Known Operator Learning for Modelling Prior Knowledge in Deep Networks

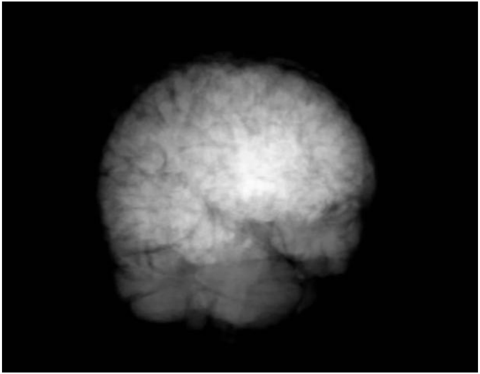
A. Maier, V. Christlein, K. Breininger, Z. Yang, L. Rist, M. Nau, S. Jaganathan, C. Liu, N. Maul, L. Folle, K. Packhäuser, M. Zinnen
Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg
April 24, 2023

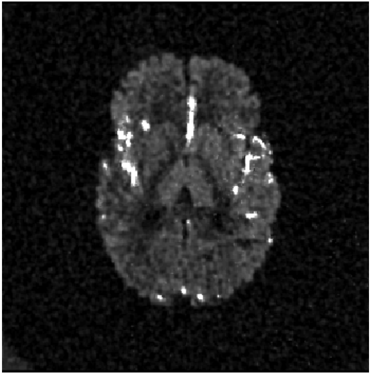
Pattern Recognition Lab

- One Professors
- One Lecturer
- Two Post-Docs
- $\approx$50 PhD Students
- Three Admins

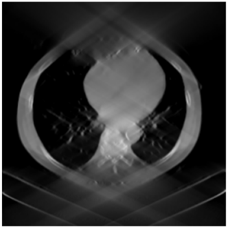
- Biomedical Image and Data Processing
- Computer Vision
- Speech Recognition
- Digital Humanities

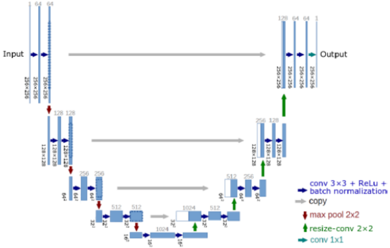
Image Reconstruction

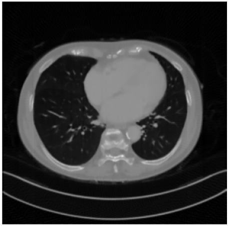




Deep Learning in Image Reconstruction?

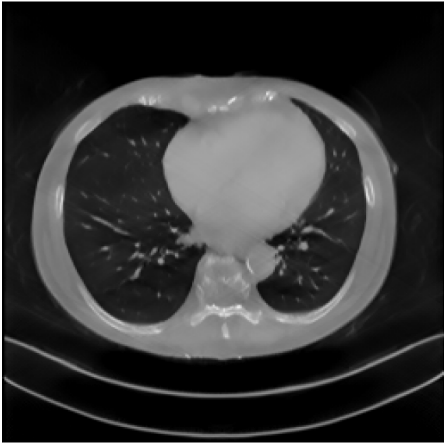






Source: Huang et al. [1]

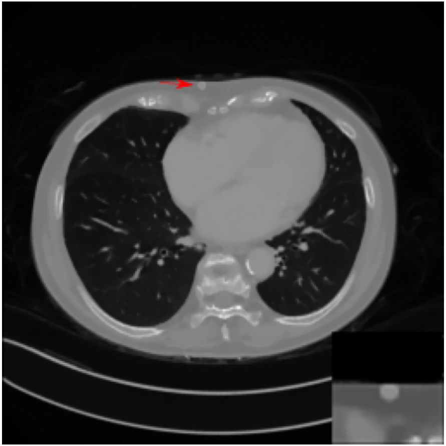
Deep Learning in Image Reconstruction?



Source: Huang et al. [1]

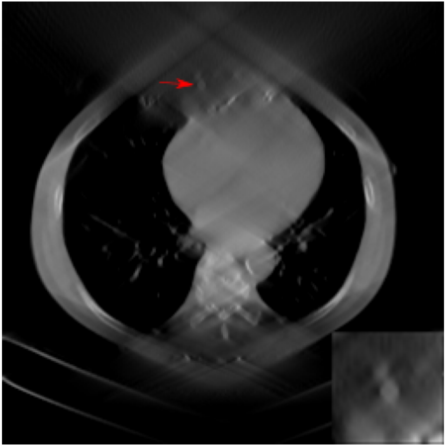
art/tf/fau-logo-english

Deep Learning in Image Reconstruction?



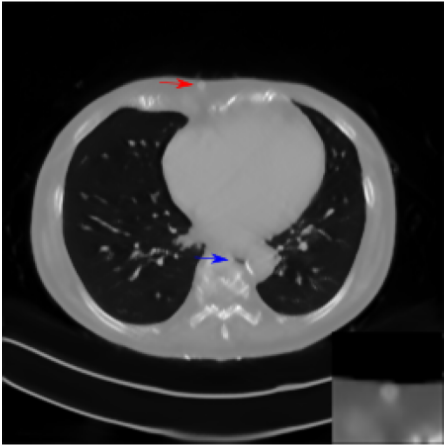
Source: Huang et al. [1]

Deep Learning in Image Reconstruction?



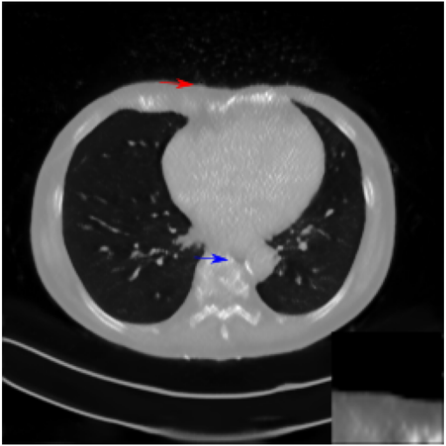
Source: Huang et al. [1]

Deep Learning in Image Reconstruction?



Source: Huang et al. [1]

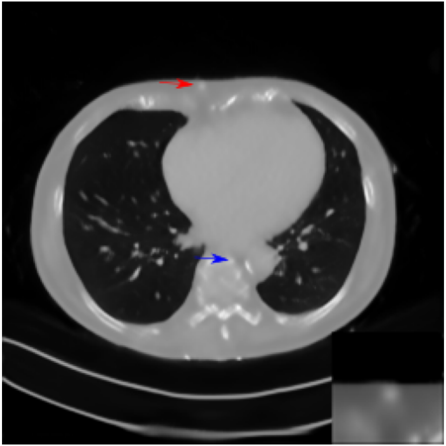
Deep Learning in Image Reconstruction?



Source: Huang et al. [1]

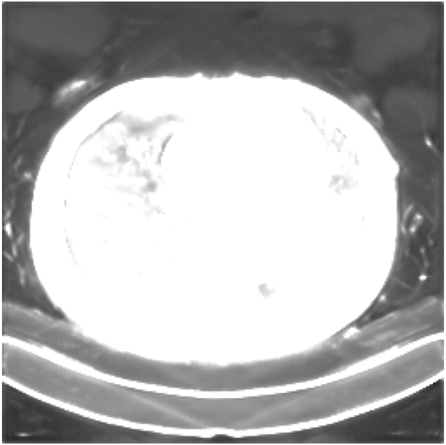
art/tf/fau-logo-english

Deep Learning in Image Reconstruction?



Source: Huang et al. [1]

Deep Learning in Image Reconstruction?



Source: Huang et al. [1]

Known Operators in Neural Networks

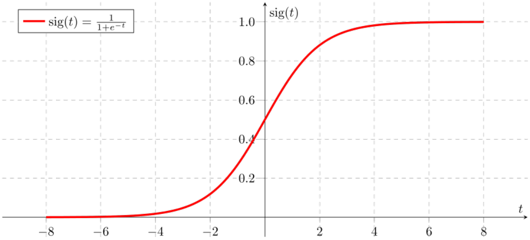
Universal Approximation Theorem

- Any continuous function can be approximated by Neural Net

$$u(\mathbf{x}) \approx U(\mathbf{x}) = \sum_i u_i s(\mathbf{w}_i^\top \mathbf{x} + w_{i,0}) \, ,$$

- The error is bound by

$$|U(\mathbf{x}) - u(\mathbf{x})| \leq \epsilon_u$$



Known Operator Learning Concept

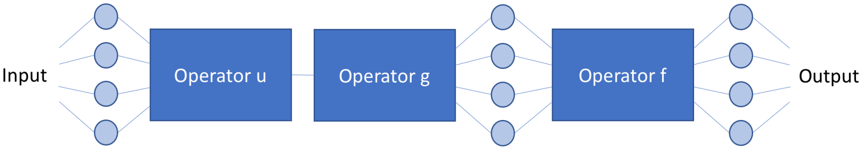


Figure: Known operators embedded in a neural network.

- Consider using known operators in learned mappings
- Specifically consider the use of two operators in sequence

$$f(\mathbf{x}) = g(u(\mathbf{x}))$$

Source: Maier et al. [5]

Approximation Sequences

- Sequential operations

$$f(\mathbf{x}) = g(u(\mathbf{x}))$$

- Can be approximated:

$$F_u(\mathbf{x}) = g(U(\mathbf{x})) = f(\mathbf{x}) - e_u$$
$$F_g(\mathbf{x}) = G(u(\mathbf{x})) = f(\mathbf{x}) - e_g$$
$$F(\mathbf{x}) = G(U(\mathbf{x})) = f(\mathbf{x}) - e_f$$

Source: Maier et al. [5]

Error of Approximation Sequences

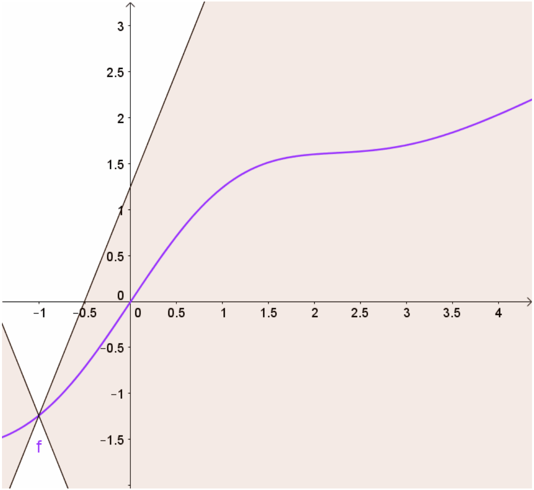
- Approximation introduces error

$$
\begin{aligned}
f(\mathbf{x}) = g(u(\mathbf{x})) &= G(u(\mathbf{x})) + e_g \\
&= \sum_j g_j s\left(u_j(\mathbf{x})\right) + g_0 + e_g \\
&= \sum_j g_j s\left(U_j(\mathbf{x}) + e_{u_j}\right) + g_0 + e_g
\end{aligned}
$$

- Can we find bounds on this error?

Source: Maier et al. [5]

Lipschitz Continous Functions



Observation on Bounds

- Bound on Error:

$$|e_f| \leq \underbrace{\sum_j |g_j| \cdot l_s \cdot |e_{u_j}|}_{\text{error } U(\mathbf{x})} + \underbrace{\epsilon_g}_{\text{error } G(x)}$$

- Observations
  - Error in $U(\mathbf{x})$ and $G(\mathbf{x})$ additive
  - Error of $U(\mathbf{x})$ amplified by $g(\mathbf{x})$
  - → Features are more important than classifiers
  - Requires Lipschitz continuity

Source: Maier et al. [5]

Computed Tomography

- Efficient solution via filtered back-projection:

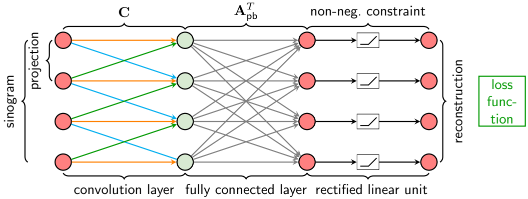
$$f(x, y) = \int_0^{\pi} p(s, \theta) * h(s)\big|_{s=x\cos\theta+y\sin\theta} d\theta$$

- Can also be derived in matrix notation:

$$\begin{aligned} \mathbf{Ax} &= \mathbf{p} \\ \mathbf{x} &= \mathbf{A}^{+}\mathbf{p} \\ \mathbf{x} &= \mathbf{A}^{T}(\mathbf{AA}^{T})^{-1}\mathbf{p} \end{aligned}$$

Reconstruction Networks

- All three steps can be modeled as a neural network:



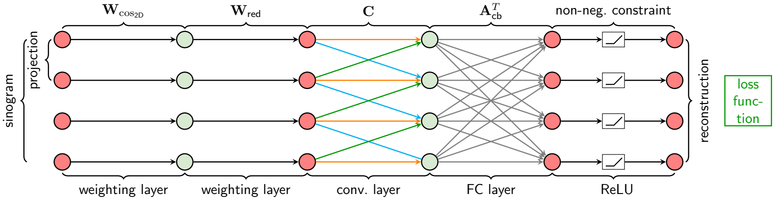
- All weights are known from FBP
- Operator **A** does not fit into memory

➔ compute on-the-fly

Source: Würfl et al. [2]

Reconstruction Networks

- Reconstruction Networks can be expanded



- Embedding of "heuristics" for artifact reduction possible

Source: Würfl et al. [3]

art/tf/fau-logo-english

Application to Incomplete Scans

Figure: Reconstruction with 360°

Application to Incomplete Scans

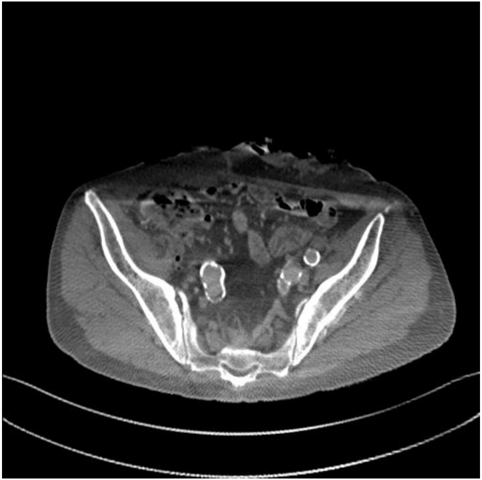


Figure: Reconstruction with 180° (FBP)

Application to Incomplete Scans

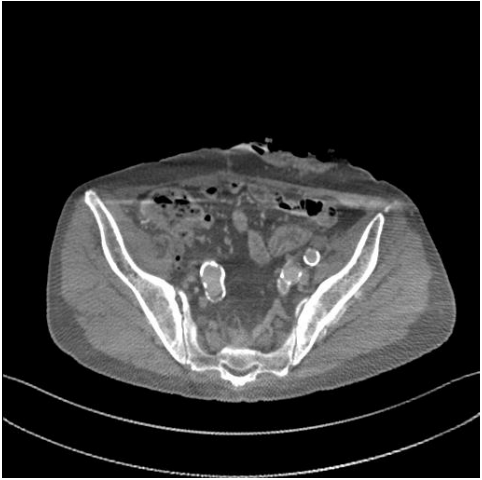


Figure: Reconstruction with 180° (NN)

Learned Weights

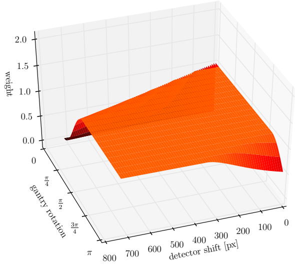


Figure: Parker et al. 1996

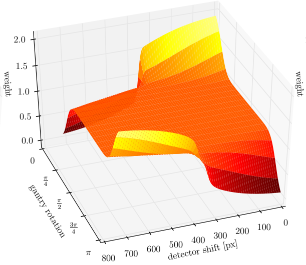


Figure: Schäfer et al. 2017

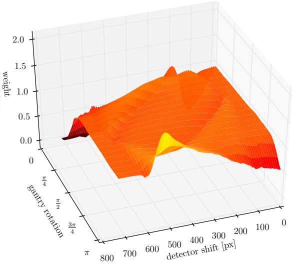


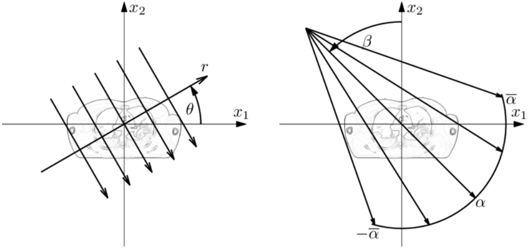
Figure: Neural Network 2018

Source: Würfl et al. [3]

Can we "derive" networks?

$$\mathbf{A}_{\angle}\mathbf{x} = \mathbf{p}_{\angle}$$

$$\mathbf{A}_{||}\mathbf{x} = \mathbf{p}_{||}$$



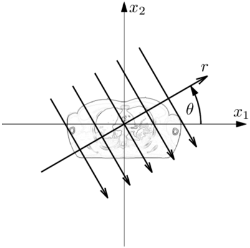
Source: Syben et al. [7]

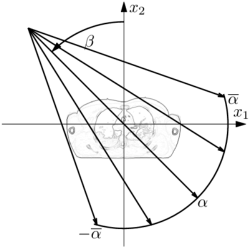
Can we "derive" networks?

$$\mathbf{A}_\angle \mathbf{x} = \mathbf{p}_\angle$$

$$\mathbf{A}_{||} \mathbf{x} = \mathbf{p}_{||}$$

$$\mathbf{x} = \underbrace{\mathbf{A}_\angle^T (\mathbf{A}_\angle \mathbf{A}_\angle^T)^{-1}}_{\text{ill-posed } \angle \text{ reconstruction}} \mathbf{p}_\angle$$





Source: Syben et al. [7]

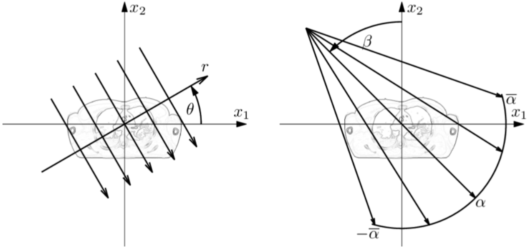
Can we "derive" networks?

$$\mathbf{A}_{\angle}\mathbf{x} = \mathbf{p}_{\angle}$$

$$\mathbf{A}_{||}\mathbf{x} = \mathbf{p}_{||}$$

$$\mathbf{x} = \underbrace{\mathbf{A}_{\angle}^{T}(\mathbf{A}_{\angle}\mathbf{A}_{\angle}^{T})^{-1}}_{\text{ill-posed } \angle \text{ reconstruction}} \mathbf{p}_{\angle}$$

$$\mathbf{p}_{||} = \mathbf{A}_{||}\mathbf{A}_{\angle}^{T} \underbrace{(\mathbf{A}_{\angle}\mathbf{A}_{\angle}^{T})^{-1}}_{\text{No efficient form}} \mathbf{p}_{\angle}$$



Source: Syben et al. [7]

Can we "derive" networks?

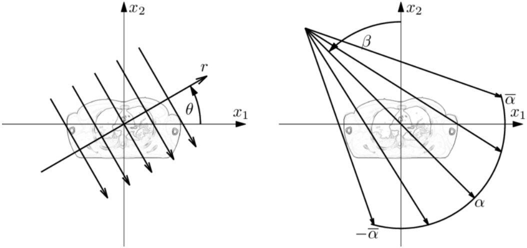
$$\mathbf{A}_\angle \mathbf{x} = \mathbf{p}_\angle$$

$$\mathbf{A}_{||} \mathbf{x} = \mathbf{p}_{||}$$

$$\mathbf{x} = \underbrace{\mathbf{A}_\angle^T (\mathbf{A}_\angle \mathbf{A}_\angle^T)^{-1}}_{\text{ill-posed } \angle \text{ reconstruction}} \mathbf{p}_\angle$$

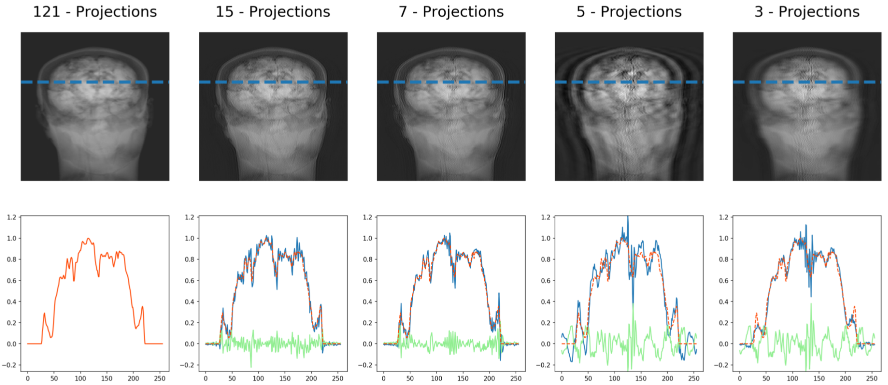
$$\mathbf{p}_{||} = \mathbf{A}_{||} \mathbf{A}_\angle^T \underbrace{(\mathbf{A}_\angle \mathbf{A}_\angle^T)^{-1}}_{\text{No efficient form}} \mathbf{p}_\angle$$

$$\mathbf{p}_{||} = \mathbf{A}_{||} \mathbf{A}_\angle^T \mathcal{F}^H \underbrace{\mathbf{K}}_{\text{model}} \mathcal{F} \mathbf{p}_\angle$$



Source: Syben et al. [7]

Learned Algorithm in Action



Source: Syben et al. [7]

art/tf/fau-logo-english

Known Operator Learning

- Many traditional approaches **equivalent** to neural networks and vice versa
- Learned algorithms are **again** traditional algorithms
- Learned parameters can be **interpreted**
- Virtually all state-of-the art methods can also be **integrated**
- Methods **efficient** and interpretable

art/tf/fau-logo-english

References

References I

[1] Y. Huang, T. Würfl, K. Breininger, L. Liu, G. Lauritsch, and A. Maier, “Some investigations on robustness of deep learning in limited angle tomography”, in International Conference on Medical Image Computing and Computer-Assisted Springer, 2018, pp. 145–153.

[2] T. Würfl, F. C. Ghesu, V. Christlein, and A. Maier, “Deep Learning Computed Tomography”, in MICCAI 2016, vol. 3, Athen, 2016, pp. 432–440.

[3] T. Würfl, M. Hoffmann, V. Christlein, K. Breininger, Y. Huang, M. Unberath, and A. Maier, “Deep Learning Computed Tomography: Learning Projection-Domain Weights from Image Domain in Limited Angle Problems”, IEEE Transactions on Medical Imaging, vol. 37, no. 6, pp. 1454–1463, 2018.

References II

[4] A. Maier, F. Schebesch, C. Syben, T. Würfl, S. Steidl, J.-H. Choi, and R. Fahrig, "Precision Learning: Towards Use of Known Operators in Neural Networks", in 2018 24rd International Conference on Pattern Recognition (ICPR), Beijing, China, 2018, pp. 183–188.

[5] A. K. Maier, C. Syben, B. Stimpel, T. Würfl, M. Hoffmann, F. Schebesch, W. Fu, L. Mill, L. Kling, and S. Christiansen, "Learning with known operators reduces maximum error bounds", Nature Machine Intelligence, vol. 1, no. 8, pp. 373–380, 2019.

[6] C. Syben, M. Michen, B. Stimpel, S. Seitz, S. Ploner, and A. K. Maier, "Technical note: Pyro-nn: Python reconstruction operators in neural networks", Medical Physics, vol. 0, no. 0, eprint: https://aapm.onlinelibrary.wiley.com/doi/pdf/10.1002/mp.13753.

References III

[7] C. Syben, B. Stimpel, J. Lommen, T. Würfl, A. Dörfler, and A. Maier, “Deriving neural network architectures using precision learning: Parallel-to-fan beam conversion”, in German Conference on Pattern Recognition, Springer, 2018, pp. 503–517.

[8] W. Fu, K. Breininger, R. Schaffert, N. Ravikumar, T. Würfl, J. Fujimoto, E. Moult, and A. Maier, “Frangi-net”, in Bildverarbeitung für die Medizin 2018, Springer, 2018, pp. 341–346.

[9] W. Fu, K. Breininger, R. Schaffert, N. Ravikumar, and A. Maier, “A divide-and-conquer approach towards understanding deep networks”, arXiv preprint arXiv:1907.06194, 2019.