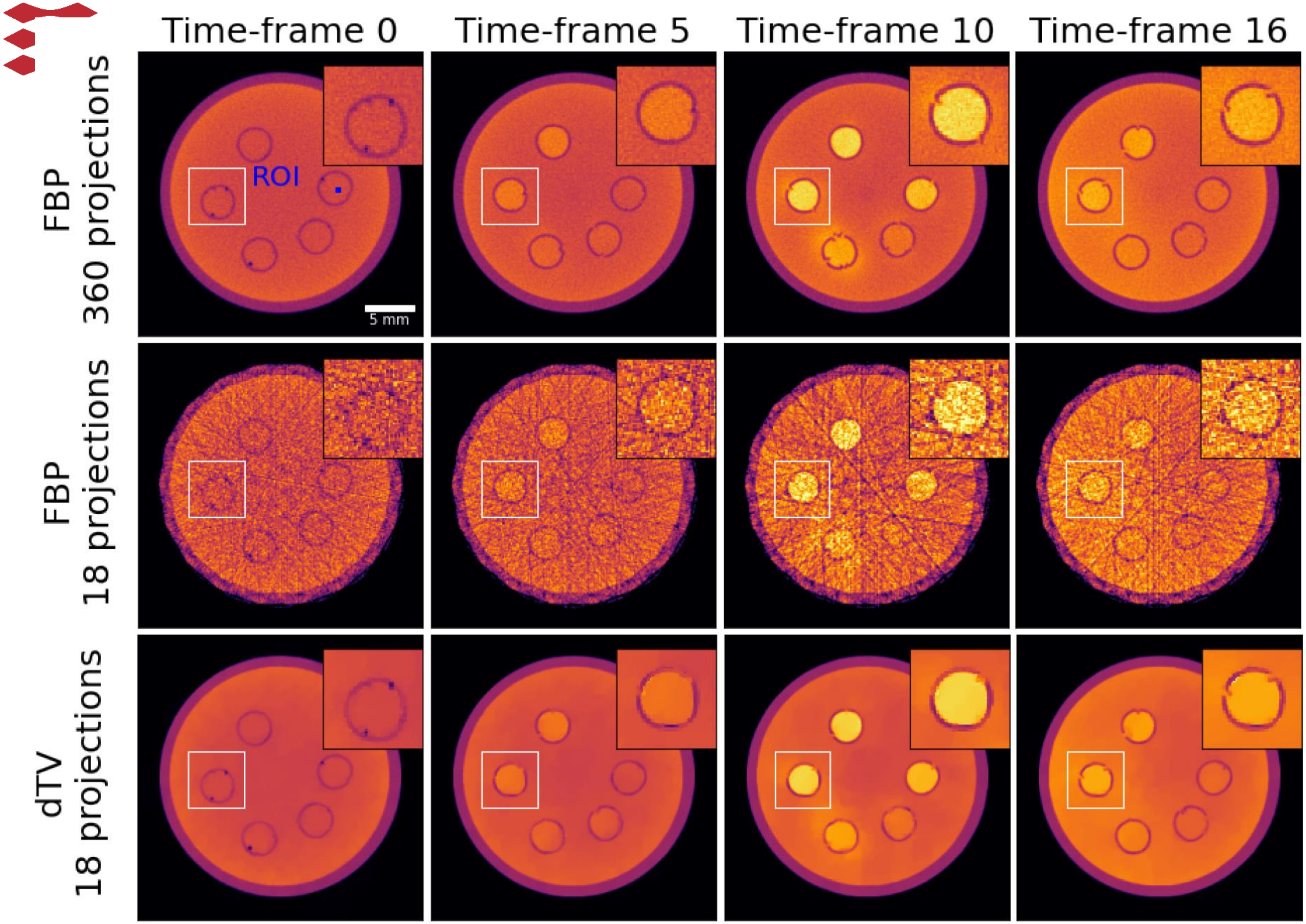






J. et al. 2021: *Core Imaging Library - Part I: a versatile Python framework for tomographic imaging,* Phil Trans A. https://doi.org/10.1098/rsta.2020.0192

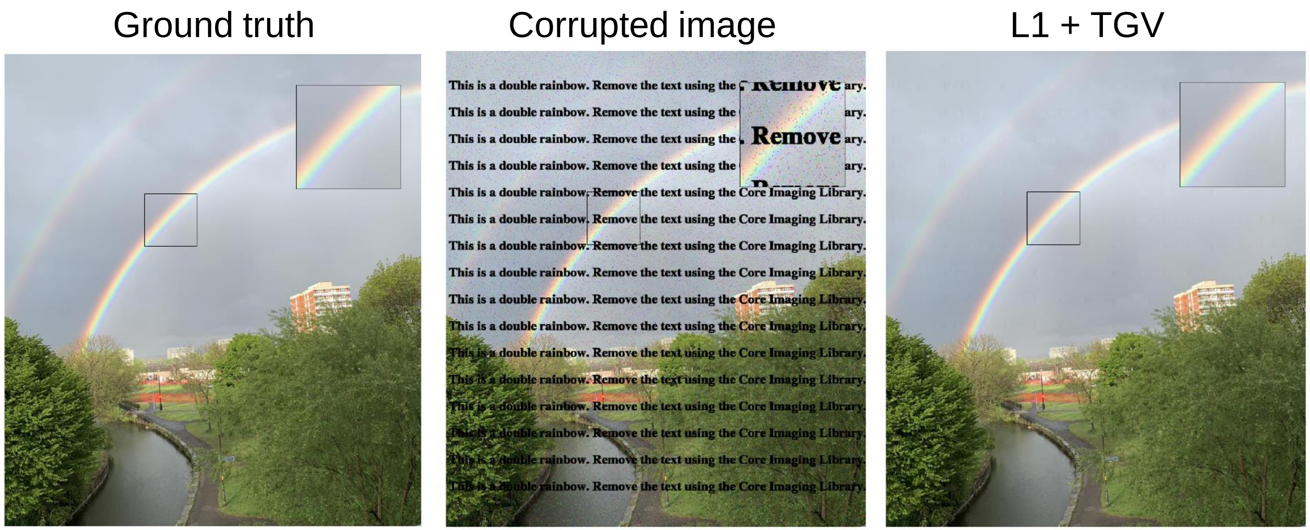
Dynamic CT with directional TV using reference images



Papoutsellis et al. 2021: *Core Imaging Library - Part II: multichannel reconstruction for dynamic and spectral tomography*, Phil Trans A. https://doi.org/10.1098/rsta.2020.0193

Colour image inpainting as an inverse problem

Colour image inpainting and salt/pepper denoising using L1-norm data fidelity and total generalized variation (TGV)



Ground truth

Corrupted image

L1 + TGV

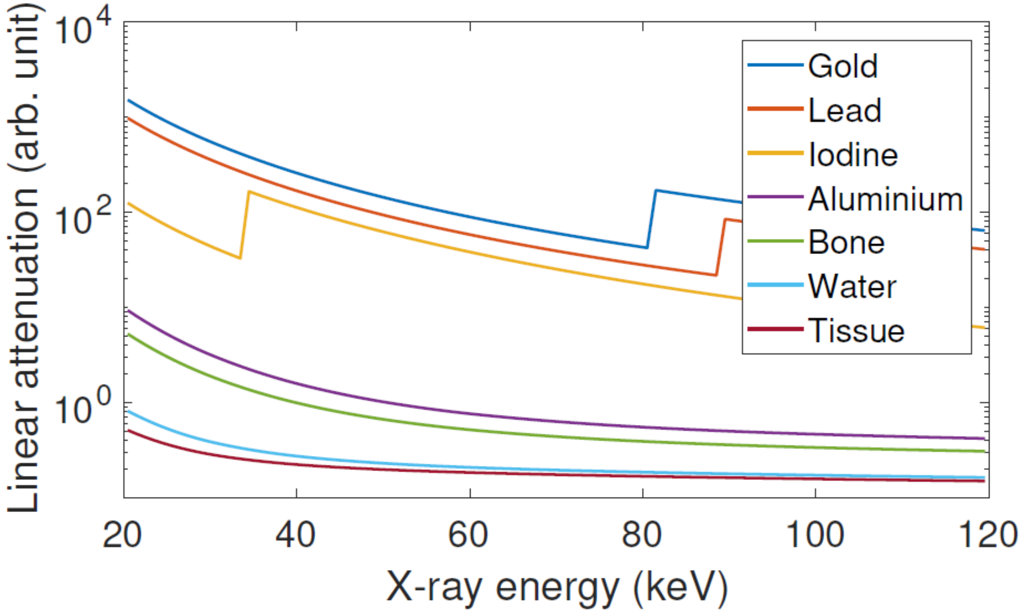
CIL supplies LinearOperators for denoising, deblurring and inpainting problems and users may write a LinearOperator wrapper for their own problem.

Papoutsellis et al. 2021: *Core Imaging Library - Part II: multichannel reconstruction for dynamic and spectral tomography*, Phil. Trans. R. Soc. A, **379**, 20200193: https://doi.org/10.1098/rsta.2020.0193

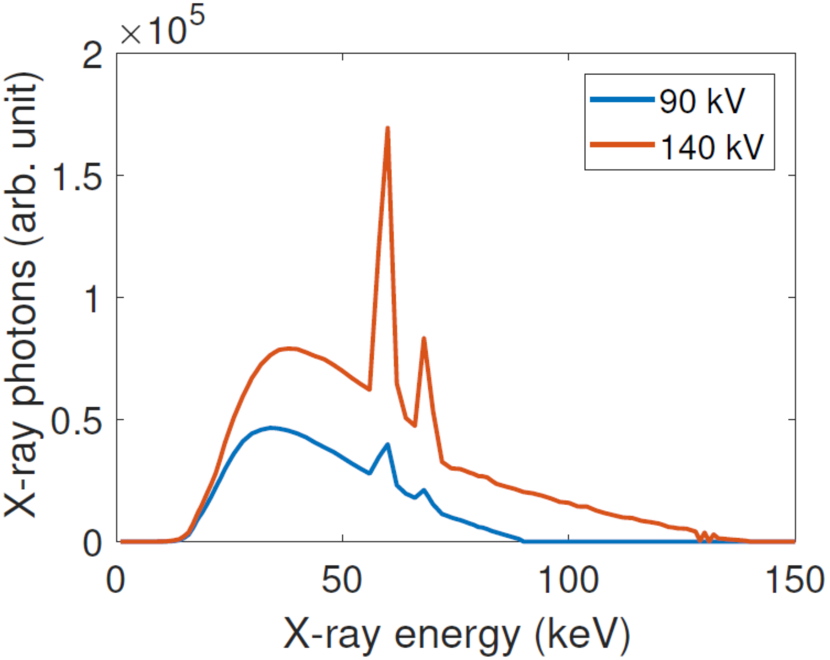
- Demo of iterative methods in CIL

X-ray beam normally not mono-chromatic

Attenuation is energy-dependent



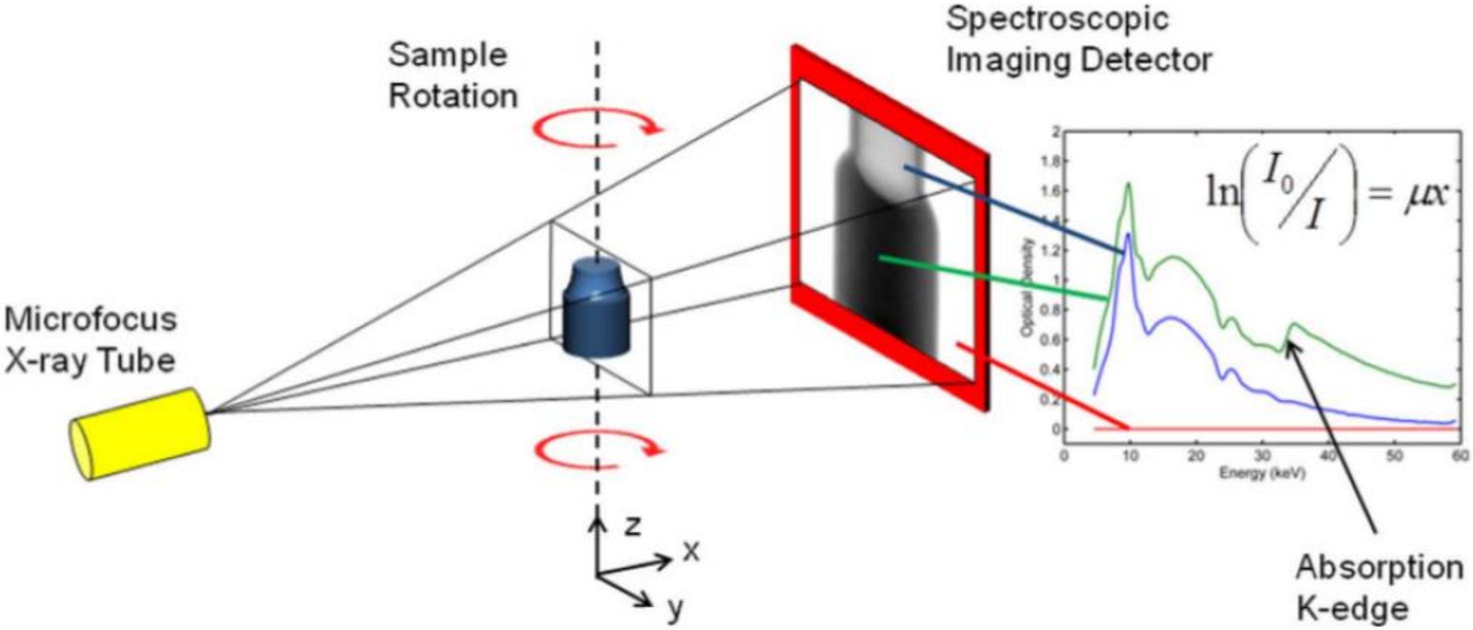
X-ray source spectrum



Hyperspectral X-ray CT @ Manchester

The Manchester Colour Bay instrument for hyperspectral X-ray imaging using a HEXITEC detector

(C. Egan et al. 3D chemical imaging in the laboratory by hyperspectral X-ray CT, 2015).



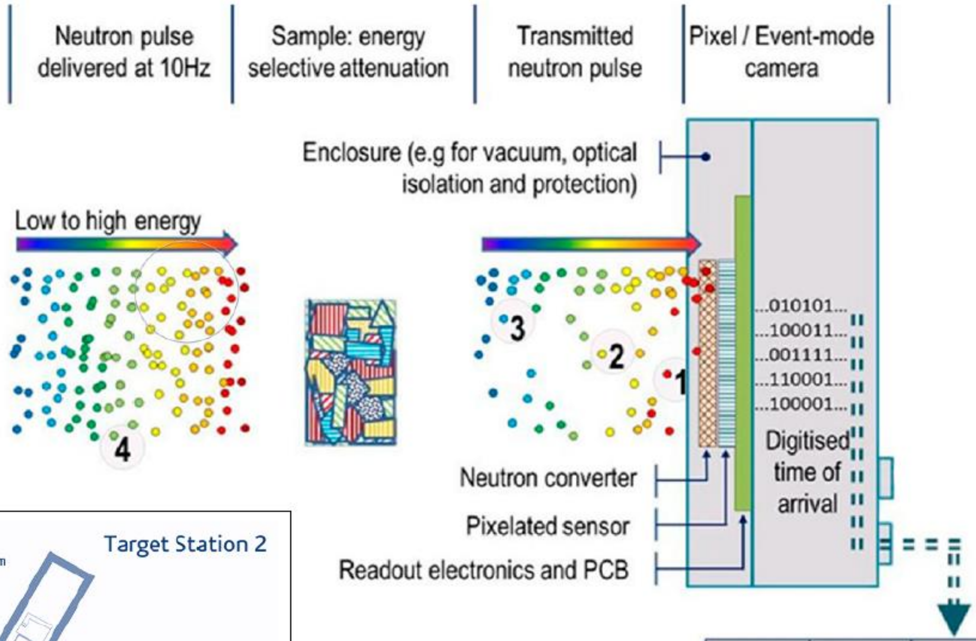
Why new reconstruction methods and software needed?

- Commerical software, e.g., Nikon CT Pro, Octopus
- Limited possibilities, mostly just FBP/FDK
- Some open source reconstruction tools: ASTRA, tomopy, Savu, ...
- Mostly parallel-beam, no software for multichannel CT

Challenges and opportunities:

- Few counts in each channel
- Naïve channel-wise reconstruction poor quality
- Neighbouring channels mostly similar
- Explore how to regularise across channels (and space) to improve reconstruction quality.

Neutron time-of-flight energy-resolved CT at ISIS Neutron and Muon Facility, UK



Neutron pulse delivered at 10Hz

Sample: energy selective attenuation

Transmitted neutron pulse

Pixel / Event-mode camera

Enclosure (e.g for vacuum, optical isolation and protection)

Low to high energy

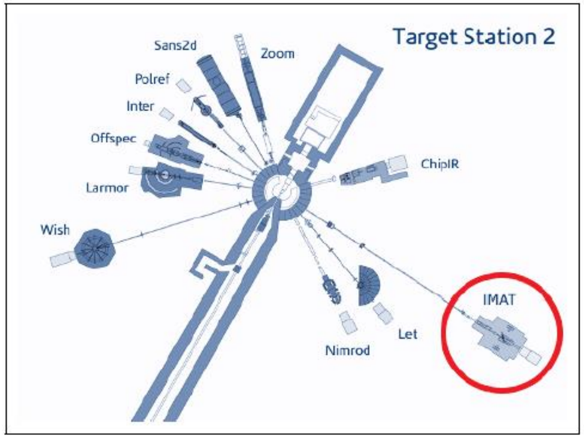
Digitised time of arrival

Neutron converter

Pixelated sensor

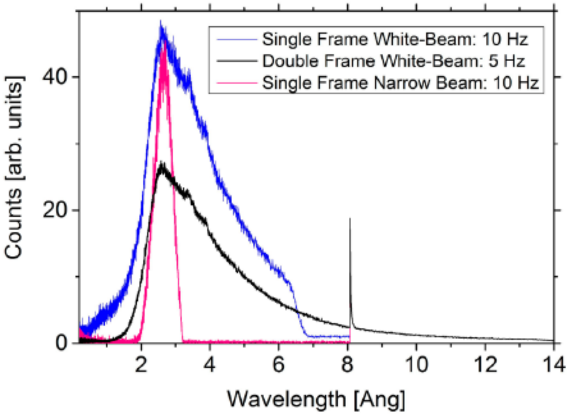
Readout electronics and PCB

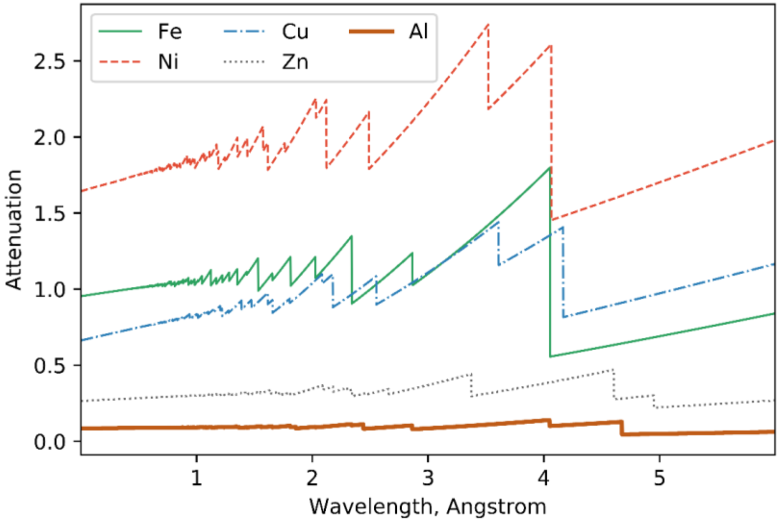
Neutron	Pulse #	ToF	Pixel
1	1	🔴	(x,y)
2	1	🟡	(x,y)
3	1	🔵	(x,y)
4	2	🟢	(x,y)

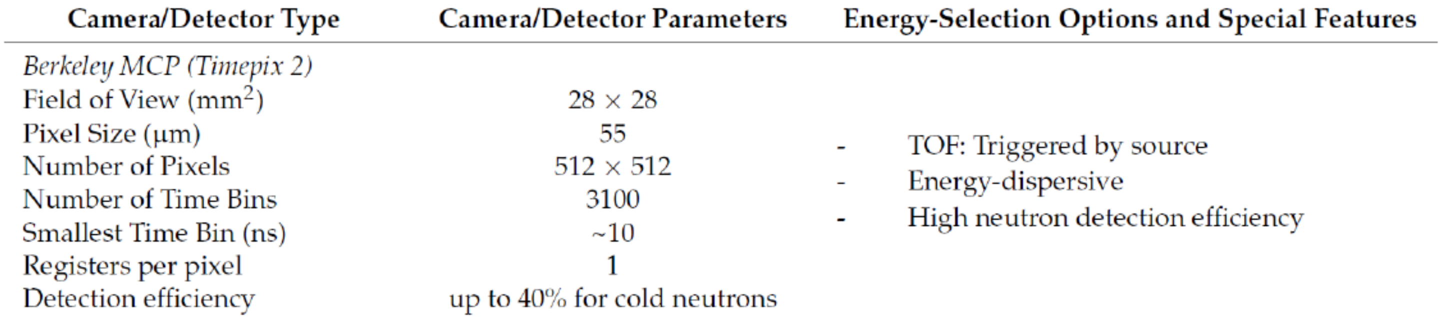


512^2 pixels, pixel size 55um, up to 3100 channels

Beam spectrum and neutron Bragg edges





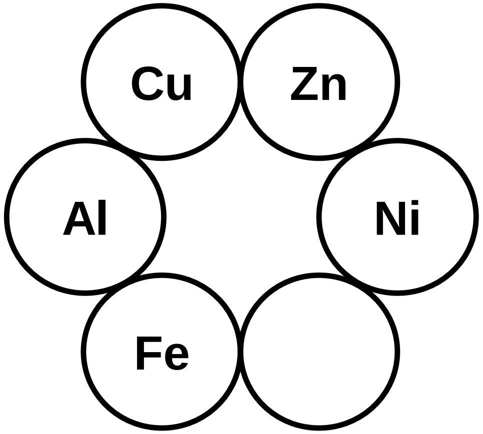


Camera/Detector Type	Camera/Detector Parameters	Energy-Selection Options and Special Features
Berkeley MCP (Timepix 2)		
Field of View (mm^2)	28 × 28	
Pixel Size (μm)	55	- TOF: Triggered by source
Number of Pixels	512 × 512	- Energy-dispersive
Number of Time Bins	3100	- High neutron detection efficiency
Smallest Time Bin (ns)	~10	
Registers per pixel	1	
Detection efficiency	up to 40% for cold neutrons	

IMAT hyperspectral neutron data

5 aluminum foil cylinders
filled with metallic powder
+ 1 empty

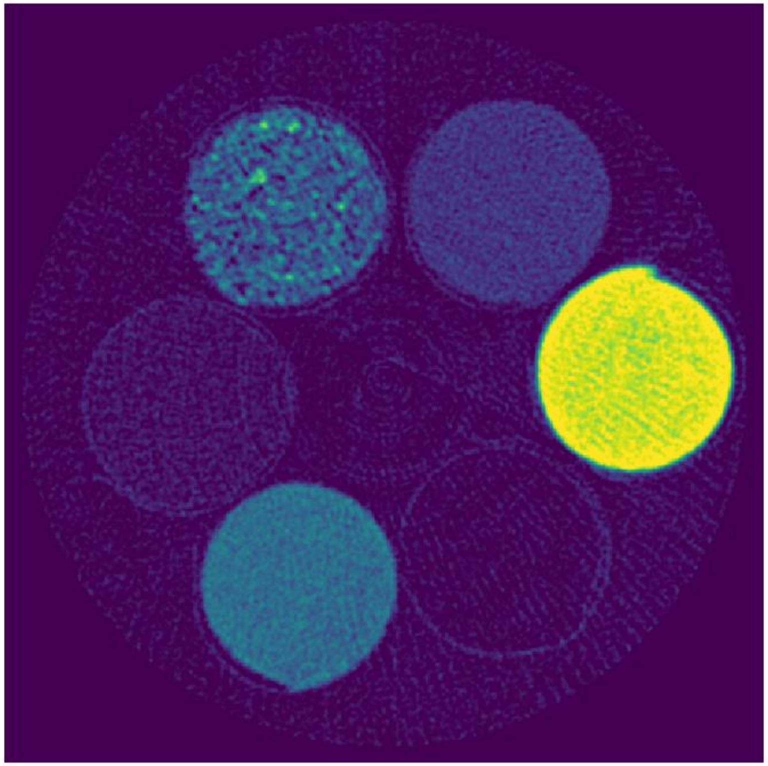
Detector size: 512x512, 0.055 mm pixel
120 projections over 180°
2840 energy channels between 1 and 5 A
15 min exposure time



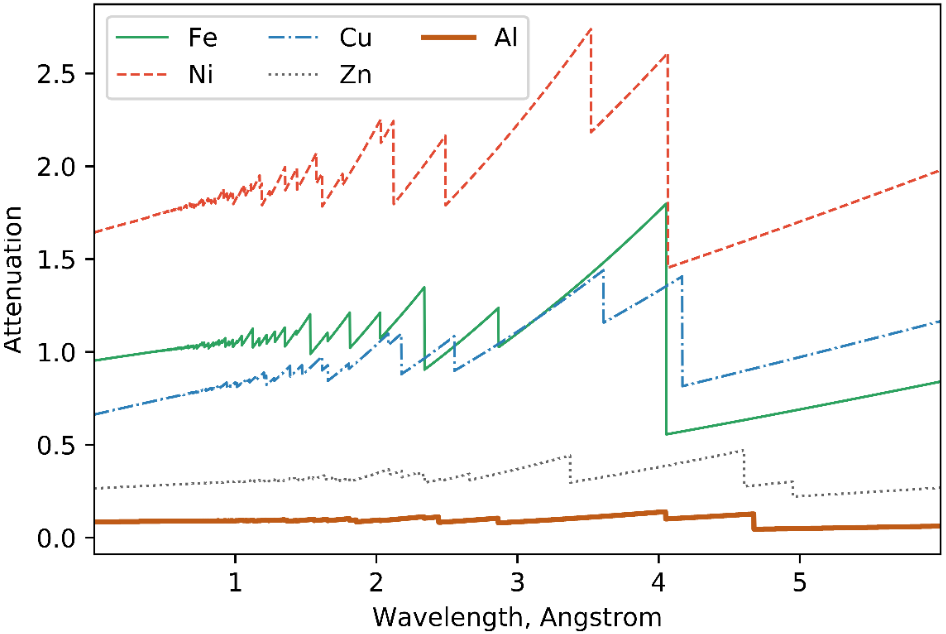
Jakob Sauer Jørgensen | jakj@dtu.dk

Propose TV spatially and TGV spectrally

Spatially:
Piecewise **constant** plus jumps



Spectrally:
Piecewise **smooth** plus jumps



Spatial TV plus spectral TGV optimisation problem

- TV: For piecewise constant plus jumps.
- TGV: For piecewise smooth plus jumps.


noisy image TV denoising TGV denoising

Spatial TV plus spectral TGV optimisation problem

$$\arg\min_{u} \; \alpha\,\mathrm{TV}(u_{space}) + \beta\,\mathrm{TGV}(u_{spt}) + \frac{1}{2}\|g-u\|_2^2$$

$$\arg\min_{u,w} \; \alpha\,\|\nabla u\|_{2,1} + \beta\left(\|\partial_{spt}u - w\|_1 + \sqrt{2}\|\partial_{spt}w\|_1\right) + \frac{1}{2}\|g-u\|_2^2$$

$$\Downarrow$$

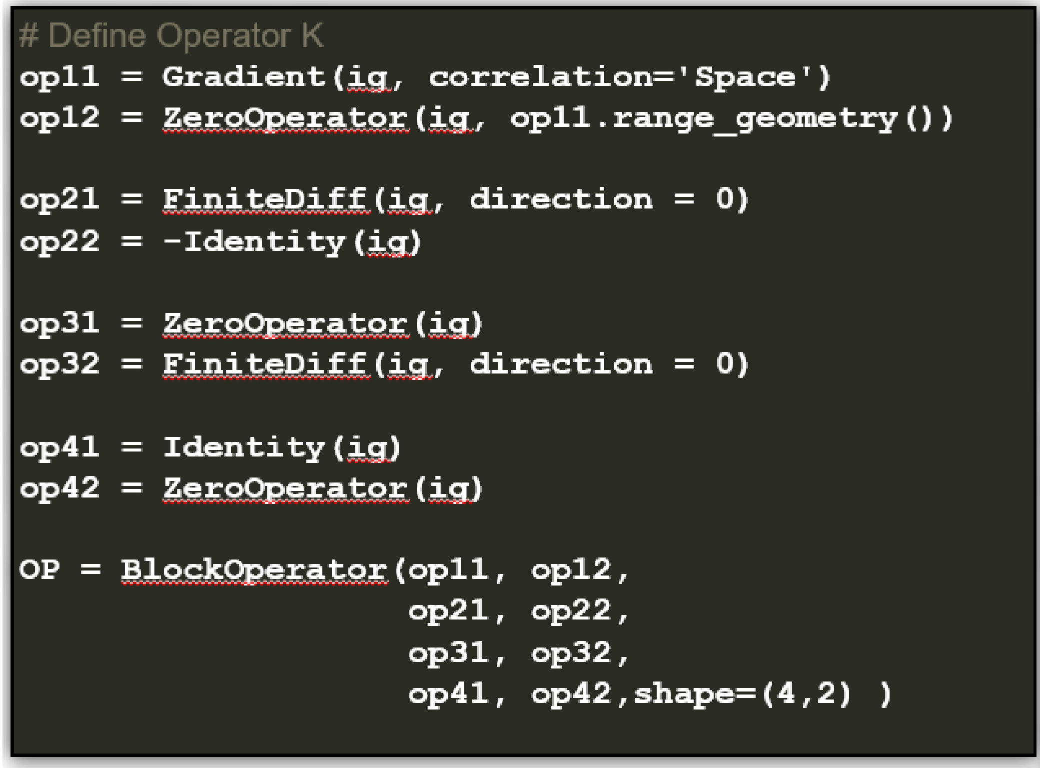
$$\arg\min_{x} \; \mathcal{F}(K\,x) + \mathcal{G}(x)$$

$$x = \begin{bmatrix} u \\ w \end{bmatrix} \qquad \mathcal{G}(x) = \mathbb{I}_{\{u>0\}}(u) \qquad K = \begin{bmatrix} \nabla & \mathbb{O} \\ \partial_{spt} & -\mathbb{I} \\ \mathbb{O} & \partial_{spt} \\ \mathbb{I} & \mathbb{O} \end{bmatrix}$$

$$\begin{aligned} \mathcal{F}(z_1, z_2, z_3, z_4) &= F_1(z_1) + F_2(z_2) + F_3(z_3) + F_4(z_4) \\ &= \alpha\,\|z_1\|_{2,1} + \beta\,\|z_2\|_1 + \beta\sqrt{2}\|z_3\|_1 + \frac{1}{2}\|g - z_4\|_2^2 \end{aligned}$$

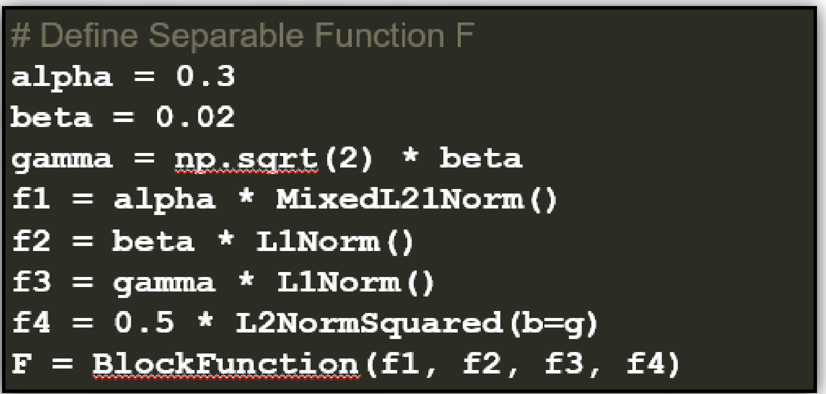
Spatial TV plus spectral TGV implementation in CIL

$$\underset{x}{\arg\min}\ \mathcal{F}(K\,x) + \mathcal{G}(x)$$

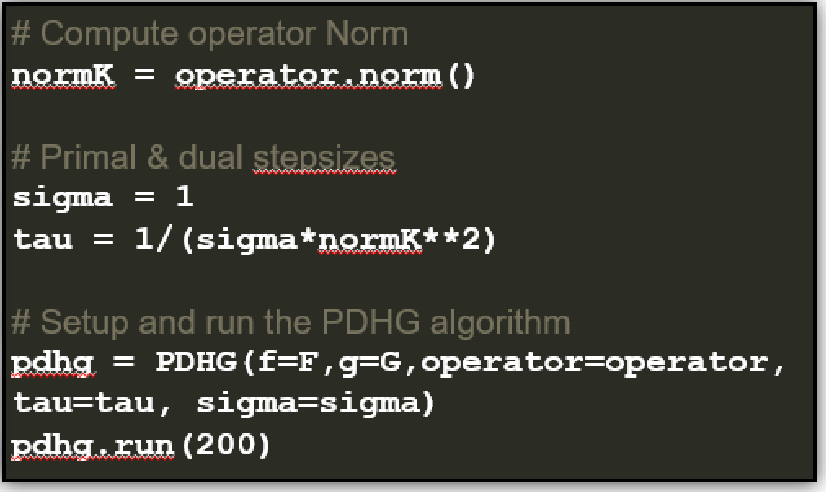


```
# Define Operator K
op11 = Gradient(ig, correlation='Space')
op12 = ZeroOperator(ig, op11.range_geometry())

op21 = FiniteDiff(ig, direction = 0)
op22 = -Identity(ig)

op31 = ZeroOperator(ig)
op32 = FiniteDiff(ig, direction = 0)

op41 = Identity(ig)
op42 = ZeroOperator(ig)

OP = BlockOperator(op11, op12,
                   op21, op22,
                   op31, op32,
                   op41, op42,shape=(4,2) )
```

```
# Define Function G, with positivity constraint
G = Indicator(lower=0)
```

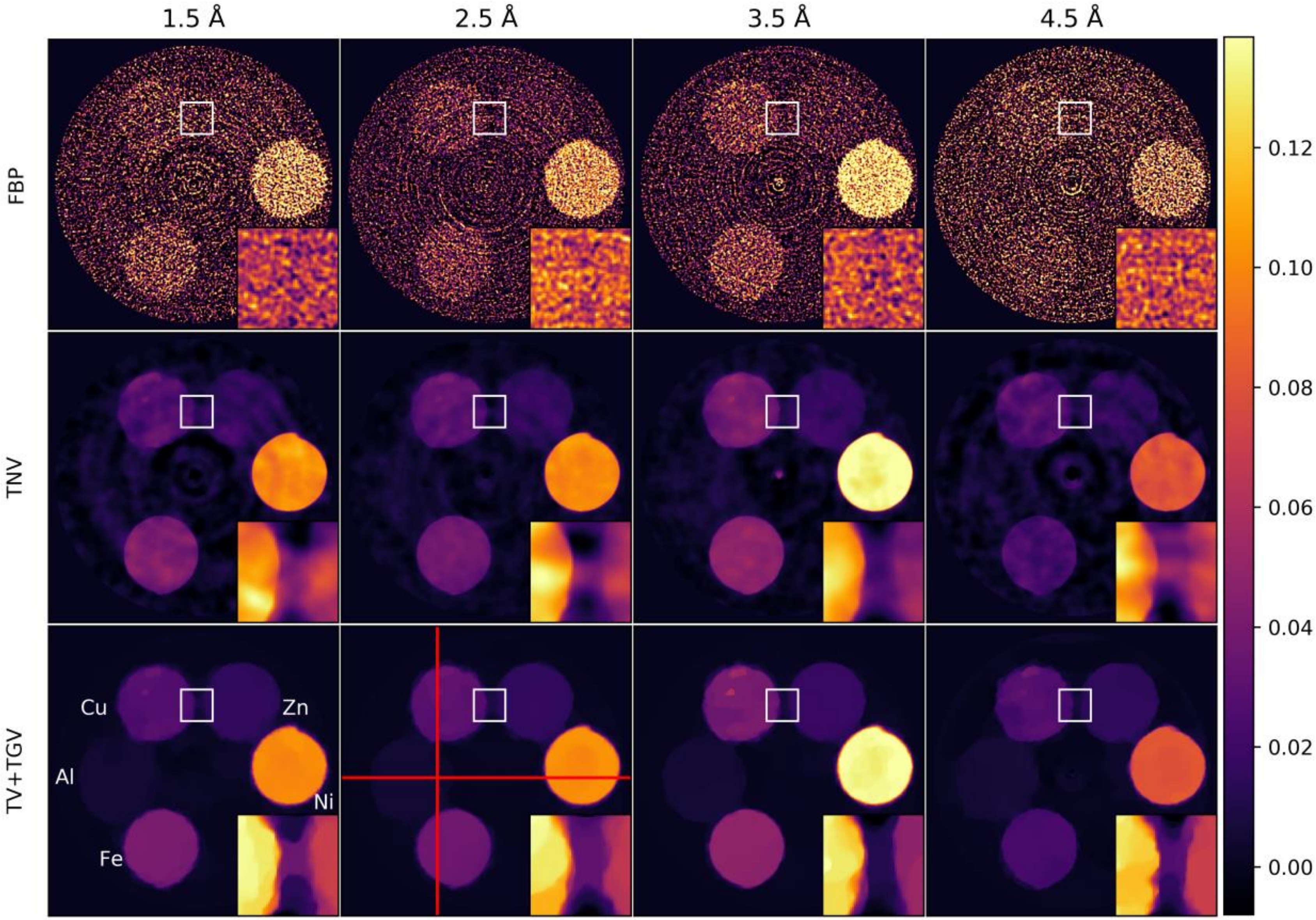


```
# Define Separable Function F
alpha = 0.3
beta = 0.02
gamma = np.sqrt(2) * beta
f1 = alpha * MixedL21Norm()
f2 = beta * L1Norm()
f3 = gamma * L1Norm()
f4 = 0.5 * L2NormSquared(b=g)
F = BlockFunction(f1, f2, f3, f4)
```



```
# Compute operator Norm
normK = operator.norm()

# Primal & dual stepsizes
sigma = 1
tau = 1/(sigma*normK**2)

# Setup and run the PDHG algorithm
pdhg = PDHG(f=F,g=G,operator=operator,
tau=tau, sigma=sigma)
pdhg.run(200)
```

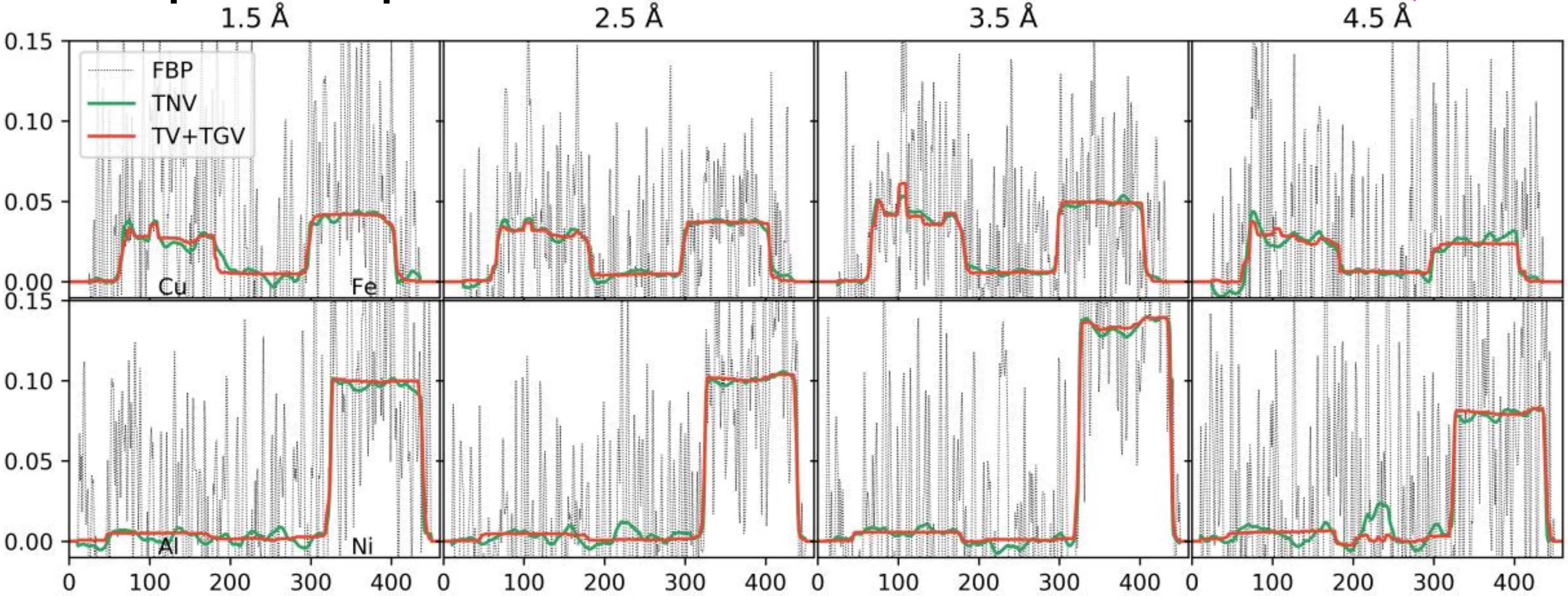
Reconstructed channel images



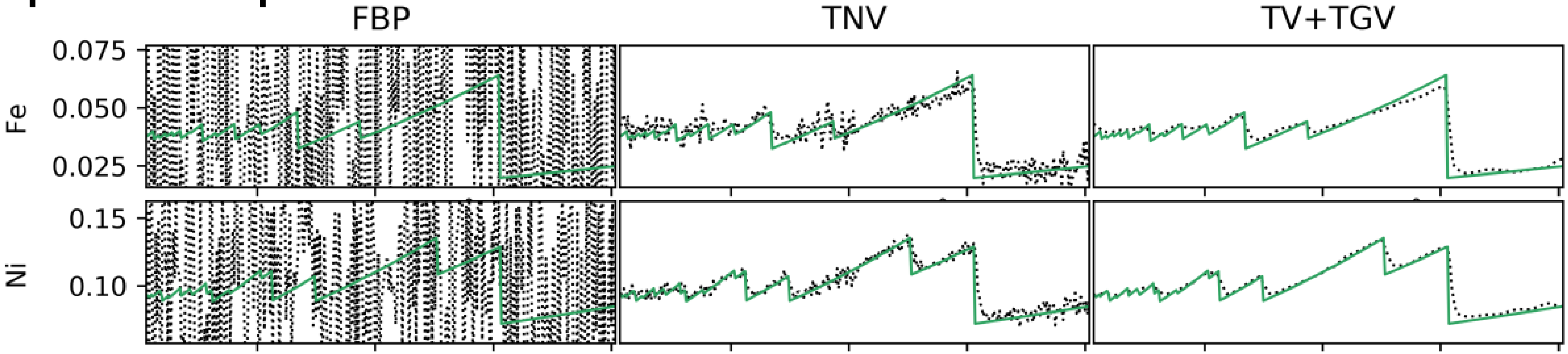
Ametova et al. 2021: *Crystalline phase discriminating neutron tomography using advanced reconstruction methods*, J. Physics D, https://doi.org/10.1088/1361-6463/ac02f9

Hyperspectral neutron CT with spatio-spectral regu.

Spatial line profiles



Spectral line profiles



Ametova et al. 2021: *Crystalline phase discriminating neutron tomography using advanced reconstruction methods*, J. Physics D, https://doi.org/10.1088/1361-6463/ac02f9

Useful links for Core Imaging Library

- Website: https://www.ccpi.ac.uk/CIL
- Documentation: https://tomographicimaging.github.io/CIL

Papers

- J. et al. 2021, Phil. Trans. R. Soc. A, **379**, 20200192:
  Core Imaging Library - Part I: a versatile Python framework for tomographic imaging
  https://doi.org/10.1098/rsta.2020.0192

- Papoutsellis et al. 2021, Phil. Trans. R. Soc. A, **379**, 20200193:
  Core Imaging Library - Part II: multichannel reconstruction for dynamic and spectral tomography
  https://doi.org/10.1098/rsta.2020.0193

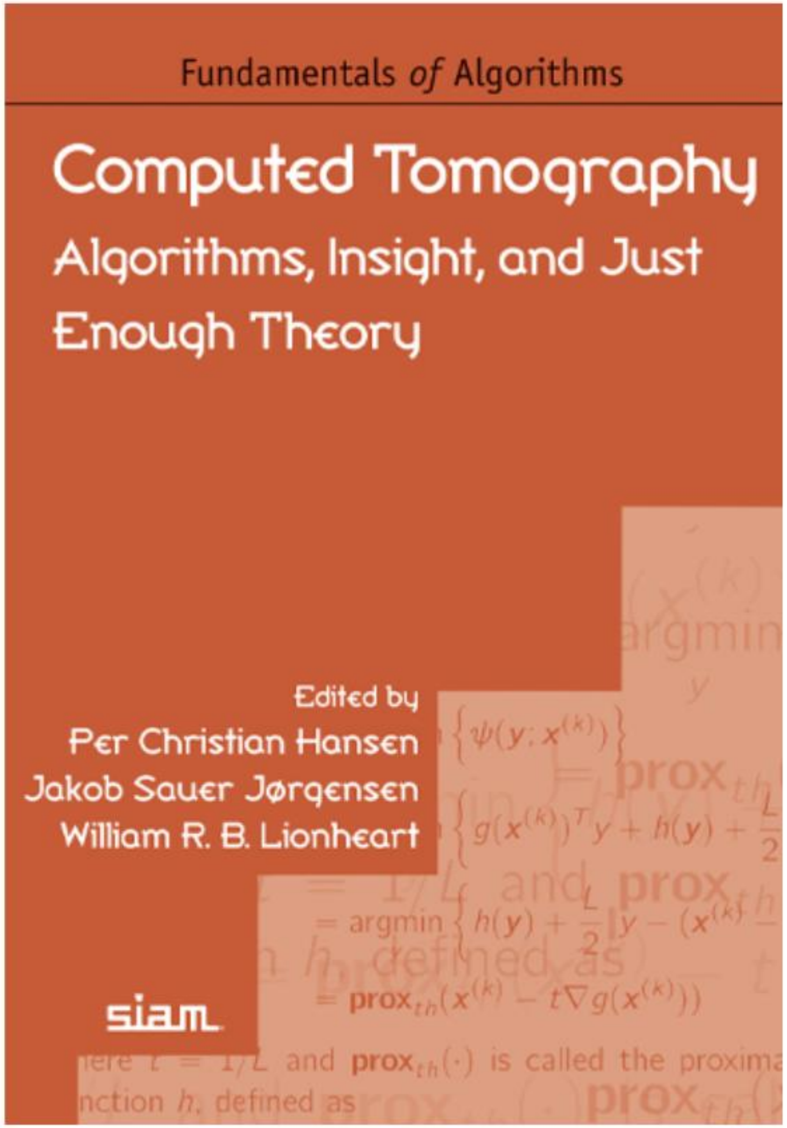
Application Papers

- Ametova et al. 2021, J. Physics D, **54**, 325502
  Crystalline phase discriminating neutron tomography using advanced reconstruction methods
  https://doi.org/10.1088/1361-6463/ac02f9

- Warr et al. 2021, Nature Scientific Reports, **11**, 20818
  Enhanced hyperspectral tomography for bioimaging by spatiospectral reconstruction
  https://www.nature.com/articles/s41598-021-00146-4

- Brown et al. 2021, Phil. Trans. R. Soc. A, **379**, 20200208
  Motion estimation and correction for simultaneous PET/MR using SIRF and CIL
  https://doi.org/10.1098/rsta.2020.0208

New book on CT reconstruction



Fundamentals *of* Algorithms

Computed Tomography
Algorithms, Insight, and Just
Enough Theory

Edited by
Per Christian Hansen
Jakob Sauer Jørgensen
William R. B. Lionheart

siam

Computed Tomography: Algorithms, Insight, and Just Enough Theory

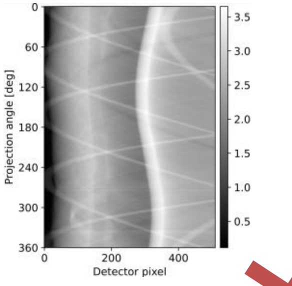
Edited by Per Christian Hansen, Jakob Sauer Jørgensen, and William R. B. Lionheart
Published: 2021
Pages: xviii + 337 pages
Softcover
ISBN: 978-1-611976-66-3
Order Code: FA18

bookstore.siam.org/fa18

CUQIpy

Forward model & data

$\mathbf{A}$ $\quad$ $\mathbf{b}$



Parameter assumptions

$$\mathbf{b} \sim \mathcal{N}(\mathbf{A}\mathbf{x}, \sigma_{\text{noise}}^2 \mathbf{I})$$
$$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \delta^2 \mathbf{I})$$
$$\delta \sim \text{Gamma}(a, b)$$

Bayesian inverse problem modelling framework

$$p(\mathbf{x}, \delta | \mathbf{b}) \propto p(\mathbf{b}|\mathbf{x})\, p(\mathbf{x}|\delta)\, p(\delta)$$

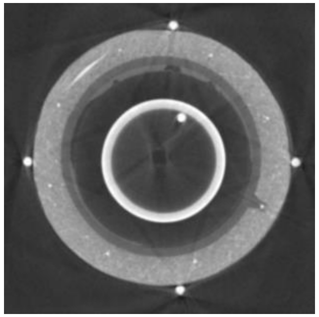
Posterior **Likelihood** **Priors**

Computational UQ engine

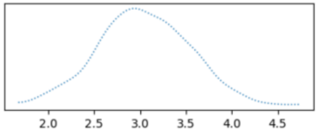
- Efficient posterior sampling using structure
- High-level interface for non-experts
- Full control for experts
- Hierarchical problem support
- Catalogue of test problems, priors, …

Parameter estimates

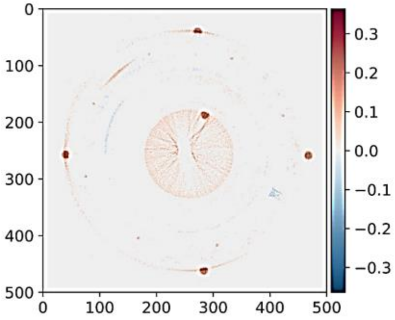
$\mathbf{x}$



δ



UQ information



https://github.com/CUQI-DTU/CUQIpy

Summary

jakj@dtu.dk

Filtered backprojection

- Many projections, full angular range, only moderate noise – look no further!

Iterative reconstruction methods

- Solve optimization problem numerically to find best image
- Trade-off between fitting data and introducing regularity
- Type of regularizer to use depends on image features

Hyperspectral tomography

- 100s or 1000s or highly noisy channels of tomographic data
- Naive channelwise reconstruction insufficient
- Spatial and spectral regularization such as TV/TGV/TNV improve image quality to allow identification of K-edges/Bragg edges on a single voxel level

Core Imaging Library reconstruction framework in Python

- Single and multichannel reconstruction methods
- Flexible: Easy to mix&match to prototype reconstruction algorithms
- www.ccpi.ac.uk/CIL

Courses and project oppotunities

- 02946 Scientific Computing for X-ray Computed Tomography
  January 2025
  https://kurser.dtu.dk/course/02946

- 47511 CINEMAX summer school
  One week, every year end of August
  https://www.conferencemanager.dk/cinemaxviii/conference

- **Open to BSc, MSc, PhD projects and specialized courses**
  - Computational methods
  - Collaboration with DTU 3D Imaging Center and other CT facilities

jakj@dtu.dk