Ill-posed Microlocal Imaging Using Targeted Deep Learning

Siiri Rautio

Doctoral dissertation

To be presented for public discussion with the permission of the Faculty of Science of the University of Helsinki, in Auditorium CK112, Exactum, Pietari Kalmin katu 5, on the 3rd of April, 2025 at 12 o'clock.

Department of Mathematics and Statistics
Faculty of Science
University of Helsinki

HELSINKI 2025

Supervisor

Professor Samuli Siltanen
Department of Mathematics and Statistics
University of Helsinki
Finland

Pre-examiners

Professor Sarah Hamilton
Department of Mathematical and Statistical Sciences
Marquette University
Wisconsin, US

Professor Kim Knudsen
Department of Applied Mathematics and Computer Science
Scientific Computing
Technical University of Denmark
Denmark

Opponent

Professor Elena Loli Piccolomini
Department of Computer Science and Engineering
University of Bologna
Italy

Custos

Professor Samuli Siltanen
Department of Mathematics and Statistics
University of Helsinki
Finland

Publisher: Helsingin yliopisto
Series: Dissertationes Universitatis Helsingiensis 92/2025

ISBN 978-952-84-0776-8 (print)
ISBN 978-952-84-0775-1 (online)
ISSN 2954-2898 (print)
ISSN 2954-2952 (online)
PunaMusta, Joensuu 2025

Abstract

Inverse problems concern the study of problems where we want to go from effect to cause, reconstructing an unknown target based on indirect measurements. Ill-posedness makes solving inverse problems difficult, as the solution might not exist, there might be multiple solutions, or the solution might not depend continuously on data. To combat the ill-posedness, we need to use *a priori* information in the form of regularization, by penalizing undesired solutions and thus guiding the result towards a suitable solution.

As computational power has grown, so has the potential and significance of deep learning in all computational sciences. In general, the more complicated the task is, the more data and learnable parameters the neural network needs to learn to solve it. So-called *targeted* deep learning takes a different approach: instead of trying to learn huge and complicated tasks, the task is decomposed into learned and non-learned parts, that are specifically tailored for the problem in hand. By targeting only a specific part of the problem with deep learning, the networks can be smaller and less expensive to train, and the end-result easier to interpret. Consequently, targeted deep learning requires specified knowledge of the problem and suitable traditional non-learned algorithms as well.

The field of inverse problems provides an interesting testing ground of problems for targeted deep learning to solve. Simply learning solutions of ill-posed inverse problems purely via deep learning would be difficult due to the ill-posedness, thus requiring a huge amount of data, and a waste of resources. There is a lot of underlying mathematical and physical information available, for example, the forward operator, that would be redundant putting effort into learning. Even after successfully training such a large model, the 'black box' nature of it would make it difficult to interpret the results. With targeted deep learning, the ill-posed part of the problem can be solved with robust mathematics, providing the learned part with already more robust features. This way, we can use deep learning to help solve even extremely complicated ill-posed inverse problems, by combining the strengths of both learned and traditional approaches, without a need for huge datasets or computational resources.

Three different ill-posed inverse problems in imaging are studied in this thesis: X-ray computed tomography, electrical impedance tomography, and image deconvolution. Publication I concerns learning a boundary estimate for limited-angle tomography, by making use of underlying microlocal information extracted with complex wavelets. In publication II, electrical impedance tomography is re-

formulated as virtual parallel-beam tomography, connecting it to X-ray computed tomography.With the method, EIT data can be processed into virtual sinograms, and reconstructed using methods from X-ray tomography. Publication III studies robust image deconvolution with bilevel learning of Fields-of-Experts filters for regularization.

Tiivistelmä

Tekoäly, erityisesti neuroverkkojen muodossa, on viime vuosikymmeninä noussut oleelliseksi osaksi laskennallista tiedettä. Kasvava laskentateho ja suuret datamäärät mahdollistavat yhä monimutkaisempien ongelmien ratkaisemisen, mutta neuroverkkojen luotettavuus ja tulkittavuus herättävät kysymyksiä. Neuroverkkomalleja luonnehditaan usein "mustiksi laatikoiksi", sillä niiden päätöksentekoa on vaikea ymmärtää ja valvoa, mikä olisi kriittistä erityisesti lääketieteellisissä sovelluksissa. Lisäksi suurten neuroverkkojen kouluttaminen vaatii merkittäviä resursseja ja laajoja opetusdatan kokoelmia, mikä on usein epäkäytännöllistä.

Näitä haasteita voidaan lieventää rajoittamalla neuroverkon osuutta menetelmässä. Jos ongelma ratkaistaan ensin tehokkain ja todistetuin matemaattisin menetelmin niin pitkälle kuin mahdollista, neuroverkon rooli voidaan kohdentaa vain kriittisiin kohtiin, joissa oppiminen tuo merkittävää lisäarvoa. Tämä mahdollistaa pienempien ja kevyempien neuroverkkojen käytön, mikä helpottaa koulutusta ja vähentää tarvittavan opetusdatan määrää. Lisäksi rajatummassa ongelmassa neuroverkon tuottaman ratkaisun arviointi ja tulkinta on selkeämpää.

Tämä väitöskirja käsittelee inversio-ongelmien ratkaisemista yhdistämällä perinteisiä algoritmeja ja tekoälyä edellä kuvaillulla tavalla. Inversio-ongelmissa kohinaisen mittaustuloksen perusteella pyritään ratkaisemaan tuntemattoman objektin ominaisuuksia. Tässä työssä tutkitaan kolmea kuvantamiseen liittyvää inversio-ongelmaa: röntgentomografiaa, sähköimpedanssitomografiaa ja valokuvan terävöittämistä eli dekonvoluutiota.

Röntgentomografiassa kohteesta otetaan röntgenkuvia eri suunnista, joiden perusteella lasketaan poikkileikkauskuva. Se paljastaa yksityiskohtaista informaatiota kappaleen sisältä, ja menetelmä onkin keskeinen esimerkiksi lääketieteellisessä diagnostiikassa. Käyttämällä parempia algoritmeja ja tekoälyn apua, voimme esimerkiksi vähentää tarvittavaa mittausdatan määrää ja samalla potilaan säteilykuormaa ilman, että kuvanlaatu heikkenee liikaa.

Sähköimpedanssitomografiassa käytetään röntgensäteilyn sijaan pientä sähkövirtaa, joka syötetään kohteeseen elektrodien kautta kohteen sähkönjohtavuuden selvittämiseksi. Koska tuntematon sähkönjohtavuus vaikuttaa sähkövirran kulkuun, tämä inversio-ongelma on epälineaarinen, tehden ratkaisusta hankalampaa. Lopputuloksena saatavat poikkileikkauskuvat eivät ole yhtä tarkkoja kuin röntgentomografiassa, mutta menetelmä on täysin harmiton, mikä avaa mahdollisuuksia esimerkiksi vauvojen keuhkokuvantamiseen ja aivoinfarktipotilaiden

jatkuvaan monitorointiin.

Valokuvan dekonvoluutiossa tavoitteena on palauttaa sumentunut signaali teräväksi. Tämä on haastavaa, sillä voimakkaan sumentumisen seurauksena osa yksityiskohdista voi olla pysyvästi menetetty. Tämä inversio-ongelma nousee kuitenkin usein esille niin perinteisessä kuin avaruuden valokuvauksessakin.

Acknowledgements

This year marks my 11th year at the University of Helsinki—a journey that has taken me from being a freshman who had never written a line of code to completing a PhD thesis in computational inverse problems. It has been an incredible experience of learning and growth, but I certainly could not have done it alone. Now, it is time to express my gratitude to those who have supported me along the way.

First and foremost, I would like to extend my deepest thanks to my supervisor, Professor Samuli Siltanen. In 2016, you welcomed me into the inverse problems research group by agreeing to supervise my bachelor's thesis, setting me on the path of academic research. Your guidance and encouragement from the very beginning sparked my passion for mathematics, and for that, I am truly grateful.

I am also profoundly grateful to Professor Elena Loli Piccolomini for traveling all the way from Italy to serve as the opponent for my defense. My sincere thanks to Professors Sarah Hamilton and Kim Knudsen for their careful pre-examination of this thesis. I also wish to thank my thesis committee members, Professors Andreas Hauptmann and Arto Klami, for their advice throughout my doctoral studies and their critical insights on this work. Additionally, I am grateful to Professors Lauri Oksanen and Petri Ola for serving as faculty representatives in my defense.

A special thank you goes to Professor Matti Lassas for all the guidance and wisdom you have shared about life as a researcher. Your patience and mentorship throughout our collaborations have meant a great deal to me. I would also like to thank Professors Melody Alsaker and Jennifer Mueller for the opportunities to collaborate and discuss electrical impedance tomography. The research visits with you were truly enriching experiences.

I deeply appreciate all of my research collaborators and co-authors. In no particular order, I would like to thank my co-authors who have not yet been mentioned: Tatiana Bubba, Rashmi Murthy, Fernando Silva de Moura, Juan Pablo Agnelli, Luca Calatroni, Thomas Pock, Ambra Catozzi, Serena Crisci, Monica Pragliola, Danilo Riccio, and Andrea Sebastiani. Research is a collaborative effort, and I am grateful to have worked alongside such talented individuals.

A heartfelt thank you to the doctoral researchers and postdocs who have been part of the inverse problems research group at the University of Helsinki during my time here. Your friendship, support, and discussions (both scientific and less scientific) have made these years unforgettable. I am especially grate-

ful to Heli Virtanen, Salla Latva-Äijö, Alexander Meaney, Markus Juvonen, Elli Karvonen, Ensio Suonperä, Tommi Heikkilä, Sara Sippola, Martin Ludvigsen, Bjørn Jensen, Otso Hyvärinen, Riina Virta, Lauri Ylinen, Anna Suomenrinne-Nordvik, Joel Enwald, Luca Ratti, and Neil Dizon. Together, we traveled the world—from Amsterdam to Milan, Rome, Bologna, Costa Rica, London, Cambridge, Bath, Dublin, Houston, New Orleans, Atlanta, Toulouse, Bonn, Göttingen, Tokyo, Copenhagen, Geneva, and Fort Collins—building friendships and memories that will last a lifetime. Particularly, Heli, we survived the Covid lockdowns together, and throughout this journey you were always there to support me, on both good days and bad. I honestly couldn't have managed without you.

Beyond the inverse problems group, I want to thank my fellow doctoral students in the Doctoral Program in Mathematics and Statistics (Domast). In particular, I would like to mention Janne Siipola. Thank you for the conference trips, seminars, collaborations, conversations, and, most importantly, for sharing a passion for machine learning.

I am also grateful to my project collaborators from the University of Oulu: Mikael Brix, Satu Inkinen, Juuso Ketola, Antti Isosalo, Mustafa Al-Rubaye, Ella Räinä, Dinidu Jayakody Arachchilage, and others. Through our work on the AIDMEI and TOMOHEAD projects, I had the opportunity to collaborate with industry partners, gaining invaluable experience along the way. A special thank you to Planmeca and all the other companies involved in these projects.

I would like to acknowledge the financial support that made this journey possible. I am grateful to the University of Helsinki for funding my salaried four-year doctoral researcher position. I was also partially supported by the Finnish Centre of Excellence of Inverse Modelling, Business Finland (funding decision 8132/31/2022), and more recently, the FAME Flagship of Advanced Mathematics for Sensing, Imaging, and Modelling. I also wish to thank the DOMAST doctoral school for travel support.

And finally, to Ossi—thank you for your unwavering support and belief in me. It truly means the world to me.

Helsinki, March 2025
Siiri Rautio

The thesis consists of this introduction and the following list of publications (I - III). The author's contributions are summarized below.

List of publications

I Rautio, S., Murthy, R., Bubba, T. A., Lassas, M., & Siltanen, S.
Learning a microlocal prior for limited-angle tomography (2023). IMA Journal of Applied Mathematics, 88(6), 888-916.

II Alsaker, M., Rautio, S., Moura, F., Agnelli, J.P., Murthy, R., Lassas, M., Mueller, J., & Siltanen, S.
CT scans without X-rays: parallel-beam imaging from nonlinear current flows (2024). (Submitted) arXiv preprint arXiv:2408.12992.

III Bubba, T. A., Calatroni, L., Catozzi, A., Crisci, S., Pock, T., Pragliola, M., Rautio, S., Riccio, D., & Sebastiani, A.
Bilevel learning of regularization models and their discretization for image deblurring and super-resolution (2022). INdAM Workshop: Advanced Techniques in Optimization for Machine learning and Imaging. Singapore: Springer Nature Singapore.

Author's contributions

I The author developed and implemented the proposed method and trained the neural networks. She was also responsible for the writing process.

II The author trained and implemented the neural networks while also contributing to the development and implementation of the method. She contributed to the writing of the paper.

III The author contributed to the numerical implementation of the method and computation of the results for learning the Fields-of-Experts regularization model.

Nomenclature

Acronyms

2D	Two-Dimensional
3D	Three-Dimensional
ADAM	ADAptive Moment estimation
AP	Astala-Päivärinta
BN	Batch Normalization
CGO	Complex Geometrical Optics
CNN	Convolutional Neural Network
CT	Computed Tomography
DN	Dirichlet-to-Neumann
DT-CWT	Dual-Tree Complex Wavelet Transform
EIT	Electrical Impedance Tomography
FBP	Filtered BackProjection
FoE	Fields-of-Experts
GMRES	Generalized Minimal RESidual
HU	Hounsfield Unit
MSE	Mean-Squared Error
PDE	Partial Differential Equation
PDFP	Primal-Dual Fixed-Point
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
TV	Total Variation
VHPT	Virtual Hybrid Parallel-beam Tomography

Mathematical notation

A	Forward operator in the model $m = Af + \epsilon$ describing the measurement process.
a_k	Wavelet approximation coefficients.
b	Bias term in a neural network.
$d_{j,k}$	Wavelet detail coefficients.
$\mathcal{F}$	Fourier operator.
f	Target that we want to solve for in the model $m = Af + \epsilon$.
$f_\theta(x_n)$	Neural network prediction for input x_n with parameter values θ.
$\widehat{f}(\xi)$	Fourier transform of $f(x)$.
$\widetilde{f}$	Solution for the model $m = Af + \epsilon$ computed using variational regularization.
$I(x)$	Intensity of X-ray radiation at point x.
(i, j)	Pixels of an image.
$J(z)$	Current-density distribution on the boundary $z \in \partial\Omega$.
k	1. Convolution kernel for the convolution model $(k * f)(x)$. 2. Complex spectral parameter $k = \tau e^{i\phi}$.
M	Number of layers in a neural network.
m	Vector containing the measurement data in the model $m = Af + \epsilon$.
N	Number of elements.
$\mathcal{R}f$	Radon transform of function f.
$R(f)$	Regularization term.
$S(i, j)$	Feature map.
$WF(f)$	Wavefront set of a signal f.
w	Weight in a neural network.
x_n	Element in the training set corresponding to the training input.
x'_n	Normalized element in the training set corresponding to the training input.
$\{(x_n, y_n)\}$	Paired dataset.

y_n	Element in the training set corresponding to the training output, also known as the ground truth.
y'_n	Normalized element in the training set corresponding to the training output, also known as the ground truth.
$\widehat{y_n}$	Predicted output of a neural network.
α	Regularization parameter.
ϵ	Error and noise term in the model $m = Af + \epsilon$.
η	Step size in gradient descent.
Λ_σ	Voltage-to-current map, also known as the Dirichlet-to-Neumann map.
$\mathcal{T}$	Soft-thresholding operator.
μ	Beltrami coefficient.
ω	Trace of the Beltrami CGO solution on the boundary.
$\overline{\partial}_z$	The D-bar operator.
Φ	Loss function measuring the performance of a learned model.
ϕ_n	Trigonometric basis function.
$\phi_{j,k}$	Wavelet scaling function.
$\psi^\nu_{j,k}$	Wavelet function with filter ν.
σ	Conductivity.
θ	1. Learned parameter values. 2. Angle, either in radians or degrees.
$\varphi(z)$	Activation function in a neural network.

Contents

1 Introduction **1**

2 Deep learning **5**
2.1 Principles of supervised learning 5
2.2 Neural network structure 7
2.3 Convolutional neural networks 9
2.3.1 U-Net architecture 10

3 Image deconvolution **13**
3.1 Mathematical formulation 13
3.2 Total variation regularization 16
3.3 Fields-of-Experts regularization 17
3.4 Bilevel learning of regularization parameters 18

4 X-ray computed tomography **21**
4.1 Mathematical formulation 21
4.2 Limited-angle tomography 24
4.3 Wavelet-based regularization 27
4.3.1 Haar wavelet transform 28
4.3.2 Complex wavelet transform 29
4.4 Primal-dual fixed-point algorithm 31

5 Electrical impedance tomography **33**
5.1 Mathematical formulation 33
5.2 Complex geometrical optics solutions 35
5.2.2 Computation from the forward problem 36
5.2.3 Computation from the inverse problem 38
5.3 Propagation of singularities 40

6 Summary of papers **45**
6.1 Learning a microlocal prior for limited-angle tomography 45
6.2 CT scans without X-rays: parallel-beam imaging from nonlinear current flows 47
6.3 Bilevel learning of regularization models and their discretization for image deblurring and super-resolution 49

Chapter 1

Introduction

In this thesis, we explore combining deep learning with robust, non-learned inverse problems algorithms for different applications in imaging. By combining the two approaches, we can harness their individual strengths. Using what we call *targeted* deep learning, we can limit the learned part of the problem to be as small as possible, and solve the rest of the problem with robust inverse problems mathematics.

By using rigorous mathematics to solve the problem as far as we can, and only solving the most difficult part with learning where traditional algorithms fail, we can limit the learned part and thus use simpler models with less training data. The results are more explainable, since we have a better grasp on what the restricted, learned part of problem is, as opposed to learning a complex problem end-to-end. For such complicated problems as ill-posed inverse problems, end-to-end learning would typically require more resources data-wise and computationally, so it makes sense to try to approach these types of problems by targeting only a specific part of it with learning.

We explore this approach of using targeted learning for three different imaging applications in ill-posed inverse problems: image deblurring, X-ray computed tomography (CT), and electrical impedance tomography (EIT). See Figure 1.1 for an illustration of each imaging problem, with the target and measurement data visualized.

In inverse problems in general, we want to reconstruct a target object based on noise-corrupted indirect measurements of the object. For image deblurring (or more mathematically: image deconvolution), this means recovering the original, sharp image from a blurry image. On the other hand, X-ray CT and EIT are medical imaging methods where the aim is to reconstruct the inner structure of an object based on external measurements. Whereas X-ray CT is linear, EIT is a nonlinear inverse problem, which makes solving it an even more difficult task.

Mathematically, an inverse problem is formulated as

$$m = Af + \epsilon, \tag{1.1}$$

where m is the measurement, A is the known forward operator describing the

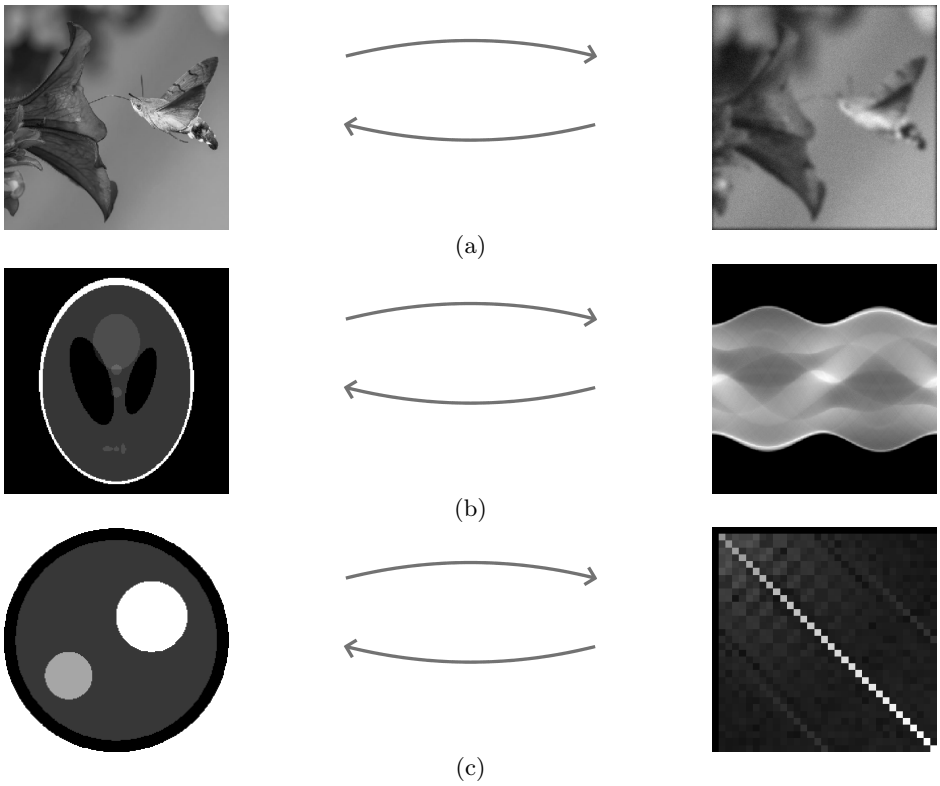


Figure 1.1: Illustrations of ill-posed inverse problems: (a) image deblurring, (b) X-ray tomography reconstruction, and (c) electrical impedance tomography reconstruction. In (a), blurring the image[1]corresponds to the *direct* problem, whereas the *inverse* problem is deblurring the image. In (b) and (c), taking the X-ray or EIT measurements of the target is the *direct* problem, while reconstructing the tomographic slice image from the data (sinogram or DN-map, respectively) corresponds to the *inverse* problem.

[1] Original photograph taken by Markus Juvonen.

measurement process, f is the unknown target we wish to reconstruct, and ϵ is an error term, modeling the unavoidable error in the measurement process not accounted for in the model.

In the field of inverse problems, we are specifically interested in ill-posed problems that violate at least one of the following Hadamard's conditions [1].

Definition 1.0.1 (Hadamard's conditions for well-posed problems)**.**

I The problem has at least one solution.

II The problem has at most one solution.

III The solution's behavior changes continuously with data.

If these conditions are not met, we cannot solve the problem in a naive manner. Hadamard's third condition can be especially tricky. Even in the case where conditions I and II are met, the operator A is linear, and its inverse A^{-1} exists, trying to solve equation (1.1) simply by inverting the operator A:

$$f = A^{-1}m \tag{1.2}$$

does not work if there is even a small amount of noise ϵ present. In that case, we have

$$A^{-1}(m + \epsilon) = A^{-1}m + A^{-1}\epsilon = f + A^{-1}\epsilon \neq f,$$

where operator A^{-1} could even boost the effect of the noise ϵ. Because of this, the naive approach in equation (1.2) is not a viable option for solving ill-posed inverse problems, as it results in a failed reconstruction if there is even a small amount of error present. The severity of the ill-posedness can be investigated by studying the singular values and condition number of matrix A.

To combat the ill-posedness, we can introduce regularization to the process, that guides the solution to a desirable outcome [2, 3, 4]. This is done by utilizing *a priori* knowledge of the solution, enforcing a penalty for unsuitable outcomes. We can then formulate the solution to the inverse problem (1.1) as

$$\widetilde{f} = \arg\min\{||Af - m||_2^2 + \alpha R(f)\},$$

where the first term in the sum is the *data fidelity* term, measuring with L^2-norm how well the solution corresponds to the measured data. The second term consists of the *regularization* term $R(f)$, penalizing undesirable solutions, where α is the regularization parameter determining how much weight we give to the penalty. The choice of the regularization term depends on what kind of results are desirable for the problem at hand. Some ill-posed inverse problems are, however, so difficult to solve that even with strong regularization the end-results are lacking. To improve the results further, we need to go beyond traditional mathematical algorithms.

In the early 2010s, machine learning has risen as an efficient approach for scientific applications alongside more conventional approaches. The past decade has been a triumph for *deep learning*, as computational power has made training large neural networks possible [5, 6, 7] together with end-to-end open source machine learning platforms like Tensorflow [8]. However, large models typically require large training datasets. In problems like medical imaging, this creates a bottleneck, as it is not easy to get large amounts of experimental data, especially patient data. Also, within deep learning, there exists the inherent problem of interpretability and trustworthiness of the results [9, 10, 11, 12, 13]. Notably, in medical imaging, it is crucial that we understand why a neural network has reached its decision, or if it has made a mistake [14]. This is why we are interested in using targeted learning and not solving the inverse problem in a black-box manner.

One of the earliest applications of this kind of approach is applying learned post-processing, where the result of an inverse problem algorithm is improved via learning to remove noise, blurring, or artefacts. See for example [15, 16, 17]

for applications in X-ray tomography and [18] for EIT. In addition to post- or pre-processing, deep learning can also be used as an intermediate step in the algorithm, see for example [19, 20]. Many authors have approached combining data-driven and model-driven inverse problems methods in different ways, see for example [21] for a comprehensive review. Some approaches include learning in approximate analytic inversion [22, 23, 24], learning a regularizer [25, 26, 27], Plug-and-Play priors [28], neural networks as regularizers [29, 30], data-driven optimization [31, 32, 33, 34], and deep inverse priors [35].

In this work, publications I and II consider targeted learning as an intermediate step in the imaging pipeline, while also making use of ideas from microlocal analysis. Microlocal analysis is a useful tool in pure and applied mathematics, and it was introduced in the 1970s by [36, 37]. Publication I studies how we can use it to recover boundary information in limited-angle tomography. Publication II introduces a curious imaging technique building upon singularity propagation via microlocal analysis. Publication III, on the other hand, approaches targeted learning by learning a regularizer with bilevel optimization.

The rest of this thesis overview is organized as follows. In Chapter 2, we explain the principles of convolutional neural networks and deep learning. This approach is used for targeted learning in publications I-II. After this, we give a mathematical background to the three inverse problems that considered. Chapter 3 gives an introduction to image deconvolution with total variation and Fields-of-Experts regularization, as well as introduces the concept of bilevel learning used in publication III. Chapter 4 goes through the mathematical formulation for X-ray computed tomography and discusses the limited-angle case with wavelet-based regularization. In Chapter 5, we explain electrical impedance tomography and the idea behind virtual hybrid parallel-beam tomography. The results and summaries of the publications are discussed in Chapter 6.

Chapter 2

Deep learning

2.1 Principles of supervised learning

As computational power has continued to grow in recent decades, so has the success of machine learning algorithms. A machine learning algorithm aims to approximate an unknown function f^* by finding optimal parameter values θ, so that the learned function f_θ is as close to the underlying function f^* as possible:

$$f_\theta \approx f^*.$$

This is done by minimizing the empirical risk

$$\min_{f_\theta} \frac{1}{N} \sum_{n=1}^{N} \Phi(y_n, f_\theta(x_n)), \tag{2.1}$$

measured with a loss function Φ over a dataset (x_n, y_n). The term *learning* refers to optimizing the model parameters θ for the specific task. In this thesis, we solely consider supervised learning in the context of neural networks.

In supervised learning, the machine learning algorithm is given a dataset including ground truth values: for each input training example x_n there exists a target result y_n, which the algorithm is instructed to output. The idea is that the algorithm learns to make a prediction $f_\theta(x_n) = \widehat{y_n}$ from the knowledge of x_n, by estimating $p(\widehat{y_n}|x_n)$. Here, the training data consists of pairs $\{(x_n, y_n)\}_{n=1}^{N}$, where N is the size of the dataset. An example of supervised learning is solving the image deconvolution problem with a neural network, where the training data includes a pair of blurry and sharp instances of the same image.

Data is, without a doubt, the most important component of machine learning. During pre-processing, all data is typically normalized to the interval $[0, 1]$:

$$x'_n = \frac{x_n - \min(x_n)}{\max(x_n) - \min(x_n)},$$

and the same normalization is performed for the ground truth values y_n. This helps the model to convergence and prevent features with large magnitudes from

dominating. The available data is split into three parts: the training, validation, and testing sets. The training process is influenced by the training and validation sets, while the testing set is left solely for final testing purposes. The training set is used to learn the model parameters θ, while the validation set aims to estimate the testing error while the training is still ongoing, enabling the hyperparameters to be optimized appropriately. Typically, a 80/20 ratio is used to split the training and validation sets.

After the training is completed and the model is set, the testing data is used to test the model on unseen data and study the generalization. To objectively assess the performance of the model, it is important that no examples in the testing data have been used to make any choices concerning the model.

To measure the performance of a machine learning algorithm, we must choose a loss function Φ that measures how well the algorithm is performing, by comparing the predicted values to the known ground truth values. The goal is to find the parameter values θ that minimize the loss function Φ. A typical choice for image processing tasks is the average of the continuous-valued L_2 loss over the training set, also known as the mean-squared error:

$$\text{MSE}(D, \theta) = \frac{1}{N} \sum_{n=1}^{N} ||y_n - f_\theta(x_n)||_2^2,$$

where y_n is the ground truth, $f_\theta(x_n)$ is the corresponding network prediction, and $D = \{(x_n, y_n)_{n=1}^N\}$ is the set of training data points.

During the learning process, the learnable parameters θ are updated in order to minimize the loss function Φ_θ. The gradient descent algorithm is used to make corrective adjustments to the learned parameters that move the model towards an optimal set of parameter values. It is a first-order iterative optimization algorithm introduced already in 1847 by Cauchy [38]. It requires only the computation of the first-order gradient in order to find a local or global minimum of a function. The gradient descent method is based on the idea that a differentiable function g at point a decreases fastest, if one goes in the direction of the negative gradient $-\nabla g(a)$. Thus, we can iterate towards the minimum of the loss function using step size η by computing

$$\theta_{i+1} = \theta_i - \eta \nabla \Phi_{\theta_i},$$

where the gradient is taken with respect to the model parameters θ. The parameters θ are initialized to random values and updated after every step towards the negative gradient, to reach the smallest possible value of Φ_θ. The loss is calculated over the entire training dataset for each iteration i, so the process becomes computationally demanding for large datasets. Then, the input data x_n is forward-propagated through the network to produce an updated output $f_{\theta_i}(x_n)$ and a new training error value. This process is iterated until a sufficiently small training error is reached.

In order to compute the gradient, the information from the loss function flows backward through a process called back-propagation. It is an efficient way of computing the gradients with automatic differentiation, by saving intermediate

states in the forward pass. To solve for the gradient, we iterate through the data points using the new parameter values θ_{i+1} and compute the corresponding partial derivatives. The new gradient gives the slope of the loss function with the current parameter values and the direction to re-update them.

Stochastic gradient descent (SGD) is a stochastic approximation of the gradient descent algorithm, where randomly selected samples are used to evaluate the gradients [39]. It is computationally more efficient, as it calculates the loss and updates the model parameters based on those selected samples. It converges almost surely to a minimum of the loss function, if the learning rate η adaptively decreases with a suitable rate. While SGD is simple and effective, other popular choices include, for example, the Adaptive moment estimation (ADAM) [40] optimization method, which while being computationally more demanding, often converges faster.

2.2 Neural network structure

Computational neural networks were introduced in the 1950s, and their construction was inspired by the principles of the biological neuron [41, 42]. In standard feed-forward neural networks, information flows from the input neurons to the output of the network through intermediate neuron connections. See Figure 2.1 for an illustration of a general feed-forward neural network. In the training process, the network $f_\theta(x)$ is driven to approximate function $f^*(x)$ by finding the optimal parameter values θ based on evaluating predicted examples at different points x_n.

A neural network consists of neurons that are grouped into layers. Each neuron takes in the sum of its inputs x_i that are multiplied by their corresponding weights w_i:

$$\begin{aligned} z &= b + \sum_{i=1}^{m} w_i x_i \\ &= b + w^T x, \end{aligned}$$

where b is known as the bias term. The weights w_i and the bias term values b are the parameter values θ that are optimized (learned) during the network training process. Positive weights activate the neuron, whereas negative weights inhibit it.

Then, the output z of the neuron is passed through a nonlinear activation function $\varphi(z)$. The activation function forces the pre-activation output to a certain range. It needs to be piece-wise differentiable for gradient-based optimization necessary for the training process. Commonly used activation functions include the rectified linear unit (ReLU) function

$$\varphi_{\text{relu}}(z) = \max(0, z),$$

that limits to output to $[0, \infty]$, and the sigmoid function

$$\varphi_{\text{sig}}(z) = \frac{1}{1 + e^{-z}},$$

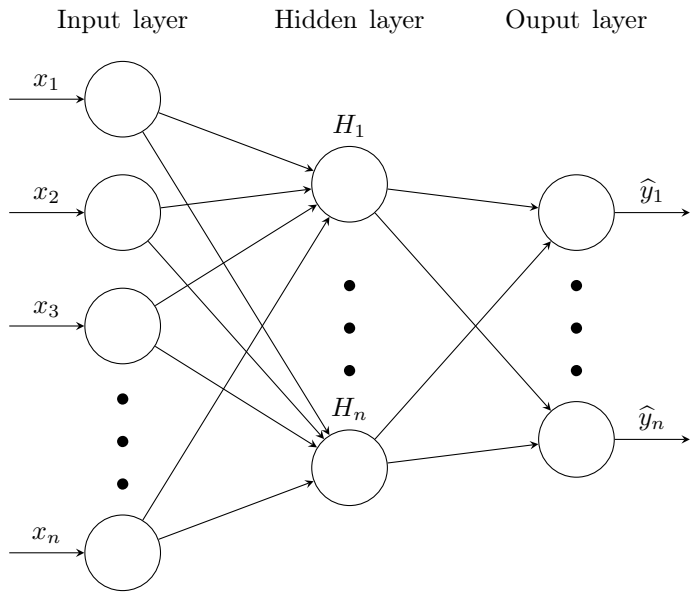


Figure 2.1: General network structure of a fully connected feed-forward neural network with one hidden layer. The number of hidden layers can vary from $1, \ldots, M$.

that ranges between $[-1, 1]$. It is important to note that while the ReLU activation function is not differentiable at $z = 0$, in practice, gradient descent can be performed since the neural network rarely arrives at the true minimum of the loss function. The gradient of ReLU is always 1 when the unit is active, so it is, nonetheless, easy to optimize.

A neural network architecture consists of an input layer, one or more hidden layers, and an output layer:

$$f_\theta(x) = f_\theta^{(M)} \circ f_\theta^{(M-1)} \cdots \circ f_\theta^{(1)}(x).$$

The inputs $x_1, x_2, \ldots, x_N$ are connected to neurons on a layer. The number of neurons on each layer can vary. The intermediate output $\widehat{y}^{(m)}$ is given as an input to the next layer of the network, until the output layer is reached. The information x_n propagates from the input neurons to the following hidden units and after flowing through all layers, the final output $f_\theta(x) = \widehat{y}$ is produced. The depth of the network is determined by the number of layers, while the width is the dimensionality of the hidden layers. The width of the network grows exponentially with problem dimension. This phenomenon is known as the curse of dimensionality [43].

According to the universal approximation theorem [44, 45], a feed-forward neural network is able to approximate any continuous function arbitrarily well if it has a linear output layer, at least one hidden layer with nonlinear activation

functions, and enough hidden units. However, in practice, there is no guarantee that during training the learning algorithm will be able to find the correct weights [46, 47]. Moreover, the no free lunch theorem [48] states that a universally superior learning algorithm does not exist, as it is impossible to ensure satisfactory generalization on unseen data.

In addition, solving the empirical risk minimization problem in equation (2.1) is ill-posed itself [49] and typically non-convex, including multiple local minima [50]. There are multiple ways of applying regularization in the learning process, including using randomly initialized stochastic gradient descent with sufficiently rapidly decreasing loss, early stopping, and training over mini-batches [51].

In practice, the more complicated the task is, the more learnable parameters the network requires to converge to a sufficient result. Supervised learning typically relies on deep neural networks that can approximate a wide range of nonlinear functions. There is empirical evidence suggesting that the network architecture choice has an influence on the ill-posedness [52, 53], in the sense that for deep networks, most local minima are almost global [54, 55]. However, the more parameters the network has, the more resources it requires to converge at all. So there exists a trade-off between model complexity and performance, especially concerning generalization on unseen data. This is one of the motivations for targeted learning: to restrict the complexity of the task to avoid unnecessarily large models, thus improving model performance.

2.3 Convolutional neural networks

Convolutional neural networks (CNNs) were introduced in 1989 [56] and are nowadays one of the standard approaches for image-to-image learning [57]. CNNs are a type of feed-forward neural network that uses weight sharing implemented by the convolution operation in place of matrix multiplication. Thus, instead of having to learn the full weight matrices (as with fully connected neural networks), CNNs optimize instead convolution kernels (filters) that are shared by each layer. This reduces the memory requirements of the model and improves its statistical efficiency.

The hidden layers of a CNN include one or more layers that perform the 2D convolution operation[2]:

$$S(i,j) = (h * I)(i,j) = \sum_m \sum_n I(i+m, j+n) k(m,n),$$

where we take a dot product of the convolution kernel $k(m,n)$ with the layer's input as matrix I. As the convolution kernel $k(m,n)$ slides along the input matrix I of the layer, the convolution operation generates a feature map $S(i,j)$ (see figure 2.2), which after passing through the activation function, becomes in

[2]Usually, the convolution operation used in a convolutional neural network is actually a related function called the cross-correlation, which corresponds to the convolution operation without flipping the kernel. From now on, in the context of neural networks, the cross-correlation operation will be referred to as the convolution operation.

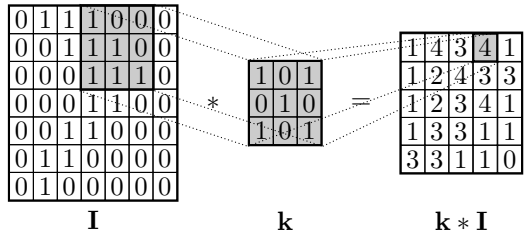


Figure 2.2: An illustration of the convolution operation on a binary pixel image. The convolution kernel $k(m, n)$ slides over the matrix I, resulting in a feature map $(k * I)(i, j) = S(i, j)$.

turn the input matrix for the next layer. As learning progresses, the network learns the optimal filters $k(m, n)$ that activate in detecting certain features at different spatial positions of the input.

Convolutions are able to encode certain image features, such as edges and corners. They act as a feature detector, as specific values in the convolution kernel $k(m, n)$ give a strong response to certain features. The aim is to find the suitable parameters θ, i.e. filters $k(m, n)$ and biases terms $b(m, n)$, so that the CNN performs the required task. The size of the learned filters $k(m, n)$ is often small, usually 3×3, to enable fast computation and easier training. Due to the convolutions, CNNs are shift invariant and provide translation-equivariant feature maps.

2.3.1 U-Net architecture

A more complicated, but regularized version of convolutional neural networks is the U-Net architecture [58], which has in practice proven to be a successful architecture choice for various image-to-image tasks, for example in the context of post-processing CT images [59, 60]. It was originally designed for image segmentation, but it can be used for other image-to-image problems as well. The properties of the U-Net allow it to achieve better generalization for complicated image-to-image tasks than standard CNNs. See Figure 2.3 for an illustration of an example architecture for 512×512 images.

The U-Net architecture consists of separate *encoder* and *decoder* sections, where the first half of the network incrementally down-samples the data (encoder), while the latter half subsequently up-scales it back to the original resolution (decoder). The two sides of the U-Net are connected via concatenated skip connections: the early layers of the network are directly connected to the deeper ones with the corresponding dimension [61].[3] This helps to propagate the information from the encoder layers to the decoder ones. As the information

[3]Note that compared to the ResNet architecture [61], the U-Net also has a large number of feature channels in the up-sampling part of the network.

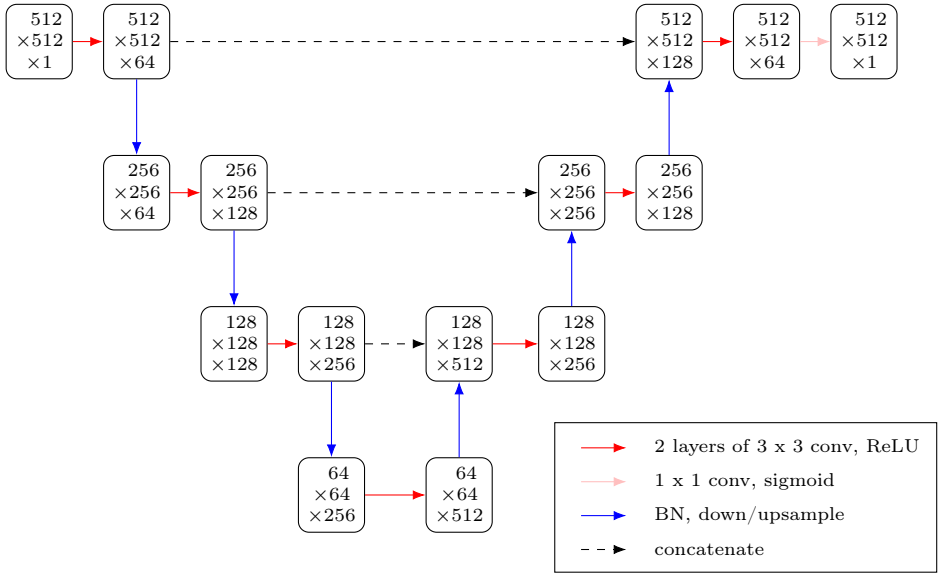


Figure 2.3: An example U-net architecture for a 512×512 greyscale image. The size of each layer is determined by the size of its feature maps and the number of channels. Note that the U-net typically has two convolutions in each layer.

passes directly through the network, vanishing gradients are usually not a problem during training. This also acts as a form of regularization for the network.

In the encoder part of the U-Net, the max-pooling operation implements a nonlinear down-sampling, that takes the maximum output within a rectangular neighborhood, thus reducing the dimensionality of the input:

$$\text{MaxPooling}(X)_{i,j} = \max_{m,n} X_{i \cdot s_x + m, j \cdot s_y + n},$$

where X is the input patch, i, j are the indices of the output feature map, and s_x, s_y are the vertical and horizontal stride values (typically $s_x = s_y = 2$). Performing the pooling maintains the most significant features while reducing the number of parameters and providing local translation invariance to the features. By doing this, it also regularizes the network weights, combating overfitting to training data.

Another form of regularization that the U-Net commonly has is batch normalization (BN) [62], that is a technique for making the training of the network more stable. It re-normalizes the values in a hidden layer, so that the subsequent input for the next layer remains normalized and the gradient descent will converge better during training. However, beyond simply normalizing the values to zero mean and unit variance:

$$x_i' = \frac{x_i - \mu_B}{\sqrt{(\sigma_B)^2 + \epsilon}},$$

where we have the empirical mean of mini-batch B of size m:

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i$$

and the variance:

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_B)^2,$$

it allows the values to be shifted and scaled by parameters γ, β that are learned during training:

$$x_i'' = \gamma x_i' + \beta.$$

The network optimizes the γ, β values to result in the best predictions, stabilizing the training process and improving the generalizability.

The general idea of the U-Net architecture can be extended and modified upon, for example, by experimenting with the number of layers (network depth) and the number of channels, depending on the application and its complexity. To learn highly nonlinear features in order to model more complicated functions, networks usually have a large number of layers and thus tend to have millions of parameters to learn. Using these types of networks is called deep learning.

There are different strategies for employing targeted learning to ill-posed inverse problems. One approach is to use a deep neural network, like the U-Net, to solve a specific part of the problem pipeline that would be otherwise difficult to solve using non-learned mathematical methods. Publications I and II use this kind of strategy. Another approach for targeted learning is to cast the inverse problem as a bilevel learning problem (see Section 3.4). Publication III explores this strategy.

Chapter 3

Image deconvolution

3.1 Mathematical formulation

In operation of convolution, each element of a signal is added to its local neighbors after being weighted a the convolution kernel. For one-dimensional signals, the continuum model of convolution is given by

$$(k * f)(x) = \int_{-a}^{a} k(x')f(x - x')dx', \tag{3.1}$$

where the signal f is blurred by the convolution kernel k.

In practical computations, we can approximate the integral in equation (3.1) by numerical quadrature. The discretized version of the convolution model thus becomes

$$(\mathbf{k} * \mathbf{f})(x) = \sum_{i=-a}^{a} \mathbf{k}(i)\mathbf{f}(x - i),$$

where $\mathbf{k}$ is the discretized version of the convolution kernel and $\mathbf{f}$ is the discretized version of the signal. For 2D signals such as images, the discretized version is

$$(\mathbf{k} * \mathbf{f})(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} \mathbf{k}(i, j)\mathbf{f}(x - i, y - j), \tag{3.2}$$

where $\mathbf{k}$ is the 2D convolution kernel and $\mathbf{f}(x, y)$ is the original image. For example, blurring an image with a 3×3 Gaussian blur is implemented by choosing the convolution kernel as

$$\mathbf{k} = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

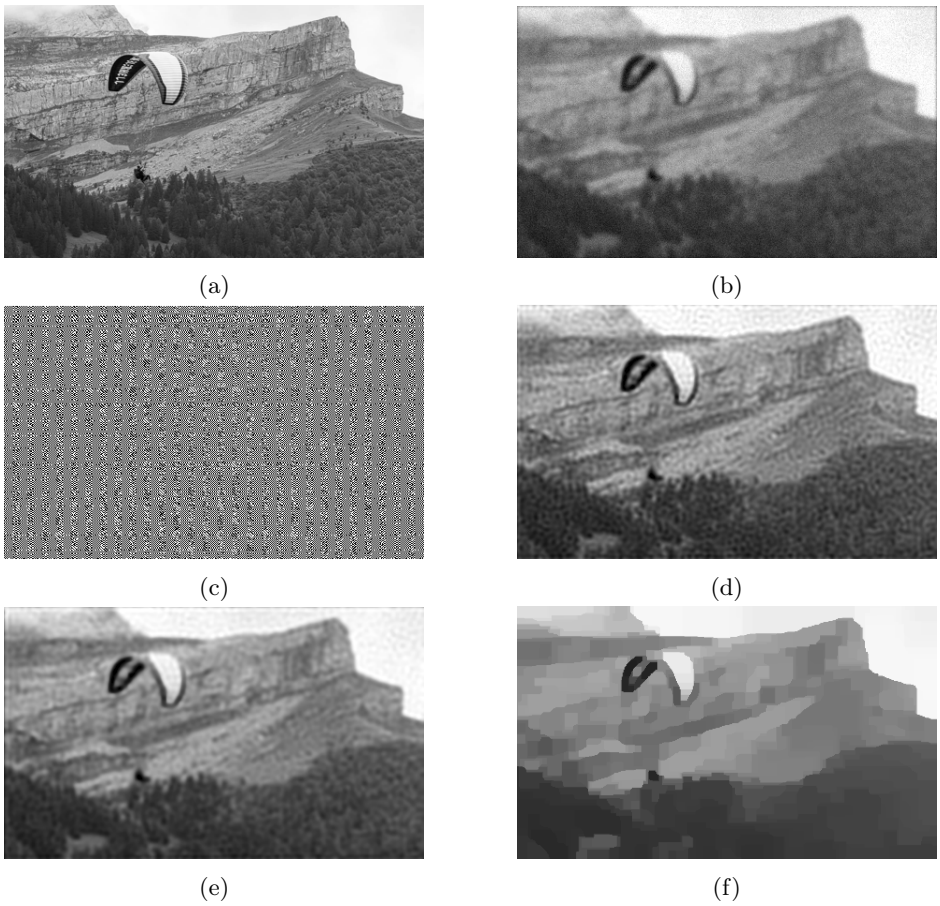


Figure 3.1: Convolving a grayscale image[4](a) with a 15×15 Gaussian blur kernel and adding 0.5% Gaussian noise to it, resulting in a blurry and noisy image (b). Attempting to deblur it using naive reconstruction fails (c), whereas with TV regularization, we are able to produce different kinds of deblurring results, depending on the choice of regularization parameter: (d) $\alpha = 0.01$, (e) $\alpha = 0.35$, (f) $\alpha = 0.15$. Choosing the best α value is a difficult problem, and even after computing the reconstruction, it is often subjective which choice is the best one. Here, it is clear to see that (f) is over-regularized, whereas the choice between (d) and (e) is not so clear.

[4] Original photograph taken by Markus Juvonen.

and performing the convolution operation with that. See Figure 3.1a and 3.1b for a visualization of the Gaussian blur example.

Deconvolution is the corresponding inverse problem of convolution: where convolving an image blurs it, deconvolving attempts to deblur the image, restoring it back to the original signal [63]. For the noise-free case, the deconvolution problem is easily solvable with division in the Fourier domain:

$$f = \mathcal{F}^{-1}(\mathcal{F}(k * f)/\mathcal{F}(k)),$$

where $\mathcal{F}$ is the Fourier transform and $\mathcal{F}^{-1}$ the inverse transform as defined below.

Definition 3.1.1 (The Fourier and inverse Fourier transform)**.** *The Fourier transform $\mathcal{F}$ takes a signal $f(x)$, $\forall x \in \mathbb{R}$ to the frequency domain:*

$$\widehat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi\xi x} dx, \tag{3.3}$$

whereas the inverse Fourier transform $\mathcal{F}^{-1}$ takes it back to the spatial domain:

$$f(x) = \int_{-\infty}^{\infty} \widehat{f}(\xi) e^{i2\pi\xi x} d\xi. \tag{3.4}$$

For the noisy case $\epsilon > 0$, deconvolution becomes an ill-posed inverse problem since it violates Hadamard's condition on uniqueness (condition II), as there can be multiple images that produce the same blurred result. Also, condition III continues to be a nuisance. We can model the inverse problem by writing

$$m = Af(x) + \epsilon = (k * f)(x) + \epsilon, \tag{3.5}$$

where the forward operator A is a circulant matrix having the elements of the convolution kernel k appearing on every row (as defined in equations 3.1 - 3.2). The measurement m is then defined as

$$m = \Delta x(k * f) + \epsilon, \quad \Delta x := x_2 - x_1.$$

As an ill-posed inverse problem, the naive approach for solving it fails completely, as seen in Figure 3.1c. We need to introduce regularization to solve the problem.

The problem of image deconvolution arises in many applications, in addition to standard photography deblurring [64] that is required after improper focusing or stabilization of the camera. It is a crucial step in astrophotography, where atmospheric turbulence and optical distortion convolve the signal [65]. An important deconvolution task arose with the Hubble Speace Telescope, after a mistake in the shape design of a mirror caused all the captured images to be blurry [66]. Also, certain image processing methods can introduce blurring as a trade-off. This can happen, for example, when removing noise or other numerical instability from a signal, as happens in publication II.

3.2 Total variation regularization

To combat measurement noise and ill-posedness, we can utilize variational regularization [67, 68, 69, 70, 71, 72, 73, 74] when solving the inverse problem. Regularization helps to combat these issues by making use of *a priori* information, so that the number of possible solutions to the problem can be limited. In the variational framework, we want to find a minimizer that combines prior information on the acquisition process in the data fidelity term with prior knowledge about the unknown target. Other approaches in tackling the problem of ill-posedness beyond variational regularization include, for example, algebraic methods [75] and Bayesian inversion [76, 77, 78, 79], not discussed here.

To consider regularized solutions to the inverse problem of image deconvolution in equation (3.5), let us first write it in a discrete form

$$\mathbf{m} = A\mathbf{f} + \epsilon.$$

We want to minimize:

$$\arg\min_{\mathbf{f}} \left\{ \frac{1}{2}\|A\mathbf{f} - \mathbf{m}\|_2^2 + \alpha R(\mathbf{f}) \right\},$$

where $R(\mathbf{f})$ is the regularization term, $\|A\mathbf{f} - \mathbf{m}\|_2^2$ is the data fidelity term, and $\alpha > 0$ is the regularization parameter. Here, the regularization term enforces prior information about the target solution and the regularization parameter balances between the *a priori* information and data fidelity term. Note that finding the optimal regularization parameter value α can be a tedious task.

In the case of total variation (TV) regularization [80], we have

$$T_{\mathrm{TV}}^{\alpha}(\mathbf{m}) = \arg\min_{\mathbf{f}} \left\{ \frac{\mathbf{1}}{\mathbf{2}}\|\mathbf{Af} - \mathbf{m}\|_{\mathbf{2}}^{\mathbf{2}} + \alpha \mathrm{TV}(\mathbf{f}) \right\},$$

where $\mathrm{TV}(f)$ is the total variation regularization term over the domain Ω, given by

$$\mathrm{TV}(f) = \int_{\Omega} |\nabla f(x)| dx,$$

assuming that f is differentiable. In the discrete setting, this becomes

$$\mathrm{TV}(\mathbf{f}) = \sup \sum_{\mathbf{i=1}}^{\mathbf{k}} |\mathbf{f}(\mathbf{x_i}) - \mathbf{f}(\mathbf{x_{i-1}})|.$$

The total variation regularized solution $T_{\mathrm{TV}}^{\alpha}(m)$ thus balances between having a small residual $AT_{\mathrm{TV}}^{\alpha}(m) - m$ and simultaneously having the regularization term $\mathrm{TV}(T_{\mathrm{TV}}^{\alpha}(m))$ be small in the ℓ_1-norm. As we minimize the total variation of the solution, the solution should have fewer oscillations. As a regularizer, total variation is edge-preserving, because it allows piece-wise constant solutions. This approach can, however, give the reconstruction a blocky texture which is not always desired. See Figure 3.1d - 3.1f for TV applied in the image deblurring problem, with different regularizer parameter values α.



Figure 3.2: A selection of learned filters of size 7×7 using the Fields-of-Experts model that was trained on the BSDS500 dataset [86] of natural images.

3.3 Fields-of-Experts regularization

The traditional non-learned approach of using variational models to solve inverse problems usually incorporates a number of free regularization parameters that are used as a trade-off between the regularization and data fidelity terms. The selection of the regularization parameter is not trivial, and it often requires testing over a large range of parameter options that can be computationally expensive. Especially if there are multiple parameters to optimize, this can quickly become infeasible.

An alternative regularization strategy is using the Fields-of-Experts (FoE) regularization [81] approach, that does not require hand-tuning the regularization parameter. Beyond basic variational regularization, it is a framework for *learning* generic and expressive image priors, inspired by sparse image coding of image patches [82, 83] and Markov random field priors [84, 85].

Now, we can cast the inverse problem in (3.5) as a minimization problem:

$$T_{\mathrm{FoE}}(m, \theta) = \arg\min_f \left\{ \frac{1}{2}\|Af - m\|_2^2 + \mathrm{FoE}(f, \theta) \right\}, \tag{3.6}$$

where the FoE regularization term is defined by a weighted sum of different convolution filters acting on f:

$$\mathrm{FoE}(f, \theta) = \sum_{\ell=1}^{L} \sum_{i=1}^{n} w_\ell \varphi((k_\ell * f)_i), \quad \varphi(x) = \log(1 + |x|^2),$$

where the weights $w_1, \dots, w_L$ and convolution kernels $k_1, \dots k_L$ of size $\kappa \times \kappa$ make up the parameter vector θ. Here, all parameters $\theta = (w, k)$ are learned from training data.

The learned filters k are able to capture important structural properties of images, such as edge and texture features at different scales and orientations. Despite that, their visual appearance lacks clear interpretability. See Figure 3.2 for examples of the learned filters k.

Compared to traditional regularization approaches such as TV, the advantage of using the FoE-model is that it requires no hand-choosing of regularization parameters. The result is a general image prior that can be applied across different imaging problems. The down-side is that training the model is computationally intensive. However, once trained, the learned filters can be applied quickly.

3.4 Bilevel learning of regularization parameters

In order to learn the weights and filters for Field-of-Experts regularization, we need an optimization strategy. Bilevel learning [87] is an optimization framework that bridges the gap between data-driven and model-driven methods, and is well-suited for estimating regularization models for ill-posed inverse problems, such as image denoising [88]. Bilevel learning casts the parameter selection task as an optimization problem: given a variational model, find the optimal parameters that minimize a loss function.

In bilevel optimization, there are a lower-level and an upper-level problem to be solved. There, the lower-level problem comes from the variational model and the upper-level problem is formulated in terms of a loss function Φ:

$$\begin{aligned} &\arg\min_{\theta} \frac{1}{s}\sum_{j=1}^{s} \Phi(f_j(\theta), g_j), \\ &\text{subject to} \quad f_j(\theta) \in \arg\min_{f_j} \left\{ \frac{1}{2}\|Af_j - m_j\|_2^2 + R(f_j, \theta) \right\}, \quad j = 1, \dots, s, \end{aligned} \tag{3.7}$$

where the variational model (as in equation (3.6) in the case of FoE regularization) acts as a constraint and θ are the unknown parameters for the regularization term we want to estimate. The loss function Φ penalizes errors between the solution of the lower level problem f and the ground truth g. The training set is composed of data pairs $\{(m_j, g_j)\}_{j=1}^{s}$, where m is the input data.

In contrast to deep learning, the bilevel learning approach does not use neural networks to learn the parameter values that are to be optimized. Instead, we use other types of optimization strategies, as the learnable parameters θ are already incorporated into the framework. Due to this, the size of the training set can be significantly smaller compared to working with neural networks, without the risk of overfitting.

If we choose, for example, MSE as the loss function $\Phi(\boldsymbol{f}_j(\boldsymbol{\theta}), \boldsymbol{g}_j)$, the bilevel problem in equation (3.7) in the context of discrete FoE regularization becomes:

$$\begin{aligned} &\min \left\{ \frac{1}{2s}\sum_{j=1}^{s} \|\boldsymbol{f}_j(\boldsymbol{\theta}) - \boldsymbol{g}_j\|_2^2 \right\} \\ &\text{s.t.} \quad \boldsymbol{f}_j(\boldsymbol{\theta}) \in \arg\min_{\boldsymbol{f}} \left\{ \frac{1}{2}\|\boldsymbol{A}\boldsymbol{f} - \boldsymbol{m}_{\boldsymbol{j}}\|_2^2 + \sum_{\ell=1}^{L}\sum_{i=1}^{n} \boldsymbol{w}_\ell \varphi((\boldsymbol{k}_\ell * \boldsymbol{f})_i) \right\}. \end{aligned}$$

To compute the gradients of the loss function with respect to $\boldsymbol{\theta}$, we can express the lower-level problem with its first-order optimality condition:

$$\nabla_f T_{\text{FoE}}(\theta, f_j) = \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{f}_j - \boldsymbol{m}_j) + \sum_{\ell=1}^{L}\sum_{i=1}^{n} \boldsymbol{w}_\ell \boldsymbol{K}_\ell^T \varphi'((\boldsymbol{K}_\ell \boldsymbol{f}_j)_i) = 0. \tag{3.8}$$

To find the optimal parameters $\boldsymbol{\theta} = (\boldsymbol{w}, \mathbf{k})$ solving the above bilevel problem,

one can utilize the Lagrange multipliers associated to equation (3.8) by iterating:

$$\begin{cases} \mathbf{f}_j^{(k+1)} \in \arg\min_{\mathbf{f}} \left\{ \frac{1}{2}\|\mathbf{A}\mathbf{f} - \mathbf{m}_j\|_2^2 + \sum_{\ell=1}^{L}\sum_{i=1}^{n} \boldsymbol{w}_\ell^{(k)} \varphi((\mathbf{K}_\ell^{(k)}\mathbf{f}_j)_i) \right\}, \\ \mathbf{p}_j^{(k+1)} = \left(\sum_{\ell=1}^{L}\sum_{i=1}^{n} \boldsymbol{w}_\ell^{(k)} (\mathbf{K}_\ell^{(k)})^T \mathbf{K}_\ell^{(k)} \varphi''((\mathbf{K}_\ell^{(k)}\mathbf{f}_j^{(k+1)})_i) + \mathbf{A}^T\mathbf{A} \right)^{-1} (\mathbf{g}_j - \mathbf{f}_j^{(k+1)}), \\ \boldsymbol{w}^{(k+1)} = P_{C_{\boldsymbol{w}}} \left(\boldsymbol{w}^{(k)} - \eta \frac{1}{s} \nabla_{\boldsymbol{w}} \Phi(\mathbf{f}_j^{(k+1)}, \mathbf{g}_j) \right), \\ \mathbf{k}^{(k+1)} = P_{C_{\mathbf{k}}} \left(\mathbf{k}^{(k)} - \eta \frac{1}{s} \nabla_{\mathbf{k}} \Phi(\mathbf{f}_j^{(k+1)}, \mathbf{g}_j) \right). \end{cases} \tag{3.9}$$

Above, $\boldsymbol{K}_\ell$ is the matrix implementing the convolution operation with filter $\boldsymbol{k}_\ell$, and we have certain projection operations $P_{C_{\boldsymbol{w}}}$, $P_{C_{\mathbf{k}}}$ that are performed on constrained sets accounting for Lagrange multipliers not explicitly included in the optimization. The Lagrange multiplier $\boldsymbol{p}_j$ is needed in order to compute the gradient components of the upper-level functional with respect to $\boldsymbol{\theta} = \boldsymbol{w}, \boldsymbol{k}$.

A gradient method can be used to solve for $\mathbf{f}_j^{(k+1)}$ in the outer iteration of (3.9) computationally. For computing $\mathbf{p}_j^{(k+1)}$, one can use the conjugate-gradient method [89]. The updates of $\boldsymbol{w}$ and $\boldsymbol{k}$ represent one step of projected gradient-descent with step-size η. A computationally efficient way for solving this is, for example, the gradient-descent minimization method with Barzilai-Borwein step-length updating strategy [90], that has both low memory requirements and low computational cost per iteration.

Chapter 4

X-ray computed tomography

4.1 Mathematical formulation

X-ray computed tomography (CT) is an imaging method that reveals the internal structure of an object by reconstructing a tomographic slice image from several X-ray projection images [91, 92]. The aim is to recover the attenuation coefficients inside the target from a collection of projection images (radiographs) that are recorded from different directions of projection. With a reconstruction algorithm, these projection images can be reconstructed into a cross-sectional tomographic slice image.

The invention of CT revolutionized medical imaging in the 1970s, and pioneers Allen M. Cormack and Godfrey N. Hounsfield [93] were awarded the Nobel Prize in medicine its development in 1979 [94]. Nowadays, CT scanning is widely used in medical imaging but also appears at airport security for luggage scanning and in nondestructive industrial testing [95].

The most simple imaging geometry is 2D parallel-beam geometry, which was used in first-generation CT scanners. There, we assume that parallel X-ray beams are shot from an X-ray source.[5] As the beams travel through the target, they get attenuated depending on the material they pass through. This is modeled with a non-negative attenuation coefficient function $f(x)$, which value depends on the relative intensity loss of the X-ray within a distance dx:

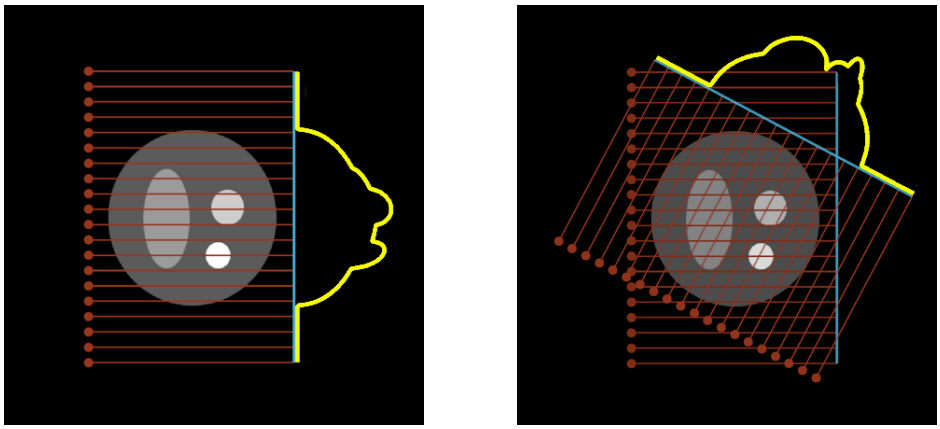
$$\frac{dI(x)}{I(x)} = -f(x)dx,$$

where $I(x)$ is the intensity of the X-ray radiation at point x. Based on this, the attenuation $f(x)$ follows an exponential decay:

$$I(x) = I_0 e^{-\int_0^x f(t)dt},$$

known as the Beer-Lambert law of attenuation [91]. Here, the initial intensity I_0 is known through calibration. The resulting attenuation $I_1 = I(1)$ is recorded by

[5] A more practical 2D imaging geometry is the fan-beam geometry, that assumes that the X-ray source is a point source, from which the X-ray beams "fan out".



(a) (b)

Figure 4.1: (a) In the parallel-beam imaging geometry, parallel X-ray beams (in red) travel through the target and their resulting attenuation is captured by the detector as a line integral (yellow). (b) These measurements are repeated over multiple angles.

the detector, so for $x = 1$, based on the previous equation, we have:

$$\int_0^1 f(t)dt + \epsilon = \ln(I_0) - \ln(I_1),$$

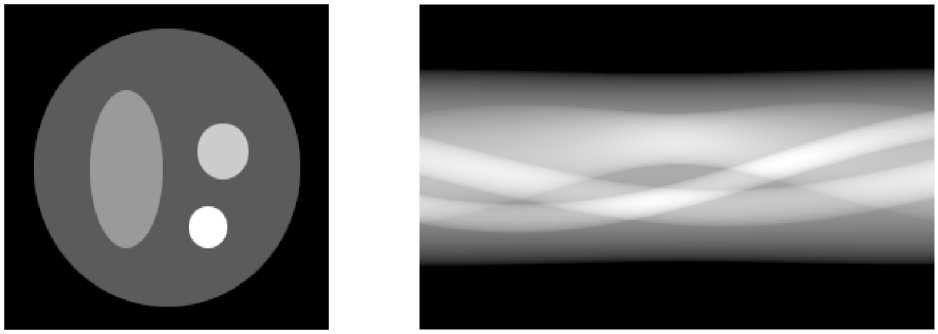
where ϵ models measurement error. There, after logarithmic transformation, the total amount of X-ray attenuation is revealed by integrating along the X-ray (from source to detector). See Figure 4.1 for an illustration of the parallel-beam imaging geometry, where the resulting line integral depicting total attenuation is visualized in yellow.

CT in medical imaging is based on different tissues having different attenuation coefficient values. The Hounsfield unit (HU) scale is a standard measurement unit that is a linear transformation of the attenuation coefficient measurement $\mu = f(x)$, where 0 HU corresponds to water and -1000 HU to air. Thus, we can compute the HU value for any linear attenuation coefficient μ by:

$$\text{HU} = 1000 \times \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}}.$$

For example, bone ($500 \leq \mu < 3400$ HU) attenuates X-rays heavier than soft tissues ($-500 \leq \mu < 500$ HU), or air ($\mu < -500$ HU). When reconstructing a tomographic image, we want to solve for these attenuation coefficient values from the measurements.

X-ray projection measurements are repeated for different directions ranging from 0 to 180 degrees, by rotating either the target object or the source-detector pair around it. The measurements from all angles are collected as columns in a



(a) (b)

Figure 4.2: (a) Computational target known as the phantom. (b) The corresponding sinogram obtained computationally by taking the Radon transform.

2D sinogram (Figure 4.2), where each measurement direction corresponds to one column in the sinogram. Mathematically, this process is described by the Radon transform.

Definition 4.1.1 (Radon transform [96])**.** *The Radon transform of a continuous function $f(x)$ is defined as*

$$\mathcal{R}f(s, \overrightarrow{\theta}) = \int_{x \cdot \overrightarrow{\theta} = s} f(x) dx^{\perp}, \tag{4.1}$$

where $dx^{\perp}$ denotes the 1D Lebesgue measure along the line s and $\overrightarrow{\theta} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$ is the unit vector with angle θ in radians.

Thus, what we have in the sinogram, is the Radon transform of the attenuation coefficient f, based on the initial and resulting intensities of the X-ray beams:

$$\mathcal{R}f(s, \overrightarrow{\theta}) = \ln\Big(\frac{I_0}{I_1}\Big).$$

There is a connection between the Radon transform (4.1) and the Fourier transform, described by the central slice theorem below.

Theorem 4.1.2 (Central slice theorem [97])**.** *If f is an absolutely integrable function defined on $\mathbb{R}$, then for a real number r and unit vector $\overrightarrow{\theta}$, the following identity holds:*

$$\int_{-\infty}^{\infty} \mathcal{R}f(s, \overrightarrow{\theta}) e^{-isr} ds = \widehat{f}(r\overrightarrow{\theta}),$$

where the left-hand side can be denoted as $\widetilde{\mathcal{R}f}(r, \overrightarrow{\theta})$, and on the right-hand side, we have the Fourier transform $\widehat{f}(\xi) = \frac{1}{(2\pi)^n} \int_{\mathbb{R}} f(x) e^{-ix\xi} dx$.

In X-ray tomography, the most widely-used reconstruction algorithm is the filtered backprojection (FBP) algorithm [97], that is based on the inverse Radon transform. With the Radon inversion formula, one can obtain a function f from its Radon transform by:

$$f(x) = \frac{1}{(2\pi)^2} \int_0^{\pi} \int_{-\infty}^{\infty} e^{isx\cdot\overrightarrow{\theta}} \widetilde{\mathcal{R}f}(s, \overrightarrow{\theta})|s| ds d\overrightarrow{\theta}, \tag{4.2}$$

where by the central slice theorem $\widetilde{\mathcal{R}f}(s, \overrightarrow{\theta}) = \widehat{f}(s\overrightarrow{\theta})$, in the ideal case where $\widehat{f}$ is absolutely integrable. The first step of the FBP algorithm is to back-project the sinogram into the spatial image domain. This produces a blurred reconstruction that is sharpened by performing high-pass filtering in the Fourier domain. The filter comes from the radial integral in the Radon inversion formula (4.2), and the output of the filter is defined as

$$\mathcal{GR}f(t, \overrightarrow{\theta}) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \widetilde{\mathcal{R}f}(r, \overrightarrow{\theta}) e^{irt}|r| dr.$$

With $|r|$, the filter suppresses low-frequency components and amplifies high-frequency components. Then, let $t = x \cdot \overrightarrow{\theta}$, and the final reconstruction is computed from:

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\pi} \mathcal{GR}f(x \cdot \overrightarrow{\theta}, \overrightarrow{\theta}) d\theta.$$

The FBP algorithm assumes densely sampled measurements between angles $[0, 180]$. If we have such measurement data, the algorithm produces high-quality reconstructions at a low computational cost. However, if the assumptions are violated, the reconstruction quality heavily deteriorates (see Figure 4.3a versus 4.3c). For such cases, we need to solve the ill-posed inverse problem of CT reconstruction using variational regularization.

4.2 Limited-angle tomography

In limited-angle tomography, we have X-ray measurements sampled only from a restricted opening angle, and not from the full range between angles $[0, 180]$. This restriction can be caused either by the scanner or the target being imaged itself. An example of limited-angle tomography is digital breast tomosynthesis, where the opening angle is only 40 degrees wide [98, 99, 100]. Other applications of limited-angle tomography include, for example, phase microscopy [101], and weld and underwater pipeline inspection [102, 103, 104].

Limiting the number of possible angles causes the inverse problem to become highly ill-posed [105, 97, 106]. Typically, limited-angle tomography reconstructions can recover only the parts of the feature boundaries that are tangented by X-rays, while other boundaries stretch out. This leads to the reconstructions generally suffering from a severe stretching of features along the central direction of projections, merging separate objects together (see Figure 4.3c). This leads to poor separation between slices perpendicular to the central direction.

The geometric structure of the ill-posedness concerning limited-angle tomography can be explained with microlocal analysis [107, 108]. Microlocal analysis provides tools to study which singularities of an image can be reconstructed in a stable manner from the limited data measurements. In practice, the singularities of an image lie in the edges of features, where there are jumps in the signal.

The wavefront set $WF(f)(x_0, \theta_0)$ of a signal $f \in L^2_{\text{loc}}(\mathbb{R}^2)$ describes the location x_0 and the direction θ_0 of singularities. The Fourier transform is a useful tool for studying the global information about a signal. If there are high frequency components in the frequency domain ($\widehat{f}(\xi) \neq 0$ for large $|\xi|$), it indicates that the signal has notable changes that occur quickly, and thus, singularities. More precisely, the signal f has a singularity in location x_0 if the localized Fourier transform of ϕf does not decay rapidly for all cutoff functions $\phi \in C_c^\infty$, $\phi(x_0) \neq 0$. The Fourier transform of ϕf decays rapidly if

$$\widehat{\phi f}(\xi) = \mathcal{O}(|\xi|^{-N}) \qquad \text{for } |\xi| \to \infty,\ N > 0.$$

To define the orientation of the singularities, we look for directions along which the localized Fourier transform $\widehat{\phi f}$ does not decay rapidly in any polar wedge $W_\delta = \{(r, w) : |w - \theta_0| < \delta\}$, where (r, w) are the polar coordinates in the frequency domain. The direction θ_0 of a singularity can be interpreted as the direction of maximum change of f at x_0. The set of all singularities of f is called the *singular support*.

In limited-angle tomography, a singularity (x_0, θ_0) of f is visible if the line $L(\theta_0, x_0 \cdot \theta_0)$ is in the measurements [109, 107, 108, 70]. Let's consider the continuous two-dimensional Radon transform $\mathcal{R}f$ of a function $f \in L^1(\mathbb{R}^2)$:

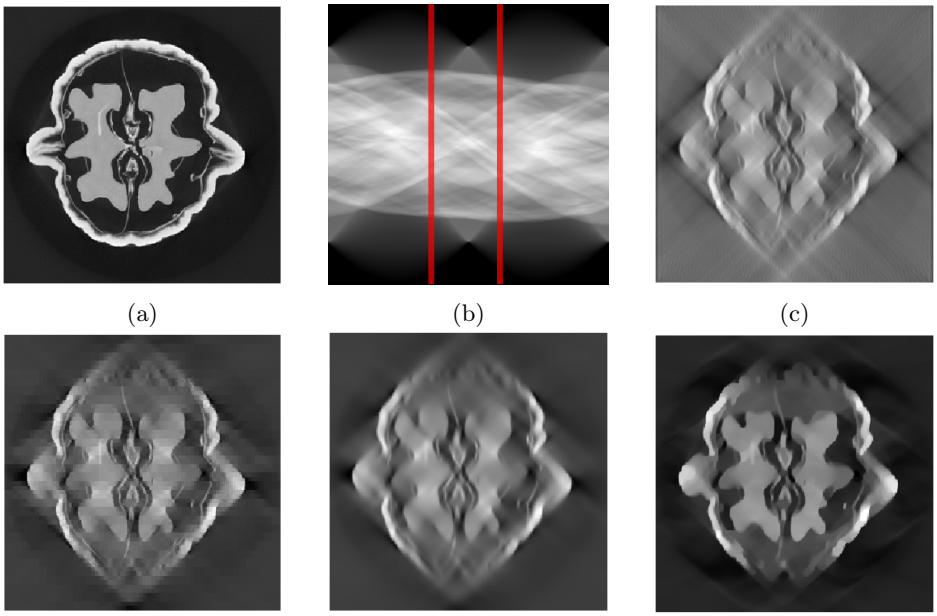
$$m(\theta, s) = \mathcal{R}f(\theta, s) = \int_{L(\theta,s)} f(x) dS(x),$$

where $L(\theta, s) = \{x \in \mathbb{R}^2 : x_1 \cos(\theta) + x_2 \sin(\theta) = s\}$ is the line with normal direction θ and signed distance from the orientation s. If we take the Radon transform data $\mathcal{R}f(\theta, s)$ arbitrarily near (θ_0, s_0), we can reconstruct singularities of f stably with location $x \in L(\theta_0, s_0)$ and direction θ_0.

In parallel-beam limited-angle tomography, we know the Radon transform $\mathcal{R}f(\theta, s)$ for a particular angular range $[\theta_0 - \delta, \theta_0 + \delta]$ for some central angle $\theta_0 \in \mathbb{R}$ and opening $0 < \delta < \pi$. The visible part of the wavefront set is $WF(f)(x_0, \alpha)$, for α ranging between

$$\pi/2 + \theta_0 - \delta \leq \alpha \leq \pi/2 + \theta_0 + \delta.$$

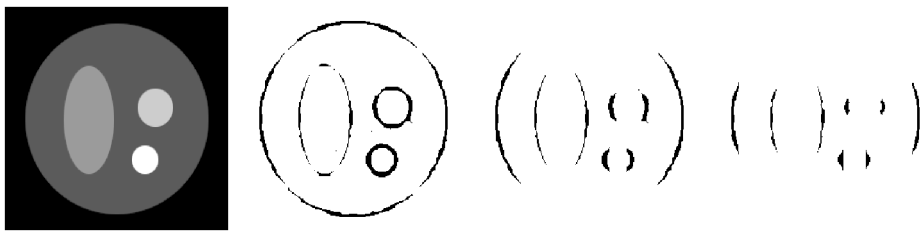
This means that certain elements of the wavefront set of the X-ray attenuation coefficient f are available, while information about the rest of the wavefront set is not present in a well-posed form. See Figure 4.4 for an example of the visible part of the wavefront set for limited-angle tomography with different opening angles. There, the implementation of the wavefront set is done by the dual-tree complex wavelet transform (see section 4.3.2).



(a) (b) (c)

(d) (e) (f)

Figure 4.3: A comparison of non-regularized and regularized reconstructions in limited-angle tomography. (a) Full-angle FBP reconstruction of a walnut of resolution 256×256, dataset from [110]. (b) Full 180-angle sinogram compared to a 45-degree limited-angle sinogram. (c) Limited-angle FBP reconstruction. (d) Limited-angle Haar wavelet -regularized reconstruction ($\alpha = 0.02$). (e) Limited-angle complex wavelet -regularized reconstruction ($\alpha = 0.03$). (f) Limited-angle total variation regularized reconstruction ($\alpha = 5$).



(a) (b) (c) (d)

Figure 4.4: A comparison of the visible singularities (wavefront set) of phantom (a) in the cases of (b) full-angle measurement and in limited-angle tomography for (c) a 90-degree opening angle ($\delta = \pi/4$) and (d) a 40-degree opening angle ($\delta = \pi/18$).

4.3 Wavelet-based regularization

For limited-angle tomography, having such incomplete data makes the inverse problem very ill-posed. The assumptions of FBP are violated and it performs poorly (Figure 4.3c). Thus, we need to use more mathematically sophisticated methods to combat the lack of measurement data: variational regularization. Let us begin with formulating the problem of computed tomography in the inverse problems setting:

$$m = Af + \epsilon,$$

where m is the sinogram, A is the forward operator matrix holding information about the imaging geometry, f is target, and ϵ is measurement noise. In the discrete setting, with target $\mathbf{f} \in \mathbb{R}^n$ based on X-ray measurements $\mathbf{m} \in \mathbb{R}^m$, the model becomes:

$$\mathbf{m} = A\mathbf{f} + \epsilon, \tag{4.3}$$

where $A \in \mathbb{R}^{m \times n}$ is the linear forward operator discretizing the Radon transform and $\epsilon \in \mathbb{R}^m$ models additive Gaussian noise.

In addition to Total Variation regularization explained in 3.2, there are also other regularization strategies suitable for limited-angle tomography reconstruction. One possibility is to use sparsity-based regularization with multidimensional representation systems, such as wavelets, with the ℓ^1-norm. There, the idea is to use regularization that promotes solutions having a sparse representation in the wavelet basis, inspired by compressed sensing [111]. Sparsity means that for a given basis, there are only relatively few nonzero coefficients and minimizing the ℓ^1-norm is a feasible way of promoting a sparse solution [112]. Sparsity-based ℓ^1-regularization for wavelet coefficients widely used in literature [113, 114, 115].

The wavelet coefficients correlate strongly with important features in the signal, such as singularities, and less so with noise. By introducing sparsity to the wavelet coefficients by keeping only the most significant ones encoding the important features, we can remove noise and other undesired features from a reconstruction.

With a sparsity-promoting wavelet-based regularization, the minimization problem in (4.3) becomes

$$\mathbf{f}_S = \underset{\mathbf{f} \geq 0}{\arg\min} \left\{ \frac{1}{2} \|A\mathbf{f} - \mathbf{m}\|_2^2 + \alpha \|W\mathbf{f}\|_1 \right\}, \tag{4.4}$$

where we take the ℓ^1-norm of the wavelet coefficients $W\mathbf{f}$ and α is the regularization parameter. Additionally, in CT we can assume a component-wise non-negativity constraint $\mathbf{f} \geq \mathbf{0}$, since we know that the linear attenuation coefficient $\mathbf{f}$ is non-negative by its nature.

In a similar manner to Equation 4.4, we can choose the sparsity-promoting ℓ_1-regularizer to use complex wavelet coefficients instead of real-valued wavelets:

$$\mathbf{f}_S = \underset{\mathbf{f} \geq 0}{\arg\min} \left\{ \frac{1}{2} \|A\mathbf{f} - \mathbf{m}\|_2^2 + \alpha \|W_{\mathbb{C}}\mathbf{f}\|_1 \right\}. \tag{4.5}$$

As a regularization strategy, real-valued Haar wavelets produce block-like shapes in the reconstruction, whereas the complex wavelets provide a more smooth texture. See Figure 4.3d and 4.3e for a comparison in the limited-angle tomography case. The next subsections will give an overview of real-valued Haar wavelets and complex wavelets.

4.3.1 Haar wavelet transform

Wavelets are functions that break up signals into shifted and scaled versions of them, making them a useful tool in signal processing [116, 117, 118]. The principles are similar to those of Fourier analysis, but the wavelet transform has advantages over Fourier methods for signals with discontinuities. Wavelets are localized and oscillatory functions, and as such, are sensitive to the shifts in the signal, whereas the Fourier coefficients are not. The most simple wavelets are square-shaped Haar wavelets [119], developed already in 1911. The general theory on multi-resolution analysis and wavelets, however, came much later than the design of Haar wavelets. For more information on multi-resolution analysis and the theoretical properties of wavelets in general, please see [120, 121, 122]. Next, we will give a short overview of Haar wavelets.

The main building blocks of wavelets are the scaling function ϕ and the wavelet function ψ, also known as the father and mother wavelet, respectively. The Haar scaling function $\phi(t)$ has value 1, if $0 \leq t < 1$, and 0 otherwise, and the wavelet function ψ is defined as

$$\psi(x) = \phi(2x) - \phi(2x - 1).$$

Haar wavelets can coarsely approximate any function $f \in L^2(\mathbb{R})$ by:

$$f(x) \approx \sum_{k\in\mathbb{Z}} a_k \phi(x - k) + \sum_{j=0}^{J} \sum_{k\in\mathbb{Z}} d_{j,k} 2^{\frac{j}{2}} \psi(2^j x - k),$$

where $j = 0, \ldots, J$ is a scaling parameter corresponding to the resolution. Here, $j = 0$ corresponds to the mother wavelet and $j = J$ corresponds to the finest scale. Above, a_k are the approximation and $d_{j,k}$ are the detail coefficients corresponding to the Haar scaling and wavelet functions:

$$a_k = \langle f, \phi_k \rangle,$$

$$d_{j,k} = \langle f, \psi_{j,k} \rangle.$$

In 2D, the wavelet and scaling functions become:

$$\psi^{LH}_{j,k}(x_1, x_2) = 2^j \phi(2^j x_1 - k_1)\psi(2^j x_2 - k_2),$$

$$\psi^{HL}_{j,k}(x_1, x_2) = 2^j \psi(2^j x_1 - k_1)\phi(2^j x_2 - k_2),$$

$$\psi^{HH}_{j,k}(x_1, x_2) = 2^j \psi(2^j x_1 - k_1)\psi(2^j x_2 - k_2),$$

$$\phi_{j,k}(x_1, x_2) = 2^j \phi(2^j x_1 - k_1)\phi(2^j x_2 - k_2),$$

for scale j. Here, H stands for high-pass and L for low-pass filter, that are related to the practical implementation of the discrete wavelet transform by convolution. For the wavelet functions, the notation LH, HL, and HH denotes the orientation of information they encode. Real-valued wavelets like Haar-wavelets are able to extract directional information with vertical $\psi_{j,k}^{LH}$, horizontal $\psi_{j,k}^{HL}$, and diagonal $\psi_{j,k}^{HH}$ wavelets. On the bottom, we have the 2D scaling function $\phi_{j,k}$.

In signal processing with wavelets, the general idea is to use the wavelet transform to decompose the signal to its approximation and detail coefficients (up to a desired scale j). Then, we can process the coefficients, for example with introducing sparsity by thresholding some of them to 0, before performing the inverse wavelet transform. There, we use the processed wavelet coefficients to reconstruct the signal. Afterwards, the signal (hopefully) enjoys some desired properties, like removed noise. Similar ideas are used for sparsity-based regularization strategies.

4.3.2 Complex wavelet transform

In addition to the real-valued Haar wavelets described above, we can also consider *complex*-valued wavelets. There, the advantage is that we can extract directional information from more than three orientations. Moreover, for real-valued inputs, complex wavelets also provide shift-invariance and directionality in 2D and higher. The dual-tree complex wavelet transform (DT-CWT) [123, 124], inspired by the Fourier transform, was introduced in the 1990s. In dual-tree CWT, the real and imaginary parts of the complex wavelets are actually the same function, just shifted out of phase by a certain amount. This makes it easier to capture features of the signal that might otherwise be lost.

In the construction of complex wavelets, the complex-valued wavelet function is defined as follows:

$$\psi_{\mathbb{C}}(x) = \psi(x) + i\mathcal{H}(\psi)(x),$$

where $\mathcal{H}$ is the Hilbert transform, defined with the Fourier transform as $\widehat{\mathcal{H}(f)}(\xi) = -\mathrm{sgn}(\xi)i\widehat{f}(\xi)$. The wavelets $\psi(x)$ used in DT-CWT are specific Q-shift wavelets, see [125] for more details.

For the 1D case, the dual-tree CWT can be implemented with two side-by-side real-valued wavelet transforms. There, one "tree" consists of the real part of the coefficients (for every scale j) and filters corresponding to the real parts of the scaling and wavelet functions, while the other tree consists of the corresponding imaginary parts. The wavelet and scaling functions themselves are complex-valued:

$$\psi_{\mathbb{C}}(x) = \psi_h(x) + i\psi_g(x),$$

$$\phi_{\mathbb{C}}(x) = \phi_h(x) + i\phi_g(x),$$

where $\psi_h(x)$ and $\phi_h(x)$ correspond to the real parts, and $\psi_g(x)$ and $\phi_g(x)$ to the imaginary parts. The complex wavelet function $\psi_{\mathbb{C}}(x)$ is associated with a high-pass filter H, and complex scaling function $\phi_{\mathbb{C}}(x)$ is associated with a low-pass filter L.



(a) Phantom (b) $\overline{L}H$ (c) $\overline{H}H$ (d) $\overline{H}L$ (e) HL (f) HH (g) LH

Figure 4.5: The finest scale of complex wavelet detail coefficients of a phantom shown in (a). The orientations of coefficient subbands are denoted below each subband. Here, we show the absolute value of the complex-valued detail coefficients.

In 2D, let us denote the complex wavelet detail coefficients by $d^{\nu}_{j,k} \in \mathbb{C}$, where the sub-index ν corresponds to the filter and thus tells us what directional information the coefficients hold:

$$\nu \in \mathcal{I} = \{\overline{L}H, \overline{H}H, \overline{H}L, HL, HH, LH\}.$$

These represent the indices of the six oriented so-called *subbands* for each scale $j = 0, \ldots, J$. The detail coefficients $d^{\nu}_{j,k}$ themselves are computed from:

$$d^{\nu}_{j,k} := \langle f, \psi^{\nu}_{\mathbb{C}_{j,k}} \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x_1, x_2) \psi^{\nu}_{\mathbb{C}_{j,k}}(x_1, x_2)\, dx_1 dx_2. \tag{4.6}$$

See Figure 4.5 for a visualization of the absolute values of each detail coefficient subband of the finest scale.

Above in equation (4.6), the six coefficient subbands $\psi^{\nu}_{\mathbb{C}_{j,k}}(x_1, x_2)$ are computed as follows:

$$\begin{aligned}
\psi^{LH}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \phi_{\mathbb{C}}(2^j x_1 - k_1)\psi_{\mathbb{C}}(2^j x_2 - k_2) \\
\psi^{HL}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \psi_{\mathbb{C}}(2^j x_1 - k_1)\phi_{\mathbb{C}}(2^j x_2 - k_2) \\
\psi^{HH}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \psi_{\mathbb{C}}(2^j x_1 - k_1)\psi_{\mathbb{C}}(2^j x_2 - k_2) \\
\psi^{\overline{H}H}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \overline{\psi_{\mathbb{C}}(2^j x_1 - k_1)}\psi_{\mathbb{C}}(2^j x_2 - k_2) \\
\psi^{\overline{L}H}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \overline{\phi_{\mathbb{C}}(2^j x_1 - k_1)}\psi_{\mathbb{C}}(2^j x_2 - k_2) \\
\psi^{\overline{H}L}_{\mathbb{C}_{j,k}}(x_1, x_2) &:= \overline{\psi_{\mathbb{C}}(2^j x_1 - k_1)}\phi_{\mathbb{C}}(2^j x_2 - k_2),
\end{aligned}$$

for each scale j. For example, the complex wavelet in subband HH (oriented at $-45°$) is computed from:

$$\begin{aligned}
\psi_{\mathbb{C}}(x_1, x_2) &= [\psi_h(x_1) + i\psi_g(x_2)][\psi_h(x_1) + i\psi_g(x_2)] \\
&= \psi_h(x_1)\psi_h(x_2) - \psi_g(x_1)\psi_g(x_2) + i[\psi_g(x_1)\psi_h(x_2) + \psi_h(x_1)\psi_g(x_2)].
\end{aligned}$$

Similarly, to obtain the wavelet in subband $\overline{H}H$ (oriented at $+45°$), we have:

$$\begin{aligned}
\psi(x_1, x_2) &= \overline{\psi_{\mathbb{C}}(x_1)}\psi_{\mathbb{C}}(x_2) \\
&= [\psi_h(x_1) + i\psi_g(x_1)][\psi_h(x_2) + i\psi_g(x_2)] \\
&= \psi_h(x_1)\psi_h(x_2) + \psi_g(x_1)\psi_g(x_2) + i[\psi_h(x_1)\psi_g(x_2) - \psi_g(x_1)\psi_h(x_2)].
\end{aligned}$$

The approximation coefficients (final-level low-pass scaling coefficients) are given by:

$$a_k = \langle f, \phi_{\mathbb{C}_k} \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x_1, x_2) \phi_{\mathbb{C}}(x_1 - k_1) \phi_{\mathbb{C}}(x_2 - k_2)\, dx_1 dx_2.$$

Finally, the function f can be recovered from the coefficients by

$$f(x_1, x_2) = \sum_{\nu \in J} \sum_{j=1}^{J} \sum_{k_1=0}^{2^j-1} \sum_{k_2=0}^{2^j-1} d_{j,k}^{\nu} S^{-1} \psi_{\mathbb{C}_{j,k}}^{\nu}(x_1, x_2),$$

where S is the frame operator

$$S = T^* T,$$

with an analysis operator T and synthesis operator T^* corresponding to S. See [124] for more details on this.

In addition to their potential with ℓ_1-regularization, complex wavelets can also be used as a computational tool to extract microlocal information about signals. Whereas real-valued wavelets offer only very limited information about the orientation of the singularities in an image, complex wavelets are able to provide directional information from six directions. Compared to more advanced shearlets [126, 127], they provide a good compromise between directionality and computational cost.

4.4 Primal-dual fixed-point algorithm

In order to solve the minimization problem in equation (4.4) or (4.5), an optimization algorithm is needed. There are multiple choices to choose from, but in this thesis, we have used the Primal-dual fixed-point (PDFP) method introduced in [128]. It is a suitable choice for wavelet-based regularization and it can be modified to include a non-negativity constraint.

One of the main difficulties in solving those minimization problems is that while both the data fidelity term and the regularization term $\alpha \|W\mathbf{f}\|_1$ are convex, the latter is non-differentiable. This is why we need to introduce the proximity operator $\text{prox}_g(x)$:

$$x \to \arg\min_y g(y) + \frac{1}{2}||x - y||_2^2.$$

For sparse regularization where $g = ||\cdot||_1$, the proximity operator can be written as the component-wise soft-thresholding operation denoted below with $\mathcal{T}$.

Thus, to solve for $\mathbf{f}$, the PDFP algorithm with a non-negativity constraint [129] iterates over:

$$\begin{cases} \mathbf{y}^{(k+1)} &= \mathbb{P}_C(\mathbf{f}^{(k)} - \tau \nabla \mathcal{G}(\mathbf{f}^{(k)}) - \lambda W^T \mathbf{v}^{(k)}), \\ \mathbf{v}^{(k+1)} &= (I - \mathcal{T}_\alpha)(W\mathbf{y}^{(k+1)} + \mathbf{v}^{(k)}), \\ \mathbf{f}^{(k+1)} &= \mathbb{P}_C(\mathbf{f}^{(k)} - \tau \nabla \mathcal{G}(\mathbf{f}^{(k)}) - \lambda (W^T \mathbf{v}^{(k+1)}), \end{cases} \tag{4.7}$$

where $\mathcal{G}(\mathbf{f}) = \frac{1}{2}\|A\mathbf{f} - \mathbf{m}\|_2^2$ is the data fidelity term, $\alpha > 0$ is the regularization parameter, and $\tau > 0$ and $\lambda > 0$ are other parameters used in the optimization. To be more precise, parameter λ can range between

$$0 < \lambda < \frac{1}{\lambda_{\max}(W(W)^T)},$$

where $\lambda_{\max}$ is the maximum eigenvalue. For parameter τ, we have

$$0 < \tau < 2/\tau_{\text{lip}},$$

where τ_{lip} is the Lipschitz constant for $\mathcal{G}(\mathbf{f})$. Next, operator $\mathcal{T}$ is the soft-thresholding operator defined radially for complex-valued inputs as

$$\mathcal{T}_\alpha(\theta) = (|\theta| - \alpha)e^{i\theta}.$$

Lastly, operator $\mathbb{P}_C$ is the Euclidean projection, where the non-negative quadrant is denoted by $C = \mathbb{R}_+^{N^2}$. In practice, the operator $\mathbb{P}_C$ simply replaces any non-negative elements in the input vector by zeroes.

Chapter 5

Electrical impedance tomography

5.1 Mathematical formulation

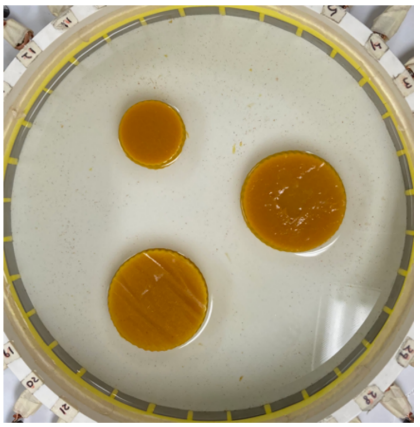
Electrical impedance tomography (EIT) aims to reconstruct the internal structure of an object, based on boundary measurements of electric current flows. In EIT, the measurement process involves injecting electrical current into the target and measuring the resulting voltages through the same electrodes. The aim is to reconstruct the internal electrical conductivity, and thus it is suitable for imaging targets that have considerably different conductivity values. As a non-invasive imaging method, it can be used in medical imaging, especially in the imaging of the lungs (pulmonary imaging) [130], the brain (stroke detection) [131, 132, 133], and the breast (tumor detection) [134]. Many organs and tissues have contrasting conductivities, enabling EIT to reveal information about medical conditions and diseases. In clinical use, it emerged in the late 1980s [135]. EIT can also be used in nondestructive testing in industrial environments and in geophysical applications, see [136, 137] for a comprehensive review of each.

To measure EIT data, electrodes are placed on the surface of the object of interest and low-frequency current is applied on the electrodes. This leads to a voltage distribution on the electrodes that can be measured. The measurement is usually repeated multiple times for certain current patterns, resulting in a current-to-voltage map, which can then be transformed to a voltage-to-current map:

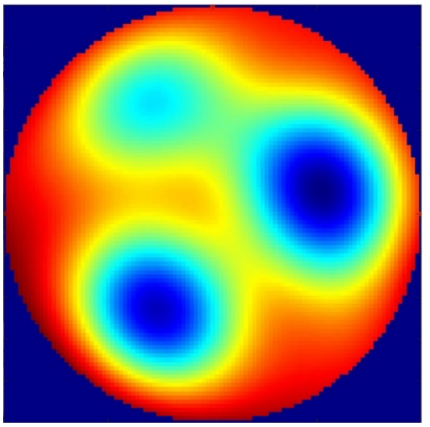
$$\Lambda_\sigma : f \mapsto \sigma \frac{\partial u}{\partial \vec{n}}|_{\partial\Omega},$$

known as the Dirichlet-to-Neumann (DN) map, which models the infinite-precision voltage and current measurements. It is a linear operator taking the given voltage distribution on the boundary to the current-density distribution, and is bounded between the Sobolev spaces:

$$\Lambda_\sigma : H^{1/2}(\partial\Omega) \to H^{-1/2}(\partial\Omega),$$



(a)



(b)

Figure 5.1: (a) Target phantom consisting of three conductive agar inclusions in a tank filled with homogeneous saline. The measurements were performed with the ACT5 EIT machine at Colorado State University [138]. (b) D-bar reconstruction of the conductivity, computation performed by Melody Alsaker.

if the conductivity $\sigma \in L^\infty(\Omega)$ is strictly positive almost everywhere in the domain Ω. The DN map Λ_σ acts as the input for the inverse conductivity problem of EIT.

Thus, we can formulate EIT as injecting voltages and measuring currents, as opposed to the other way around. When we apply a voltage distribution f on the boundary of the domain Ω, it corresponds to the Dirichlet boundary condition

$$u(z) = f(z), z \in \partial\Omega.$$

We can measure the current-density distribution J on the boundary:

$$J(z) = \sigma(z)\frac{\partial u}{\partial v}(z), z \in \partial\Omega,$$

where v is the outward unit normal on the boundary. Due to the conservation of charge (Kirchhoff's law), the current density is conserved over the boundary. After using Green's formula, we have:

$$\int_{\partial\Omega} J ds = \int_{\partial\Omega} \sigma \frac{\partial u}{\partial v} ds = \int_{\partial\Omega} \nabla \cdot \sigma \nabla u dz = 0.$$

Thus, applying voltage f at the boundary $\partial\Omega$ leads to the elliptic partial differential equation (PDE)

$$\begin{cases} \nabla \cdot \sigma \nabla u &= 0 \text{ in } \Omega, \\ u|_{\partial\Omega} &= f, \end{cases} \tag{5.1}$$

where u is the electrical current.

Recovering the inner unknown conductivity distribution σ from given electric boundary measurements on the surface $\Omega \in \mathbb{R}^2$ in equation (5.1) is a severely ill-posed nonlinear inverse problem. The nonlinearity stems from the forward map $\sigma \to \Lambda_\sigma$ being nonlinear, due to the fact that the target we are imaging affects the travel paths of the measuring energy, as electrical current flows mostly through where the conductivity is highest. The ill-posedness arises from the facts that the solution fails to depend continuously on data, the discrete measurements are of finite precision, and usually only partial data is available. This leads to the fact that two very different targets can produce almost exact measurements (Hadamard's condition III).

While the problem of EIT reconstruction is mathematically demanding, there are several benefits of using EIT as an imaging method. Electric currents can penetrate several materials without causing any damage or harming them, and the contrast resolution of EIT reconstructions is good. EIT equipment is cost-effective compared to other medical imaging apparatus, and the method has potential for real-time applications. However, there are also some downsides to it: mainly the fact that EIT is extremely sensitive to modeling errors and measurement noise, requiring proper regularization techniques that can be computationally costly. Additionally, to use optimal current patterns, one must be able to both inject and measure currents from the same electrodes. The reconstructions have, by nature, low spatial resolution, which can cause difficulties in interpretation. Also, an important restriction is that under finite measurements, objects that are smaller than a certain size cannot be distinguished [139].

The inverse conductivity problem from the complete knowledge of the DN map is known as the *Calderón problem*, after mathematician Alberto Calderón who studied the problem extensively [140]. In 1980, he posed two fundamental questions:

- Is it possible to uniquely determine the electrical conductivity of an unknown object by making voltage and current measurements on its boundary?
- How to reconstruct this unknown conductivity?

In his work, Calderón proved that the answer to both questions is yes, in the case of a linearized problem, and he also proposed a linearized reconstruction method. His work has inspired research on the inverse conductivity problem, leading to practical EIT algorithms and results in using complex geometrical optics solutions to study the aforementioned questions.

5.2 Complex geometrical optics solutions

One way of approaching EIT reconstruction is based on exponentially behaving complex geometrical optics (CGO) solutions. This approach leads to the theoretical insight necessary for virtual hybrid parallel-beam tomography (VHPT) imaging (publication II). CGO solutions have exponential growth in certain di-

rections while also decaying exponentially in others. They are an important technique in both uniqueness proofs and in computational algorithms.

Before moving further, let us define an important operator for computing derivatives with respect to a complex variable z.

Definition 5.2.1 (The ∂_z and $\overline{\partial}_z$ operator)**.** *For a complex variable $z = x + iy$, we define*

$$\partial_z = \frac{1}{2}\left(\frac{\partial}{\partial x} - i\frac{\partial}{\partial y}\right), \quad \textit{and} \quad \overline{\partial}_z = \frac{1}{2}\left(\frac{\partial}{\partial x} + i\frac{\partial}{\partial y}\right).$$

CGO solutions were first introduced in 1966 by Faddeev in the context of the Schrödinger's equation [141], and later introduced to inverse problems in 1987 for the three-dimensional EIT problem [142]. Later in 1996, a study of the 2D case [143] made a CGO-based EIT algorithm called the D-bar method possible [144, 145, 146, 3, 147]. There, the exponential behavior of CGO solutions is used to define a nonlinear Fourier transform, and image reconstruction is made robust against noise by applying a low-pass filter in the nonlinear frequency domain, with the cutoff frequency depending on the signal-to-noise ratio. The D-bar method is demonstrated to work well with experimental data [148]. See Figure 5.1 for a D-bar reconstruction of an experimental phantom.

Astala and Päivärinta transformed the construction of the CGO solutions in 2D by reducing the conductivity equation to a Beltrami equation [149]. Following their work, let us assume a strictly positive conductivity $\sigma \in L^\infty(\Omega)$ and define a function $\mu : \mathbb{R}^2 \to (-1, 1)$, known as the Beltrami coefficient:

$$\mu = \frac{1 - \sigma(z)}{1 + \sigma(z),}$$

where $z = x_1 + ix_2$. Next, let us consider the Beltrami equation:[6]

$$\overline{\partial}_z f_\mu = \mu \overline{\partial_z f_\mu}. \tag{5.2}$$

The CGO solutions $f_\mu = f_\mu(z, k)$ of the Beltrami equation can be written in the form

$$f_\mu(z, k) = e^{ikz}(1 + \omega(z, k)), \tag{5.3}$$

where $ikz = ik(x_1 + x_2)$ and the functions ω have the asymptotics

$$\omega(z, k) = \mathcal{O}(z^{-1}) \quad \text{as } |z| \to \infty.$$

5.2.2 Computation from the forward problem

The existence and uniqueness of the Beltrami CGO solutions $f_\mu(z, k) = e^{ikz}(1 + \omega(z, k))$ was proved in [149]. A numerical computation of the trace of the CGO solution ω was first introduced in [150], based on the original construction in [149]. In 2012, Huhtanen and Perämäki introduced an efficient method to compute the

[6] Note that the Beltrami equation (5.2) is not complex-linear but only real-linear because the solution f_μ is complex-conjugated on the right-hand side.

CGO solutions f_μ in the forward problem [151]. From [149], we know that ω satisfies the following equation:

$$\overline{\partial}\omega - v\overline{\partial\omega} - \alpha\overline{\omega} - \alpha = 0, \tag{5.4}$$

where the derivatives are taken with respect to z, and we have

$$\begin{aligned}\alpha(z,k) &= -i\overline{k}e_{-k}(z)\mu(z),\\ v(z,k) &= e_{-k}(z)\mu(z),\end{aligned}$$

where $e_k(z) = e^{i(kz+\overline{k}\overline{z})}$.[7]

Next, let us define $u \in L^p(\Omega)$ for which $\overline{u} = -\overline{\partial}\omega$. Now, we have $\omega = -P\overline{u}$ and $\partial\omega = -S\overline{u}$, with the solid Cauchy transform $P = \overline{\partial}^{-1}$ and the Beurling transform $S = \partial\overline{\partial}^{-1}$, that are defined as follows:

$$Pf(z) = -\frac{1}{\pi}\int_{\mathbb{C}} \frac{f(\lambda)}{\lambda - z} d\lambda_1 d\lambda_2, \tag{5.5}$$

$$Sg(z) = -\frac{1}{\pi}\int_{\mathbb{C}} \frac{g(\lambda)}{(\lambda - z)^2} d\lambda_1 d\lambda_2, \tag{5.6}$$

where points $(\lambda_1, \lambda_2) \in \mathbb{R}^2$ and $\lambda_1 + i\lambda_2 \in \mathbb{C}$. We can now rewrite equation (5.4) with the newly defined u as the real-linear integral equation

$$\begin{aligned} -\overline{u} - v\overline{(-S\overline{u})} - \alpha\overline{(-P\overline{u})} &= \alpha\\ \Leftrightarrow \quad u + (-\overline{v}S - \overline{\alpha}P)\overline{u} &= -\overline{\alpha}\\ \Leftrightarrow \quad (I + A\rho)u &= -\overline{\alpha}, \end{aligned} \tag{5.7}$$

where we have $A := (-\overline{v}S - \overline{\alpha}P)$ and ρ denotes the complex conjugation operation as $\overline{f} = \rho(f)$. It is proven in [151] that $I + A$ is invertible in $L^p(\Omega)$, and that we can transform the real-linear equation (5.7) into the complex-linear equation

$$(I - A\overline{A})v = \overline{\alpha} \qquad \text{in } L^p(\Omega). \tag{5.8}$$

This allows us to use the iterative Generalized minimal residual (GMRES) method [152] to computationally solve for v in equation (5.8). Then, once we know v, we can write the solution u of equation (5.7) as

$$u = (I - A\rho)v.$$

Finally, we get ω from

$$\omega = -P\overline{u}.$$

This is the trace of the Beltrami CGO solution on the boundary, computed from the known conductivity.

[7] Note that $\overline{\partial}\omega$ is supported in Ω, since we can write equation (5.4) as $\overline{\partial}\omega = v\overline{\partial\omega} + \alpha\overline{\omega} + \alpha$.

5.2.3 Computation from the inverse problem

In the inverse problem setting with unknown conductivity $\sigma(z)$ and thus unknown Beltrami coefficient $\mu(z)$, we have to take a different approach in recovering the traces of the CGO solutions on the boundary $\partial\Omega$. Let's follow the Astala-Päivärinta (AP) method [149] to compute ω from the DN map.

We need to make the following assumptions. Let the domain $\Omega \subset \mathbb{R}^2$ be the unit disc, and let the conductivity $\sigma : \Omega \to (0, \infty)$ be an essentially bounded measurable function, that satisfies $\sigma(z) \geq c > 0$ for almost every $z \in \Omega$. To define global solutions, let us set that the outside of the domain Ω the conductivity is $\sigma \equiv 1$.[8] Let $k \in \mathbb{C}$ be a complex parameter, seen as a variable in the frequency domain, as the exponential behavior of the CGO solutions is used for constructing a nonlinear Fourier analysis for the inverse conductivity problem.

Following AP-method, let us define:

$$M_\mu(z, k) = 1 + \omega(z, k), \tag{5.9}$$

where ω is as defined in equation (5.3). Recovering ω is done by the means of solving the following boundary integral equation:

$$M_\mu(\cdot, k)|_{\partial\Omega} + 1 = (\mathcal{P}_\mu^k + \mathcal{P}_0)M_\mu(\cdot, k)|_{\partial\Omega}, \tag{5.10}$$

where projection operator $\mathcal{P}_\mu^k$ is defined as:

$$\mathcal{P}_\mu^k = e^{-ikz}\mathcal{P}_\mu e^{ikz},$$

with operator $\mathcal{P}_\mu : H^{1/2}(\partial\Omega) \to H^{1/2}(\partial\Omega)$ connected to the DN map Λ_σ by:

$$\mathcal{P}_\mu g = \frac{1}{2}(I + i\mathcal{H}_\mu)g + \frac{1}{2}\mathcal{L}g,$$

where $\mathcal{H}_\mu$ is the μ-Hilbert transform uniquely determined by the DN map (see [149] for details) and $\mathcal{L}$ is an averaging operator $\mathcal{L}\phi = |\partial\Omega|^{-1} \int_{\partial\Omega} \phi \mathrm{ds}$.

In the discrete setting, the approximate solution to the boundary integral equation (5.10) can be obtained from the Fourier transform coefficients of M_μ by solving:

$$(I - \boldsymbol{P}_\mu^k - \boldsymbol{P}_0)\widehat{M}_\mu(\cdot, k)|_{\partial\Omega} = -\widetilde{\mathcal{F}}(1), \tag{5.11}$$

where $\boldsymbol{P}_\mu^k$ and $\boldsymbol{P}_0$ are the approximate implementations of the corresponding operators, defined in the transform domain as:

$$\boldsymbol{P}_\mu^k \widehat{g} = \widetilde{\mathcal{F}}(e^{-ikz}\mathcal{F}^{-1}(\boldsymbol{P}_\mu \widetilde{\mathcal{F}}(e^{ikz}(\widetilde{\mathcal{F}}^{-1}\widehat{g})))), \tag{5.12}$$

$$\boldsymbol{P}_\mu \widehat{g} = \frac{1}{2}\big(I + i \begin{bmatrix} \boldsymbol{H}_\mu & 0 \\ 0 & \boldsymbol{H}_{-\mu} \end{bmatrix}\big)\widehat{g} + \frac{1}{2}\boldsymbol{L}\widehat{g}. \tag{5.13}$$

We can now find $\widehat{M}_\mu$ numerically by using GMRES to iteratively solve equation (5.11), and finally use equation (5.9) to calculate ω.

[8] This makes the CGO solutions harmonic in $\mathbb{R}^2 \setminus \Omega$.

Above in equation (5.13), the μ-Hilbert transform $\mathcal{H}_\mu$ in the discrete setting is defined with submatrices

$$\boldsymbol{H}_\mu = \begin{bmatrix} 0 & 0 \\ 0 & \widetilde{\boldsymbol{H}}_\mu \end{bmatrix}, \quad \boldsymbol{H}_{-\mu} = \begin{bmatrix} 0 & 0 \\ 0 & -\widetilde{\boldsymbol{H}}_\mu^{-1} \end{bmatrix},$$

where

$$\widetilde{\boldsymbol{H}}_\mu = \boldsymbol{D}_T^{-1}\boldsymbol{L}_\sigma,$$

with

$$\boldsymbol{D}_T = \begin{bmatrix} 0 & 1 & & & & & & \\ -1 & 0 & & & & & & \\ & & 0 & 2 & & & & \\ & & -2 & 0 & & & & \\ & & & & \cdots & & & \\ & & & & & & 0 & N \\ & & & & & & -N & 0 \end{bmatrix},$$

and the approximate representation of the DN-map:

$$(\boldsymbol{L}_\sigma)_{m,n} = \langle \Lambda_\sigma \phi_n, \phi_m \rangle,$$

using the trigonometric basis functions

$$\phi_n(\theta) = \begin{cases} \pi^{-1/2}\cos((n+1)\theta/2) & \text{for odd } n, \\ \pi^{-1/2}\sin((n\theta)/2) & \text{for even } n. \end{cases}$$

In equation (5.13), the $(4N+2)\times(4N+2)$ matrix representation of the averaging operator is given by

$$\boldsymbol{L} = \frac{1}{\sqrt{2\pi}}\mathrm{diag}[1, 0, \cdots, 0, 1, 0, \cdots, 0],$$

where the ones correspond to the matrix elements $(1,1)$ and $(2N+2, 2N+2)$.

And finally, a complex-valued function $g \in H^{1/2}(\partial\Omega)$ in equations (5.12), (5.13) can be represented in the Fourier domain by expanding the real and imaginary parts separately, and then organizing the coefficients as the following $\mathbb{R}^{4N+2}$-vector:

$$\widehat{g} = \begin{bmatrix} \langle \mathrm{Re}\ g, \phi_0 \rangle \\ \langle \mathrm{Re}\ g, \phi_1 \rangle \\ \vdots \\ \langle \mathrm{Re}\ g, \phi_{2N} \rangle \\ \langle \mathrm{Im}\ g, \phi_0 \rangle \\ \langle \mathrm{Im}\ g, \phi_1 \rangle \\ \vdots \\ \langle \mathrm{Im}\ g, \phi_{2N} \rangle \end{bmatrix}.$$

The direct Fourier transform is denoted by $\widehat{g} = \widetilde{\mathcal{F}} g$ and the inverse by $g = \widetilde{\mathcal{F}}^{-1}\widehat{g}$.

Reconstructing the boundary values of the CGO solutions is exponentially ill-posed, meaning that it only satisfies logarithmic stability estimates [146]. The main source of ill-posedness comes from equation (5.12). Note that there we conjugate with exponential functions, which leads to the errors in $\boldsymbol{H}_{\pm\mu}$ getting multiplied with numbers exponentially large in k. This is why we can numerically only solve the boundary integral equation in (5.10) reliably for small k, that is, for k ranging in a disc $D(0, R)$, where the radius $R > 0$ depends on the noise level.

5.3 Propagation of singularities

As introduced in [153], we can use the propagation of singularities for complex principal-type PDEs [154] to detect singularities of the conductivity.[9] Such singularities are caused by and carry important information about conductivity changes to the boundary. By analyzing the CGO solutions, we can exploit a complex principal-type geometry, leading to the fact that the leading-order term in a Born series is an invertible nonlinear generalized Radon transform of the conductivity. This is the idea that makes virtual hybrid parallel beam tomography (introduced in publication II) possible.

The boundary values of the CGO solutions ω can be transformed in a manner that allows extraction of the leading singularities of the conductivity. This is done by performing a separation of variables and a partial Fourier transform in the radial component of the spectral variable.

The Beltrami equation in (5.2) can be viewed as a scattering equation, where μ is the scatterer in the constant function 1. Thus, the traces of the CGO solutions ω can be written as a scattering series:

$$\omega \sim \sum_{n=1}^{\infty} \omega_n.$$

In the case of a known conductivity σ, we can compute each scattering term ω_n from:

$$\begin{aligned} \omega_n &= -\overline{\partial}_z^{-1}\overline{u_n}, \\ u_n &= -A\overline{u_{n-1}}, \quad u_1 = -\overline{\alpha}, \\ A: &= (-\overline{\nu}S - \overline{\alpha}P), \\ \alpha(z,k) &= -i\overline{k}e_{-k}(z)\mu(z), \\ e_k(z) &= \exp(i(kz + \overline{k}\overline{z})), \end{aligned}$$

where operators P, S are defined as in equations (5.5), (5.6). For the leading-order term of the series ω_1, the above computation simplifies as:

$$\omega_1 = \overline{\partial}_z^{-1}\alpha.$$

[9] For complex principal-type operators, the singularities propagate along 2D surfaces called *leaves*.

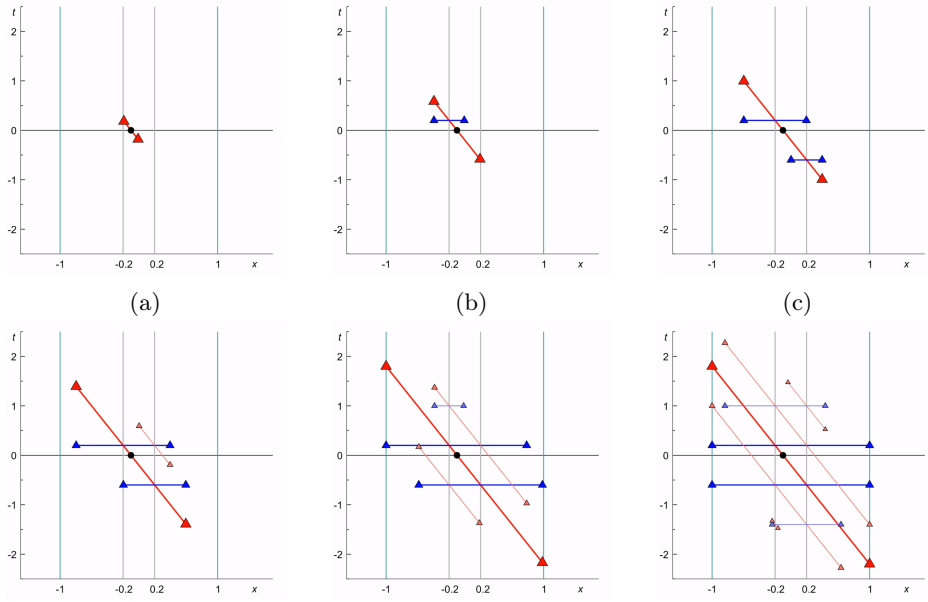


Figure 5.2: Reflection and propagation of singularities in pseudo-spacetime: starting from (a) and advancing to (f). Consider that the conductivity lies between $[-0.2, 0.2]$. As a virtual plane wave $\widehat{\omega}_1$ (red) interacts with the discontinuity of the conductivity σ, it causes scattering, propagating the singularities (b - c). When these secondary singularities hit a discontinuity, they scatter again (d) and so forth (f). This behavior continues to infinity, and what we can measure, is only the end result. Note that in reality, all the scattering happens at once, as the waves travel at an infinite speed. Visualization courtesy of Nea Nordlund.

Following [153], the complex spectral parameter k can be decomposed as $k = \tau e^{i\phi}$. Now, we can form the partial Fourier transform of the n-th order scattering term ω_n in τ, resulting in

$$\widehat{\omega}_n^{\pm}(z, t, e^{i\phi}) = F_{t\to\tau}(\omega_n^{\pm}(z, \tau e^{i\phi})).$$

Singularities in the Beltrami coefficient μ (and thus in the conductivity σ) can be then detected by taking a complex contour integral of the leading term $\widehat{\omega}_1^{\pm}(z, t, e^{i\phi})$ over $z \in \partial\Omega$:

$$T_1^{\pm}\mu(t, e^{i\phi}) = \frac{1}{2\pi i}\int_{\partial\Omega} \widehat{\omega}_1^{\pm}(z, t, e^{i\phi})dz.$$

With this, we can recover something that resembles the parallel-beam X-ray projection data of the singularities of μ. See Figure 5.2 for a visualization of the singularity scattering. Based on this, an approximate reconstruction formula for μ arises:

$$\mu = (-\triangle)^{-1/2}(T_1^{\pm}) * T_1^{\pm}\mu.$$

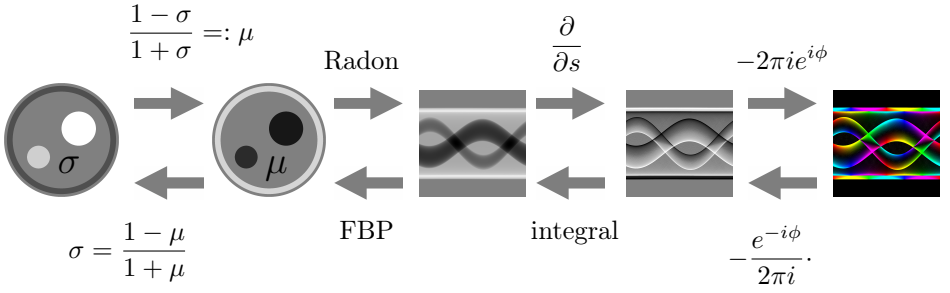


Figure 5.3: The right-most figure corresponds to the operator $T_1^+\mu(t, e^{i\phi}) = -2\pi i e^{i\phi}\big(\frac{\partial}{\partial s}R\mu\big)(\frac{1}{2}t, e^{i\phi})$.

The flowchart in figure 5.3 demonstrates the connection between Beltrami coefficient μ and the operator $T_1^{\pm}\mu(t, e^{i\phi})$.

However, in the inverse problem setting an with unknown conductivity, there is no formula for solving the leading-order term ω_1 that determines the singularities of μ. What we can compute from the DN map is the full scattering series w (as presented in section 5.2.3).

In practice, due to noise in the data, the solutions ω grow inaccurate as the magnitude of $|\tau|$ increases. To combat the inaccuracy, we can regularize the solutions by multiplying the traces of the CGO solutions with a Gaussian window that suppresses the signals to 0 as the numerical inaccuracy grows. See Figure 5.4 for a more detailed explanation of this process. However, even with a regularized ω, there is no formula for extracting the leading-order term ω_1 from the scattering series, that allows theoretically sound reconstruction of μ and the conductivity σ. Publication II addresses this issue, allowing conductivity reconstruction even from experimental data.

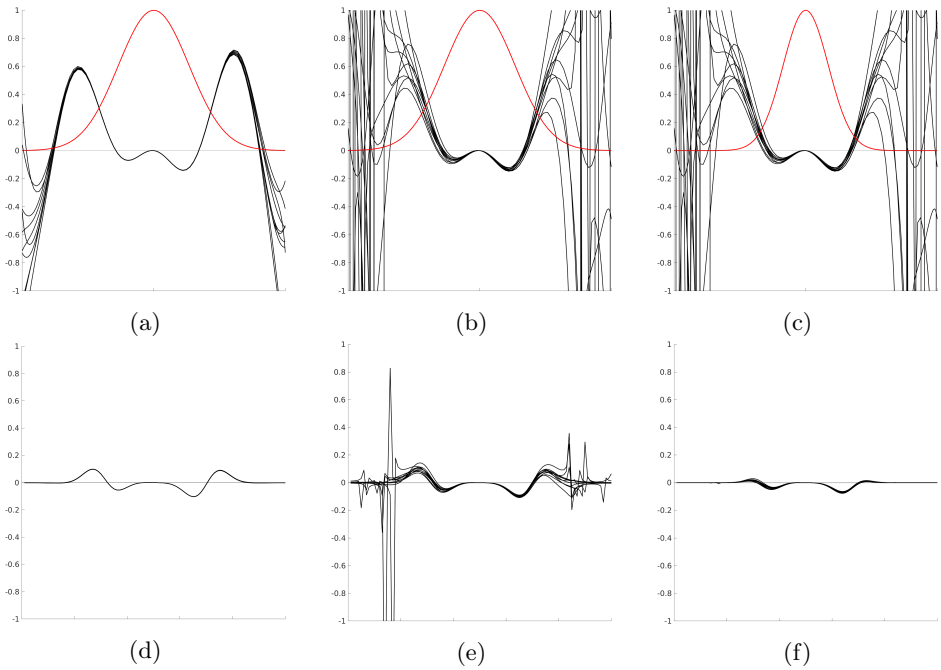


Figure 5.4: (a) $\mathrm{Re}(\omega^+(1,\tau))$ from many realizations of low-noise Λ_σ. The Gaussian window is depicted in red. (b) $\mathrm{Re}(\omega^+(1,\tau))$ from many realizations of high-noise Λ_σ. Now the previously used Gaussian window is too wide. (c) $\mathrm{Re}(\omega^+(1,\tau))$ from many realizations of high-noise Λ_σ. The Gaussian window width is now reduced to accommodate for the higher noise. (d) Low-noise $\mathrm{Re}(\omega^+(1,\tau))$ after multiplication with a suitable Gaussian window as shown in (a), removing noise but causing blurriness. (e) High-noise $\mathrm{Re}(\omega^+(1,\tau))$ after multiplication with a too narrow Gaussian window shown in (b), not removing enough noise. (f) High-noise $\mathrm{Re}(\omega^+(1,\tau))$ after multiplication with a suitable Gaussian window shown in (c), removing noise but causing blurriness.

Chapter 6

Summary of papers

The goal of this thesis was to develop algorithms for inverse problems in imaging that combine deep learning with traditional mathematical methods.

In publication I, we propose a method for boundary estimation for limited-angle CT features based on microlocal analysis via complex wavelets and deep learning. We demonstrate the method with simple experimental data.

Publication II introduces the surprising fact that EIT data can be nonlinearly processed into a parallel-beam sinogram that has the same geometry as X-ray tomography sinograms produced by first-generation CT scanners. We demonstrate this finding with experimental data.

Publication III studies how bilevel learning can be harnessed in different image restoration tasks. We learn Fields-of-Experts regularizer filters for image denoising and deblurring, and also a discretization for total variation regularization. The methods are demonstrated with natural images.

6.1 Learning a microlocal prior for limited-angle tomography

Using a limited-angle imaging geometry in X-ray tomography leads to the reconstruction problem being severely ill-posed, resulting in stretching artefacts heavily distorting the reconstruction. We propose a feature boundary estimation method that is able to differentiate between artefacts and true features. The limited-angle reconstructions suffer from their features being stretched along the central direction of projections, but with the proposed overlay, it is possible to reveal the true form and extent of the features.

In the method, we first use the Primal-dual fixed-point algorithm together with complex wavelet-based regularization to compute initial slice-by-slice reconstructions of 3D phantoms. That is, we use the PDFP algorithm to solve the minimization problem

$$\mathbf{f}_S = \operatorname*{arg\,min}_{\mathbf{f}\geq 0} \left\{ \frac{1}{2}\|A\mathbf{f} - \mathbf{m}\|_2^2 + \alpha\|W_{\mathbb{C}}\mathbf{f}\|_1 \right\},$$

where α is the regularization parameter for the regularization term $\|W_{\mathbb{C}}\mathbf{f}\|_1$ promoting solutions having a sparse representation in the complex wavelet basis. The PDFP algorithm also enforces the non-negativity constraint. Here, we treat each slice as a separate 2D reconstruction problem.

Due to the limited-angle measurement, the initial reconstructions suffer from stretching artefacts. Next, we take the complex wavelet transform of the reconstruction in order to extract the known part of the wavefront set. We process this information by performing a morphological opening of the data and binarizing it into a mask with a neural network.

Complex wavelets have directional information about the wavefront set in six subbands. In the 40-degree opening angle case, we have information in two of the six subbands, while the rest are empty. The goal is to fill in the wavefront set information in the rest of the subbands, to understand how the known boundaries extend into the unknown area. Theoretically, we know the directional information that each subband can hold and, based on this, we designed a specific structuring element $\widetilde{S}$. After morphologically dilating the known parts of the wavefront in the subbands $\mathbf{D}$:

$$(\mathbf{D}_n \oplus \widetilde{S}_n)(x) = \sup_{y \in E}[\mathbf{D}_n(y) + \widetilde{S}_n(x-y)],$$

we get an estimation of where the unknown parts lie. By summing over subbands $\sum_{j=1}^{6}(\mathbf{D}_j)$, we get an estimate for the full singular support of the target. This initial estimation is given to a convolutional neural network to refine, filling in the missing parts of the wavefront set. The morphological skeleton of this predicted singular support can be used as an overlay for the reconstruction, giving an estimation of the feature boundaries, separating artefacts from actual features.

The slice-by-slice reconstructions and boundary estimations are computed for vertical slices of the 3D targets, which are then stacked together to form a three-dimensional reconstruction. The results can also be viewed by slicing horizontally, where the boundary estimate aids with poor slice separation.

6.2 CT scans without X-rays: parallel-beam imaging from nonlinear current flows

We introduce a new reconstruction algorithm that provides a connection between electrical impedance tomography and traditional X-ray tomography, based on the idea of *virtual* X-rays. We divide the exponentially ill-posed and nonlinear inverse problem of EIT into separate steps, where all ill-posedness is confined in two linear steps. We call the emerging imaging method Virtual hybrid parallel-beam tomography. We demonstrate the method with both simulated and experimental data, creating the first-ever virtual sinograms from real EIT data.

In practical VHPT, we begin by forming a discrete matrix approximation of the DN map Λ_σ from current and voltage measurements. This is done via a calibration approach, similar to the theoretical work in [155]. Next, we calculate the so-called virtual X-ray projection data from the DN map, by first solving for the CGO traces $\omega^\pm$ from the boundary integral equation:

$$M_\mu(\,\cdot\,,k)|_{\partial\Omega} + 1 = (\mathcal{P}_\mu^k + \mathcal{P}_0)M_\mu(\,\cdot\,,k)|_{\partial\Omega},$$

where $M_\mu(z,k) = 1 + \omega^+(z,k)$, and $\mathcal{P}_\mu^k$, $\mathcal{P}_0$ are certain projection operators computed from the DN map Λ_σ.

Now that we have the traces of the CGO solutions ω, we can compute the scattering operator

$$\begin{aligned}\widetilde{T}_{\text{odd}}\mu(\tau,\varphi) &= \frac{1}{2\pi i}\int_{\partial\Omega}\big(\omega^+(z,\tau,\varphi) - \omega^-(z,\tau,\varphi)\big)dz \\ &= \frac{1}{2\pi i}\int_{\partial\Omega}\big(\omega_{\text{odd}}(z,\tau,\varphi)\big)dz.\end{aligned}$$

Next, we apply a one-dimensional Fourier transform to the scattering operator, with windowing to suppress inaccuracies caused by measurement error and noise. We choose the width of the window depending on the noise amplitude, as the inaccuracy grows with $|\tau|$. We restrict the integration to the interval $[-R,R]$ and insert a Gaussian function $e^{-a\tau^2}$ with suitable parameter $a > 0$ into the Fourier integral:

$$T_{\text{odd}}\mu(t,\varphi) = \frac{1}{2\pi i}\int_{-R}^{R} e^{-a\tau^2}e^{-it\tau}(\widetilde{T}_{\text{odd}}\mu(\tau,\varphi))d\tau.$$

By multiplying with the Gaussian window, the noisy parts of the signal will be set to zero, but as a trade-off, the resulting complex-valued $T_{\text{odd}}\mu(t,\varphi)$ has blurring along its columns.

Following this, we multiply with a complex phase factor and then integrate to get the real-valued CGO sinogram:

$$R_{\text{odd}}\mu(s,\varphi) = \int_{-\infty}^{s}\frac{e^{-i\varphi}}{2\pi i}T_{\text{odd}}\mu(t,\varphi)\,dt.$$

There are still two problems with the CGO sinogram $R_{\text{odd}}\mu$: it is blurry (due to the windowed Fourier transform) and the fact that it does not match with the theoretical concept from [153] that holds instead for the T_1 operator:

$$R_1\mu = \int \frac{-e^{i\varphi}}{2\pi i}(g * T_1^+)\mu(t,\varphi),$$

where g is a convolution kernel.

To deal with these problems, we train a convolutional neural network NN_θ to perform deconvolution in the ideal case

$$NN_\theta(R_1\mu) \to R\mu,$$

where θ is the learned parameter vector. The network is trained with training data pairs $(R_1\mu, R\mu)$, where $R\mu$ is the Radon transform of μ. As the CGO solutions form a scattering series, where $\omega_1 \approx \omega_{\text{odd}}$ roughly holds, also the CGO sinograms $R_1\mu \approx R_{\text{odd}}\mu$ are somewhat close to each other. Because of this, the network is able to generalize to work with $R_{\text{odd}}\mu$ as input data

$$NN_\theta(R_{\text{odd}}\mu) \to \widehat{R}\mu,$$

providing an estimate of the corresponding sharp CGO sinogram $\widehat{R}\mu$.

The sinogram $\widehat{R}\mu$ itself can be a useful tool in analyzing the underlying conductivity. Further, in order to reconstruct the conductivity, we can simply use the filtered back-projection (FBP) formula to reconstruct first the Beltrami coefficient μ. Then, we can get conductivity σ from the simple nonlinear transformation

$$\sigma = \frac{1-\mu}{1+\mu}.$$

6.3 Bilevel learning of regularization models and their discretization for image deblurring and super-resolution

In this work, we apply bilevel learning for learning image regularization models in image restoration. The images have been corrupted with Gaussian noise and different blur kernels: Gaussian blur, disk blur, and motion blur. First, we learn the scalar weights and convolutional filters defining a Fields-of-Experts regularizer. We demonstrate the methods with numerical results, studying their generalizability. The bilevel setup is also applied to learn a discretization of total variation regularization for image deblurring and super-resolution, but this part of the work is omitted here due to author's contributions focusing on the first case.

In the deblurring task, we use FoE regularization with a bilevel learning setup to restore the images, by adapting the regularization model to explicitly account for the blur operator. This is done by including it in the formulation of the lower-level problem. We estimate the optimal weights w and convolution filters k denoted with parameter $\theta = (w_1, \dots, w_L, k_1, \dots, k_L)$, by solving the bilevel problem:

$$\min_{\theta \in \mathcal{C}} \left\{ \frac{1}{s} \sum_{j=1}^{s} \Phi(f_j(\theta), g_j) = \frac{1}{2s} \sum_{j=1}^{s} \|f_j(\theta) - g_j\|_2^2 \right\}$$

$$\text{s.t.} \quad f_j(\theta) \in \arg\min_{f} \left\{ T_{\text{FoE}}(m_j, \theta, f) = \frac{1}{2}\|Af - m_j\|_2^2 + \sum_{\ell=1}^{L} \sum_{i=1}^{n} w_\ell \varphi((k_\ell * f)_i) \right\},$$

where $A \in \mathbb{R}^{n \times n}$ is a circulant convolution matrix and function $\varphi : \mathbb{R} \to \mathbb{R}$ is defined as

$$\varphi((k * f)_i) = \log(1 + |(k_\ell * f)_i|^2) \quad i = 1, \dots, n.$$

We set a constraint $\mathcal{C}$ for the unknown parameters θ in the minimization:

$$\mathcal{C} = C_w \times C_k \,,\; C_w = \mathbb{R}_+^L \,,\; C_k = \bigotimes_{\ell=1}^{L} C_{k_\ell} \,,\; C_{k_\ell} := \{k_\ell \mid 1^T k_\ell = 0\} \,,$$

where $\bigotimes$ denotes the Cartesian product.

We choose the mean squared error as the loss function Φ. In order to compute the gradients of the loss with respect to parameter vector θ, we can exploit the smoothness of the lower-level problems by replacing them with their first-order optimality conditions,

$$\min_{\theta \in \mathcal{C}} \; \frac{1}{2s} \sum_{j=1}^{s} \|u_j(\theta) - g_j\|_2^2$$

$$\text{s.t.} \; \nabla_f T_{\text{FoE}}(m_j, \theta, f) = A^T(Af_j - m_j) + \sum_{\ell=1}^{L} \sum_{i=1}^{n} w_\ell K_\ell^T \varphi'((K_\ell f_j)_i) = 0,$$

where the convolution products are expressed as matrix-vector products. We utilize the Lagrange multipliers strategy, where we use a combination of the conjugate gradient method and a gradient descent scheme combined with a non-monotone line search with the Barzilai-Borwein rule in order to iteratively find the optimal parameter values θ for the FoE regularizer.

The training examples $\{(g_j, m_j)\}_{j=1}^{s}$ consist of small patches of natural images that are corrupted by blurring and noise, and the corresponding ground truths. Using the bilevel learning method, we iterate over the outer and inner schemes until convergence. The higher-level solver calls to the lower-level solver, whose outputs $f^{(k+1)}$ are fed to the upper-level solver. After convergence, the final output is the optimal set of parameters $\theta^* = (w, k)$.

After the optimal weights and convolution filters are learned, we test the method on natural images. The numerical results prove satisfying, even with only 9 images in the training set, trained on different kinds of blur. The robustness of the bilevel learning scheme is justified by the fact that increasing the training set size to 75 images does not improve the results significantly.

Bibliography

[1] Jacques Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton university bulletin*, pages 49–52, 1902.

[2] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.

[3] Jennifer L Mueller and Samuli Siltanen. *Linear and nonlinear inverse problems with practical applications*. SIAM, 2012.

[4] Martin Benning and Martin Burger. Modern regularization methods for inverse problems. *Acta numerica*, 27:1–111, 2018.

[5] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[6] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[7] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[8] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. {TensorFlow}: a system for {Large-Scale} machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pages 265–283, 2016.

[9] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[10] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688, 2019.

[11] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831, 2010.

[12] Yinpeng Dong, Hang Su, Jun Zhu, and Fan Bao. Towards interpretable deep neural networks by leveraging adversarial examples. *arXiv preprint arXiv:1708.05493*, 2017.

[13] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

[14] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps: Automation of decision making*, pages 323–350, 2018.

[15] Daniël M Pelt, Kees Joost Batenburg, and James A Sethian. Improving tomographic reconstruction from limited data using mixed-scale dense convolutional neural networks. *Journal of Imaging*, 4(11):128, 2018.

[16] Ji Zhao, Zhiqiang Chen, Li Zhang, and Xin Jin. Few-view ct reconstruction method based on deep learning. In *2016 IEEE Nuclear Science Symposium, Medical Imaging Conference and Room-Temperature Semiconductor Detector Workshop (NSS/MIC/RTSD)*, pages 1–4. IEEE, 2016.

[17] Hu Chen, Yi Zhang, Weihua Zhang, Peixi Liao, Ke Li, Jiliu Zhou, and Ge Wang. Low-dose ct denoising with convolutional neural network. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 143–146. IEEE, 2017.

[18] Sarah Jane Hamilton and Andreas Hauptmann. Deep d-bar: Real-time electrical impedance tomography imaging with deep neural networks. *IEEE transactions on medical imaging*, 37(10):2367–2377, 2018.

[19] Tatiana A Bubba, Gitta Kutyniok, Matti Lassas, Maximilian März, Wojciech Samek, Samuli Siltanen, and Vignesh Srinivasan. Learning the invisible: A hybrid deep learning-shearlet framework for limited angle computed tomography. *Inverse Problems*, 35(6):064002, 2019.

[20] Hector Andrade-Loarca, Gitta Kutyniok, Ozan Oktem, and Philipp C Petersen. Extraction of digital wavefront sets using applied harmonic analysis and deep neural networks. *SIAM Journal on Imaging Sciences*, 12(4):1936–1966, 2019.

[21] Simon Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numerica*, 28:1–174, 2019.

[22] CE Floyd. An artificial neural network for spect image reconstruction. *IEEE transactions on medical imaging*, 10(3):485–487, 1991.

[23] Tobias Würfl, Mathis Hoffmann, Vincent Christlein, Katharina Breininger, Yixin Huang, Mathias Unberath, and Andreas K Maier. Deep learning computed tomography: Learning projection-domain weights from image domain in limited angle problems. *IEEE transactions on medical imaging*, 37(6):1454–1463, 2018.

[24] Eline Janssens, Jan De Beenhouwer, Mattias Van Dael, Thomas De Schryver, Luc Van Hoorebeke, Pieter Verboven, Bart Nicolai, and Jan Sijbers. Neural network hilbert transform based filtered backprojection for fast inline x-ray inspection. *Measurement Science and Technology*, 29(3):034012, 2018.

[25] Marshall F Tappen. Utilizing variational optimization to learn markov random fields. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[26] Juan Carlos De los Reyes and Carola-Bibiane Schönlieb. Image denoising: learning the noise model via nonsmooth pde-constrained optimization. *Inverse Problems & Imaging*, 7(4), 2013.

[27] Yunjin Chen, Wei Yu, and Thomas Pock. On learning optimized reaction diffusion processes for effective image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5261–5269, 2015.

[28] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE global conference on signal and information processing*, pages 945–948. IEEE, 2013.

[29] Sebastian Lunz, Ozan Öktem, and Carola-Bibiane Schönlieb. Adversarial regularizers in inverse problems. *Advances in neural information processing systems*, 31, 2018.

[30] Housen Li, Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. Nett: Solving inverse problems with deep neural networks. *Inverse Problems*, 36(6):065005, 2020.

[31] Daniel Maria Pelt and Kees Joost Batenburg. Fast tomographic reconstruction from limited data using artificial neural networks. *IEEE Transactions on Image Processing*, 22(12):5238–5251, 2013.

[32] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th international conference on international conference on machine learning*, pages 399–406, 2010.

[33] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016.

[34] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017.

[35] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018.

[36] Mikio Sato. Regularity of hyperfunction solutions of partial differential equations. *Actes Congr. Int. Nat. Nice*, 2:785–794, 1970.

[37] Lars Hörmander. Fourier integral operators, *i*. *Acta Math*, 127:79–183, 1971.

[38] Augustin Cauchy et al. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.

[39] Jack Kiefer and Jacob Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, pages 462–466, 1952.

[40] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[41] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[42] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.

[43] R. Bellman, Rand Corporation, and Karreman Mathematics Research Collection. *Dynamic Programming*. Rand Corporation research study. Princeton University Press, 1957.

[44] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[45] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[46] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[47] Hugo Larochelle, Yoshua Bengio, Jérôme Louradour, and Pascal Lamblin. Exploring strategies for training deep neural networks. *Journal of machine learning research*, 10(1), 2009.

[48] David H Wolpert. The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390, 1996.

[49] Jan Kukačka, Vladimir Golkov, and Daniel Cremers. Regularization for deep learning: A taxonomy. *arXiv preprint arXiv:1710.10686*, 2017.

[50] P Auer, M Herbster, and MK Warmuth. Advances in neural information processing systems 8, 1996.

[51] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM review*, 60(2):223–311, 2018.

[52] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

[53] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR, 2018.

[54] Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial intelligence and statistics*, pages 192–204. PMLR, 2015.

[55] Simon Becker, Yao Zhang, and Alpha A Lee. Geometry of energy landscapes and the optimizability of deep neural networks. *Physical review letters*, 124(10):108301, 2020.

[56] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[57] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[58] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[59] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE transactions on image processing*, 26(9):4509–4522, 2017.

[60] Eunhee Kang, Junhong Min, and Jong Chul Ye. A deep convolutional neural network using directional wavelets for low-dose x-ray ct reconstruction. *Medical physics*, 44(10):e360–e375, 2017.

[61] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[62] Sergey Ioffe. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[63] Sedki M Riad. The deconvolution problem: An overview. *Proceedings of the IEEE*, 74(1):82–85, 1986.

[64] Ruxin Wang and Dacheng Tao. Recent progress in image deblurring. *arXiv preprint arXiv:1409.6838*, 2014.

[65] Jean-Luc Starck, Eric Pantin, and Fionn Murtagh. Deconvolution in astronomy: A review. *Publications of the Astronomical Society of the Pacific*, 114(800):1051, 2002.

[66] Robert J Hanisch, Richard L White, and Ronald L Gilliland. Deconvolution of hubble space telescope images and spectra. *Deconvolution of images and spectra*, 2, 1997.

[67] Alexander H Delaney and Yoram Bresler. Globally convergent edge-preserving regularized reconstruction: an application to limited-angle tomography. *IEEE Transactions on Image Processing*, 7(2):204–221, 1998.

[68] Emil Y Sidky, Chien-Min Kao, and Xiaochuan Pan. Accurate image reconstruction from few-views and limited-angle data in divergent-beam ct. *Journal of X-ray Science and Technology*, 14(2):119–139, 2006.

[69] Ludwig Ritschl, Frank Bergner, Christof Fleischmann, and Marc Kachelrieß. Improved total variation-based ct image reconstruction applied to clinical data. *Physics in Medicine & Biology*, 56(6):1545, 2011.

[70] Jürgen Frikel. Sparse regularization in limited angle tomography. *Applied and Computational Harmonic Analysis*, 34(1):117–141, 2013.

[71] Zhiqiang Chen, Xin Jin, Liang Li, and Ge Wang. A limited-angle ct reconstruction method based on anisotropic tv minimization. *Physics in Medicine & Biology*, 58(7):2119, 2013.

[72] Yixing Huang, Oliver Taubmann, Xiaolin Huang, Viktor Haase, Günter Lauritsch, and A Maier. A new weighted anisotropic total variation algorithm for limited angle tomography. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 585–588. IEEE, 2016.

[73] Yixing Huang, Oliver Taubmann, Xiaolin Huang, Viktor Haase, Guenter Lauritsch, and Andreas Maier. Scale-space anisotropic total variation for limited angle tomography. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 2(4):307–314, 2018.

[74] E Loli Piccolomini and E Morotti. A fast total variation-based iterative algorithm for digital breast tomosynthesis image reconstruction. *Journal of Algorithms & Computational Technology*, 10(4):277–289, 2016.

[75] Anders H Andersen. Algebraic reconstruction in ct from limited views. *IEEE transactions on medical imaging*, 8(1):50–55, 1989.

[76] Kenneth M Hanson and George W Wecksung. Bayesian approach to limited-angle reconstruction in computed tomography. *JOSA*, 73(11):1501–1509, 1983.

[77] Ville Kolehmainen, Samuli Siltanen, Seppo Järvenpää, Jari P Kaipio, P Koistinen, M Lassas, J Pirttilä, and E Somersalo. Statistical inversion for medical X-ray tomography with few radiographs: II. Application to dental radiology. *Physics in Medicine & Biology*, 48(10):1465–1491, 2003.

[78] Maaria Rantala, Simopekka Vänskä, Seppo Järvenpää, Martti Kalke, Matti Lassas, Jan Moberg, and Samuli Siltanen. Wavelet-based reconstruction for limited-angle x-ray tomography. *IEEE Transactions on Medical Imaging*, 25(2):210–217, 2006.

[79] Babak Maboudi Afkham, Yiqiu Dong, and Per Christian Hansen. Uncertainty quantification of inclusion boundaries in the context of x-ray tomography. *SIAM/ASA Journal on Uncertainty Quantification*, 11(1):31–61, 2023.

[80] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

[81] Stefan Roth and Michael J. Black. Fields of experts: a framework for learning image priors. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2:860–867 vol. 2, 2005.

[82] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[83] Max Welling, Simon Osindero, and Geoffrey E Hinton. Learning sparse topographic representations with products of student-t distributions. *Advances in neural information processing systems*, 15, 2002.

[84] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984.

[85] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27:107–126, 1998.

[86] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings 8th International Conference on Computer Vision*, volume 2, pages 416–423, July 2001.

[87] Caroline Crockett, Jeffrey A Fessler, et al. Bilevel methods for image reconstruction. *Foundations and Trends® in Signal Processing*, 15(2-3):121–289, 2022.

[88] Karl Kunisch and Thomas Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM Journal on Imaging Sciences*, 6(2):938–983, 2013.

[89] Magnus Rudolph Hestenes, Eduard Stiefel, et al. *Methods of conjugate gradients for solving linear systems*, volume 49. NBS Washington, DC, 1952.

[90] Jonathan Barzilai and Jonathan M Borwein. Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1):141–148, 1988.

[91] Thorsten M Buzug. Computed tomography. In *Springer handbook of medical technology*, pages 311–342. Springer, 2011.

[92] Frank Natterer and Frank Wübbeling. *Mathematical methods in image reconstruction*. SIAM, 2001.

[93] Allan Macleod Cormack. Representation of a function by its line integrals, with some radiological applications. *Journal of applied physics*, 34(9):2722–2727, 1963.

[94] Allan M Cormack and Godfrey N Hounsfield. The nobel prize in physiology or medicine 1979. *Nobelprize. org*, 9, 2011.

[95] John M Boone and Cynthia H McCollough. Computed tomography turns 50. *Physics Today*, 74(9):34–40, 2021.

[96] Johann Radon. Uber die bestimmung von funktionen durch ihre integralwerte langs gewissez mannigfaltigheiten, ber. *Verh. Sachs. Akad. Wiss. Leipzig, Math Phys Klass*, 69, 1917.

[97] Frank Natterer and F Natterer. Computerized tomography. *The Mathematics of Computerized Tomography*, pages 1–8, 1986.

[98] Loren T Niklason, Bradley T Christian, Laura E Niklason, Daniel B Kopans, Donald E Castleberry, BH Opsahl-Ong, Cynthia E Landberg, Priscilla J Slanetz, Angela A Giardino, Richard Moore, et al. Digital tomosynthesis in breast imaging. *Radiology*, 205(2):399–406, 1997.

[99] Srinivasan Vedantham, Andrew Karellas, Gopal R Vijayaraghavan, and Daniel B Kopans. Digital breast tomosynthesis: state of the art. *Radiology*, 277(3):663–684, 2015.

[100] Tao Wu, Richard H Moore, Elizabeth A Rafferty, and Daniel B Kopans. A comparison of reconstruction algorithms for breast tomosynthesis. *Medical Physics*, 31(9):2636–2647, 2004.

[101] Rongli Guo, Itay Barnea, and Natan T Shaked. Limited-angle tomographic phase microscopy utilizing confocal scanning fluorescence microscopy. *Biomedical Optics Express*, 12(4):1869–1881, 2021.

[102] Misty I Haith, Peter Huthwaite, and Michael JS Lowe. Defect characterisation from limited view pipeline radiography. *NDT & E International*, 86:186–198, 2017.

[103] Weslley Silva, Ricardo Lopes, Uwe Zscherpel, Dietmar Meinel, and Uwe Ewert. X-ray imaging techniques for inspection of composite pipelines. *Micron*, 145:103033, 2021.

[104] Nicolai André Brogaard Riis, Jacob Frøsig, Yiqiu Dong, and Per Christian Hansen. Limited-data X-ray CT for underwater pipeline inspection. *Inverse Problems*, 34(3):034002, 2018.

[105] Mark E Davison. The ill-conditioned nature of the limited angle tomography problem. *SIAM Journal on Applied Mathematics*, 43(2):428–448, 1983.

[106] Samuli Siltanen, Ville Kolehmainen, Seppo Järvenpää, JP Kaipio, P Koistinen, Matti Lassas, J Pirttilä, and Erkki Somersalo. Statistical inversion for medical X-ray tomography with few radiographs: I. General theory. *Physics in Medicine & Biology*, 48(10):1437–1464, 2003.

[107] Eric Todd Quinto. Singularities of the X-ray transform and limited data tomography in $\mathbb{R}^2$ and $\mathbb{R}^3$. *SIAM Journal on Mathematical Analysis*, 24(5):1215–1225, 1993.

[108] Jürgen Frikel and Eric Todd Quinto. Characterization and reduction of artifacts in limited angle tomography. *Inverse Problems*, 29(12):125007, 2013.

[109] Allan Greenleaf and Gunther Uhlmann. Nonlocal inversion formulas for the X-ray transform. *Duke Mathematical Journal*, 58(1):205–240, 1989.

[110] Keijo Hämäläinen, Lauri Harhanen, Aki Kallonen, Antti Kujanpää, Esa Niemi, and Samuli Siltanen. Tomographic x-ray data of a walnut, May 2018.

[111] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[112] David L Donoho. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(6):797–829, 2006.

[113] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 57(11):1413–1457, 2004.

[114] Zenith Purisha, Juho Rimpeläinen, Tatiana Bubba, and Samuli Siltanen. Controlled wavelet domain sparsity for x-ray tomography. *Measurement Science and Technology*, 29(1):014002, 2017.

[115] Diego Vidaurre, Concha Bielza, and Pedro Larranaga. A survey of l1 regression. *International Statistical Review*, 81(3):361–387, 2013.

[116] Ingrid Daubechies. Ten lectures on wavelets. *Society for industrial and applied mathematics*, 1992.

[117] Mallat Stephane. A wavelet tour of signal processing, 1999.

[118] Albert Boggess and Francis J Narcowich. *A first course in wavelets with Fourier analysis*. John Wiley & Sons, 2015.

[119] Alfred Haar. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 71(1):38–53, 1911.

[120] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on pure and applied mathematics*, 41(7):909–996, 1988.

[121] Stephane G Mallat. Multiresolution approximations and wavelet orthonormal bases of $l^2(r)$. *Transactions of the American mathematical society*, 315(1):69–87, 1989.

[122] Yves Meyer. *Wavelets and operators: volume 1*. Number 37. Cambridge university press, 1992.

[123] Nick Kingsbury. The dual-tree complex wavelet transform: a new efficient tool for image restoration and enhancement. In *9th European Signal Processing Conference (EUSIPCO 1998)*, pages 1–4. IEEE, 1998.

[124] Ivan W Selesnick, Richard G Baraniuk, and Nick C Kingsbury. The dual-tree complex wavelet transform. *IEEE signal processing magazine*, 22(6):123–151, 2005.

[125] Nick Kingsbury. Design of q-shift complex wavelets for image processing using frequency domain energy minimization. In *Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429)*, volume 1, pages I–1013. IEEE, 2003.

[126] Gitta Kutyniok and Demetrio Labate. *Shearlets: Multiscale analysis for multivariate data*. Springer Science & Business Media, 2012.

[127] Demetrio Labate, Wang-Q Lim, Gitta Kutyniok, and Guido Weiss. Sparse multidimensional representation using shearlets. In *Wavelets XI*, volume 5914, pages 254–262. SPIE, 2005.

[128] Peijun Chen, Jianguo Huang, and Xiaoqun Zhang. A primal–dual fixed point algorithm for convex separable minimization with applications to image restoration. *Inverse Problems*, 29(2):025011, 2013.

[129] Peijun Chen, Jianguo Huang, and Xiaoqun Zhang. A primal-dual fixed point algorithm for minimization of the sum of three convex separable functions. *Fixed Point Theory and Applications*, 2016:1–18, 2016.

[130] Margaret Cheney, David Isaacson, and Jonathan C Newell. Electrical impedance tomography. *SIAM review*, 41(1):85–101, 1999.

[131] DS Holder. Detection of cerebral ischaemia in the anaesthetised rat by impedance measurement with scalp electrodes: implications for non-invasive imaging of stroke by electrical impedance tomography. *Clinical Physics and Physiological Measurement*, 13(1):63, 1992.

[132] DS Holder. Electrical impedance tomography with cortical or scalp electrodes during global cerebral ischaemia in the anaesthetised rat. *Clinical Physics and Physiological Measurement*, 13(1):87, 1992.

[133] Emma Malone, Markus Jehl, Simon Arridge, Timo Betcke, and David Holder. Stroke type differentiation using spectrally constrained multifrequency eit: evaluation of feasibility in a realistic head model. *Physiological measurement*, 35(6):1051–1066, 2014.

[134] Michel Assenheimer, Orah Laver-Moskovitz, Dov Malonek, David Manor, Udi Nahaliel, Ron Nitzan, and Abraham Saad. The t-scantm technology: electrical impedance as a diagnostic tool for breast cancer detection. *Physiological measurement*, 22(1):1, 2001.

[135] Brian H Brown and Andrew D Seagar. The sheffield data collection system. *Clinical physics and physiological measurement*, 8(4A):91, 1987.

[136] Trevor York. Status of electrical tomography in industrial applications. *Journal of Electronic imaging*, 10(3):608–619, 2001.

[137] A Abubakar, TM Habashy, M Li, and J Liu. Inversion algorithms for large-scale geophysical electromagnetic measurements. *Inverse Problems*, 25(12):123012, 2009.

[138] Omid Rajabi Shishvan, Ahmed Abdelwahab, Nilton Barbosa da Rosa, Gary J. Saulnier, Jennifer L. Mueller, Jonathan C. Newell, and David Isaacson. Act5 electrical impedance tomography system. *IEEE Transactions on Biomedical Engineering*, 71(1):227–236, 2024.

[139] David Isaacson. Distinguishability of conductivities by electric current computed tomography. *IEEE transactions on medical imaging*, 5(2):91–95, 1986.

[140] Alberto P Calderón. On an inverse boundary value problem. *Comput. Appl. Math*, pages 2–3, 2006.

[141] Lyudvig Dmitrievich Faddeev. Increasing solutions of the schrödinger equation. In *Fifty Years of Mathematical Physics: Selected Works of Ludwig Faddeev*, pages 34–36. World Scientific, 2016.

[142] John Sylvester and Gunther Uhlmann. A global uniqueness theorem for an inverse boundary value problem. *Annals of mathematics*, pages 153–169, 1987.

[143] Adrian I Nachman. Global uniqueness for a two-dimensional inverse boundary value problem. *Annals of Mathematics*, pages 71–96, 1996.

[144] Samuli Siltanen, Jennifer Mueller, and David Isaacson. An implementation of the reconstruction algorithm of a nachman for the 2d inverse conductivity problem. *Inverse Problems*, 16(3):681, 2000.

[145] David Isaacson, Jennifer L Mueller, Jonathan C Newell, and Samuli Siltanen. Reconstructions of chest phantoms by the d-bar method for electrical impedance tomography. *IEEE Transactions on medical imaging*, 23(7):821–828, 2004.

[146] Kim Knudsen, Matti Lassas, Jennifer L Mueller, and Samuli Siltanen. Regularized d-bar method for the inverse conductivity problem. *Inverse Problems & Imaging*, 3(4):599, 2009.

[147] Thiago de Castro Martins, André Kubagawa Sato, Fernando Silva de Moura, Erick Dario León Bueno de Camargo, Olavo Luppi Silva, Talles Batista Rattis Santos, Zhanqi Zhao, Knut Möeller, Marcelo Brito Passos Amato, Jennifer L. Mueller, Raul Gonzalez Lima, and Marcos de Sales Guerra Tsuzuki. A review of electrical impedance tomography in lung applications: Theory and algorithms for absolute images. *Annual Reviews in Control*, 48:442–471, 2019.

[148] Jennifer L Mueller. Evaluation of pulmonary structure and function in patients with cystic fibrosis from electrical impedance tomography data. *Biomedical Engineering Technologies: Volume 1*, pages 733–750, 2022.

[149] Kari Astala and Lassi Päivärinta. Calderón's inverse conductivity problem in the plane. *Annals of Mathematics*, pages 265–299, 2006.

[150] Kari Astala, Jennifer L Mueller, Lassi Päivärinta, and Samuli Siltanen. Numerical computation of complex geometrical optics solutions to the conductivity equation. *Applied and Computational Harmonic Analysis*, 29(1):2–17, 2010.

[151] Marko Huhtanen and Allan Perämäki. Numerical solution of the r-linear beltrami equation. *Mathematics of Computation*, 81(277):387–397, 2012.

[152] Youcef Saad and Martin H Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[153] Allan Greenleaf, Matti Lassas, Matteo Santacesaria, Samuli Siltanen, and Gunther Uhlmann. Propagation and recovery of singularities in the inverse conductivity problem. *Analysis & PDE*, 11(8):1901–1943, 2018.

[154] JJ Duistermaat JJ and L Hörmander. Fourier integral operators ii. *Acta Math*, 128:183–269, 1972.

[155] Henrik Garde and Nuutti Hyvönen. Mimicking relative continuum measurements by electrode data in two-dimensional electrical impedance tomography. *Numerische Mathematik*, 147:579–609, 2021.