Saarland University

A Dissertation Submitted Towards the Degree Doctor of Engineering (Dr.-Ing.)
of the Faculty of Mathematics and Computer Science of Saarland University

3D Exemplar-based Image Inpainting in Electron Microscopy

by

Patrick Trampert

Saarbrücken

– 2019 –

Tag des Kolloquiums: 13.01.2020

Dekan der Fakultät für Mathematik und Informatik
Prof. Dr. Sebastian Hack

Prüfungsausschuss
Prof. Dr. Hans-Peter Lenhof (Vorsitzender)
Prof. Dr.-Ing. Philipp Slusallek
Prof. Dr. rer. nat. habil. Joachim Weickert
Prof. Dr. rer. nat. habil. Michael Breuß
Dr. Tim Dahmen (Akademischer Assistent)

1

Short Abstract

In electron microscopy (EM) a common problem is the non-availability of data, which causes artefacts in reconstructions. In this thesis the goal is to generate artificial data where missing in EM by using exemplar-based inpainting (EBI). We implement an accelerated 3D version tailored to applications in EM, which reduces reconstruction times from days to minutes.

We develop intelligent sampling strategies to find optimal data as input for reconstruction methods. Further, we investigate approaches to reduce electron dose and acquisition time. Sparse sampling followed by inpainting is the most promising approach. As common evaluation measures may lead to misinterpretation of results in EM and falsify a subsequent analysis, we propose to use application driven metrics and demonstrate this in a segmentation task.

A further application of our technique is the artificial generation of projections in tilt-based EM. EBI is used to generate missing projections, such that the full angular range is covered. Subsequent reconstructions are significantly enhanced in terms of resolution, which facilitates further analysis of samples.

In conclusion, EBI proves promising when used as an additional data generation step to tackle the non-availability of data in EM, which is evaluated in selected applications. Enhancing adaptive sampling methods and refining EBI, especially considering the mutual influence, promotes higher throughput in EM using less electron dose while not lessening quality.

Kurzzusammenfassung

Ein häufig vorkommendes Problem in der Elektronenmikroskopie (EM) ist die Nichtverfügbarkeit von Daten, was zu Artefakten in Rekonstruktionen führt. In dieser Arbeit ist es das Ziel fehlende Daten in der EM künstlich zu erzeugen, was durch Exemplar-basiertes Inpainting (EBI) realisiert wird. Wir implementieren eine auf EM zugeschnittene beschleunigte 3D Version, welche es ermöglicht, Rekonstruktionszeiten von Tagen auf Minuten zu reduzieren.

Wir entwickeln intelligente Abtaststrategien, um optimale Datenpunkte für die Rekonstruktion zu erhalten. Ansätze zur Reduzierung von Elektronendosis und Aufnahmezeit werden untersucht. Unterabtastung gefolgt von Inpainting führt zu den besten Resultaten. Evaluationsmaße zur Beurteilung der Rekonstruktionsqualität helfen in der EM oft nicht und können zu falschen Schlüssen führen, weswegen anwendungsbasierte Metriken die bessere Wahl darstellen. Dies demonstrieren wir anhand eines Beispiels.

Die künstliche Erzeugung von Projektionen in der neigungsbasierten Elektronentomographie ist eine weitere Anwendung. EBI wird verwendet um fehlende Projektionen zu generieren. Daraus resultierende Rekonstruktionen weisen eine deutlich erhöhte Auflösung auf.

EBI ist ein vielversprechender Ansatz, um nicht verfügbare Daten in der EM zu generieren. Dies wird auf Basis verschiedener Anwendungen gezeigt und evaluiert. Adaptive Aufnahmestrategien und EBI können also zu einem höheren Durchsatz in der EM führen, ohne die Bildqualität merklich zu verschlechtern.

Abstract

In electron microscopy a common problem is the non-availability of data. For tilt-based tomography for example, it is in general not possible to gather projections from the whole angular range. This causes artefacts in the corresponding reconstructions. Aiming at higher throughput and smaller electron dose when acquiring measurements of a specimen leads to additional reduction of information available for a reconstruction. In this thesis the goal is to generate artificial data where it is missing in electron microscopy by utilising exemplar-based inpainting (EBI), a technique known from image processing that is used e.g. for reconstructing sparsely acquired data.

We implement a three-dimensional version of EBI tailored to applications in electron microscopy. Nanoscale imaging of specimens with high resolution, especially in a three-dimensional setting, requires a lot of data, which must be processed as fast as possible for a reasonable acquisition and reconstruction time. Thus, we also propose ways to speed-up EBI. We design a carefully optimised high-performance version on graphics processing units. Furthermore, we introduce a new acceleration structure based on space filling curves and dimensionality reduction. Our novel acceleration structure yields a maximal speed-up factor of 660 compared to a parallel approach on a central processing unit, so that reconstruction times of several days can be reduced to minutes.

Inpainting methods inspired by compressed sensing (CS) and sparse coding gained much interest during the last years in electron microscopy. Random sparse sampling strategies are used for data acquisition followed by non-linear optimisation methods for reconstruction of the non-sampled data. We evaluate several such methods to assess their applicability in electron tomography. Furthermore, we compare alternative approaches to reduce electron dose and acquisition time, particularly denoising, super-resolution (SR), and our version of EBI. The investigation shows that sparse sampling followed by inpainting is the most promising approach. Due to lengthy runtime of the best performing sparse coding algorithms and the inability to handle three-dimensional data, we concentrate mainly on our high-performance EBI for further research.

In contrast to sparse coding inpainting, where random sampling is used for data acquisition, there are more sophisticated ways to adaptively sample data. We develop and investigate adaptive sampling procedures that infer knowledge in an iterative scanning

procedure and compare the performance of EBI reconstructions on adaptively sampled data against randomly sampled data. The evaluation indicates that adaptive sampling strategies combined with EBI can enhance the reconstruction quality significantly.

A further application of our technique is the artificial generation of projections in tilt-based electron tomography. EBI is used to generate missing projections, such that the full angular range is covered. Subsequent reconstructions are significantly enhanced in terms of resolution, which facilitates further analysis of samples.

While executing experiments in the course of this thesis it became evident that common evaluation measures may not help to judge reconstruction quality or even lead to misinterpretation of results. This can also falsify subsequent analysis and interpretation of the reconstructed data. Hence, we propose to use an application driven evaluation procedure to ensure the applicability of a method for a workflow. As proof of concept we apply such a process to a segmentation task showing its benefits compared to traditional evaluation techniques.

In conclusion, EBI proves promising when used as an additional data generation step to tackle the non-availability of data in electron microscopy, which was evaluated in selected applications. Further use cases in electron microscopy may appear in the future. Enhancing adaptive sampling methods and refining the EBI, especially considering the mutual influence, will promote higher throughput in electron microscopy using less electron dose while not lessening quality, which will also reduce acquisition times remarkably.

Zusammenfassung

Ein häufig vorkommendes Problem in der Elektronenmikroskopie ist die Nichtverfügbarkeit von Daten. Ein Beispiel ist die neigungsbasierte Tomographie bei der es im Allgemeinen nicht möglich ist Projektionen aller Winkelbereiche aufzunehmen. Dies führt zu Artefakten in korrespondierenden Rekonstruktionen. Höherer Datendurchsatz, sowie die Verringerung der eingesetzten Elektronendosis bei der Aufnahme von Messwerten einer Probe führen zu einer weiteren Reduktion von verfügbaren Informationen, die für eine Rekonstruktion verwendet werden. In dieser Arbeit ist es das Ziel fehlende Daten in der Elektronenmikroskopie künstlich zu erzeugen. Dies realisieren wir mit Hilfe von Exemplar-basiertem Inpainting (EBI), einer Technik, die aus der Bildverarbeitung bekannt ist. Eine Beispielanwendung ist die Rekonstruktion von Daten, welche mit einer sehr geringen Abtastdichte aufgenommen wurden.

Wir entwerfen ein dreidimensionales EBI, welches für eine Anwendungen in der Elektronenmikroskopie angepasst wird. Hochauflösende Bildgebung von Proben auf der Nano-Ebene, insbesondere in einem dreidimensionalen Setting, benötigt viele Daten. Diese müssen mit höchstmöglicher Geschwindigkeit verarbeitet werden, um in angemessener Zeit verfügbar zu sein. Darauf basierend stellen wir auch Methoden zur Beschleunigung von EBI vor. Zu diesem Zweck implementieren wir eine sorgfälltig optimierte, hoch-performante Version, welche auf einer Grafikkarte läuft. Weiterhin führen wir eine neue Beschleunigerstruktur ein, welche auf raumfüllenden Kurven und Dimensionsreduktion basiert. Verglichen mit einer parallel ausgeführten Version auf einer Zentraleinheit können wir durch unsere neuartige Beschleunigerstruktur die benötigte Zeit einer Rekonstruktion um einen maximalen Faktor von 660 reduzieren. Dies ermöglicht Rekonstruktionszeiten von mehreren Tagen auf wenige Minuten zu verringern.

Inpainting Methoden, die von Compressed Sensing und Sparse Coding inspiriert sind, erhielten in den letzten Jahren eine erhöhte Aufmerksamkeit in der Elektronenmikroskopie. Randomisierte Unterabbtastungsstrategien werden zur Datenbeschaffung verwendet. Die nicht abgetasteten Daten werden dann mit Hilfe von nicht linearen Optimierungsverfahren rekonstruiert. Wir evaluieren verschiedene solche Methoden, um ihre Anwendbarkeit in der Elektronenmikroskopie beurteilen zu können. Weiterhin vergleichen wir alternative Ansätze zur Reduktion von Elektronendosis und Aufnahmezeit, insbesondere Denoising, Super-Resolution und unser EBI. Die Untersuchungen zeigen, dass Unterabtastung

gefolgt von Inpainting zur Rekonstruktion der erfolgsversprechendste der evaluierten Ansätze ist. Wir konzentrieren uns hauptsächlich auf unser hoch-performantes EBI, da die am besten abschneidenden Sparse Coding Algorithmen sehr lange Laufzeiten haben und keine dreidimensionalen Daten verarbeiten können.

Bei Inpainting Algorithmen, die auf Sparse Coding beruhen, werden Daten randomisiert abgetastet. Es existieren allerdings ausgefeiltere Methoden zur adaptiven Datenabtastung. Wir entwickeln und untersuchen adaptive Abtastungsverfahren, welche Wissen auf Basis von iterativen Scanvorgängen erschließen. Wir vergleichen die Rekonstruktionsgüte von EBI auf Basis von adaptiven sowie randomisierten Scanverfahren. Die Evaluierung weist darauf hin, dass adaptive Scanverfahren kombiniert mit EBI die Rekonstruktionsqualität erheblich verbessern können.

Eine weitere Anwendung unserer Technik ist die künstliche Erzeugung von Projektionen in der neigungsbasierten Elektronentomographie. EBI wird verwendet um fehlende Projektionen zu generieren, so dass alle Winkelbereiche abgedeckt sind. Daraus resultierende Rekonstruktionen weisen eine deutlich erhöhte Auflösung auf. Dies vereinfacht weitere Auswertungen der Daten.

Während der Ausführung der Experimente für diese Arbeit zeigte sich, dass Evaluationsmaße zur Beurteilung der Rekonstruktionsqualität oft nicht helfen oder sogar zu falschen Schlüssen führen können. Dies kann folgende Analysen und Interpretationen der rekonstruierten Daten verfälschen. Deswegen schlagen wir eine Methodik vor, die einer anwendungsbasierten Evaluation eines Workflows zugrunde liegt. Als Konzeptnachweis wenden wir solch einen Prozess auf einen Segmentierungsworkflow an und zeigen die Vorteile gegenüber der traditionellen Methodik.

Abschließend ist EBI ein vielversprechender Ansatz, um nicht verfügbare Daten in der Elektronenmikroskopie zu generieren. Dies wurde auf Basis verschiedener Anwendungen gezeigt und evaluiert. Weitere Anwendungsfälle in der Elektronenmikroskopie werden sich in Zukunft sicherlich ergeben. Adaptive Aufnahmestrategien und EBI müssen unter Berücksichtigung des wechselseitigen Einflusses weiterentwickelt werden. Dies wird eine höhere Durchsatzleistung in der Elektronenmikroskopie unter Verwendung einer geringeren Elektronendosis fördern, ohne dabei die Bildqualität merklich zu verschlechtern. Aufnahmezeiten können darüber hinaus erheblich reduziert werden.

Acknowledgement

> *Don't cry because it's over, smile because it happened.*
>
> — **Ludwig Jacobowski**
> German Lyricist

I would like to express my great thankfulness to my supervisors Prof. Dr. Philipp Slusallek and Dr. Tim Dahmen for the opportunity to do a PhD in the Agents and Simulated Reality group as part of the German Research Centre for Artificial Intelligence. They offered me a very valuable working environment and allowed me a lot of freedom in research.

Most of all, I would like to express my very deep gratitude to Dr. Tim Dahmen for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time so generously had a lot of value. His encouragement and support were invaluable and allowed me to grow both professionally and personally.

I also would like to thank Prof. Dr. Joachim Weickert and Prof. Dr. Michael Breuß for paving the way to start a PhD and their willingness to review this thesis in addition to Prof. Dr. Philipp Slusallek.

I am particularly grateful to Pascal Peter, Tim Dahmen, and Christian Schlinkmann for proofreading my thesis and providing much valuable feedback.

Furthermore, I would like to express my appreciation for many fruitful discussions and support to persons who have not been mentioned above: Pavel Potocek, Dr. Faysal Bourghorbel, Dr. Sabine Schlabach, Dr. Delei Chen, Sviatoslav Bogachev, Prof. Dr. Niels de Jonge, Georg Demme, Stefan Nickels, Dr. Stefan Helfrich, Dr. Daniel Stöckel, Lara Schneider, Tim Kehl, Prof. Dr. Hans-Peter Lenhof, Pia Scherer-Geiss, and Dr. Michelle Carnell.

Not to forget our system administration team around Jennifer Schenkenberger and her Adminions, who always were available if technical support was necessary. They steadily made sure that I could use my time for researching, a big thanks again to all of them.

I would like to thank Thermo Fisher Scientific, the Karlsruhe Institute for Technology, and the Institute of Nanoscopy at the University of Maastricht for providing data that was used in the course of this thesis.

My special thanks are extended to my family. Words can not express my gratefulness to my parents for all their sacrifices that they made on my behalf. Without their permanent support and guidance I would not be where I am today. Once more I thank my late father, who unfortunately did not stay long enough in this world to see his son become a PhD.

I would also like to thank my beloved wife Sabrina for supporting and encouraging me, I can not thank you enough. Not to forget our dog Bertie. You two always cheered me up when my constitution and mood were at the bottom. Without both of you and the many things you sustained me during difficult times I would not be the person that I am today!

Contents

Introduction

2

> *It's much easier to get a reception from someone if there is an introduction versus randomly trying to get in front of people.*
>
> — **Brad Feld**
> (American entrepreneur and author)

In this chapter we give a motivation, why exemplar-based inpainting (EBI) has an impact on electron microscopy (EM). We introduce problems that can be tackled with EBI and briefly discuss previous approaches and their limitations. We show contributions that help to deal with shortcomings of other methods and state the goals of this thesis, including an overview of the structure.

2.1 Background and Motivation

In EM many applications suffer from restrictions that prevent the collection of all data of a present specimen. Such restrictions are for example a limited field of view, technical constraints of the microscope, or time and electron dose. Corresponding reconstructions of the data often contain artefacts. For very small structures, interesting parts may not be visible at all. Time-consuming acquisition schemes prevent imaging at high resolution, and dose sensitive materials do not allow the use of high electron doses, which leads to a low signal-to-noise ratio (SNR) and unsatisfactory image quality. Such shortcomings narrow the collectable amount of data. In this thesis we introduce and evaluate methods to tackle this non-availability of data.

One example application is related to scanning electron microscopy (SEM). SEM plays a central role for analysing the composition of materials and structures by imaging centimetres of a sample at nanometre scales. At such a fine resolution an analysis of samples requires a vast amount of data, which can take months of recording [8]. A conventional recording scheme for SEM is the raster scan, i.e. scanning each grid point on a regular grid. Each pixel location of a sample is visited for a given amount of dwell time per pixel that corresponds to a desired SNR. Consider a typical volume resolution of 4096^3. The corresponding acquisition time via raster scanning of a cube with edge length 0.02 mm and a resolution of 5 nm per pixel takes nearly eight days when the dwell time per pixel is $10\,\mu$s. This example shows that the data acquisition of cubes with an edge length of a millimetre with the same parameters is already infeasible for a raster scan.

A further application is related to electron tomography (ET). ET is one of the main methods for three-dimensional (3D) imaging at the nano-scale. It is extensively used both in structural biology and material science. The resolution is in the range of 1 nm to 20 nm, thereby filling a critical length scale gap for biological applications between atomic resolution of X-ray crystallography and single particle electron tomography [93] on the one hand, and high-resolution confocal light microscopy [117] and X-ray microscopy [175] on the other. In material science, the methods fill the gap between atom probe tomography [135] on the one side and focussed ion beam scanning electron microscopy (FIB-SEM) [42] or X-ray Tomography techniques on the other.

In ET, projections from different directions are acquired by means of transmission electron microscopy (TEM) while tilting the sample. A 3D volume is numerically generated from these projections. Electron tomographic reconstructions often suffer from different restrictions, like opacity, limited field of view, limited number of projections, or missing wedge artefacts. A comprehensive review of computational methods involved in electron tomography studies with all these problems can be found in [92], and an evaluation of accuracy of several reconstruction algorithms with tubular objects as example was conducted in [50].

One of the most severe problems is the non-availability of projections covering the whole angular range. Most electron tomography datasets have a maximum angular range of about –75° to 75°. Main reasons are mechanical limitations of the specimen holder, shading of the area of interest by the support grid, and the increasing projected thickness of the tilted specimen. The restricted angular range leads to elongation artefacts of reconstructed details along the symmetry centre of projections. This is also visible in the Fourier domain as missing wedge, because no data is present in the angular range not imaged during data acquisition, as predicted by the Fourier slice theorem [280]. This so called missing wedge problem introduces prominent elongation artefacts along the beam direction of reconstructed objects [134] leading to an anisotropic spatial resolution [199].

We identified use cases in EM, where data is missing due to technical restrictions or physical limits. The influence of the missing data in subsequent data analysis tasks makes interpretation of the data difficult, e.g. due to very low SNR or artefacts caused by reconstruction algorithms. Increasing the quality of gathered data may be possible, but at the cost of infeasible acquisition times. Scanning a whole human brain, which has an average size of $1,200\,\text{cm}^3$ [6], at an isotropic pixel resolution of 10 nm requires $2.1 \cdot 10^{24}$

pixels to be scanned. A single electron beam that scans each pixel for $30\,\mu$s would take $1.1 \cdot 10^{12}$ years for the whole acquisition. Such acquisition sizes are well justified in brain imaging, as tissue sections need to be 30 nm to 40 nm thick to see cell bodies and axons. For the goal of resolving fine processes, the sections can be only 10 nm to 20 nm thick. Otherwise, thinner branches may be missed, which increases the risk of inaccuracies when reconstructing 3D circuits of connected neurons [172]. Obviously, this is a very high pixel resolution and may not be needed for all kinds of applications, but even for an isotropic pixel size of $1\,\mu$m with a scanning time of $1\,\mu$s per pixel, a single electron beam would take 38 years. These examples demonstrate that a reduction of scanning time is mandatory to be able to scan such big specimens as a human brain within a reasonable timeframe.

Multi-beam approaches already make it possible to scan with 61 beams in parallel, so that $1 \cdot 10^{9}$ pixels can be scanned per second [172] [1]. Splitting the specimen into several chunks and scanning these in parallel is a further option to speed up the acquisition process. However, the equipment needed for such a parallel acquisition is very expensive and only few can afford this. An alternative approach to reduce acquisition times is changing the scanning process and subsequent reconstruction methods. Based on these requirements, we need to develop an algorithm that can reconstruct data based on a small subset of the original data and prior knowledge, e.g. from already available data. For this task it may also be needed to design appropriate sampling schemes that support the reconstruction process.

Most methodologies applied for sparse sampling of EM data are based on ideas from compressed sensing (CS). However, to the best of our knowledge there is no hardware yet, that supports the acquisition of linear combinations of measurements, which is a fundamental requirement in CS. Reconstruction schemes mainly work based on ideas from sparse coding, where convex optimisation methods are applied. Sparse coding relies on basis transformations like wavelets or discrete cosine transforms. These can produce artefacts during reconstruction, which are misleading for subsequent interpretation of the data. Furthermore, measurements are no real linear combinations in SEM as a pixel is either scanned or not. This fact leads to quality worsening perturbations in the reconstruction process.

In this thesis, we propose a heuristic method based on EBI to generate missing data. EBI is a technique used in image processing, e.g. for the restoration of damaged images, which uses available data as a basis for the estimation of non-available data. We also concentrate on speeding up EBI to make it feasible for interactive use in combination with a microscope. Furthermore, we investigate possible approaches to reduce the amount of

---

[1] The newest ZEISS MultiSEM 506 is even capable to scan with 91 beams in parallel (`https://applications.zeiss.com/C125792900358A3F/0/E22B336878B463FCC1257EDC004FFFF1/$FILE/EN_41_011_078_MultiSEM_rel2-4.pdf`).

data that has to be acquired and corresponding sampling approaches to support the algorithmic reconstruction.

2.2 Own Contributions

We evaluate methods for the reconstruction of sparsely sampled data, respectively of data with missing information. We identify different approaches to explore which workflow is most suitable for estimating the missing data. Based on the results, we investigate sparse coding approaches and select one of the best performing methods for further studies.

For the presented use cases inpainting approaches inspired by CS were found to be lacking applicability, which shifts the focus to EBI. Based on the work of Criminisi et al. [58] we modify the original approach, which is intended only for two-dimensional (2D) data, to work on 3D data, such as electron tomography tilt series and SEM. That means, we extend the 2D EBI to work on 3D data using 3D patches, which has not yet been explored to the best of our knowledge.

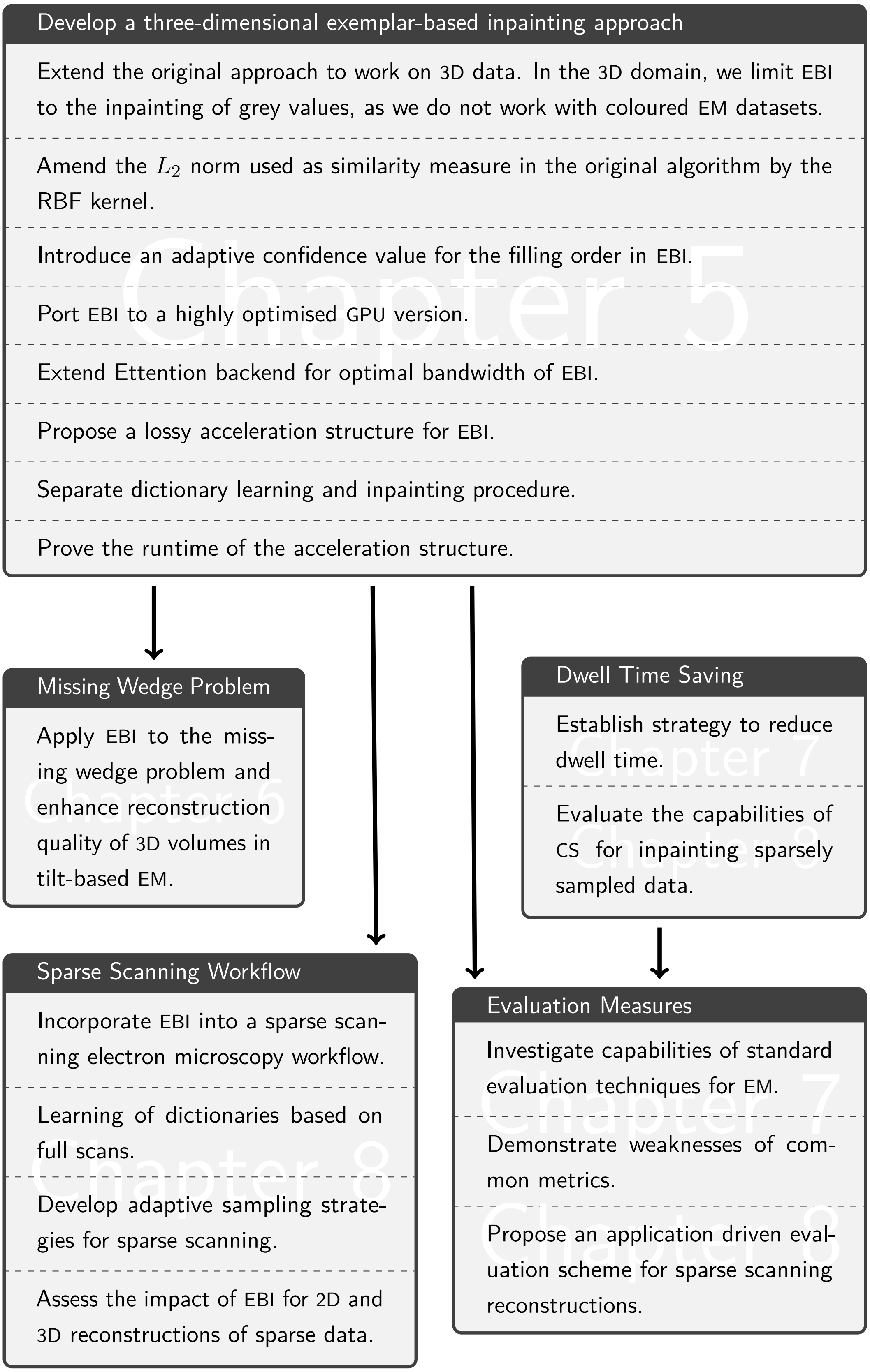
Our in-house developed software for tomographic reconstruction in electron tomography, called Ettention [64], is used as basis for the implementation of 3D EBI. This highly optimised library makes it possible to implement EBI much more efficiently in terms of performance compared to the original Matlab implementation. As brute force EBI is already very slow in 2D, we port the whole algorithm to graphics processing unit (GPU). We extend Ettention with low-level hardware access tailored to a minimal amount of data transfer for optimal bandwidth of EBI. However, the speed-up on GPU may not suffice for very demanding 3D EM datasets. Hence, we develop an acceleration structure based on z-curves that enables for a further performance gain, such that EBI becomes applicable in 3D.

The initial application of EBI in EM is the generation of missing projections in tilt-based tomography. This missing data causes elongation artefacts along the beam direction, which is reflected as a missing wedge in Fourier space. We use EBI to artificially generate the missing projections. The complemented stacks are then used to perform tomographic reconstructions. The impact of artificial projections is afterwards evaluated visually and by means of appropriate evaluation measures.

A second application is the reconstruction of sparsely sampled date in SEM to reduce the required time and electron dose for acquisitions. We develop a workflow for sparse SEM, which incorporates dictionary learning and sparse sampling strategies. A pre-learned dictionary is then used to inpaint the sparsely sampled data. The impact of EBI for 2D and 3D reconstructions is finally investigated.

Common workflows to assess algorithm performance for comparing outcomes against a ground truth were often misleading in the course of our experiments. Hence, besides visual comparison and widely used figures of merit, we propose to use an application driven evaluation scheme. Segmentation of membranes and mitochondria in a mouse brain serves as a proof of concept experiment.

A summary of our contributions is depicted in Figure 2.1.



Develop a three-dimensional exemplar-based inpainting approach

- Extend the original approach to work on 3D data. In the 3D domain, we limit EBI to the inpainting of grey values, as we do not work with coloured EM datasets.
- Amend the L_2 norm used as similarity measure in the original algorithm by the RBF kernel.
- Introduce an adaptive confidence value for the filling order in EBI.
- Port EBI to a highly optimised GPU version.
- Extend Ettention backend for optimal bandwidth of EBI.
- Propose a lossy acceleration structure for EBI.
- Separate dictionary learning and inpainting procedure.
- Prove the runtime of the acceleration structure.

Missing Wedge Problem

- Apply EBI to the missing wedge problem and enhance reconstruction quality of 3D volumes in tilt-based EM.

Dwell Time Saving

- Establish strategy to reduce dwell time.
- Evaluate the capabilities of CS for inpainting sparsely sampled data.

Sparse Scanning Workflow

- Incorporate EBI into a sparse scanning electron microscopy workflow.
- Learning of dictionaries based on full scans.
- Develop adaptive sampling strategies for sparse scanning.
- Assess the impact of EBI for 2D and 3D reconstructions of sparse data.

Evaluation Measures

- Investigate capabilities of standard evaluation techniques for EM.
- Demonstrate weaknesses of common metrics.
- Propose an application driven evaluation scheme for sparse scanning reconstructions.

Fig. 2.1.: Own contributions.

2.3 Thesis Structure

After having introduced problems in EM and the goals we are pursuing above, the further thesis is structured as follows.

Chapter 3 addresses and explains mathematical concepts needed in this thesis. We fix the notations used and explain vector space properties and how images are processed in a mathematical sense. We further concentrate on sampling, especially sparsity and compressibility of signals as well as corresponding reconstructions. Finally, we elaborate evaluation measures that are used in this thesis for assessing algorithm outcomes.

We give an overview of work that is related to this thesis in Chapter 4. We discuss different inpainting approaches and possibilities to speed them up, especially concentrating on EBI. We present a short overview of CS and sparse coding to make clear the difference of the two approaches. For the sake of completeness, we also show deep learning approaches related to EM and the work of this thesis. At the end of the chapter, we present approaches that try to tackle the missing wedge problem, as this is one of the applications we address with the help of EBI.

EBI as core algorithm of this thesis is presented in Chapter 5. An explanation how the selected version from Criminisi et al. [58] works, and modifications to the algorithms that are made to work with 3D EM data are elucidated. Furthermore, ways to accelerate EBI in general are introduced and their impact is demonstrated based on a photo processing case study.

Chapter 6 deals with the application of EBI to tackle the missing wedge problem. A workflow for generating projections heuristically for tilt-based EM is described and evaluated on both artificial and real data. Quality and runtime performance are judged based on the evaluation results. Finally, the applicability and properties of datasets where EBI helps to reduce missing wedge artefacts are discussed.

Different methodologies for an efficient usage of dwell time in EM are evaluated in Chapter 7. Effects of reducing dwell time in combination with super-resolution (SR), sparse coding and interpolation, respectively inpainting, algorithms are analysed to identify an optimal framework for application in EM. In the course of the corresponding evaluation we also show problems of common workflows that are based on figures of merit for assessing reconstructions in EM.

In Chapter 8, selected algorithms that are inspired by CS theory are presented and compared. After having identified sparse coding as optimal strategy for saving dwell time in Chapter 7, the goal of this chapter is to determine which algorithm is most appropriate for EM applications, especially which algorithm can be advantageous for SEM. Furthermore, we propose an application driven evaluation scheme for sparse data reconstruction based on a segmentation problem. Following the findings, we present a sparse scanning workflow, which includes dictionary learning, adaptive sampling, and subsequent reconstruction using EBI.

We conclude with Chapter 9, where we provide a summary of the work performed in the course of this thesis. Concluding about the capabilities of inpainting in EM and possible future work finishes the thesis.

Mathematical Background

3

> *All models are wrong, but some are useful.*
>
> — **George E. P. Box**
> (British statistician)

In this chapter important mathematical concepts that are needed in this thesis are explained. We introduce notations and explain how images are used for computations, followed by vector space properties that are needed for further explanations. Then we concentrate on signal processing and sampling, which includes sparsity and compressibility as well as the concept of a dictionary for the reconstruction of signals. Finally, we give an overview of evaluation measures used to assess the quality of algorithms developed and applied in this thesis.

3.1 Notations

A clean and clear mathematical notation is mandatory to facilitate reading and understanding. We differentiate between different entities by capitalisation and bold face. Scalar values and functions are depicted in lower case and regular font, e.g. $c, \mu \in \mathbb{R}$ or $f : \mathbb{R}^2 \to \mathbb{R}$, vectors in lower case and bold face, e.g. $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, matrices in upper case and bold face, e.g. $\boldsymbol{\Phi}, \boldsymbol{A} \in \mathbb{R}^{n \times m}$, sets in upper case and regular face, e.g. $D \subset \mathbb{R}^n$. Table 3.1 summarises the formatting of mathematical symbols, and Table 3.2 depicts all mathematical symbols and their meaning as used in this thesis.

entity	capitalisation	bold face	examples
scalar value	✗	✗	$c, \mu \in \mathbb{R}$
vector	✗	✓	$\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$
matrix	✓	✓	$\boldsymbol{\Phi}, \boldsymbol{A} \in \mathbb{R}^{n \times m}$
set	✓	✗	$D \subset \mathbb{R}^n$
function	✗	✗	$f : \mathbb{R}^2 \to \mathbb{R}$

Tab. 3.1.: Overview of the notation used in this thesis.

semantic	symbol
mean of $\boldsymbol{x}$	$\mu_{\boldsymbol{x}}$
standard deviation of $\boldsymbol{x}$	$\sigma_{\boldsymbol{x}}$
covariance of $\boldsymbol{x}$ and $\boldsymbol{y}$	$\sigma_{\boldsymbol{xy}}$
$\mathrm{L_p}$ norm of $\boldsymbol{x}$	$\lVert\boldsymbol{x}\rVert_p$
gradient of $\boldsymbol{x}$	$\nabla\boldsymbol{x}$
orthogonal vector to $\boldsymbol{x}$	$\boldsymbol{x}^{\perp}$
transposed vector to $\boldsymbol{x}$	$\boldsymbol{x}^{T}$
regularisation factors	c, λ
logarithm of x	$\log(x)$
exponential function of x	$\exp(x)$
summation symbol / operator	Σ
cardinality of S	$\lvert S\rvert$
image domain / region	$\mathcal{I}$
Inpainting	
patch size (x, y, z)	Ψ
source region	Φ
target region	Ω

Tab. 3.2.: Overview of the symbols used in this thesis and their meaning.

3.2 Vector Spaces

Images in a mathematical sense. In modern image processing signals are often modelled as vectors in an appropriate vector space. For a two-dimensional (2D) image $\mathcal{I} \in \mathbb{R}^{n\times m}$ all rows, or columns, are stacked together via a vectorisation function $v : \mathbb{R}^{n\times m} \to \mathbb{R}^{nm}$ that represents an index transformation, w.l.o.g. we build vectors row-wise. That means the first M elements of the vector are composed of the first row of the image, the next m elements from position $m+1$ to $2m$ comprise the second row and so forth (Figure 3.1). The vectorised images are used in many algorithmic setups, which is the reason we cover some basic background about vector spaces in the following.

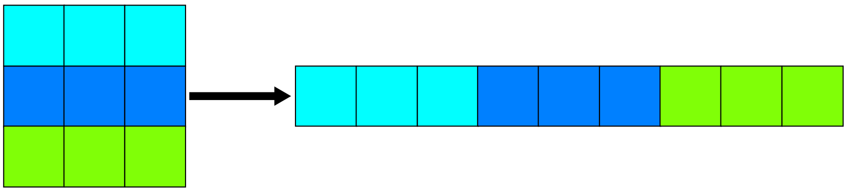


Fig. 3.1.: Stacking together the rows of an image $\mathcal{I} \in \mathbb{R}^{3\times 3}$ via a vectorisation function $v : \mathbb{R}^{3\times 3} \to \mathbb{R}^{9}$, which is a simple index transformation.

In optimisation algorithms a cost function, also called distance measure, is needed. In general, L_p norms are used for this purpose, where $p \geq 1$. For $p \in (0,1)$ L_p is called a quasi-norm. The L_p norm of a vector $\boldsymbol{x}$ is defined as

$$\|\boldsymbol{x}\|_p = \sqrt[p]{\sum_{i=1}^{n} x_i^p} \, . \tag{3.1}$$

The unit spheres of some L_p norms and quasi-norms in 2D are depicted in Figure 3.2 for $p \in \{0.2, 0.5, 0.7, 1, 2, 3, 10, 20, \infty\}$. Many algorithms need to compare a ground truth against a reconstructed version. This is represented by vectors inside the cost function, so that the equation becomes

$$L_p(\boldsymbol{x}, \boldsymbol{y})^n = \sum_{i=1}^{n} (x_i - y_i)^n \tag{3.2}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are two quantities that are compared, e.g. images or signals in general. The square root can be omitted as it is a monotone function and does not change the order of values. In this thesis we predominantly use L_2, which is also called sum of squared differences (SSD).

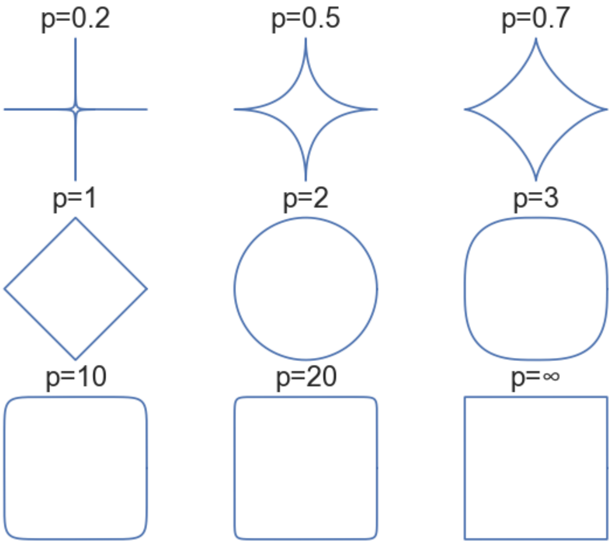


Fig. 3.2.: Unit spheres in $\mathbb{R}^2$ for the L_p quasi-norms with $p \in \{0.2, .0.5, 0.7\}$ and for the L_p norms with $p \in \{1, 2, 3, 10, 20, \infty\}$.

3.3 Nyquist-Shannon Theorem

Signal processing, which also incorporates image processing, underlies one central theorem for analogue to digital signal conversion, the Nyquist-Shannon sampling theorem [223]. Its main statement is that the bandwidth w determines the number of required samples that are needed to reconstruct a signal without error. That means the shortest interval W that contains the support of the spectrum from a signal, must fulfil the theorem for unique and error-free acquisition of an analogue signal in the time domain. Shannon's original theorem states: "If a function $f(t)$ contains no frequencies higher than W, it is completely determined by giving its ordinates at a series of points spaced $\frac{1}{2w}$ apart." [129] With other words the theorem states that the sampling rate must be at least twice the highest frequency in the signal to be reconstructed without an error, provided an equidistant sampling rate. This sampling rate is also called Nyquist rate.

Sampling in the image domain. The Nyquist-Shannon theorem has consequences for continuous signals $f : D \to \mathbb{R}$ on a rectangular grid D that are sampled into a discrete image domain $\mathcal{I} \in \mathbb{R}^{n \times m}$ (Figure 3.3). The sampling density, which is constituted by the step size on a regular grid, is responsible for the maximally representable information in terms of resolution. This means the step size must be smaller than the smallest structure that should be visible, so that no important information is lost due to under-sampling as stated by the Nyquist-Shannon theorem. In the field of electron microscopy (EM), this is not always possible because of technical restrictions and physical limits, which makes a sparse acquisition of data even more challenging.

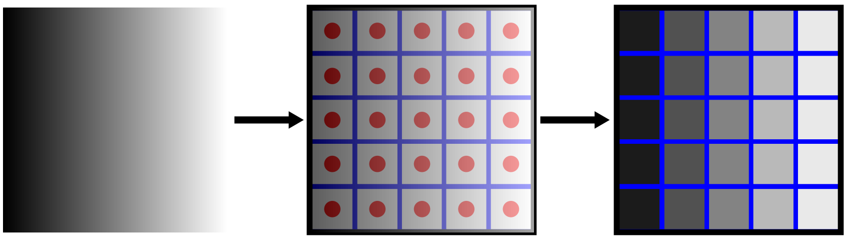


Fig. 3.3.: A continuous signal $f : D \to \mathbb{R}$ on a rectangular domain D is sampled at the red dots of a regular grid, which results in a discrete image $\mathcal{I} \in \mathbb{R}^{5 \times 5}$.

3.4 Sparsity and Compressibility of Signals

Opposing the Nyquist-Shannon sampling theorem an alternative theory was put forth at the beginning of the millennium, called compressed sensing (CS) or compressive sampling [32, 75]. The theory shows that under strict assumptions far less samples than typically considered necessary are required to reconstruct a signal without error. In this section, we give a short summary on the theory, especially covering sparsity and compressibility of signals.

Sparse representation means that a signal of length n can be represented with $s \ll n$ non-zero coefficients, whereas a compressible representation is an approximation of a signal with s non-zero coefficients that comes close to the original. Preserving values and locations of the largest coefficients of a transformed signal suffices to represent a compressible signal with low error, which is also the basis for standards like JPEG [261] or MP3 [30].

The connection between CS theory and sparse/compressible signals is, that instead of sampling at high rates and compressing the acquired data it would be beneficial to "sense" the data in a compressed form, which reduces sampling rates and makes compression redundant. Indeed, this is possible, which has been shown in several publications [17, 32, 36–38, 40, 75]. However, establishing measurement schemes and extending these to practical applications via meaningful acquisition systems is not straight forward and remains a central point in CS research. The main reasons are the assumptions and properties that must hold to ensure that CS works, which we clarify shortly in the following.

There are three main differences between classical sampling approaches and CS:

1. Traditional methods are mostly based on infinite-length, continuous-time signals. CS is focused on finite-dimensional signals in $\mathbb{R}^n$, e.g. images that are transformed into a vector.
2. Traditional signal theory samples signals at specific points in time, whereas in CS measurements are inner products between signals and more general functions, e.g. random sampling. That means CS measurements are linear combinations of several distinct measurement points.
3. The reconstruction of a complete signal underlies different algorithmic procedures. Approaches based on the Nyquist-Shannon theory use sinc interpolation, whereas CS methods rely on highly non-linear methods [253]. It has also been shown that such non-linear reconstruction methods can be applied when non-linear and non-ideal sampling processes are present [82]. An example is an acquisition system where a sensor introduces static non-linearity before the signal is sampled by an analogue-to-digital converter.

Restricted Isometry Property. CS relies on properties of the sensing matrix $\mathbf{\Phi}$, which is a pattern that determines how data is acquired, e.g. which pixels are measured and which not. Suppose we want to reconstruct an object $\boldsymbol{x} \in \mathbb{R}^n$ while measurements $\boldsymbol{y} \in \mathbb{R}^m$ and a known sensing matrix $\mathbf{\Phi} \in \mathbb{R}^{m \times n}$ are available. To solve the under-determined system of equations $\boldsymbol{y} = \mathbf{\Phi}\boldsymbol{x}$, a second assumption is required to make the ill-posed problem's solution feasible. This assumption is that $\boldsymbol{x}$ is sparse, i.e. $\boldsymbol{x}$ only depends on a small number of coefficients. It has been shown in [39] that $\boldsymbol{x}$ can be recovered exactly provided that (i) $\boldsymbol{x}$ is sufficiently sparse, and (ii) $\boldsymbol{\Phi}$ obeys the restricted isometry property (RIP) by solving

$$\underset{\tilde{\boldsymbol{x}} \in \mathbb{R}^n}{\arg\min} \|\tilde{\boldsymbol{x}}\|_1 \qquad s.t. \qquad \boldsymbol{\Phi}\tilde{\boldsymbol{x}} = \boldsymbol{y} \, . \tag{3.3}$$

Definition 3.4.1. Restricted Isometry Property. A matrix $\boldsymbol{\Phi}$ satisfies the s-restricted isometry property if there exists a restricted isometry constant δ_s such that

$$(1 - \delta_s) \|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{\Phi x}\|_2^2 \leq (1 + \delta_s) \|\boldsymbol{x}\|_2^2 \tag{3.4}$$

holds for every integer $s \in \{1, 2, \dots\}$ and for every s-sparse vector $\boldsymbol{x_s}$. A vector is called s-sparse if it has at most s non-zero entries.

For non-sparse vectors $\boldsymbol{x} \in \mathbb{R}^n$ we can approximate the solution with the best sparse solution by keeping the s-largest entries of $\boldsymbol{x}$ and setting all other entries to 0, denoted by $\boldsymbol{x_s}$. Assume that $\delta_{2s} < \sqrt{2} - 1$. Then the following inequalities hold for the solution $\tilde{\boldsymbol{x}}$ of Equation 3.3 in a noiseless recovery:

$$\|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_1 \leq c_0 \|\boldsymbol{x} - \boldsymbol{x_s}\|_1 \qquad \text{and} \qquad \|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq c_0 \frac{1}{\sqrt{s}} \|\boldsymbol{x} - \boldsymbol{x_s}\|_1 \tag{3.5}$$

for some $c_0 \in \mathbb{R}$. This equation states that the recovery is exact if $\boldsymbol{x}$ is s-sparse. Further considerations and corresponding proofs can be found in [33].

Mutual Coherence. The sensing matrix $\boldsymbol{\Phi}$ must also fulfil the property that all columns are uncorrelated. Measuring this property can be performed with the so called mutual coherence [76].

Definition 3.4.2. Mutual Coherence. Let $\boldsymbol{a_1}, \dots, \boldsymbol{a_m} \in \mathbb{C}^n$ be the normalised columns of the matrix $\boldsymbol{A}$, such that $\boldsymbol{a_i}^T \boldsymbol{a_i} = 1$. The mutual coherence of $\boldsymbol{A}$ is defined as

$$\text{MC} = \max_{1 \leq i,j \leq m} \left| \boldsymbol{a_i}^T \boldsymbol{a_j} \right| \tag{3.6}$$

According to [267], a lower bound for MC is

$$\text{MC} \geq \sqrt{\frac{m - n}{n(m - 1)}} \tag{3.7}$$

In case there are highly correlated columns in $\boldsymbol{\Phi}$, it is not possible to distinguish which of the columns explains a measurement, so that a sparse signal cannot be reconstructed exactly. However, in [35] it was shown that there are situations where signals are not sparse in an orthonormal basis, but in a truly redundant dictionary, which may also suffice to guarantee exact recovery.

3.5 Dictionary as Learned Basis

A dictionary, as used in this thesis, is always some prior knowledge that has been determined or learned in advance of any application in an algorithm. There are two kinds of dictionaries we use. In the first case a patch dictionary is a fixed set of possible image patches, where each patch is a point in a high-dimensional space. That means, for a fixed patch size $\boldsymbol{\Psi} \in \mathbb{R}^{\Psi_x \times \Psi_y}$, a corresponding entry in the dictionary is a point $\boldsymbol{p} \in \mathbb{R}^{\Psi_x \Psi_y}$. In the second case a basis dictionary is built of dictionary atoms that form a basis, as known from linear algebra, or a frame, which is a generalisation of the concept of a basis to linear dependent sets, also called overcomplete basis. In this case, image patches that should be inserted in the course of an algorithm application are linear combinations of all basis elements of the dictionary. In this section we concentrate only on the basis dictionaries, as patch dictionaries are a straight forward concept, where only possible image patches are stored in a dictionary. From now on we only differentiate between the two dictionary versions if the selection becomes unclear from the context.

The straight forward choice for a dictionary consists of selecting any already proposed dictionary that fits to the application domain. Examples are wavelets [187], steerable wavelets [258], curvelets [34], shearlets [140], countourlets [74], discrete cosine transform basis functions [205], and many more. Most of these dictionaries are tailored to images establishing sparsity of the representation coefficients. Also there are tunable dictionaries, which are controlled based on one or more parameters, like bandlets [169] or wavelet packets [53].

A disadvantage of such pre-constructed dictionaries is their lack of signal sparsification, meaning they are in general only well suited for the domain they were designed for. Here, dictionary learning can help to overcome such limitations. Based on some training dataset coming from the domain that should be reconstructed afterwards, the basis elements can be learned from the statistics of the data.

Let $\boldsymbol{D} \in \mathbb{R}^{m \times m}$ be a dictionary and $\boldsymbol{y_i} \in \mathbb{R}^m, i \in \{1, \ldots, n\}$ the training signals coming from some fixed, but unknown model with a model deviation of ϵ. Olshausen et al. [182] have shown that it is possible to identify the generating model, which means it is possible to compute $\boldsymbol{D}$. At the same time the sparsest representation $\boldsymbol{x_i}$ over the unknown dictionary $\boldsymbol{D}$ of the corresponding signal $\boldsymbol{y_i} \, \forall i \in \{1, \ldots, n\}$ is computed.

Further extensions can be found in [88, 150]. The following optimisation problem has to be solved for computing $\boldsymbol{D}$:

$$\underset{\boldsymbol{D},\boldsymbol{x_i}}{\arg\min} \sum_{i=1}^{n} \|\boldsymbol{x_i}\|_0 \qquad s.t. \;\; \|\boldsymbol{y_i} - \boldsymbol{D}\boldsymbol{x_i}\|_2 \leq \epsilon, \; i \in \{1, \dots, n\} \tag{3.8}$$

Solving Equation 3.8 delivers a proper representation of $\boldsymbol{x}$ as well as the dictionary $\boldsymbol{D}$ at the same time. Aharon et al. [3] have shown that for $\epsilon = 0$ there exists a unique dictionary under some constraints for the sparsity of the representation. There are many algorithms to solve this optimisation problem, e.g. K-SVD [2], the method of optimal directions [87], stochastic gradient descent [1], or an online learning approach [167].

From the above, it becomes evident that there is a clear relation between sparsity and the value of p of the L_p norm. Figure 3.2 shows that for smaller p the corresponding search space becomes sparser, which is the reason that Equation 3.8 uses $p = 0$, even if in practise it is infeasible to compute the L_0 norm in high-dimensional spaces. As computing L_p for $p < 1$ is NP-hard [99], the L_1 norm is used as approximation in CS optimisation settings [201].

3.6 Evaluation Measures for Image Quality

Image processing algorithms are evaluated both qualitatively, by comparing images and image structures visually, and quantitatively, by computing figures of merit, or evaluation metrics, that measure the similarity between a reconstruction and the corresponding ground truth. In this thesis we use several figures of merit, which are introduced in this section.

Let $\boldsymbol{x} \in \mathbb{R}^n$ be ground truth data and $\boldsymbol{y} \in \mathbb{R}^n$ estimated, respectively reconstructed, data. Each image dataset, which can be 2D images or three-dimensional (3D) image stacks, consists of n pixels, respectively voxels.

A standard evaluation metric for the comparison of two datasets is the mean squared error (MSE) [210] defined as

$$\text{MSE}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{n} \cdot \sum_{i=1}^{n} (x_i - y_i)^2 \quad \in [0, \infty) \, , \tag{3.9}$$

respectively the root mean squared error (RMSE)

$$\text{RMSE}(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\text{MSE}(\boldsymbol{x}, \boldsymbol{y})} \quad \in [0, \infty) \tag{3.10}$$

where x_i are the values of the ground truth data and y_i are the values of the estimated data.

In electron tomography acquired datasets are in general quite noisy. As a result of that signal-to-noise ratio (SNR) [210] is a useful indicator for the quality of estimated data compared to ground truth data. SNR hereby is defined as

$$\mathsf{SNR}(\boldsymbol{x},\boldsymbol{y}) = 20 \cdot \frac{\frac{1}{n}\sum_{i=1}^{n} x_i^2}{\mathsf{RMSE}(\boldsymbol{x},\boldsymbol{y})} \quad \in [0,\infty) \tag{3.11}$$

Note, that SNR can also be computed for images where no ground truth is available by

$$\mathsf{SNR}(\boldsymbol{y}) = \frac{\mu_{\mathsf{signal}}}{\sigma_{\mathsf{noise}}} \quad \in [0,\infty) \tag{3.12}$$

where μ_{signal} is the mean of the image signal $\boldsymbol{y}$ and σ_{noise} is the standard deviation of the noise contained in $\boldsymbol{y}$.

Rather than SNR, an adapted version that normalises for the dynamic range of images is used as image quality metric, the peak signal-to-noise ratio (PSNR) [210] defined as:

$$\mathsf{PSNR}(\boldsymbol{x},\boldsymbol{y}) = 20 \cdot \log_{10}(\mathcal{I}_{\mathsf{max}}) - 10 \cdot \log_{10}(\mathsf{MSE}(\boldsymbol{x},\boldsymbol{y})) \quad \in [0,\infty) \tag{3.13}$$

where $\mathcal{I}_{\mathsf{max}}$ is the dynamic range of the pixel values, that means $2^{\#bits} - 1$, which is the maximal possible grey value of a given image, e.g. for an 8 bit image it is 255. PSNR measures the deviation of a reconstructed image from its ground truth by relating signal power and corrupting noise. A higher reconstruction quality results in higher PSNR values. Adapted versions of PSNR especially tailored to specific situations like the improved SNR [262] were proposed for usage in electron tomography (ET). However, we use PSNR as it is widely applied and accepted in the image processing community for evaluation tasks.

The presented measures so far all have the disadvantage that they do not capture the quality of images as perceived by humans [123]. Wang et al. [264] have shown examples of distortions that led to similar or equal MSE, where the images look very different. As a consequence, many evaluation measures that serve the purpose of incorporating the human visual system have been proposed [81, 196, 264]. Furthermore, there are approaches based on supervised learning in the context of convolutional neural networks [28, 203, 224].

Currently, the structural similarity index (SSIM) is one of the most frequently used metrics with nearly 20,000 citations at the time of writing. Due to its high acceptance in the image processing community and the proven usefulness in applications [158, 212, 265], we use SSIM, defined as

$$\text{SSIM}(\boldsymbol{x}, \boldsymbol{y}) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad \in [0, 1] \tag{3.14}$$

Here, μ_x and μ_y are mean values of ground truth data $\boldsymbol{x}$ and estimated data $\boldsymbol{y}$, σ_x and σ_y are corresponding standard deviations, and σ_{xy} is the covariance of $\boldsymbol{x}$ and $\boldsymbol{y}$, and $c_1 = 0.01 \cdot \mathcal{I}_{\text{max}}$ and $c_2 = 0.03 \cdot \mathcal{I}_{\text{max}}$ are two variables to stabilise the division with weak denominator. SSIM relies on the assumption that high-quality is perceived by the human visual system (HVS) as structural consistency, i.e. the dependencies of neighbouring pixels. Hence, SSIM can be regarded as a measure for the loss of image structure, that means the closer SSIM is to 1, the more similar is the structure of the two compared images. In this thesis, SSIM is computed by an averaged sliding window approach. That means SSIM values are computed for every image patch of size 11×11. All values computed this way are averaged. This approach captures local structural similarities more reliably than computing SSIM for the whole image as a global measure. All selected parameters for computing SSIM are the default values proposed in [264], as our intention was not to evaluate SSIM parameters.

During our research presented in this thesis we could confirm that PSNR and SSIM do not always correspond to the visual quality of final images in a reliable way, compare [77, 263]. Therefore, we additionally used two further measures in some of the experiments. The complex wavelet structural similarity index (CW-SSIM) [211]

$$\text{CW-SSIM}(\boldsymbol{x}, \boldsymbol{y}) = \frac{2 \cdot \left|\sum_{i=1}^{n} c_{x,i} c_{y,i}^*\right| + k}{\sum_{i=1}^{n} \left|c_{x,i}\right|^2 + \sum_{i=1}^{n} \left|c_{y,i}\right|^2 + k} \quad \in [0, 1] \tag{3.15}$$

where $c_{x,i}$ and $c_{y,i}$ are complex wavelet coefficients of the images to be compared extracted at the same spatial location in the same wavelet subbands, c^* is the complex conjugate of c, and k is a small positive constant for stabilisation. Analogously to SSIM, the measure is computed for image patches with a sliding window approach, and the computed values are averaged afterwards. A value of 1 implies no structural distortion, even if small spatial shifts are present. Hence, CW-SSIM is insensitive to geometric distortions that do not influence structure, which is an important property for some of the applications in this thesis.

Further, we included PSNR-HVS-M [194] for some experiments. PSNR-HVS-M $\in [0, \infty)$ is essentially the PSNR adapted to the HVS.

$$\text{PSNR}_{\text{HVS}}(\boldsymbol{x}, \boldsymbol{y}) = 20 \cdot \log_{10}(\mathcal{I}_{\text{max}}) - 10 \cdot \log_{10}(\text{MSE}_{\text{HVS}}(\boldsymbol{x}, \boldsymbol{y})) \tag{3.16}$$

where

$$\mathsf{MSE}_{\mathsf{HVS}}(\boldsymbol{x}_{\mathsf{DCT}}, \boldsymbol{y}_{\mathsf{DCT}}) = k \cdot \sum_{i=1}^{64}((x_i - y_i)t_i)^2 \tag{3.17}$$

$\mathcal{I}_{\mathsf{max}}$ is the dynamic range as above, k is a normalisation factor, and $\boldsymbol{x}_{\mathsf{DCT}}$ and $\boldsymbol{y}_{\mathsf{DCT}}$ are vectorised discrete cosine transform coefficients of the corresponding image blocks. The values of the correcting factors t can be found in [85]. Additionally, PSNR-HVS-M uses a between-coefficient contrast masking of the discrete cosine transform basis functions to reduce the masking effects of edges by proportionally reducing local variances. The whole computation can be found in [194]. These extensions enhance the capabilities of the metric to capture the visual appearance to the HVS, which has been shown in [193] by conducting subjective experiments.

We want to emphasise that not only quantitative measures should be used for judging the performance of algorithms. Quantitative measures can be very misleading, which has been demonstrated by Wang and Bovik [263]. Hence, visual comparisons of the images as well as important properties like line profiles should be incorporated into an evaluation additionally. Application driven evaluations may further facilitate the assessment of performance, as will be exemplified later in Section 8.2.

For ET data a way to measure resolution for assessing the reconstruction quality is needed. We estimate the resolution with help of the full-width at half-maximum (FWHM) of the point spread function at different positions in a reconstructed volume, because FWHM is a good approximation of the possible resolution [60]. Higher resolutions are reflected by smaller FWHM values. FWHM of a function f that has an extremum is the difference of the two argument values that have half the value of the extremum:

$$\mathsf{FWHM}(f) = |x_1 - x_2| \qquad \text{s.t. } f(x_1) = f(x_2) = \frac{1}{2} \cdot f(x_{\mathsf{extremum}}) \tag{3.18}$$

where $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ is a function, e.g. line profiles through particles as in Figure 3.4. Before estimating FWHM, the noise level of reconstructed volumes is normalised to a mean of 0 and a standard deviation of 1 based on image background. Half the value of the extremum is selected as half the maximum of the line profile, as normalisation leads to a value of 0 as baseline for the measurements. Both half the maximum and the width at half the maximum are linearly interpolated based on enclosing pixels, which is needed when working with discrete data.

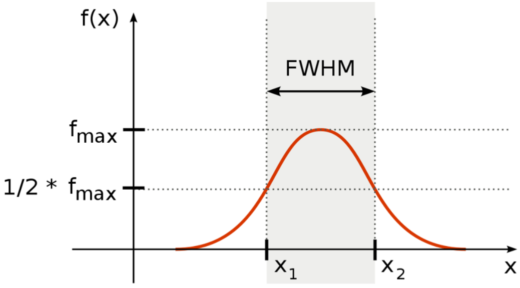


Fig. 3.4.: full-width at half-maximum illustration.

4

Related Work

> *I'm reverent toward my sources. History is a team sport, and references are how you support your teammates.*
>
> — **Rick Perlstein**
> (American historican)

In this chapter we give an overview of existing work that is related to the contents of this thesis. As the main ingredient for the applications, inpainting is covered first with a focus on exemplar-based inpainting (EBI) and performance enhancement. Then, sparse coding applications that can also be applied for the reconstruction, respectively inpainting, of microscopic data are covered. Lastly, we focus on the so called missing wedge problem and solutions that have been proposed to cope with this problem.

4.1 Inpainting Techniques

The term inpainting refers to the reconstruction of missing parts of an image. The most common applications for inpainting are the restoration of damaged photos, the synthetic filling of holes in images where unwanted parts have been deleted, or texture synthesis. Examples are shown in Figure 4.1.

The first time inpainting appeared in literature was 1998, called disocclusion at that time [173]. Disocclusion was performed by means of a level lines structure, which according to the authors offers a reliable, complete, and contrast-invariant representation of the image. The term inpainting instead of disocclusion was created two years later by Bertalmio et al. [23].

There are several kinds of inpainting methods, each based on different assumptions and theories. Most common approaches are patch based, such as EBI covered in Section 4.2, inpainting based on diffusion utilising partial differential equations (PDEs) [45–47, 149, 226], inpainting methods inspired by compressed sensing (CS) that assume image sparsity priors [86, 109, 110, 153, 235], or mixtures of the above mentioned [12]. Furthermore, the recent rise of deep learning has led to an increasing number of inpainting methods

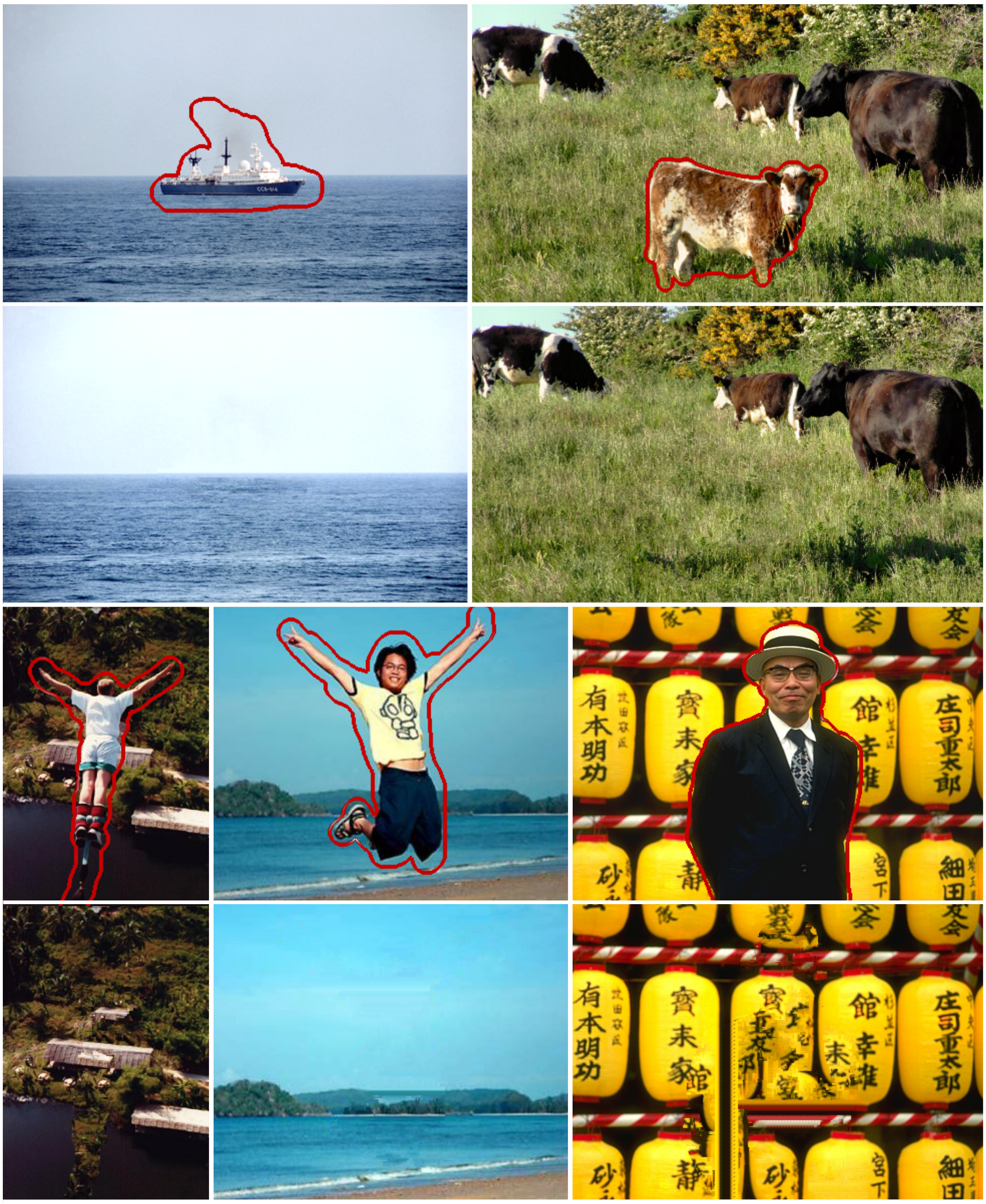


Fig. 4.1.: Examples for exemplar-based inpainting. Each image in the 1^{st} and 3^{rd} row contains an object that should be deleted, marked with a red border. The 2^{nd} and 4^{th} row shows the inpainted results after having removed the objects. Images from: `https://github.com/AmirooR/CriminisiInpaint`, `https://github.com/PetterS/patch-inpainting`, `http://vgl-ait.org/cvwiki/doku.php?id=opencv:tutorial:object_removal_by_exemplar-based_image_inpainting`.

based on convolutional neural networks [125, 151, 186, 257, 274]. The area of application for inpainting methods varies across photo processing for reconstructing damaged areas of an image, image and video compression, reconstruction of missing data and many more.

Stitching together small patches of existing images was the basis of some early inpainting algorithms [83, 84] and was used to generate novel visual appearances. This so called image quilting was used for fast and simple texture synthesis and for texture transfer. Modern deep learning approaches for the same task, termed as style transfer, have been developed for images [98, 130, 272] and videos [206]. These neural network based approaches are typically superior to traditional methods.

One of the first variational approaches was based on joint interpolation of image grey levels and gradient directions [16]. Solving the variational model with the help of its gradient descent flow results in coupled second order PDEs that are used for inpainting, especially for the restoration of old photographs and removal of text in images.

The inpainting technique is typically applied to two-dimensional (2D) images such as photos or digital micrographs, but can also be applied to three-dimensional (3D) datasets such as videos or tomograms [243]. The most common applications for inpainting are the restoration of damaged photos and the synthetic filling of holes in images where unwanted parts have been deleted. The basic principle is quite versatile and can be applied to less common domains as well. In inpainting-based super-resolution (SR) algorithms, the resolution of an image is enhanced a-posteriori by synthetically generating additional pixels using inpainting methods [145]. In sparse scanning microscopy, only a subset of the pixels of a digital micrograph is acquired and missing parts of the image are generated synthetically using inpainting algorithms [230].

Reconstruction of missing data is highly related to our work. Considered applications in electron microscopy (EM) share requirements also present in a compression setup. In image processing, a lot of work regarding compression utilising inpainting techniques has already been conducted. The quality of various PDE based inpainting operators for compression has been evaluated in [217]. The authors were able to beat the quality of JPEG 2000 using edge enhancing diffusion employing a specific subdivision on rectangles. The choice of interpolation points thereby is an important ingredient, which has also been shown in [96]. Furthermore, PDE based inpainting has also been used for 3D data, e.g. for video compression [9]. For inpainting based compression methods, sparse image representations and optimisation methods to find optimal interpolation points play an important role [120, 165, 190, 218]. However, computing optimal points for compression underlies the assumption that a ground truth image is available. In EM this is not the case, so that compression algorithms cannot be used. Nevertheless, ideas from that domain can be transferred and modified, especially using EBI for the reconstruction of incomplete data.

4.2 Exemplar-based Inpainting

Most relevant to the work of this thesis are algorithms from the class of EBI. Besides the texture synthesis approach from [84], EBI was first proposed in [23], even if this naming was used for the first time three years later [57]. Since then much research has been conducted on EBI [12, 14, 31, 43, 49, 58, 69, 89, 90, 142, 145, 156, 157, 171, 191, 197, 225, 270, 271, 278, 283, 284], as well as a patent on EBI [59]. EBI works iteratively on small, often rectangular, respectively cubic in the 3D case, parts of the image, called patches. In a first step, a target patch is selected for processing. The target patch is always selected at the border of the missing region of the image, such that it overlaps both the missing region of the image and a region of the image where pixel values are known. The selection is performed heuristically, whereby the applied heuristic has a crucial impact on the quality of the inpainting result. The work of Criminisi et al. uses a heuristic that aims to continue linear structures [58].

The approach has been extended to also detect incomplete salient structures, which incorporates curvature [152]. These structures are inpainted first to guarantee that the most important edges of an image are connected correctly. The remaining sub-regions are then inpainted using the approach from Criminisi and co-workers. Instead of detecting structures automatically, a user can also interactively specify salient structures as input [237]. Applying belief propagation, specified structures are reconstructed first, followed by the remaining missing parts. The concept of structure sparsity of a patch was introduced in [273]. Structure sparsity is a measure for the sparseness of the non-zero similarities to neighbouring patches. Patches with higher structure sparseness are filled first. In [43], an inpainting order is generated by interpolating level lines with Euler spirals. In [278] a filling order based on colour values was proposed.

Once a target patch has been selected, the algorithm searches known parts of the image, called the dictionary, for a suitable patch to be inserted into the missing area. The selection is performed using a cost function, for example the L_2 norm between known pixels of the target patch and corresponding pixels of a patch in the dictionary. The cost function is another degree of freedom that allows tailoring an inpainting algorithm to different situations. The L_2 norm can be weighted with the Hellinger distance [171], replaced with the structural similarity index [180], extended to a bidirectional similarity measure that better captures non-stationary visual data [229], or converted so that it is equivalent to a Lagrangian relaxation of a strictly guided reconstruction [43]. Rather than inpainting individual patches, a linear combination of several most similar patches, whose coefficients are computed by a constrained optimisation problem, can be used [23]. A completely different approach is a variational framework based on PDEs to address EBI [11]. Ibarrola et al. [124] proposed a two-step procedure. First, curvature-driven diffusion inpainting is used to build a pilot image that is used to infer a-priori structural information on the image gradient. Second, the image is inpainted by minimising a mixed spatially variant anisotropic functional that uses the pilot image.

No matter which cost function is used, the algorithm always selects the patch from the dictionary that evaluates to the global minimum of the cost function value relative to the selected target patch. Typically, this selection is performed by means of a full scan, i.e. the cost function is evaluated for every patch in the dictionary and the patch corresponding to the minimum cost function value is selected. A dictionary can be huge, with about 10^6 entries in the 2D case and more than 10^8 in the 3D case if each dimension has a resolution of $1,024$, such that the full scan becomes computationally expensive and is the performance limiting factor of the algorithm.

4.2.1 Accelerating Exemplar-based Inpainting

Different solutions have been proposed to improve the runtime performance of EBI, including probabilistic methods and local search strategies. Sangeetha et al. [213] have proposed a cost function that includes a diameter of the surrounding region of the patch to be inpainted. Depending on this diameter, only the local area around the patch is taken into consideration for finding a patch to be inserted. Ruzic et al. [207, 208] used contextual features to guide the search for image patches. The image is divided into non-overlapping blocks that are used to reduce the search space for each patch to be inpainted, which also reduces the computation time. Alilou et al. [5] introduced a new search strategy that does not need to calculate the distance of all pixels contained in a patch. Combined with an early stopping criterion this reduced computation time by a factor of two to five. Kwok et al. [141] proposed to decompose patches into frequency coefficients. Using only the most significant coefficients to compute matching scores they were able to reduce computation time by a factor of around 11. He et al. [115] used offset statistics to obtain a subset of the source region as the optimal search area for a patch to be inserted. Their method performs two to five times faster than a full scan.

Currently, Patch Match is one of the state of the art EBI methods [18, 19, 159], especially in terms of computation time. Patch Match uses a randomised algorithm to quickly find approximate nearest-neighbour matches between image patches that are then used to iteratively inpaint the image. The speed-up of 20 to 100 times against previous state of the art algorithms has even made it possible to use Patch Match in real-time interactive editing tools, which is its main application in Adobe Photoshop for 2D image inpainting based purely on the image to be inpainted.

Patch Match works well on natural images as it based on the assumption that pixel grey values in a local neighbourhood have a high correlation. In EM this assumption cannot be guaranteed, especially when acquiring images very fast, as this leads to highly noisy images with little signal of the sampled specimens. Differently from EBI, even if Patch Match uses whole patches for distance computations, in each step only one pixel is inpainted. EM data is highly textured, hence, the aim is to inpaint textures, which is inherent to EBI approaches. The non-locality of patch-based approaches supports this goal, whereas it is questionable if Patch Match's local neighbourhood assumption helps here.

Furthermore, the available implementation from Barnes et. al [18] is not applicable to inpainting problems where data is missing. The implementation only supports the reconstruction of an image using all possible patches from another image that serves as dictionary. Unfortunately, we were not able to run a real inpainting based on the available software, and the inpainting integrated into Adobe Photoshop is not open source. Based on the differing assumptions of natural images and EM images, we did not take Patch Match into consideration for further experiments.

Our ideas for reducing the runtime of EBI were influenced by several other concepts. The concept to consider patches as data points in a high-dimensional space was first developed by Buyssens et. al. [31]. However, the work does not apply techniques from high-dimensional point search, but instead uses a search-window based strategy. The concept to order patches in a one-dimensional array according to some similarity-measure was first mentioned in [200], but used for the purpose of filtering, not searching. The idea of building an index structure over such data points was first proposed in [266]. However, the work used tree structured vector quantisation as the search algorithm and was applied to texture synthesis, not inpainting.

4.3 Sampling in Electron Microscopy

Ideas from compression for the selection of optimal data points may be interesting for adaptive sampling strategies in EM. A common compression setting aims to reduce the amount of data needed to represent some input data. A probabilistic sparsification approach was used in [120, 166]. Starting at the complete image, a fraction of pixels is removed at random. The sparser image is then inpainted and those removed pixels that result in the largest mean squared error (MSE) compared to the original are put back into the image. This procedure is repeated iteratively until a selected percentage of pixels has been removed. In EM there is no original image, hence, the opposite way of working is needed going from sparse to dense. Probabilistic densification [121] seems more suited for our tasks. The algorithm starts with an empty set of pixels. Several random pixel sets are added. For each set an inpainting algorithm is applied and the MSE with the original is computed. The pixel set resulting in the largest error is kept, all other sets are discarded. This procedure is repeated until a predetermined percentage of pixels has been added, which then resembles the sparsely sampled data. Again, this sampling strategy assumes that the ground truth image is available. An extended probabilistic densification was used to optimise data for EBI [132].

The presented approaches are very inspiring for developing adaptive sampling strategies for EM, even if not directly applicable. Identifying important pixels for a subsequent reconstruction of sparsely acquired EM data is key for high quality reconstructions. Currently, most sampling methods are not adaptive in an iterative way of working. The common approach is to randomly sample a fixed percentage of pixels that are used for CS inspired reconstructions using sparse coding.

4.4 Compressed Sensing

The application of compressed sensing (CS) methods is growing in many fields. In medical imaging, magnetic resonance (MR) tomography can benefit from CS. Lustig et al. [162–164] used randomly perturbed spirals to approximate random sampling in MR angiography. 3D Cartesian trajectories, respectively phase encodes, of blood vessel data were undersampled. The acquired sparse data was reconstructed using L_1 total variation minimisation (TVM). With this procedure the authors were able to reduce scan times by a factor of five with a similar quality compared to full scanning.

Trzasko et al. [254] proposed a generalisation of CS theory based on homotopic approximations of the lscr$_0$ quasi-norm. They showed how their results can be used to reconstruct clinical MR images with sampling rates far below the Nyquist limit and closer to theoretical bounds.

Vasanawala et al. [259] developed a method that combines parallel imaging and CS to enable faster MR imaging with high spatial resolution. The authors applied the method in a pediatric clinical setting to show its feasibility. Cardiovascular, abdominal and knee MR scans were acquired with a 3D gradient-echo sequence at high acceleration factors. Obtained sparse data was reconstructed using L_1 iterative self-consistent parallel imaging reconstruction, also called SPIR-iT, which resulted in a speed-up factor of four.

Utilizing CS theory, sub-Nyquist sampling systems have been built. Gedalyahu et al. [101] presented a sampling theory for analogue signals defined over an infinite union of subspaces. With the method it is possible to perfectly recover multipath delays from samples of single channel outputs at the lowest possible rate, which was realised by incorporating CS theory into traditional theory.

Mishali et al. [176] developed a sub-Nyquist analogue-to-digital converter of widespread inputs. The hardware is able to reconstruct 2 GHz signals at a sampling rate of 280 MHz. Tur et al. [256] applied their sub-Nyquist sampling approach to ultrasound imaging. The authors were able to reduce the sampling rate by two orders of magnitude from $4,160$ to 33 samples. Further sub-Nyquist systems can be found in [249, 252].

New imaging architectures were developed with the help of CS. Duarte et al. [79] presented a new CS-based camera architecture that uses digital micromirror devices. The approach made it possible to build simpler, smaller, and cheaper digital cameras that can operate efficiently across a broad spectral range, called single-pixel camera. Many other groups have developed and enhanced similar systems [48, 192, 214], which have also been used for vision tasks like steering a robot [25].

Marcia et al. [170] built a CS imaging technique that improves the performance of video imaging systems in the presence of constraints in the focal plane array size. Coded aperture imaging, which enhances pixel resolution, was combined with superimposing subframes of a scene onto a single focal plane array to increase the field of view. Wavelet-based sparsity recovery algorithms were used to reconstruct the original subframes. The authors showed that their method offers an effective means to improve reconstruction accuracy.

Robucci et al. [204] implemented CS imaging on a CMOS sensor. Instead of acquiring raw pixels, the image is projected onto a separable 2D basis set. There, the corresponding expansion coefficients are measured. Then, the inner products are computed in the analogue domain using a computational focal plane and an analogue vector-matrix multiplier. The approach compresses the image without any digital computation, so that the reduction of throughput can reduce power consumption of the camera and increase the frame rate at the same time.

A further application domain for CS are sensor networks. Haupt et al. [113] used universal sampling and decentralised encoding to encode network data into a compressed form. Efficient and scalable monitoring of large-scale distributed networks without the need to reconstruct the complete network becomes possible with this approach, which is a step towards a theory for distributed signal processing. Sarvotham et al. [80] expanded their distributed CS theory with the so called joint sparsity of a signal ensemble, which enabled new distributed coding algorithms for multi-signal ensembles that exploit both intra- and inter-signal correlation structures. Davenport et al. [70] proposed a joint manifold framework for data ensembles that exploits multi sensor dependencies. The authors showed that signal processing algorithms for applications like classification and manifold learning can benefit from the incorporation of CS.

Compression, as may seem obvious, has also been applied using CS for synthetic aperture radar imaging systems [26]. As satellites have low computational resources, a simple 2D fast Fourier transform (FFT) based encoder was used to compress raw radar signals detected by the satellite before transmission.

In EM there are various proposals that try to integrate CS for both hardware, respectively microscopes, and software. For hardware implementation, complex blankers/mask/shutters and expensive instrument modifications are needed to perform CS measurements and reconstructions [131]. In [238] CS reconstructions are applied to energy dispersive spectroscopy (EDS) data collected on a scanning electron microscopy (SEM). The authors claim that this technique may also be generalisable to scanning transmission electron microscopy (STEM) and similar spectroscopic techniques. In their CS reconstructions, the spectra are fully-featured and have substantially reduced noise.

Further, CS was used to reduce the data volume in ptychography [178], which helps to cope with the challenges of data streaming and storage due to higher speeds. Even with only $1\,\%$ of detector pixels used for CS reconstructions, the image maintained strong visibility. Reed et. al [202] have developed a CS based system operating on a programmable zero-hysteresis post-sample deflection unit that sends images to multiple regions on a camera. This system is able to acquire CS video data in a transmission electron microscopy (TEM), which generates kHz-scale video using a conventional Hz-scale camera, which results in 50 to 100 output frames per input frame.

Recently, a new acquisition scheme called controlled rotation tomography has been proposed [154]. Projections of a specimen are acquired while rotating the specimen to maximise the time spent recording pixels with different angular views of the specimen, which maximises the information per unit of electron exposure. The acquired sparse projections are then recovered using CS reconstructions of the corresponding sinograms. The recovered projections are finally used to reconstruct the 3D volume with an iterative reconstruction scheme. Similar schemes, like the high-speed fly-scanning scheme [111], may be used as well.

4.5 Sparse Coding

As in other fields, CS theory, respectively the sparse coding theory underlying the reconstruction capabilities, has gained growing interest in the microscopy community. The ability to reconstruct huge 3D volumes from sparse electron tomography datasets, which means that only a small percentage of a specimen is scanned, has many advantages including lowering acquisition time and electron dose [29]. Binev et al. [27] examined which parts of CS may be useful for electron microscopy. The authors concentrated on two applications concerning STEM. Identifying the notions of sparsity and measurements are the challenging tasks to develop new imaging concepts in the context of electron tomography that can benefit from CS. This challenge is exemplified in [108]. This work explored the application of CS theory for visualising 3D structures of biological specimens from tomographic tilt series in STEM. However, only numerical simulations of a traditional sampling process are evaluated as CS sampling theory was not yet applicable to this acquisition technique.

Contrary to biological specimens, CS could be used in material science to reconstruct very sparsely sampled strontium titanate and zeolites using only $5\,\%$ of scanned data [230]. According to the authors, CS reconstructions can be implemented as a low-dose acquisition method with no change in the electron optics, which can dramatically reduce acquisition time and electron dose.

Based on these findings it is worth to investigate algorithms for reconstructing sparsely sampled data to evaluate the capabilities of sparse coding in the field of electron microscopy. Examples for such methods are Joint Statistical Modelling (JSM) [277], geometric analysis operator learning (GOAL) [114], Basis Pursuit (BP) [51] implemented in the software package YALL [279], beta process factor analysis (BPFA) [282] [281], Group-based Sparse Restoration (GSR) [276], or Convolutional Sparse Coding (CSC) [116]. In [126] the strength and weaknesses of CS methods, especially TVM, were evaluated based on STEM in electron energy loss spectroscopy.

4.5.1 Sparse Coding is not Compressed Sensing

First, we want to emphasise the difference between sparse coding and CS, so that no confusions arise. In literature and in the microscopy community, these two terms are often used as synonyms, but they are not!

It is correct that CS relies on the sparsity of a signal, so that optimisation methods, especially techniques based on the L_1 norm, can be used to reconstruct the original signal by solving an underdetermined system of equations. CS is a mathematical theory that describes and proves, when signals can be recovered error-free from their sparse representation. Ignoring all the details, the main trick in CS is that so called compressive measurements of a signal are used. These compressive measurements are linear combinations of basis elements from a basis in which the original signal is known to be sparse. CS is akin to acquiring measurements of a projection from the original signal into a much lower dimensional space. Special hardware is needed to perform such measurements, which means hardware that is able to acquire compressive measurements. For an image a single measurement must constitute a linear combination of several pixels, however, values of individual single pixels are not recorded. An example for such hardware is the single-pixel camera [122].

This is not the usual mode of acquisition in EM, at least in current state-of-the-art microscopes. For SEM each pixel is scanned individually. Scanning a random pattern of pixels, such that only a small percentage of pixels had been scanned, is not CS, as no compressive measurements have been acquired.

CS and sparse coding are only related in terms that a underdetermined system of equations has to be solved. Inspired by the ideas from CS, sparse coding uses a sparsifying transformation, which is essentially a matrix multiplication of the acquired signal with some basis coefficients in a sparse domain like wavelets. The transformed underdetermined system of equations can then be solved with the same L_1 minimisation methods that are used for recovering CS acquisitions.

Summing up, in CS a signal $\boldsymbol{x}$ is reconstructed error-free from observed data $\boldsymbol{y}$, whereas in sparse coding it is the goal to find a representation of an observed signal $\boldsymbol{y}$ as a sparse linear combination of elements from an overcomplete dictionary [2, 144]. As a consequence, algorithms from both domains may be very competitive as long as the signal $\boldsymbol{x}$ is very sparse, respectively compressible. However, if this assumption does not hold for $\boldsymbol{x}$, sparse coding in general is preferable to CS. Furthermore, in the EM setups we work with, even sparse coding may be used with caution due to the way data is acquired.

4.6 Deep Learning in Electron Microscopy

Deep learning gained much attention since the paper from Krizhevsky et al. [137] from 2012 with over $31,000$ citations, which is also reflected by the Nature publication about deep learning [147] from 2015 with nearly $11,000$ citation. This impact has also hit EM. For the sake of completeness we also show some deep learning approaches that relate to EM and some of the work of this thesis. However, we will not go into any details regarding deep learning. For readers interested in deep learning we recommend further reading in [52, 102, 104, 179].

In [143] a convolutional neural network (CNN) was used to determine the local 3D structure, e.g. tilt parameters or symmetry and magnitude of octahedral rotations, from STEM images of perovskites. The CNN was trained using simulated data, but also validated on experimental data resulting in remarkable accuracy. Structure determination with a CNN was also proposed in [219], which means estimation of class and location of objects. In [100] a CNN was trained to segment atomic columns in atomic resolution annular dark-field STEM based on synthetic data. A CNN for image segmentation of SEM data was used in [107]. An approach for identifying atoms based on CNN segmentations in high-resolution transmission electron micrographs was presented in [216].

A further application for deep learning in EM is real time event tracking and automation of imaging, diffraction, and spectroscopy acquisitions, which has been demonstrated on a classification problem of STEM data [103]. An integrated platform for performing deep learning tasks that should facilitate the application of CNN based algorithms in EM has been proposed recently [168], which, according to the authors, should make it possible to apply deep learning like image filters in any image processing toolbox.

In [275] deep learning and CS have been used to build a lensless camera. The developed block-wise lensless CS camera is able to capture scenes compressively and in parallel using multiple sensors. The compressed measurement are reconstructed using a generative model based on a CNN in real time.

End-to-end deep learning based inpainting methodologies have been published recently. Based on many examples these systems can learn adaptive image features, which are used to generate data for inpainting areas [125, 151, 186, 274]. Inpainting without any prior knowledge is also a very promising approach [257].

4.7 Missing Wedge Problem

Different approaches on tackling the missing wedge problem have been proposed, based mainly on (i) data collection, i.e. the way images are taken like using dual axis schemes, (ii) enabling a complete rotation of the specimen by using cylindrical sample geometries, and (iii) specialised reconstruction methods that address reconstructing missing information.

Changing the way of data acquisition by extending standard approaches of single axis tomography leads to better results regarding the missing wedge issue. Lee et al. [148]

experimentally demonstrated, that equally sloped tomography (EST) can reduce missing wedge artefacts by constraining acquired images in a tilt series to have equally sloped angle increments. Another approach is double-tilt tomography, also called dual axis tomography [13, 189], where data is acquired from two perpendicular tilt axes. This technique reduces the missing wedge to a missing pyramid due to the fact that more information is present because of the two mutually perpendicular tilt series for one reconstruction. Dahmen et al. [60, 63, 66] proposed a combined tilt- and focal series as a new recording scheme for high-angle annular dark-field (HAADF) STEM tomography. Hereby, a tilt series with limited tilt range is supplemented by a through-focal series per tilt direction. The method results in a better sampling of the Fourier space and, thus, to a reduction of axial elongation artefacts.

Instead of using slab-like flat specimens for electron tomography, which restricts viewing angles, Kawase et al. [134] used a cylindrical specimen. The cylindrical shape allows imaging from a full range of viewing angles around the tilt axis, so that no missing wedge exists. They quantitatively analysed the effect of the missing wedge on the volume fraction in a composite and the elongation due to the missing wedge. The work was extended by Kato et al. [133] who used rod-shaped specimens with different diameter in TEM to evaluate the maximum usable rod diameter for the tomography specimen. While effective at avoiding the missing wedge problem altogether, the approach is not applicable in many situations, because manufacturing the rod-shaped specimen requires making a decision on the field of view prior to initial imaging and, thus, selecting the area of interest becomes difficult and the statistical sampling is limited. Furthermore, fabrication of such needle-shaped samples requires the material to be mechanically sufficiently strong, which is not the case for many materials.

Adapted reconstruction methods to handle missing information may also be capable of fighting the missing wedge. Baba et al. [15] performed a topographic analysis by stereo-photogrammetry to get pre-determined voxel arrangements for thin film-like replica-type specimens with some known properties. Using this prior knowledge as constraint in the reconstruction process reduces an infinite number of solutions due to the missing wedge to a finite number. Qualitatively enhanced reconstructions revealing fine structural parts showed no missing wedge effects. Zürner et al. [285] also used prior knowledge by adding a mask in each reconstruction step setting all voxels known to be vacuum in the samples to zero. A stronger assumption is made in the field known as discrete tomography. Hereby, the assumption is that the solution of the reconstruction problem has a sparse gradient, i.e. the sought tomogram consists of only a limited number of a priori known materials. The assumption obviously simplifies the problem and can be used to reduce missing wedge artefacts. A heuristic approach to discrete tomography is taken by the discrete algebraic reconstruction technique (DART) [20] and consecutive work. Another group of algorithms based on the sparse gradient assumption are TVM techniques [105]. Contrary to DART, TVM techniques do not require the grey values to be known a priori, which can be an advantage for some applications.

Some algorithms that work without additional assumptions have also been reported to improve on missing wedge artefacts. A combination of the weighted back projection and simultaneous iterative reconstruction technique (SIRT), also called weighted SIRT, claims an improved reconstruction with reduced elongated point spread in the direction of the missing tilt angles for a simple geometrical phantom, i.e. an improved reconstruction of spatial frequencies in the vicinity of the missing wedge [269]. The DIRECTT method [138, 139] is an iterative algorithm based on a sequence of alternating reconstructions and virtual projections of intermediate results, which are subsequently enhanced using residual sinograms instead of filtered back projections. DIRECTT reduces the typical missing wedge artefacts by variation of reconstruction parameters in the course of iterations.

5

Three-Dimensional Exemplar-based Inpainting

> *The only time I feel alive is when I'm [in]painting.*
>
> — **Vincent van Gogh**
> (Dutch Post-Impressionist painter)

In this chapter we present the exemplar-based inpainting (EBI) approach, which constitutes the core of the developed applications. Besides the original algorithm from Criminisi et al. [58], extensions and adaptations are explained. Furthermore, we show ways to speed up EBI approaches. A case study in photo processing is used to evaluate the impact on the runtime for these acceleration approaches.

5.1 Criminisi's Approach

Our algorithm is inspired by the approach of Criminisi et al. [58]. The method originates from the field of image processing and was originally developed to fill missing regions in photographs in a way that this is not perceivable to the human eye after using the so called EBI. A simplified illustration of the original approach is depicted in Figure 5.1.

EBI is an iterative process that inserts small image parts, called patches, of the known discrete source region $\Phi \subset \mathcal{I}$ into the unknown discrete target region $\Omega = \mathcal{I} - \Phi$. As input the algorithm takes an image and a corresponding mask to discriminate between source and target region (Figure 5.1a, 5.1b). A discrete template window $\Psi = \Psi_x \times \Psi_y$ must be specified as size for the patches. Ψ is highly dependent on the dataset to be inpainted. In practice Ψ should be set larger than the largest distinguishable texture element in the source region. Each patch is centred at a pixel position $\boldsymbol{p} = (x, y)$, which will be used as patch identifier $\Psi_{\boldsymbol{p}}$ from now on.

The order of filling is crucial as texture and structure should be propagated from source region to target region. Hence, patches that are on the continuation of strong edges and additionally are surrounded by many known pixels must be filled first, because such positions have highest confidence to result in meaningful insertions. This means a prioritisation order for the positions where patches should be inserted must be determined. Each source region pixel $\boldsymbol{p}$ that is a direct neighbour of a target region pixel is a possible candidate position and is assigned to the so called fillfront (Figure 5.1c). The computed priority $P(\boldsymbol{p})$ consists of a confidence term $C(\boldsymbol{p})$ and a data term $D(\boldsymbol{p})$.

The computation of the confidence $C(\boldsymbol{p})$ of pixel $\boldsymbol{p}$ is defined as

$$C_{\text{init}}(\boldsymbol{p}) = \begin{cases} 0, & \text{if } \boldsymbol{p} \in \Omega \\ 1, & \text{if } \boldsymbol{p} \in \Phi \end{cases}, \qquad C(\boldsymbol{p}) := \frac{\sum_{\boldsymbol{q} \in \Psi_{\boldsymbol{p}} \cap \Phi} C(\boldsymbol{q})}{|\Psi_{\boldsymbol{p}}|} \tag{5.1}$$

where $\Psi_{\boldsymbol{p}}$ is the patch centred at pixel $\boldsymbol{p}$, $\boldsymbol{q}$ are all pixels in the intersection of a patch and the source region, and $|\Psi_{\boldsymbol{p}}|$ is the number of pixels of the patch. The data term $D(\boldsymbol{p})$ of a pixel $\boldsymbol{p}$ is computed by

$$D(\boldsymbol{p}) = \frac{|\nabla \mathcal{I}_{\boldsymbol{p}}^{\perp} \cdot n_{\boldsymbol{p}}|}{\mathcal{I}_{\text{max}}} \tag{5.2}$$

where $\nabla I_{\boldsymbol{p}}^{\perp}$ is the isophote at pixel $\boldsymbol{p}$ of Image $\mathcal{I}$, that means orthogonal ($\perp$) to the gradient ∇, $n_{\boldsymbol{p}}$ is a unit vector orthogonal to the fillfront in pixel $\boldsymbol{p}$ (Figure 5.1d), and $\mathcal{I}_{\text{max}}$ is the dynamic range of the pixel values, which is the biggest possible grey value in a given image, i.e. $2^{(\#\text{bits})-1}$, e.g. for 8 bit images it is 255.

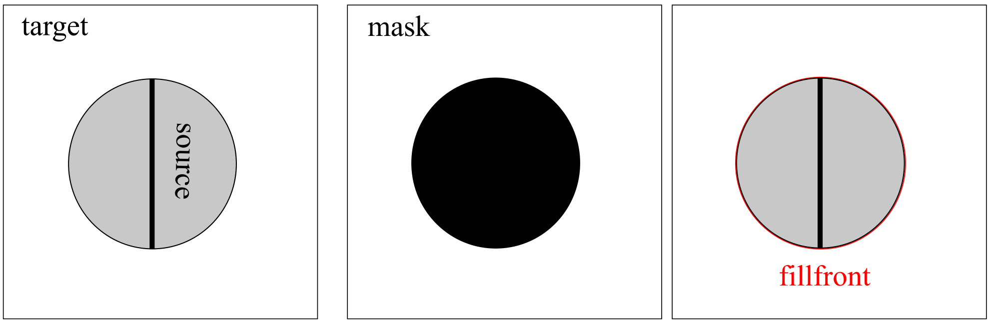
Having computed the confidence term and the data term the priority value $P(\boldsymbol{p})$ of a pixel $\boldsymbol{p}$ is computed by

$$P(\boldsymbol{p}) = C(\boldsymbol{p}) \cdot D(\boldsymbol{p}) \tag{5.3}$$

The position of the maximal value $\boldsymbol{p'} = \underset{\boldsymbol{p}}{\text{argmax}}\, P(\boldsymbol{p})$ determines where the next patch is inserted (Figure 5.1e). $\Psi_{\boldsymbol{p'}}$ is then compared to every patch that completely fits into the source region by computing a similarity measure between both. The original approach uses the sum of squared differences (SSD), which is equivalent to a L_2 norm as defined in Section 3.2. For EBI it is defined as

$$\text{SSD}(P_t, P_s) = \|P_t - P_s\|_2^2 = \sum_{\substack{i=1 \\ t \in P_t \cap \Phi, s \in P_s \cap \Phi}}^{|\Psi_{\boldsymbol{p}}|} (t_i - s_i)^2 \tag{5.4}$$

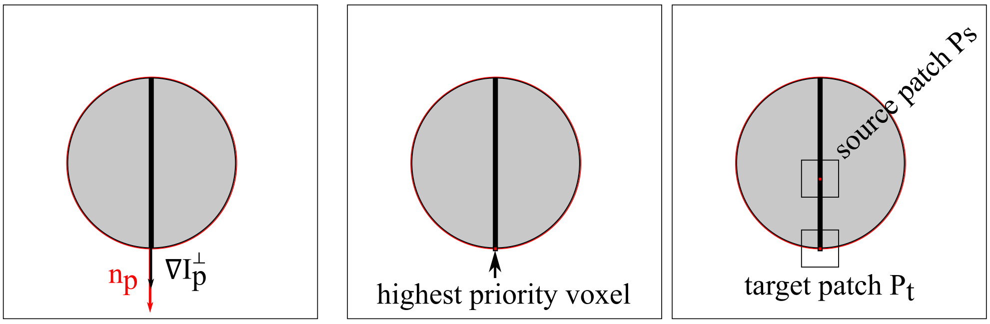
where P_t is the target patch, i.e. the patch where data will be inserted, and P_s is a source patch, i.e. a candidate patch whose pixels will be inserted where P_t has no values yet (Figure 5.1f).



(a) Input image, the known part is the grey circle with the black bar (source) and the unknown part is the white area around it (target).

(b) Mask, black – source, white – target, this is also the initial confidence whereby source is 1 and target is 0.

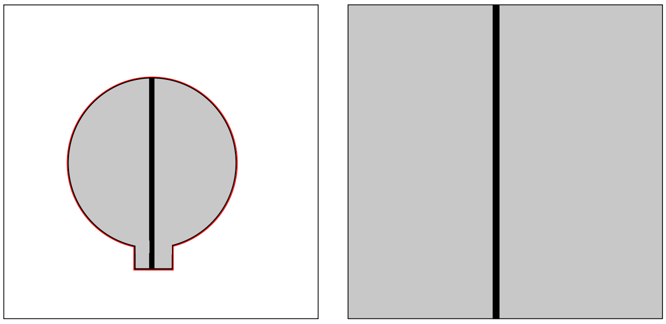
(c) Fillfront, the red line around the object.



(d) Input for data term computation, the black arrow is the orthogonal gradient at the end of the bar, the red arrow is the normal of the fillfront at that pixel.

(e) Example for a highest priority pixel, which will be the centre coordinate of the target patch P_t.

(f) Target patch P_t and a possible source patch P_s with minimal distance in terms of a similarity measure.



(g) Result after one iteration, where the first patch has been inserted and the fillfront has been updated to fit the situation after the step.

(h) Final inpainted image.

Fig. 5.1.: Exemplar-based inpainting procedure.

The candidate patch with minimal SSD constitutes the most similar patch P_{best}. Every pixel of P_t that has no information yet is then replaced by the corresponding pixel of P_{best} (Figure 5.1g).

At the end of each iteration the fillfront must be updated, which means pixel positions that were inpainted are deleted from the fillfront and pixel positions that are adjacent to the inserted patch are added to the fillfront. Having updated the priority values, the next patch to be inserted can be computed. This is done until there are no pixels left in the fillfront and the whole target region has been filled (Figure 5.1h).

5.2 Modifications and Extensions to Criminisi's Algorithm

Grey values. As most microscopic datasets do not contain colour, we concentrated on the inpainting of grey values. Nevertheless, the developed method can also handle single RGB images and an extension for three-dimensional (3D) colour stacks is also possible, if this may be needed.

Extension for 3D data. The original algorithm only works on two dimensional (2D) data. We extended the ideas to work on 3D data, as electron tomography (ET) tilt series are 3D datasets. That means we extended the two-dimensional (2D) method to work on 3D data using 3D patches, which has not been done yet to the best of our knowledge. The template window changes to $\Psi = \Psi_x \times \Psi_y \times \Psi_z$. We have set $\Psi = 9 \times 9 \times 9$ as the default window. For the sake of completeness, pixels are also called voxels in 3D.

Priority and Confidence computation. We adapted the computation of the priority of a voxel $\boldsymbol{v}$ to

$$P(\boldsymbol{v}) = \begin{cases} C(\boldsymbol{v}) \cdot D(\boldsymbol{v}), & \text{if } D(\boldsymbol{v}) > 0 \\ C(\boldsymbol{v}), & \text{if } D(\boldsymbol{v}) = 0 \end{cases} \tag{5.5}$$

Through this change also voxels where the data term is 0 are taken into consideration for the priority. Additionally, this gives a ranking when all voxels in the fillfront have a data term of 0. Different from the original approach, new priority values are computed for each voxel of the fillfront that was affected by the inserted patch. Instead of freezing already computed confidence values, we changed the computation of the confidence. Added voxels are assigned a confidence value

$$C_{\text{adaptive}}(v_{\text{new}}) = \min\left(1, \frac{1}{\lambda \cdot \#\text{iterations}}\right) \quad , \lambda \in (0, 1] \tag{5.6}$$

Decreasing λ helps with stability issues, as $C_{adaptive}$ quickly becomes very small when $\lambda = 1$. Our experiments were performed with $\lambda = 0.1$. Afterwards, each affected pixel is recomputed. This ensures that the target region is inpainted slice by slice, which stabilises the whole inpainting process in 3D.

Adapted metric. We extended the originally proposed SSD to a kernel based similarity measure. Inspired from Kernel Learning in Machine Learning, we used the well-known radial basis function (RBF) kernel [95] to define a similarity measure, which is defined as

$$K(P_t, P_s) = \exp\left(-\frac{\mathsf{SSD}(P_t, P_s)}{2 \cdot \mathcal{I}_{\mathsf{max}}}\right) \tag{5.7}$$

where $K \in (0, 1]$ with 0^+ meaning no similarity and 1 meaning the patches are identical.

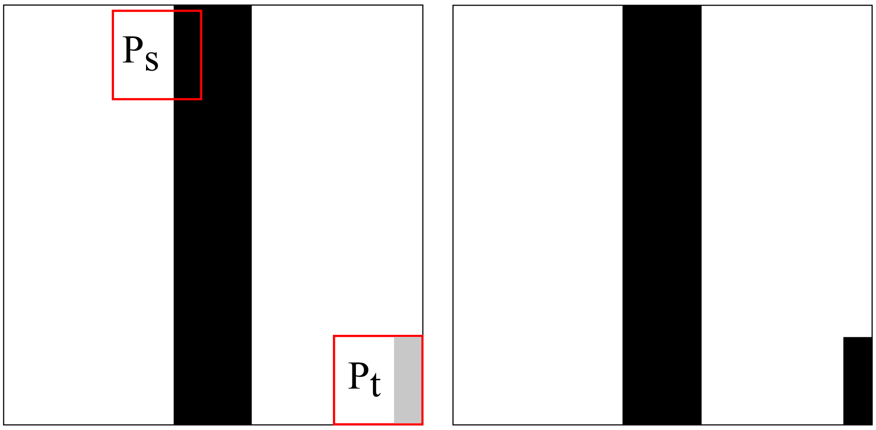
As P_t consists of known voxels from the source region as well as voxels with no information, it is not straightforward to compute a similarity measure between P_t and a candidate patch. Furthermore, voxels that have no value yet must not be ignored in the similarity computation, because doing so can lead to the introduction of artefacts during inpainting that can propagate further, as shown in Figure 5.2, which may severely influence the resulting inpainting.

The new similarity measure consists of two parts: (i) a RBF Kernel $K_1(P_t \cap \Phi, P_s)$ for the part of the target patch where data is available denoted as $P_t \cap \Phi$, and (ii) a RBF Kernel $K_2(\mu_{t|\mathsf{known}}, P_s)$ for the part of the target patch that has no information yet, where $\mu_{t|\mathsf{known}}$ is the mean value of the known input patch voxels. This procedure simulates voxel values for empty target patch voxels that fit to the available data of that patch. The sum of K_1 and K_2 is then calculated as the convex combination

$$K(c) = c \cdot K_1 + (1 - c) \cdot K_2 \tag{5.8}$$

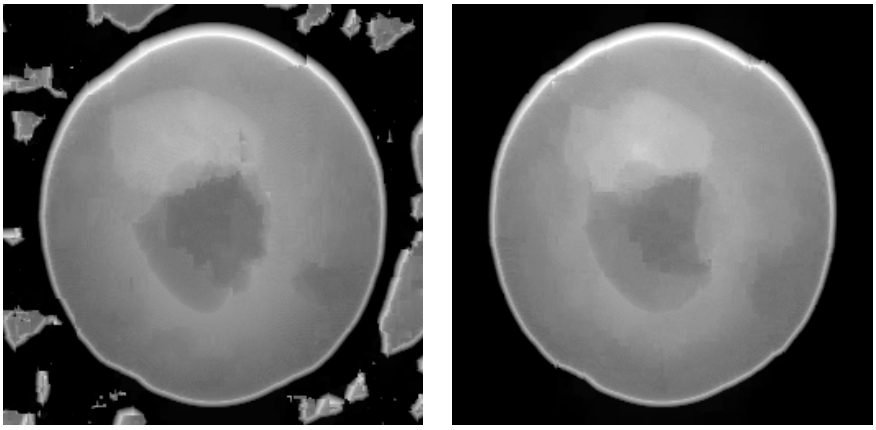
with $c \in [0, 1]$ resulting in $K_c \in [0, 1]$, which is still a kernel function and hence a norm [95]. We have set $c = 0.999$, as conducted experiments showed that a small contribution from K_2 is enough to prevent the occurrence of artefacts like those in Figure 5.2c. The candidate patch with maximal K_c constitutes the most similar patch P_{best}.

Input image independent dictionary. We extended EBI to work on source regions not present in the input image. An additional input can be selected as dictionary, so that our EBI is not reliant on image regions of an input, which is important when inpainting sparse data that does not contain full patches. Nevertheless, image regions of an input image that contain full patches can also be selected as dictionary, respectively source region.



(a) P_t is the patch where data should be inserted. Here white and black areas are already filled and the grey area has no values yet. The patch P_s is a possible candidate for the insertion when leaving out voxels that have no value yet in the similarity computation.

(b) This leads to the introduction of an artefact. As example an inpainted Shepp-Logan phantom projection is shown.



(c) Result using $K(1)$ with ignored target region, which has introduced many artefacts.

(d) Result using $K(0.999)$.

Fig. 5.2.: Example of the possible introduction of artefacts when ignoring the target region in the similarity measure.

5.3 Acceleration Approaches

EBI is very time consuming, especially in a 3D setup. Applying the method to a volume with a resolution of $256 \times 256 \times 181$ pixels, where the source region had a size of $256 \times 256 \times 125$ pixels took over 90 minutes. For datasets with a resolution of $1,024 \times 1,024$ per image the method took over 60 hours, which made it infeasible for high-resolution image stacks. This raised the need to accelerate the algorithm. We implemented a high-performance computing (HPC) version of the algorithm based on graphics processing unit (GPU), which delivers the same result as the original algorithm in a full scan fashion.

This means, that the cost function is evaluated for every possible patch with a brute force approach. Then, the patch corresponding to the minimum cost function value is selected. Furthermore, we developed an acceleration structure based on high-dimensional indexing techniques [68], which runs on a central processing unit (CPU) and turns the original approach into a lossy one that does not deliver the same results as the full scan approach, as only a small percentage of possible patches is examined. In the following we present the two approaches and evaluate their performance.

5.3.1 Exemplar-based Inpainting on GPU

The new method is computationally very demanding on 3D data. During one iteration, many similarity measure computations for voxels, respectively patches, have to be computed by the algorithm. As this task is highly parallelisable and most performance critical in the method, we implemented this part on GPU via OpenCL [233]. Furthermore, we minimised the amount of data that must be transferred from CPU to GPU and vice versa, as bandwidth plays a crucial role in GPU programming.

As backend for the GPU based inpainting we used and adapted Ettention [64]. Ettention was originally developed for the reconstruction of electron microscopy (EM) data. Ettention's set of modular building-blocks for tomographic reconstructions already featured a modular, object-oriented software design and an optimised access to HPC platforms, in our case the GPU. Ettention offered an ideal starting point for developing a HPC EBI due to its high-level abstractions of low-level access to hardware via a well-structured interface design.

Ettention combines algorithmic innovation as well as immediate applicability to experimental data. It has the ability to provide suitable hardware-independent and domain-specific abstractions over the underlying many-core platforms. This not only made it an ideal basis for the GPU version, but also for the development of the following space-filling curve (SFC) approach.

5.3.2 Acceleration Structure for Exemplar-based Inpainting

We propose to use high-dimensional indexing techniques as typically applied in the field of multimedia databases as acceleration structure for the patch search. We show how the problem of searching the best patch for a target location corresponds to a nearest neighbour search in an ad-hoc subspace of the dictionary. As the most general form of this problem is suspected to be linearly hard, we use additional knowledge on the inpainting algorithm to map the patch search to the much simpler problem of a nearest neighbour search in a subspace that is approximately known a-priori. This problem can then be addressed using techniques that are well known from the field of high-dimensional indexing, such as dimensionality reduction by means of a principal component analysis (PCA), filter-and-refine approaches using k-nearest neighbour (knn) queries, and multi-dimensional indexing using SFCs. Our solution has the advantage that it is independent of most model specific aspects of EBI, so that it is applicable to a large variety of inpainting algorithms.

The developed inpainting process starts by building acceleration structure indices. The basic idea is to split the dictionary into constant sized patches and arrange them in a one-dimensional array, sorted by some measure of similarity, such that similar patches are stored close to each other. The indices are built over patches of constant size $K \times K$. For coloured RGB images, red, green and blue values are treated separately, resulting in $K^2 \cdot 3$ values per patch. Typical patch sizes are $K = 9$ to $K = 13$, so a patch corresponds to at least $9 \cdot 9 \cdot 3 = 243$ values. In the following, we consider a patch as a point in a K^2-dimensional (or $K^2 \cdot 3$, respectively) space, called the patch space. During the inpainting procedure, some values of the processed patch are unknown, as they are about to be inpainted. Patch similarity is therefore always defined on a subset of pixels, and the cost function must be defined in such a way that it can handle unknown pixel values. In the example of the L_2 norm, one simply assumes that the difference between unknown pixel values is zero. Geometrically, this can be interpreted as a parallel projection into a lower-dimensional sub-space. If a pixel value is unknown, the data set is projected parallel to the dimension that corresponds to the unknown pixel value, and distance is measured in the resulting, lower-dimensional space. As different pixels are known in every inpainting step, we say that we measure patch similarity in an ad-hoc subspace of the patch space.

We build the indices using a three-stage technique. First, we select a subset of the pixels to be considered in an index. Second, we reduce the dimensionality of the resulting data points by means of a PCA. Finally, we arrange them in a one-dimensional array according to their positions on a SFC (Figure 5.3). The defined parameters for the acceleration structure are depicted in Table 5.1.

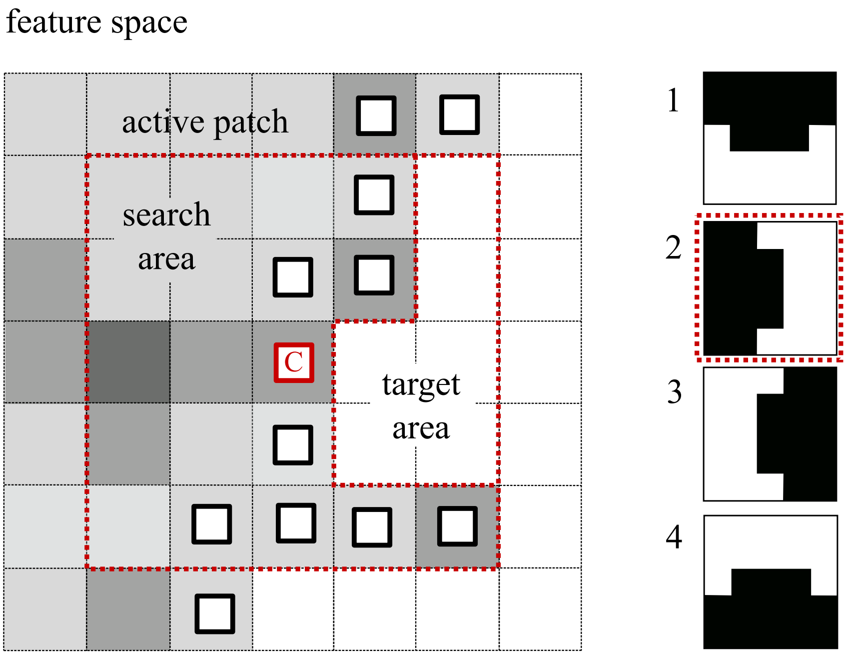
Tab. 5.1.: Annotations and typical values for the parameters used in the acceleration approach.

symbol	meaning	typical values
K	patch size	7 to 13
s	fraction of pixels used in index	0.6
D	dimensionality of principal space	6 to 12
k	size of candidate set (filter-and-refine)	40 to 160
μ	minimal interval size to use recursion	128 to 512
v	minimal interval size for parallelisation	512 to $2,048$

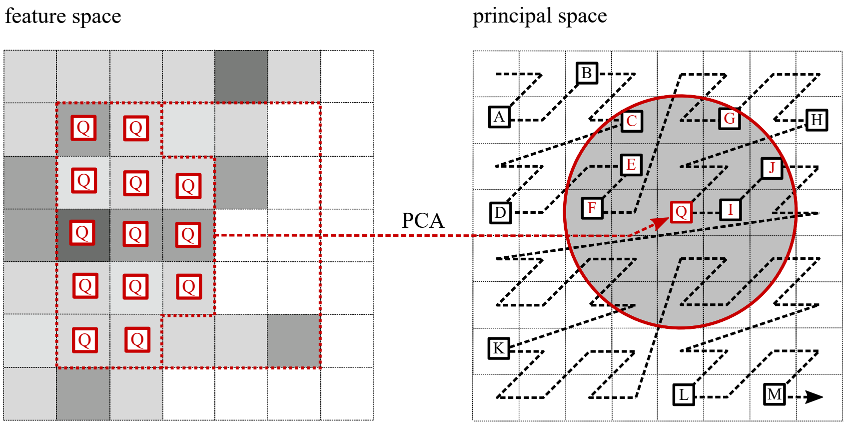
In order to approximate the ad-hoc subspaces, we build several indices, where each index covers a different subset of the pixels in the patch. Each index uses a fraction c, typically about $60\,\%$, of these values. See the result section for a determination of optimal values of c. For inpainting 2D images, we build eight separate indices, where each index contains the pixels nearest to the middle of one of the edges or one of the corners of the patch (Figure 5.4).

Dimensionality Reduction. All known index structures with support for efficient nearest neighbour queries suffer from the curse of dimensionality, which means they scale well only for a total number of 8 to 16 dimensions. In the field of high-dimensional indexing, techniques of dimensionality reduction are therefore used prior to building the index. Dimensionality reduction can be performed by several techniques. Common approaches are PCA and linear discriminant analysis. Recently, a technique called random projections (RP) has also received wide interest, as it is computationally cheaper than a PCA, particularly in high dimensional spaces. However, we implemented and tested RP and found that RP does not preserve distances sufficiently well for our application. Therefore, we reduce the dimensions of the index to a constant number D by using a conventional PCA of which the first D principal components are kept for further processing.

High-Dimensional Indexing. After dimensionality reduction, an index can be built based on a moderate number of dimensions. A large number of index structures have been proposed for this purpose, including the well-known R-tree [112] and its versions R^{+}-tree [222] and R^{*}-tree [21]. Slightly better scalability up to 20 dimensions was reported for the structures X-tree [22], SS-tree [268], TV-tree [155] and the proxy-based indexing method called iDistance [127]. More recently, the idea of locality preserving hashing [255] has also gained some interest. However, most of the work on multi-



(a) The centre of the active patch (marked C) is selected on the fillfront (white squares). The fillfront is defined as pixels with known value adjacent to pixels with unknown value. The known pixels inside the active patch are called search area, the unknown pixels are called target area. Index 2 is selected as it best covers the search area.



(b) The query object is formed by all pixels covered by this index, marked Q. It is by definition a subset of the search area. The query object is transformed to principal space and a knn search is performed to select k candidate patches. The best patch is then selected by means of a full scan in image space on those k candidates.

Fig. 5.3.: Overview of the patch search algorithm.

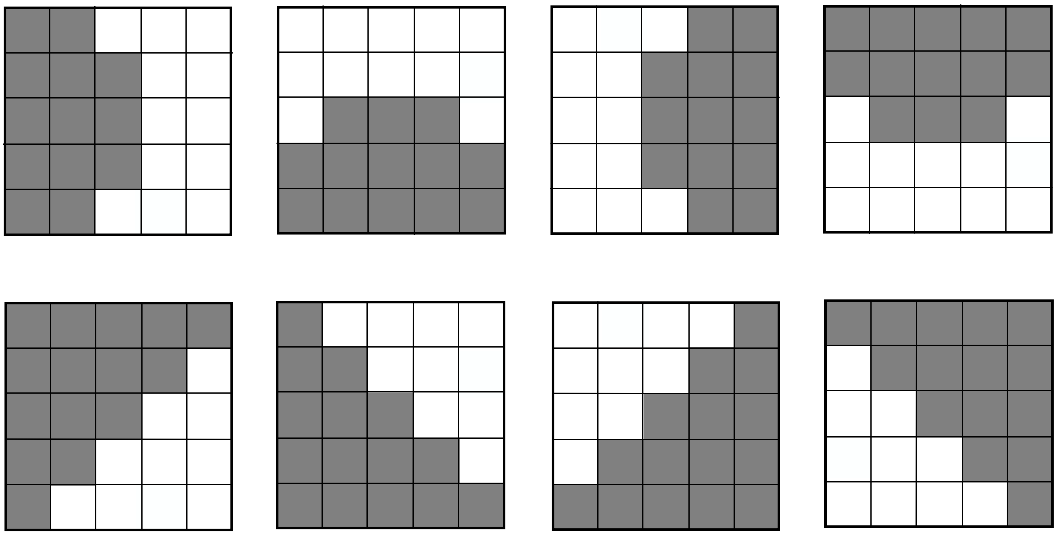


Fig. 5.4.: We build a total of eight indices in 2D. Each index uses approximately 60% of the pixel values of the patch. Four indices use pixel values closest to the middle of a patch edge (upper row). The remaining four indices use pixel values closest to a corner of the patch (lower row).

dimensional indexing comes from the field of database design and considers out-of-core situations in the sense that the data is stored in the form of fixed-size pages. An index should optimise the number of page accesses in this case. In our situation, several indices are stored in main memory, such that the memory consumption per index as well as the computational cost for traversing the index become relevant factors. Memory locality is still an important factor as cache hit rates have a high impact on CPU performance, but to a much lesser extent than for disc-based access schemes. Therefore, we selected an index structure that has low memory consumption per element and supports computationally efficient knn queries. The basic idea is to use a SFC to map the dictionary space to a one-dimensional sequence. A large number of SFCs have been investigated in the past, with early work including the famous Hilbert SFC [119], Peano SFC [188], and Lebesgue SFC [146]. A good introduction to the topic of SFCs is given in [209]. Of most practical relevance are z-curves [177], which can be considered to be polygonal approximations of the Lebesgue SFC on a discrete space. Z-curves have the distinct advantage that the computation of positions on the curve is computationally very cheap. In our approach, we use a z-curve to map the dictionary space to a one-dimensional structure. This structure is sorted and can be used by a binary search to find points in the set efficiently.

Sorting Relative to Z-Curve Position. After performing a PCA and transferring all patches in the dictionary to principal space, a vector with the patch positions must be sorted according to the position of the principal space representation of the vector on the z-curve. This can be achieved by a conventional Quick-sort algorithm, assuming the availability of a comparison operator that compares points relative to their position on a z-curve. Efficient implementations of many operations on z-curves depend on properties of the representation of the curve position on bit-level, called Morton encoding [177]. We first present a short overview of this encoding based on two pixels. The Morton-encoded address of a position on the curve is called its z-address. It consists of the interleaved bits of the two pixel values. In higher dimensions, the operation is performed analogously, such that the highest bit of each pixel is used before returning to the 2^{nd} highest bit of the first pixel and so on. Summing together, in multi-dimensions the z-address of a point is computed by interleaving the binary representation of its coordinate values.

An interesting observation is that the z-curve does not treat all dimensions equally. It rather introduces the concept of "more significant" or "less significant" dimensions. This is expressed by the order in which the different dimensions are processed when forming the z-address. In the 2D example, it is possible to either use the x-address first, then the y-address, or the other way around. In fact, the two options result in two different z-curves. In the n-dimensional case, this leads to $n!$ possible curves. It is known from database research that indices give best performance if the curve uses dimensions that have the highest variability in a given dataset first. In our case, the dataset is generated using a PCA, so this property is guaranteed. Interestingly, the use of a PCA prior to building the z-curve also establishes rotational invariance of the method as any permutation of dimensions is removed by the PCA. Comparing two entries can be implemented efficiently without computing the entire z-address. Hence, only the most significant bit that differs between the two entries needs to be calculated, and the bits of corresponding dimension have to be compared. This is made possible by the encoding procedure, which ensures that all bits of a z-address are ordered by dimension. An *exclusive or* operation is used to mask out all higher order bits that are identical, so that identifying the first bit that differs is easy. Figure 5.5 shows a C++ code fragment to perform this operation.

K-Nearest-Neighbours Queries. We now explain how to use this structure to perform a knn search in the dictionary. We use a traversal technique called outside-in search [251]. Hereby, the acceleration structure is split into recursive regions, which are traversed depth-first. In each traversal step the current region is split into two sub-regions. Then, the distance of both regions to the query object is computed. The sub-region closest to the query object is recursively traversed first, the sub-region further from the query object is traversed second. Hereby, the visit to the second sub-region is pruned if its distance to the query object is higher than that of the k closest neighbours found so far.

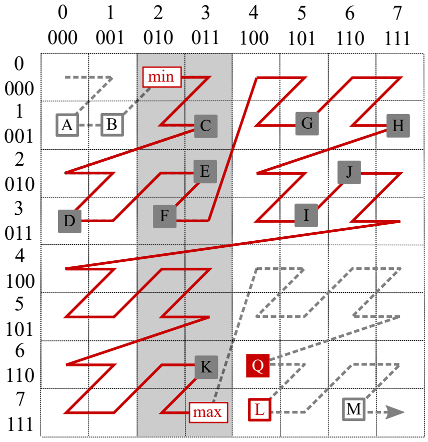
In detail, the search operates by recursively subdividing the search space into axis-aligned regions. One property of the z-curve is that for each axis aligned box specified by its upper left corner min and its lower right corner max, it holds that all points inside the box are also inside the interval $[\text{min}, \text{max}]$ on the z-curve. With each axis-aligned region, we can associate an interval on the z-curve (Figure 5.6a). The opposite is not true,

```cpp
inline bool lessMostSignificantBit(byte a, byte b)
{
return (a < b) && (a < (a ^ b));
}
bool lessRelativeToZCurve(const byte* a, const byte* b)
{
  unsigned int bitWiseDifferenceSoFar = 0;
  unsigned int bestDimensionSoFar = 0;
  unsigned int size = (unsigned int) a.size();

  for (unsigned int dim = 0; dim < size; dim++)
  {
   auto bitWiseDifference = a[dim] ^ b[dim];

   if (lessMostSignificantBit(bitWiseDifferenceSoFar,
       bitWiseDifference))
   {
     bitWiseDifferenceSoFar = bitWiseDifference;
     bestDimensionSoFar = dimension;
   }
  }
 return a[bestDimensionSoFar] < b[bestDimensionSoFar];
}
```

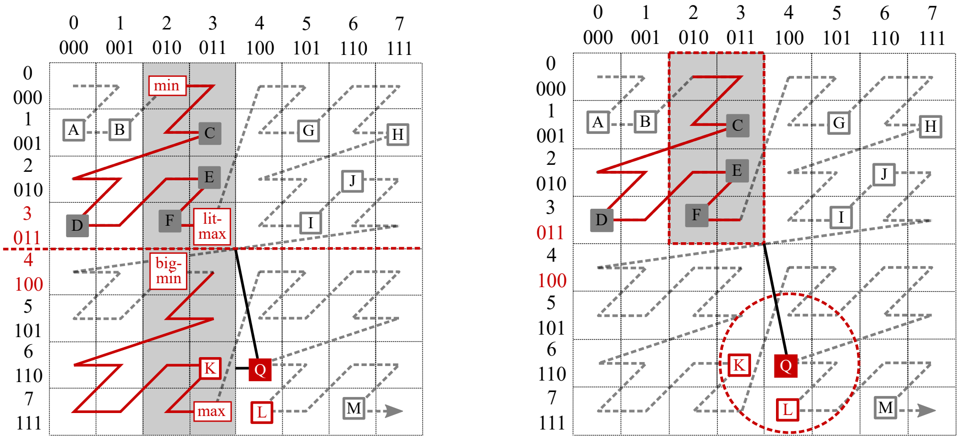
Fig. 5.5.: C++ code fragment for *exclusive or* operation.

so that in general the interval $[\mathrm{min}, \mathrm{max}]$ contains some points outside the associated axis-aligned region. When recursively splitting the search regions, the goal is to minimise this overhead, i.e. find split positions where the unification of the intervals associated with the two sub-regions contains significantly less patches than the interval associated with the full region. This is achieved by the litmax/bigmin algorithm first presented in [255]. In short, the algorithm splits the region always at the position and in the dimension where most significant bits differ between the coordinates of min and max. Intuitively, this can be understood by inspecting the query region in Figure 5.6b. It can be observed that taking a step of size one at the position where the most significant bit of the coordinate changes corresponds to a large path along the curve. A proof that this scheme is in fact optimal is given in [119].

After the position of the split is determined, the algorithm computes the z-curve intervals associated with the two sub-regions. This is done by using a binary search on the dictionary to find the positions of litmax and bigmin on the z-curve. The intervals are then $[\mathrm{min}, \mathrm{litmax}]$ and $[\mathrm{bigmin}, \mathrm{max}]$. The two sub-regions are traversed recursively, whereby the order of traversal depends on the distance of the sub-region to the query object. The sub-region with the smaller distance is traversed first. After returning from the visit to the near sub-region, the algorithm checks if the far sub-region is farther away than the k nearest neighbours found so far. If this is the case, the sub-region cannot contain a part of the result and the interval is culled (Figure 5.6c). At the lower



(a) The example consists of 14 patches (A to M), sorted according to their position on the z-curve. One property of the z-curve is that all patches inside a bounding box are also inside the interval [min, max] on the z-curve. Those active patches are marked in grey, patches outside the active interval are marked white.



(b) The active region is split at litmax $= (3, 3)$ and bigmin $= (2, 4)$. The unification of the two resulting intervals [min, litmax] and [bigmin, max] does not cover the full interval [min, max], so patches G to J are culled. The two sub-regions are traversed according to their distance to the query object.

(c) After visiting the sub-region [bigmin, max], the distance of the query object to the remaining sub-region is compared to distance of the k-nearest neighbours found so far. The distance is higher, so the sub-region is not visited and patches C to F are culled.

Fig. 5.6.: A knn query on the z-curve index structure. The active search region is specified by an axis aligned bounding box [min $= (2, 0)$, max $= (3, 7)$], indicated in grey. The query object at 4/6 is marked solid red, patches in the knn list are marked white with red border.

levels of recursion, the sizes of the intervals fall below the defined threshold μ. All patches contained in these intervals are checked using a full scan. For each point in these intervals, the distance of the point to the query object is computed. If it is smaller than the distance to the k nearest neighbours so far, the corresponding patch is added to the knn list. If the list overflows, i.e. contains more than k entries, the entry with the largest distance is removed from the list.

Performance Optimisation. While the described approach is straightforward, it turns out that a naïve implementation of the algorithm is quite slow. In order to optimise performance, we avoid storing a separate copy of litmax and bigmin on the stack for each recursion step and store only the dimension and position of the split. We can then re-use the memory already reserved for the query range, so that only one byte in the structure must be changed. A second optimisation concerns the access to the dictionary. A binary search on a sorted vector of length L is typically initialised from 0 to $L-1$ and then uses bisection to find an element. For the query algorithm, however, we already know an interval that must contain the searched element from the previous level of recursion. Therefore, it is wasteful to initialise the search from 0 to $L-1$ every time. Instead, the search can be initialised from the beginning to the end of the already computed interval. This interval can be substantially smaller than the dictionary, particularly on deeper levels of recursion, which is the case for the majority of query steps. In Figure 5.7 we give C++ code for the recursive query algorithm including the discussed optimisations.

```cpp
bool knnSearch(const Point& queryObject, HyperCube&
activeRange, Interval interval, DataSet* dataset)
{
  if (interval.last >= dataset->size () ||
  interval.last < interval.first)
  {
    return false;
  }

  if (interval.length() <= minimumIntervalLengthToUseRecursion)
  {
    return fullScanSearch(queryObject, activeRange,
           interval, dataset);
  }

  int splitDim = activeRange.dimensionWhereHighestBitDiffers();
  int splitPos = activeRange.positionWhereHighestBitDiffers();

  float leftDist = distanceToLeftRange(queryObject,
                   activeRange, splitDim, splitPos);

  float rightDist = distanceToRightRange(queryObject,
                    activeRange, splitDim, splitPos);

  if (rightDist >= leftDist)
  {
    return traverseLeftRight(queryObject, activeRange, interval,
           dataset, splitPos, splitDim, rightDist);
  }
  else
  {
    return traverseRightLeft (queryObject, activeRange, interval,
           dataset, splitPos, splitDim, leftDist);
  }
}
```

Fig. 5.7.: C++ code fragment for the recursive query algorithm.

```cpp
bool traverseLeftRight(const Point& queryObject, HyperCube&
activeRange, Interval interval, DataSet* dataset)
{
  const byte maxInDimension = activeRange.last[splitDimension];
  const byte minInDimension = activeRange.first[splitDimension];

  activeRange.last[splitDimension] = splitPosition;

  Interval lower = dataset->dataPointsInInterval(activeRange,
                   interval, KNOW_LOWER);

  bool didImprove = knnSearch(queryObject, activeRange, lower,
                              dataset);

  activeRange.last[splitDimension] = maxInDimension;

  if (rightDistance >= distanceToKthEntry())
    return didImprove;

  activeRange.first[splitDimension] = splitPosition + 1;

  auto borders = (didImproveDistance) ? KNOW_NONE : KNOW_UPPER;

  Interval upper = dataset->dataPointsInInterval(activeRange,
                   interval, borders);

  didImprove = knnSearch(queryObject, activeRange,
                         upper, dataset);

  activeRange.first[splitDimension] = minInDimension;

  return didImprove;
}
```

Fig. 5.8.: C++ code fragment for the recursive routine that visits the sub-ranges in left-right order.

Figure 5.8 shows code for the recursive routine that visits the sub-ranges in left-right order. Code for the right-left traversal works analogously.

```
bool fullScanSearch(const Point& queryObject, HyperCube&
activeRange, Interval interval, DataSet* dataset)
{
  bool didImprove = false;
  for (int i = interval.first; i <= interval.last; i++)
  {
    Point& dataPoint = dataset->getDataPointAtIntex(i);
    float distance = norm->distance(queryObject, dataPoint);
    if (distance < kDistanceEntry())
    {
      addToKnnList(KNNCandidate(distance, dataPoint.key));
      didImprove = true;
    }
  }
  if (didImprove)
  {
    HyperCube newBoundingRange(queryObject, queryObject);
    newBoundingRange.extend((int) std::ceilf(
                              distanceToKthEntry()));
    activeRange.crop(newBoundingRange);
  }
  return didImprove;
}
```

Fig. 5.9.: C++ code fragment for cropping of the active range.

The code segment depicted in Figure 5.9 is used at the lower levels of the recursion, when the number of patches in the active interval falls below the threshold μ. Whenever the result is improved, i.e. when a new entry is added to the knn list, the active range is cropped. As the active range is passed by reference, this also reduces the active range on the previous (higher) levels of recursion.

Parallelisation. The traversal code given above can be parallelised to run efficiently on multi-core processors. However, a naïve implementation results in poor performance as the subdivision into regions gives a strongly unbalanced partitioning of the search space. In order to achieve a reasonable load balancing, a fine granular parallelisation is used that searches two sub-regions in parallel only if both regions are above a minimum size. This fine granular subdivision means, however, that the overhead to starting a thread per sub-region makes it prohibitively slow. The problem can be solved by applying a thread pool design pattern [97]. Different from the original design pattern, we implemented a prioritised job queue where the distance between the query object and the traversed sub-region is used as a priority. This means closer regions are always traversed first, independent of the order in which they were inserted into the queue. This increases the chance of pruning regions and improves performance. The remaining issue is that access to the knn list needs to be synchronised between threads. As read access to this list for finding the current relevant search radius is much more frequent than write access, which only occurs when a new candidate is inserted into the list, the synchronisation can be solved efficiently using the read-write lock design pattern [118].

5.3.3 Evaluation

We evaluated the acceleration approaches based on photographs where regions should be deleted and inpainted afterwards, which is one of the standard applications for inpainting algorithms.

Experimental Environment. We implemented the algorithm described above in the C++ 11 programming language. The programme was compiled with the native compiler of Microsoft Visual Studio 2015 using AVX instruction set (/arch:AVX), fast floating point arithmetic (/fp:fast) and full compiler optimisation (/Ox). Computation of the PCA was implemented based on the Intel DAAL library, which in turn links against the BLAS and LAPACK implementations provided by the Intel MKL library. All runtime experiments were conducted on an Intel Core i7-6700K at 4 GHz with 16 GB memory. The brute force, i.e. full scan from the beginning, experiments on GPU were performed on an Nvidia GeForce GTX 970 programmed using the OpenCL API [106].

Datasets for Evaluation. We used a set of colour images with resolutions of 800×600, $1,600 \times 1,200$, $2,048 \times 1,536$, and $2,560 \times 1,920$ pixels to evaluate our approach. For each size, a set of 10 different images was selected from a publicly available image database [128]. Each image was deteriorated with a mask of text covering $20\,\% \pm 0.1\,\%$ of the pixels.

Results. The performance of the presented acceleration structure for EBI depends on several parameters. Before gathering overall results we analysed quality and performance parameters. We then selected values that constitute a good trade-off for the evaluated image resolutions between both satisfying results of the inpainting quality and fast runtime. Example results are shown in Figure 5.10a-d. A texture synthesis example as a further application of EBI is depicted in Figure 5.10e-g.

Overall Performance. We compared the runtime for several inpainting problems, shown in Table 5.2. As a baseline, we used a highly optimised implementation of the brute-force search on a GPU and an implementation of the original algorithm [24]. All experiments were conducted using the following parameters: $K = 9 \times 9$, $D = 10$, $k = 80$, $\mu = 256$, $\nu = 2,048$, and $c = 0.6$. Explanations of how these parameters were obtained are given below.

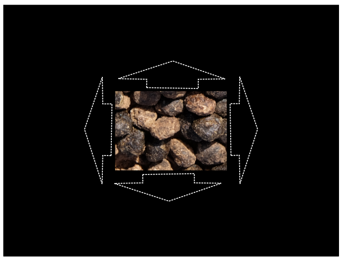
Sorting the Dictionary. The runtime of the acceleration structure also depends on the time it takes to sort the dictionary. We therefore evaluated this sorting time based

(a) Original image.

(b) Example with overlaid mask used to deteriorate the image. The mask consists of $80\,\%$ white pixels and $20\,\%$ black pixels.

(c) Result of inpainting with the GPU optimised brute force inpainting.

(d) Result of inpainting with the accelerated z-curve inpainting.



(e) Example of texture synthesis based on EBI. The image is extended in all directions.

(f) Result of inpainting with the GPU optimised brute force inpainting.

(g) Result of inpainting with the accelerated z-curve inpainting.

Fig. 5.10.: Examples for accelerated inpainting.

Tab. 5.2.: Overall runtime of the algorithm with and without acceleration approaches. Depicted are the average runtime for ten images of each resolution ± the standard deviation of the runtime. The acceleration structure runtime includes the time needed for dictionary learning. Corresponding bar charts are depicted in Figure 5.11.

resolution	original CPU 8 cores	brute force GPU	z-curve CPU 8 cores
800×600	10.9 ± 2.1 min	20.2 ± 0.4 sec	9.7 ± 0.6 sec
$1,600 \times 1,200$	160 ± 6.5 min	153.0 ± 2.0 sec	38.6 ± 1.7 sec
$2,048 \times 1,536$	7.7 ± 0.7 h	6.0 ± 0.05 min	1.1 ± 0.1 min
$2,560 \times 1,920$	22.1 ± 1.4 h	13.3 ± 0.2 min	2.0 ± 0.2 min

on all 40 images contained in the evaluation dataset (Table 5.4). For the standard application, where damaged parts of an image are inpainted with the available parts of the image, the index structures must be built prior to the inpainting procedure. However, there are also applications like sparse scanning microscopy inpainting [246], where a generic dictionary is used, so that index structures can be pre-computed and stored on disc, reducing execution time considerably.

Impact on Inpainting Quality. The proposed multi-index structure computes the best match for a patch approximately. We investigated the impact of this approximation on the quality of the inpainting by computing the L_2 norm of both the z-curve patch and the brute force patch for each iteration. The acceleration error (AE) is then defined as

$$\text{AE} = \frac{E_z}{E_b} - 1 \quad \in [0, \infty) \tag{5.9}$$

where E_z is the z-curve error, and E_b is the brute force error. For normalisation one is subtracted such that a value of zero corresponds to no AE.

Number of Dimensions of Principal Space D and Size of Candidate Set k. Two parameters determine the quality of the inpainting algorithm, the number of kept principal components D and the number of entries k in the knn list. We evaluated values from $D = 4$ to $D = 16$ and from $k = 1$ to $k = 320$ on an image of $1,600 \times 1,200$ pixels resolution. The resulting acceleration error is shown in Table 5.3. The influence of D and k on the runtime of the inpainting is shown in Table 5.4.

We selected $D = 10$ and $k = 80$ for all further experiments, as it represents a reasonable trade-off between quality and performance. This combination had the best runtime under all evaluated combinations with an AE below $1\,\%$.

Optimal Recursion Thresholds μ. The parameter μ controls when the recursion stops. The algorithm then processes the remaining patches in the dataset sequentially.

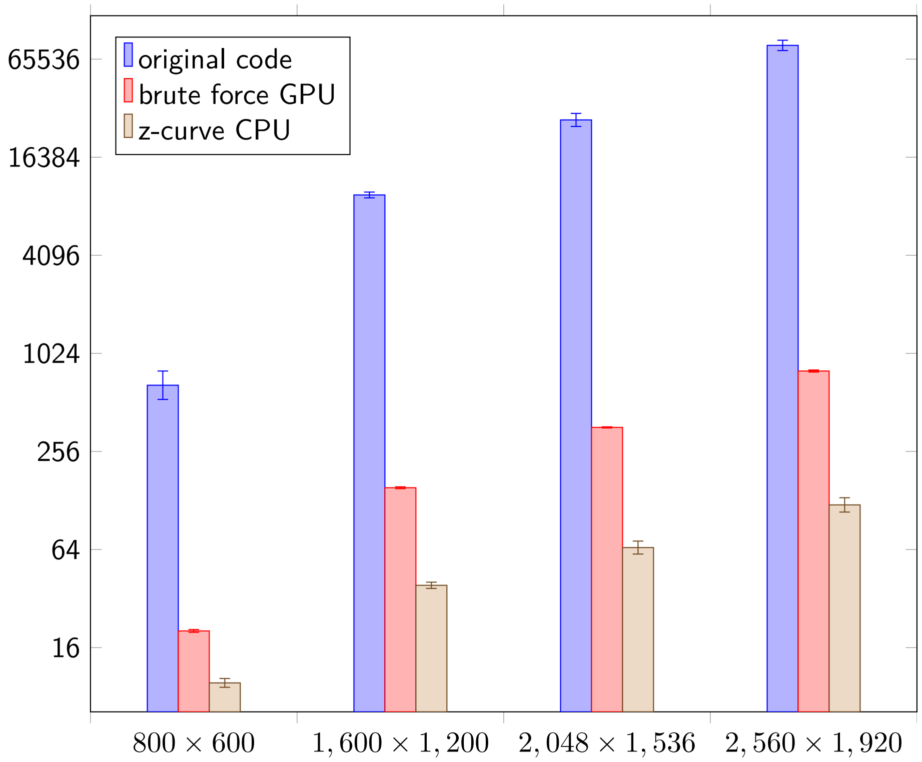


Fig. 5.11.: Overall runtime of the algorithm with and without acceleration structure as bar charts on a logarithmic scale ($\log_2$) corresponding to Table 5.2

We determined an optimal value by running inpainting experiments on one image for each resolution for values of μ ranging from 16 to $4,096$ (Table 5.6). We found that the best choice for the recursion threshold reduces the runtime by up to $40\,\%$. The optimal value was consistently at $\mu = 256$ for any of the evaluated images, so that this value was selected for all further experiments.

Optimal Parallelisation Thresholds ν. Another important performance parameter is the threshold for parallelisation ν. We determined the value by running inpainting experiments on one image for each resolution for values of ν ranging from 16 to $8,192$. An optimal choice of the parallelisation threshold can also speed up the computation by up to $50\,\%$. Optimal values for ν vary between 512 and $4,096$ (Table 5.6). We selected $\nu = 2,048$ for all further experiments, as this value resulted in the fastest runtime for most resolutions.

Optimal Value c. The last performance parameter we determined was the percentage of pixels that are covered by the applied indices. We determined the value for c by running an experiment based on a $2,560 \times 1,920$ resolution image for values between 0.2 and 0.8. Runtime increased monotonically from 120 seconds for $c = 0.2$ to 263 seconds for $c = 0.8$. As values below $c = 0.6$ created clearly visible artefacts we decided to set c to 0.6. Depending on the missing parts in an image that should be inpainted, c

Tab. 5.3.: Acceleration error for different combinations of D and k. Combinations of D and k that result in an AE below $1\,\%$ are considered acceptable by us and are marked light grey. The combination $D = 10$ and $k = 80$ has the best runtime performance of the combinations with AE $< 1\,\%$ and is indicated dark grey (see also Table 5.4).

	D						
k	4	6	8	10	12	14	16
1	72.37	43.93	31.86	25.58	23.19	21.57	20.98
10	32.54	13.82	7.53	5.54	4.64	4.27	4.12
20	25.1	9.13	4.6	3.14	2.64	2.48	2.29
40	19.0	5.77	2.66	1.77	1.47	1.37	1.26
80	14.69	3.58	1.49	0.95	0.85	0.58	0.7
160	9.74	2.03	0.82	0.5	0.44	0.41	0.39
320	7.39	1.08	0.41	0.23	0.25	0.26	0.22

might be set to lower values for faster runtime without much quality loss, e.g. for clear textures or homogeneous areas.

Results for iDistance Index. Additionally to the z-curve acceleration structure, we investigated the use of iDistance [127], which is one of the state of the art multi-dimensional indexing structures. It is used for knn queries, especially when the dimension of the data is very high. We tested iDistance on an image with a resolution of $1,600 \times 1,200$ pixels. We found that the runtime performance was clearly inferior compared to the optimised GPU accelerated implementation of EBI. Hence, we abandoned the idea to use iDistance as a multi-dimensional indexing structure and decided to use the knn search on the SFCs.

Theoretical Performance. For a constant number of dimensions of the principal space, the algorithm has a theoretical runtime of $O(\log(n)^2)$ where n is the size of the dictionary. A proof of the result is provided in the Appendix A. This result holds under the assumption that the cost function is a norm. Furthermore, we obtain that the algorithm scales with $O\left(\sqrt[D]{\left(\frac{D}{2}\right)!}\right)$ in the number of dimensions of the principal space. It therefore seems advisable to select a moderate number of dimensions for the principal space, which is consistent with our experimental results (Table 5.3).

Performance for L_1 norm as Cost Function. We experimentally investigated the performance of the acceleration structure if the L_2 norm is replaced by the L_1 norm. As expected, the acceleration settings had to be adjusted in order to keep the AE in the acceptable range. We changed the values from $D = 10$ and $k = 80$ (Table 5.3) to $D = 14$ and $k = 160$. Using these settings, we obtained an acceleration error of

Tab. 5.4.: Runtime in seconds for different combinations of D and k. Combinations of D and k that result in an AE below 1% are considered acceptable by us and are marked light grey. Additionally, the average time for learning the dictionary is given as separate row. The combination with the best runtime performance and AE $< 1\,\%$ is indicated dark grey.

	D						
k	4	6	8	10	12	14	16
1	11.26	12.21	13.96	16.23	18.78	21.82	25.45
10	12.27	13.83	16.15	19.3	22.15	25.94	30.0
20	13.53	15.11	17.65	21.24	24.4	28.44	32.94
40	15.64	17.72	20.12	24.11	28.39	32.83	37.91
80	21.15	23.85	27.06	31.66	36.41	41.24	46.45
160	35.6	40.3	44.2	50.0	55.35	61.05	66.13
320	65.73	73.46	79.9	90.09	98.43	105.57	112.25
sorting	6.7	7.4	7.7	8.3	8.3	8.6	8.7

approximately $1\,\%$, while taking 62.7 plus 8.0 seconds for reconstruction plus building the indices to inpaint our test image. Thus, the acceleration structure is approximately a factor of 2 slower when using the L_1 norm compared to the L_2 norm.

5.3.4 Acceleration Impact

The implemented acceleration approaches enhance the runtime performance significantly. From a theoretical point of view, the acceleration structure has a runtime in $O(\log(n)^2)$ per query, where n is the size of the dictionary. This is a fundamental improvement over the original inpainting proposed by Criminisi et al. [58], which has a runtime in $O(n)$ per query.

In terms of practical performance gains, implementation and dictionary size play an important role. As a baseline, we implemented a highly-optimised version of the original algorithm that runs in parallel on a GPU. Compared to this baseline, our method achieved a speed-up of a factor of 2 to 7 depending on the size of the inpainted images and hence also the dictionary size. Compared to the original algorithm, we obtained a speed-up factor of up to 660. The real importance of the acceleration structure lies in its logarithmic scalability. Using the proposed method, extending EBI from 2D to 3D datasets with much larger dictionaries becomes a feasible option. This will allow for novel applications, including EBI of video data and EBI of volumetric data, for example in

Tab. 5.5.: Influence of the recursion threshold μ on the runtime of the accelerated inpainting algorithm in seconds. For each resolution, a single image was processed.

	Resolution [pixels]			
μ	800×600	$1,600 \times 1,200$	$2,048 \times 1,536$	$2,560 \times 1,920$
16	9.9	38.2	70.0	135.5
32	8.6	34.1	63.1	115.2
64	7.7	31.1	58.2	105.3
128	7.2	29.2	56.0	100.1
256	7.1	28.5	55.2	97.9
512	7.2	29.0	56.3	98.1
$1,024$	7.4	30.7	58.6	99.0
$2,048$	7.9	32.8	62.0	104.2
$4,096$	8.4	35.2	66.1	110.5

3D microscopy. Furthermore, EBI can be used as a texture synthesis algorithm as shown in [68] and in Figure 5.10e-g.

An interesting aspect of the acceleration structure is that the inpainting requires the construction of several index structures as a preprocessing step. For conventional inpainting, the undamaged parts of the image are used as dictionary. In such scenarios, the index structures need to be built for each inpainting as it was the case for the performance evaluation given in Table 2. For other applications, for example in sparse microscopy, it is possible to reuse dictionaries, which opens a potential for additional performance improvements. On the downside, the need to initialise the indices means that the proposed acceleration structure is virtually useless for cases where only few pixels need to be inpainted, as the cost for the initialisation cannot be amortised over too small a number of query operations.

The proposed acceleration structure is independent of the exact inpainting algorithm in two ways. First, the structure is used to accelerate individual queries. As such, it can be used in combination with any EBI algorithm, independent of the heuristic that defines the order in which patches are processed. Second, the theoretical result of logarithmic runtime was obtained with very mild assumptions about the cost function. Thus, the acceleration structure can be used with any norm as cost function, at least in theory. In practice, we found that the computational cost of evaluating a norm plays an important role. For example, a general L_p norm requires the evaluation of logarithmic and exponential functions and is non-trivial to implement efficiently.

Tab. 5.6.: Influence of the parallelisation threshold ν on the runtime of the accelerated inpainting algorithm in seconds. For each resolution, a single image was processed.

	Resolution [pixels]			
ν	800 × 600	$1,600 \times 1,200$	$2,048 \times 1,536$	$2,560 \times 1,920$
16	10.5	52.7	111.6	159.2
32	9.9	49.7	104.0	153.8
64	9.1	46.4	95.8	139.2
128	8.0	40.4	80.9	125.5
256	7.3	35.0	67.4	105.6
512	7.0	30.7	59.3	91.7
$1,024$	7.1	28.9	56.4	85.1
$2,048$	7.1	28.4	55.6	82.3
$4,096$	7.4	28.9	56.7	82.2
$8,192$	7.6	31.0	58.3	85.7

The main contribution lies in the fact that the acceleration structure is independent from most model-based aspects of the inpainting algorithm. Thus, it can be combined with a large number of existing and future EBI algorithms. The acceleration structure makes the generalisation of EBI to 3D datasets such as video or volumetric data a feasible option.

Inpainting the Missing Wedge

6

> *Sometimes the hardest pieces of a puzzle to assemble, are the ones missing from the box.*
>
> — **Dixie Waters**
> (American Author)

In this chapter the application of exemplar-based inpainting (EBI) to the missing wedge problem is demonstrated. We describe the missing wedge inpainting procedure and evaluate the approach based on different datasets, afterwards results on both an artificial dataset and a real dataset are discussed.

6.1 Workflow for Missing Wedge Data Generation

We found that the capability of EBI to replicate both texture and structure can also be utilised to generate data for the missing tilt angle range. This seemingly surprising result can intuitively be understood from the fact that sinograms generated from a tilt series consist of the superposition of continuous curves. A reasonable guess on how the data in the missing wedge looks should respect continuity, particularly in the angular dimension. Additionally, many samples exhibit some degree of local self-similarity. In the absence of additional data, continuing structures with patterns observed elsewhere in the sinogram results in better estimates than statistical methods are able to achieve. We tried other statistical approaches such as filling in the average grey value or using random distributions of grey values, but both approaches worsened reconstruction results. The EBI method generates data in the missing wedge by computing a likely continuation of the sinogram under the assumption that tilt series are both continuous and to some degree redundant.

In many cases, the transmission electron microscopy (TEM) sample, and thus the electron tomography (ET) data, has an approximately slab-like geometry, which means that the thickness of a specimen as seen by the electron beam differs with the tilt angle. As the missing wedge inpainting assumes a cylindrical geometry, ET data has to be normalised in a preprocessing step. By dividing each projection through its mean intensity, the original geometry is mapped to a cylindrical geometry with uniform distances.

Hereby, one must consider the imaging mode. In high-angle annular dark-field (HAADF) scanning transmission electron microscopy (STEM), the intensity is proportional to the sample thickness, so simple division is sufficient. In bright-field transmission electron microscopy (BF-TEM) the intensity is proportional to the log of the sample thickness and would have to be linearised prior to the division. The normalised data is then used as input for the inpainting algorithm.

Having acquired an ET image stack, empty projections are added to this stack, such that each missing tilt angle is initialised with an empty projection. All empty projections together constitute the target volume Ω that will be filled during the inpainting procedure. The acquired projections that are available as prior knowledge constitute the source volume Φ. An additional mask is given as input to indicate which projections have to be inpainted. The inpainting works as explained in Chapter 5.

The following pseudo-code (Figure 6.1) gives a summary of the inpainting algorithm for three-dimensional (3D) ET data. The final inpainted projections can be used with any reconstruction algorithm to compute a reconstructed volume.

```
Input  : ET image stack $S$, patch size $\Psi$
Output : extended ET image stack that covers full tilt range
normalise $S$ to comply with a cylindrical geometry;
extend $S$ with empty projections;
extract fillfront $F$;
while $F \neq \emptyset$ do
    compute priorities $P(\boldsymbol{v}) \, \forall \boldsymbol{v} \in F$;
    identify $\boldsymbol{v}' = \arg\max_{\boldsymbol{v}} P(\boldsymbol{v})$;
    find $\boldsymbol{w} \in \Phi$ that minimises cost function $K(c)$;
    copy image data from $\Psi_{\boldsymbol{w}}$ to $\Psi_{\boldsymbol{v}'} \, \forall \boldsymbol{v} \in \Psi_{\boldsymbol{v}'} \cap \Omega$;
    update $F$ and $P(\boldsymbol{v}) \, \forall \boldsymbol{v} \in \Psi_{\boldsymbol{v}'}$;
end
```

Fig. 6.1.: Pseudo-code of the 3D missing wedge inpainting algorithm.

6.2 Datasets for Evaluation

For the evaluation of our approach we used datasets of size $256 \times 256 \times N$ and $450 \times 450 \times N$, where N is the number of source region projections. The datasets were (i) the well-known Shepp-Logan phantom [227], which is a standard synthetic test dataset for reconstruction algorithms, (ii) a downsized catalyst dataset, and (iii) a dataset containing colloidal gold particles. The artificial Shepp-Logan phantom is well suited to compare parameters and reconstruction conditions more easily and with higher validity, whereas the real datasets are used to evaluate the algorithms capabilities under real conditions. The 3D Shepp-Logan phantom was created using Matlab [174] and ground truth data was generated for the angular range of ± 90°. All projections of ± 90° in 1° steps constitute the ground truth dataset, and the projections of ± 62° were used as input for the inpainting and constitute the incomplete data. All projections from −90° to −63° and from 63° to 90° were used for the comparison of generated projections and ground truth projections.

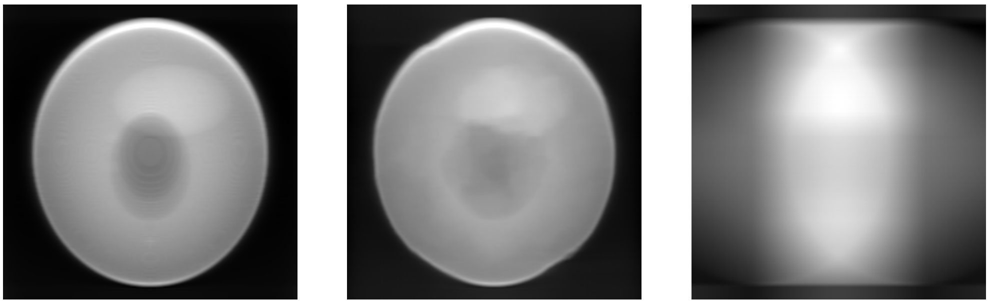
The second dataset was a catalyst dataset. The powder of the $1\,\text{wt.}\,\%$ Au/TiO_2 catalyst was directly dispersed on copper grid with carbon support film. ET was performed using a FEI Titan 80-300 microscope operated at an acceleration voltage of $300\,kV$ in STEM mode. The $2,048 \times 2,048$ pixels STEM images with a pixel size of $0.23\,nm$ were collected with a high-angle annular dark-field detector using the Xplore3D software [221] over a tilt range of ± 76° with a tilt increment of 1°. Alignment of the tilt series was performed using IMOD [136] with the Au nanoparticles as fiducial markers resulting in a residual mean error for the alignment of 0.8 pixel. The aligned projections were downsized to 450×450 pixels resolution to compensate for a long runtime of the algorithm.

For the third dataset, colloidal gold particles with a diameter of $15\,nm$ were applied from aqueous suspension to a TEM carbon coated copper grid, followed by drying in air. Tomography was performed in a BF-TEM electron microscope, operated at $120\,kV$. A tilt series was acquired over a tilt range from −70° to 68° with step increment of 1° and a defocus of $-5\,\mu m$, using the TOMO4 software [220]. The nominal magnification was $23,000$ corresponding to a pixel size of $0.48\,nm$. The alignment of the tilt series was performed in IMOD based on fiducial markers. The aligned tilt series was binned to $0.96\,nm$ (binning factor 2) and then a region of interest of 256×256 was cut out for our study.

6.3 Results

For the evaluation of our algorithm we compared (i) inpainted projections with ground truth projections, (ii) reconstructions of hybrid data, which is the original data with added inpainted projections, and ground truth data, (iii) reconstructions of hybrid data and incomplete data, (iv) the Fourier power spectra of reconstructions of hybrid data and ground truth data, and (v) the Fourier power spectra of reconstructions of hybrid data and incomplete data. Missing wedge effects are clearly visible in the Fourier power spectrum, which makes it a great visual indicator for the applicability of our new method.

We applied the developed algorithm with a fixed patch size of $\Psi = 9 \times 9 \times 9$ to the Shepp-Logan phantom dataset and to the catalyst dataset. Reconstructions were performed



(a) Shepp-Logan phantom, reconstructed from the full tilt range.

(b) Shepp-Logan phantom, reconstructed from hybrid data.

(c) Shepp-Logan phantom, reconstructed from incomplete data (tilt range of ± 62°).

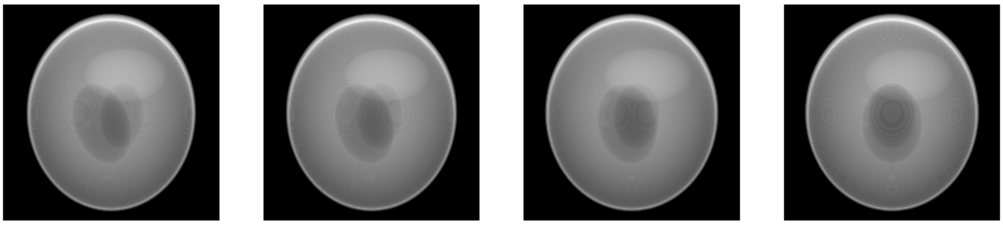
Fig. 6.2.: Comparison of projections from Shepp-Logan phantom reconstructions. For all cases, the image shows the projection at −90°.

with the software package Ettention [32] using simultaneous iterative reconstruction technique (SIRT) with 300 iterations. The reconstruction algorithm and iteration choice were inspired by [21]. Outcomes were evaluated on the basis of the introduced evaluation metrics peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM), introduced in Section 3.6, as well as visual inspection of inpainted projections, reconstructions in xy-direction, resliced reconstructions in xz-direction, and the corresponding Fourier power spectra. Furthermore, we applied the algorithm to the gold particle dataset with varying patch sizes from $\Psi = 5 \times 5 \times 3$ to $\Psi = 21 \times 21 \times 21$. For the gold particle dataset we only compared sinograms of inpainted projections visually to show how particle size and corresponding choice of Ψ influence the inpainting algorithm.

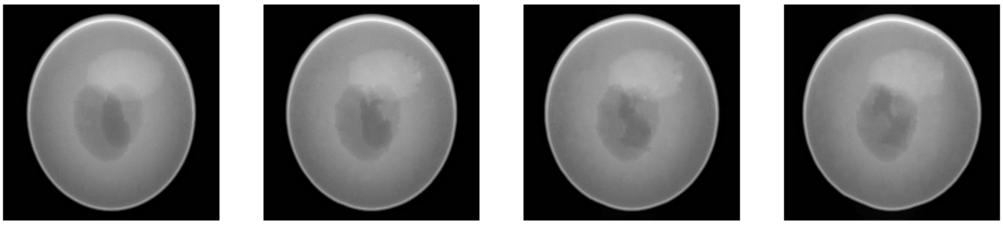
Shepp-Logan phantom. We compared a Shepp-Logan phantom reconstructed from the full tilt range (Figure 6.2a) with one reconstructed from hybrid data (Figure 6.2b) and one reconstructed from incomplete data (Figure 6.2c).

Figure 6.3 shows selected inpainted projections from −63° to −90°. The further away the inpainted projections were from the input of range of ± 62°, the more the inpainted projections deviated from the real ones. Nevertheless, even the projection at −90° showed visual similarity to Shepp-Logan phantom projections. Projections in the upper tilt angle range between 63° and 90° behaved analogously.

Taking the evaluation metrics confirmed the visual inspection results. SSIM and PSNR of the 56 inpainted projections were 0.75 ± 0.1 and 30.91 ± 5.54. Looking at reconstructions reinforced the first impressions from the results of inpainted projections (Figure 6.4). The xy-direction of reconstructions of ground truth data, hybrid data, and incomplete data are shown in Figure 6.4a-c. The reconstruction with incomplete data showed clear artefacts originating from the missing wedge. The round shape of the phantom was



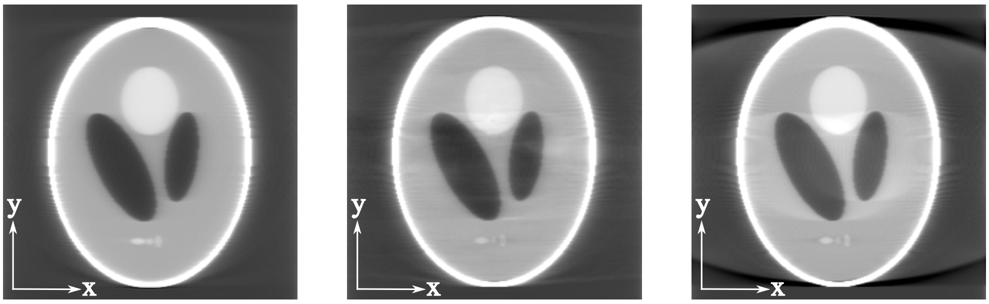
(a) GT −63° (b) GT −70° (c) GT −78° (d) GT −90°



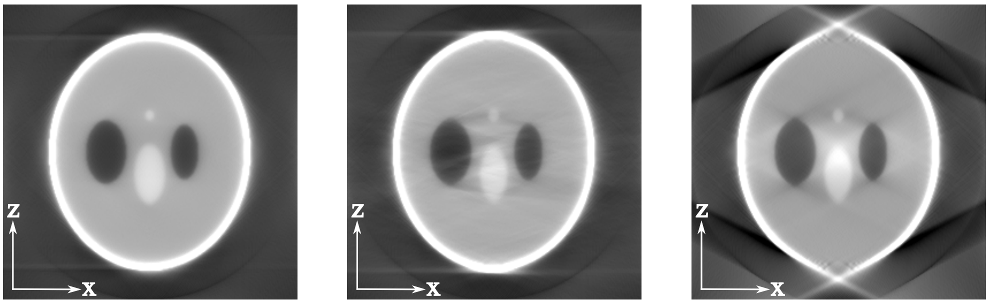
(e) inpainted −63° (f) inpainted −70° (g) inpainted −78° (h) inpainted −90°

Fig. 6.3.: Comparison of several projections between ground truth data (GT) and inpainted data of Shepp-Logan phantom.

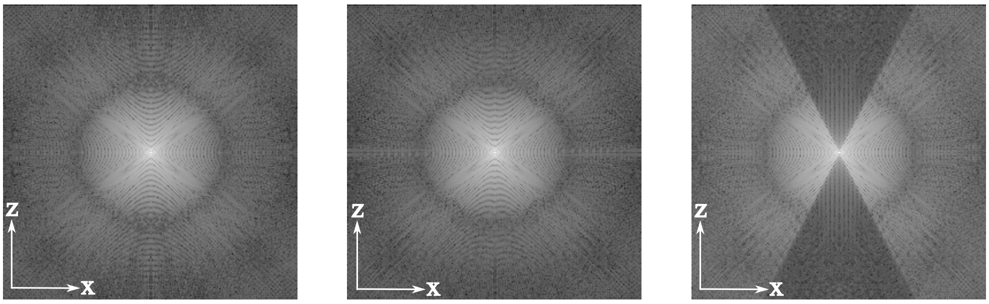
deformed in the missing wedge area. There were also artefacts in the area around the phantom (Figure 6.4f). In the reconstruction from hybrid data (Figure 6.4e) there were hardly any artefacts left from the missing wedge. However, the interior of the phantom shows a new type of artefact, expressed as smearing. A PSNR of 35.53 for the reconstruction from hybrid data confirmed the good quality compared to the incomplete data reconstruction with a PSNR of 18.76. SSIM was higher for the hybrid data reconstruction with 0.9833 than for the incomplete data reconstruction with 0.7624. root mean squared error (RMSE) was reduced by a factor of around 14. Finally, the Fourier power spectrum of the incomplete data (Figure 6.4i), which reflects the missing data by a missing wedge, was completely filled, as can be seen in Figure 6.4g-h. For the Fourier power spectra of the hybrid data reconstruction and the ground truth data reconstruction we acquired a PSNR of 44.36 and a SSIM of 0.9611.



(a) GT in xy-direction **(b)** hybrid in xy-direction **(c)** incomplete in xy

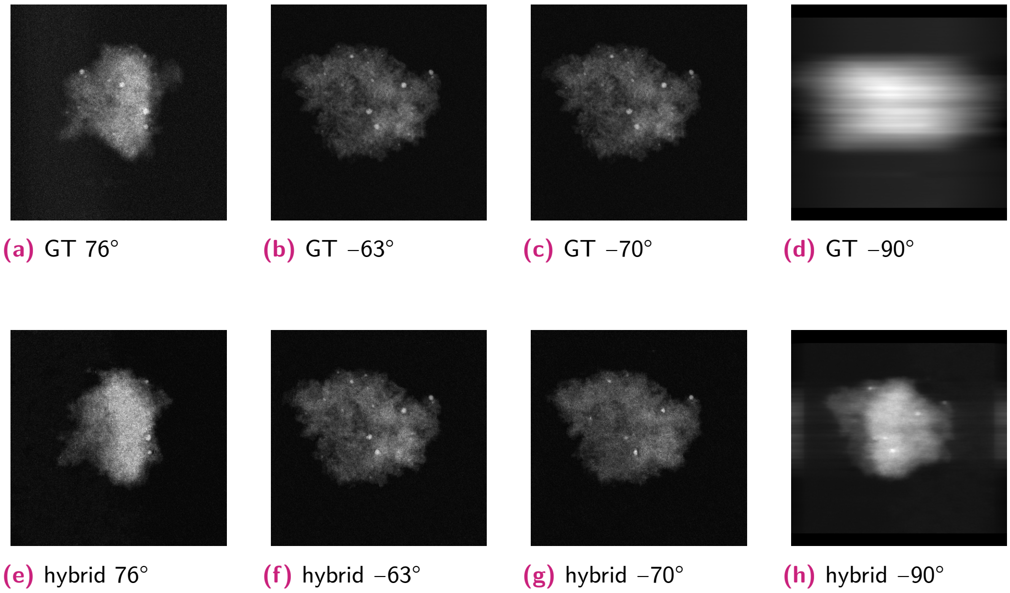


(d) GT in xz-direction **(e)** hybrid in xz-direction **(f)** incomplete in xz



(g) GT Fourier Spectrum in xz-direction **(h)** hybrid Fourier Spectrum in xz-direction **(i)** incomplete Fourier Spectrum in xz-direction

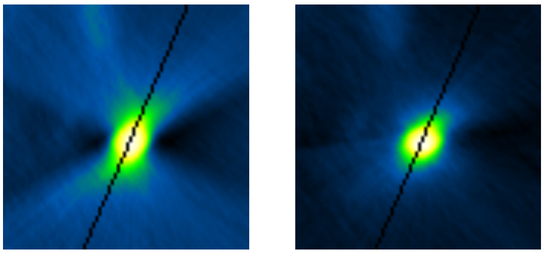
Fig. 6.4.: Comparison of reconstruction results of Shepp-Logan phantom. Shown are reconstructions from ground truth data (GT), from hybrid data, and from incomplete data.


(a) GT 76° (b) GT −63° (c) GT −70° (d) GT −90°

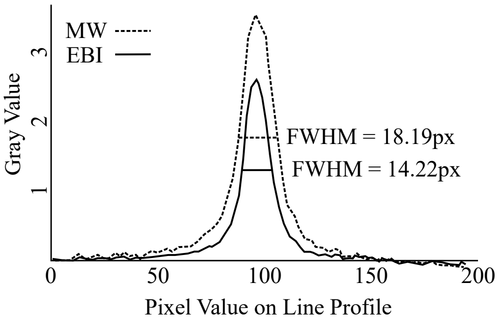
(e) hybrid 76° (f) hybrid −63° (g) hybrid −70° (h) hybrid −90°

Fig. 6.5.: Comparison of several projections between ground truth (GT) and hybrid catalyst data. The −90° projections were acquired from a virtual projection of the reconstructed data.

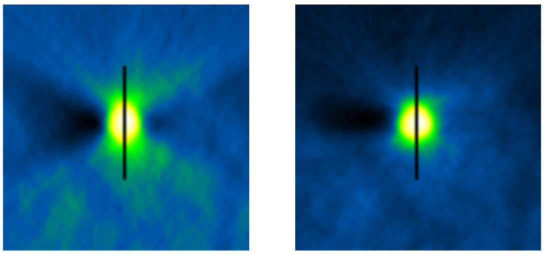
Catalyst data. The catalyst data had only projections in the range of ±76° due to mechanical limitations of the specimen holder and the supporting copper grid. From this dataset, we selected a range of projections from ±62° to make the dataset comparable to our other experiments. Figure 6.5 shows selected inpainted projections and the corresponding ground truth, respectively virtual ground truth. As there is no ground truth we measured full-width at half-maximum (FWHM) of each particle in the reconstructed datasets to compare resolution. The compared datasets were (i) the reconstruction of the original data with a tilt range of ±62° and (ii) the inpainted hybrid data, which constitutes a tilt range of ±90°. The dataset contained 25 particles that were used to measure FWHM in the reconstructions in xz-direction. The original data reconstructions FWHM was 11.2 ± 3.5 pixels, whereas the inpainted data reconstructions FWHM was 9.9 ± 2.6 pixels. To ensure that the result is statistically significant, a paired two-tailed t-test [234] was conducted, which resulted in a p-value of $3 \cdot 10^{-5}$. Two particles and the corresponding line profiles are depicted in Figure 6.6. Additionally, we conducted a size analysis of the particles in 2D based on the 0° projection. The average particle size was 8.8 ± 2.6 pixels, which shows that the smaller particle sizes of the hybrid data reconstructions are closer to the real sizes.



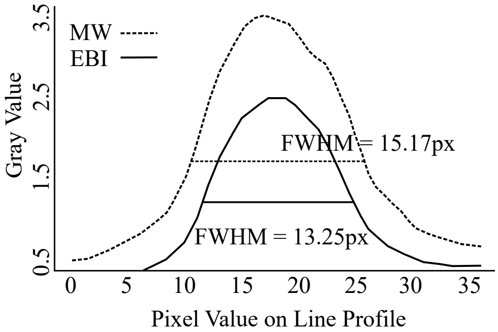
(a) incomplete (b) inpainted



(c) Line profiles and FWHM corresponding to Figures 6.6a and 6.6b.



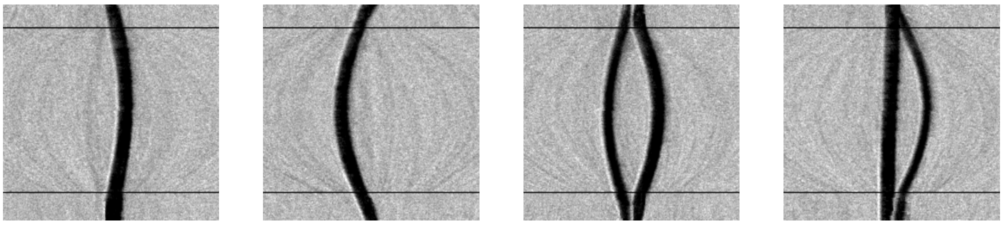
(d) incomplete (e) inpainted



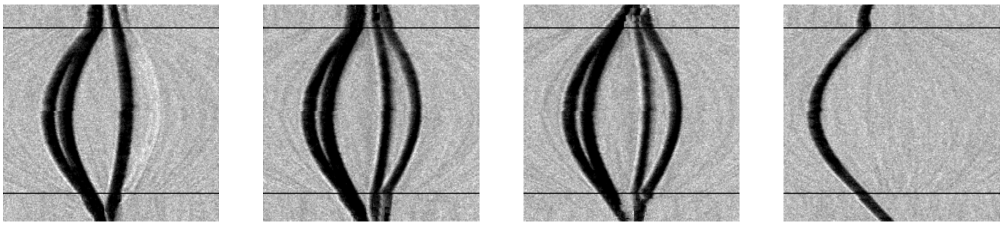
(f) Line profiles and FWHM corresponding to Figures 6.6d and 6.6e.

Fig. 6.6.: Particle comparison and corresponding line profiles with FWHM values. The particles in xz-direction were reconstructed with the incomplete dataset ranging from −62° to 62° depicted as MW and with the inpainted hybrid dataset depicted as EBI. FWHM values are much smaller for the inpainted reconstructions.

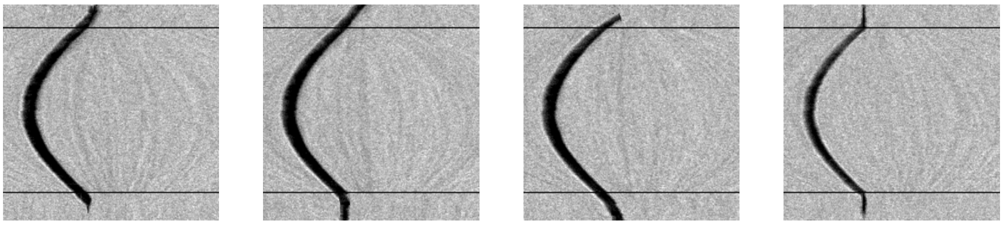
Colloidal gold particles. Furthermore, we acquired a real dataset of gold particles and evaluated the inpainting algorithm with different patch sizes $\Psi_x \times \Psi_y \times \Psi_z$. We hand-picked different worst-case situations with inpainting artefacts (Figure 6.7) to show that results were heavily influenced by the patch size selection. One black curve constitutes the movement of one gold particle during tilting in the sinogram. The structures in between the two horizontal black lines in each sinogram constitute the original data. The inpainted data is above and below the black lines. We observed different behaviour of the inpainting algorithm when looking at varying patch sizes and individual gold particles. For curves with low curvature, the inpainting continued the curves (Figure 6.7a-e). For all these examples Ψ_z was at least 15, whereas Ψ_x and Ψ_y varied significantly. This means the patch size in the z-direction plays an important role and must be chosen at least as large as the particles in the projections. Figures 6.7f-h show examples of insufficient sizes for Ψ_z. For particles that resemble a high curvature in the sinogram (Figure 6.7i-p) the inpainting algorithm shows problems. These ranged from unreasonable continuation of a curve at one (Figure 6.7j) or both boundaries (Figure 6.7l) of the range to stopping or no continuation of a curve (Figure 6.7p) as well as the introduction of artefacts (Figure 6.7o). The presented approach mainly aims to recover linear structures, that means the higher the curvature of an object is, the less likely it is that the algorithm works. Hence, the left curve in Figure 6.7m-p was never recovered, no matter how the patch size had been chosen.



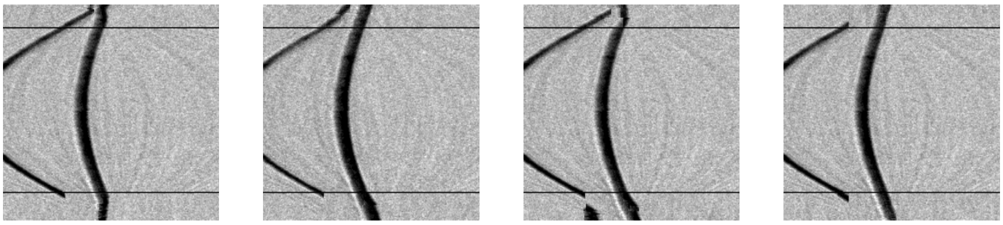
(a) $\Psi = 7 \times 7 \times 15$, slice 103 **(b)** $\Psi = 13 \times 13 \times 19$, slice 216 **(c)** $\Psi = 5 \times 5 \times 21$, slice 132 **(d)** $\Psi = 13 \times 13 \times 15$, slice 47



(e) $\Psi = 9 \times 9 \times 15$, slice 72 **(f)** $\Psi = 11 \times 11 \times 11$, slice 67 **(g)** $\Psi = 15 \times 15 \times 9$, slice 66 **(h)** $\Psi = 19 \times 19 \times 9$, slice 174



(i) $\Psi = 9 \times 9 \times 11$, slice 243 **(j)** $\Psi = 13 \times 13 \times 21$, slice 246 **(k)** $\Psi = 19 \times 19 \times 9$, slice 248 **(l)** $\Psi = 7 \times 7 \times 7$, slice 251



(m) $\Psi = 19 \times 19 \times 9$, slice 224 **(n)** $\Psi = 17 \times 17 \times 21$, slice 223 **(o)** $\Psi = 19 \times 19 \times 15$, slice 224 **(p)** $\Psi = 11 \times 11 \times 21$, slice 224

Fig. 6.7.: Effect of the selection of the patch size parameters on the inpainting results. Inpainted data is situated above and below the black horizontal lines generated with projections from −70° to 68°, which is depicted in between the black horizontal lines.

6.4 Discussion

We developed a new method to reduce missing wedge artefacts and evaluated it on several datasets. The evaluation was conducted by visual comparison of projections and reconstructions as well as by taking several evaluation metrics for judging the quality of the results quantitatively.

Quality performance

The Shepp-Logan phantom served as an artificial ground truth dataset to be able to assess the performance of the inpainting algorithm. The inpainted projections continued the tilt series reasonably. This was also confirmed by the enhanced reconstruction results, both visually and by means of the evaluation metrics. The algorithm enhanced structure identification of the ellipses that the phantom is built of. However, there were also new artefacts in the hybrid data reconstruction, which could be seen as smearing inside the phantom. This is a concern, as the structure of typical missing wedge artefacts is familiar to most tomography users and can easily be recognised as such. The introduction of new, per se unfamiliar artefacts introduces the risk of mistaking them for structures inside the sample. However, the newly introduced artefacts are much less pronounced than the typical missing wedge artefacts and thus can be acceptable or not depending on the application.

Applying the new method to a real catalyst dataset supported the positive results of the Shepp-Logan phantom. The catalyst dataset contained 25 particles whose resolution was enhanced by applying the inpainting procedure as preprocessing to the reconstruction. This was visible when comparing the reconstruction results and confirmed by the significantly reduced FWHM values for the hybrid data reconstruction. The shape of the particles was ellipsoidal in the incomplete data reconstructions, whereas the particles in the hybrid data reconstruction appeared spherical, which shows that elongation artefacts caused by the missing data were significantly reduced due to the inpainting procedure. Even small particles gained resolution, which facilitated detection. Furthermore, we could not identify new artefacts as for example the smearing in the Shepp-Logan phantom.

An exploration of the parameter space, which means different patch sizes, was conducted to give further insights on improvements of the method. In general, the patch size should be at least as large as the size of the features in question. Recovered areas of the gold particle dataset seemed to depend on the patch size, i.e. even in the same dataset, different areas achieved best results with different patch sizes. A deeper understanding of this phenomenon could lead to improvements in the method.

Runtime performance

The experiments also showed that runtime is an issue that must be addressed. On an AMD Radeon R9 390, the method took around 90 minutes on $256 \times 256 \times 181$ resolution datasets, where 56 projections of size 256×256 had to be inpainted. Further experiments showed that this scales quadratically with increasing resolution. In addition to implementing the main parts of the algorithm on graphics processing unit (GPU), there is much potential for further improvements towards shorter execution times. Preliminary results indicate that the issue can alternatively be addressed with the introduction of novel acceleration structures for EBI [68], as described in Section 5.3.

Application domain

In the performed experiments, inpainted and real projections complemented each other reasonably. Also the evaluation metrics showed the gain of the method based on artificial and real datasets. For the Shepp-Logan phantom, visual results already sufficed to see that the new method has a positive effect. However, while experimenting with the Shepp-Logan phantom data, it also showed that the new method became less advantageous when the input dataset contained few projections. Such datasets are acquired when tilt angle steps are quite large, e.g. 3° steps or higher, or when the tilt angle range is very small, e.g. from −30° to 30°. Therefore, this kind of data cannot provide enough information, respectively a big enough source region. Furthermore, large tilt increments lead to gaps between two projections that falsify neighbourhood relationships, which impedes a meaningful inpainting. In such situations the new method brings no benefits and should not be applied as is.

For the catalyst data, comparing the reconstructions of the particles visually as well as with FWHM revealed the positive impact on a material science dataset. This may be an ideal use case for our method, where the focus lies in analysing contained particles. The parameter space exploration further showed that there is a lot of potential to enhance performance for such kinds of data.

For example, in many biological samples the inpainting artefacts pose a risk for misinterpretation. Contrary, in material science samples, where the primary interest is the morphology of a microstructure, the typical missing wedge artefacts hide crucial questions, such as the connection or separation of phase grains.

The new algorithm is capable of heuristically generating appropriate data for the missing wedge. The method acts as preprocessing before a subsequent tomographic reconstruction and adds further projections to the original input data. Integrating these into a reconstruction enhances results by reducing the missing wedge, which leads to fewer artefacts in the final reconstruction.

Furthermore, elongation artefacts in xz-direction are reduced, which enhances resolution. Performance issues and the possible introduction of new artefacts are the main concerns of the new method. Regarding performance, we developed acceleration approaches to speed up EBI, as shown in Section 5.3.

Adaptive algorithms that work on the basis of a local neighbourhood may be advantageous. Extending the inpainting algorithm, such that it becomes specialised for electron microscopy (EM) data, in the sense that the sinusoidal movement of individual particles in the projections as seen in a sinogram is taken into consideration, could also enhance results when used as prior for salient structure completions.

When using the method, the artefacts that are introduced must be kept in mind. An additional comparison with original reconstructions may help to prevent wrong conclusions due to introduced artefacts. As soon as runtime performance has reached an acceptable level for daily application, the missing wedge inpainting may become an important preprocessing step before subsequent reconstructions of material science tasks that involve limited angle electron tomography.

Considering the conducted experiments, the new method currently works well for datasets with small equidistant tilt angle steps of around 1° and a range that includes at least the range from $-60°$ to $60°$. Further research must be performed to improve runtime performance on bigger datasets and to get a better understanding of EBI for solving the missing wedge problem.

7

Optimal Dose Distribution in Scanning Electron Microscopy

> *Give me six hours to chop down a tree and I will spend the first four sharpening the axe*
>
> — **Abraham Lincoln**
> (16^{th} President of the USA)

We investigated how dwell time can be used optimally [242, 244]. To this end, we assumed a fixed budget of total dwell time for the acquisition of an image. We compared three strategies to acquire the best possible image with a fixed budget. Furthermore, we evaluated each of the strategies with different beam currents. The strategies were: (1) scan each pixel in a raster scan with shorter dwell time and address the resulting noise using denoising algorithms, (2) increase the pixel size and pixel spacing and address the reduced resolution using super-resolution (SR) algorithms, and (3) scan a randomly selected subset of all pixels and synthetically generate the missing data using inpainting algorithms. Instead of concentrating on low dose imaging [44], we fixed the available time per frame for the whole acquisition in this study. The main focus of this study was to determine an optimal way for the distribution of dwell time to maximise the information content of final reconstructions.

7.1 Strategies to Reduce Dose and Time

There are different approaches that can reduce the acquisition time of a volume. We applied the following strategies.

Strategy 1: Reduced Dwell Time Raster Scan plus Denoising
The simplest way of reducing the dwell time per pixel scales linearly with the whole acquisition time. However, as each pixel value is determined from a decreasing number of electrons, the statistics of the measurements leads to decreased precision manifested as shot noise. This problem can be addressed in software by applying a denoising algorithm to the image. Recently, algorithms that use prior knowledge in the form of previously learned dictionaries have been established as state-of-the-art in denoising.

One such algorithm, inspired by compressed sensing (CS) is geometric analysis operator learning (GOAL) [114]. In GOAL, image reconstruction underlies the assumption that natural images have a sparse representation over a so called dictionary. A dictionary is a collection of small image patches of a fixed size, e.g. 8×8 pixels. GOAL uses the dictionary as basis of the image patch space, so that linear combinations of only few dictionary entries resemble patches to be inserted. Instead of using mathematical models like Wavelets [53] or Curvelets [34] as dictionary, GOAL uses a representative training set to learn a maximally sparse representation over this training set as dictionary. This is conducted via convex optimisation and conjugate gradients. The learned dictionary, respectively analysis operator, is then used to denoise an image.

Strategy 2: Low Resolution Raster Scan plus super-resolution

Rather than reducing the dwell time per pixel, it is also possible to decrease the number of pixels. By increasing the pixel size and scanning the specimen on a regular grid in larger steps, the sample is discretised to fewer pixel values and acquisition time can be substantially lowered while maintaining the same measurement precision per pixel. Obviously, image resolution is lost as high frequency details in the image fall below the Nyquist threshold. This issue can be addressed by SR approaches, which may help to retrieve the original resolution. Given a low resolution image of size $M \times N$, SR image reconstruction approaches aim to increase the resolution to a higher resolution image of, for example, size $2M \times 2N$, while not decreasing quality. In our study, the method from Villena et al. [260], which is based on the robust SR method using bilateral Total Variation priors [91], was applied. The authors used a hierarchical Bayesian framework to minimise a linear convex combination of the Kullback-Leibler divergence to find a unique approximation of the high resolution image.

Strategy 3: Random Sparse Scan plus Inpainting

Instead of reducing dwell time per pixel or increasing pixel size, it is also possible to reduce the number of pixels that are scanned without increasing the pixel size. This is done by randomly selecting a subset of the image pixels and scanning only this subset. These pixels are then used to reconstruct the image using inpainting algorithms. Similar concepts have already been demonstrated in the context of scanning transmission electron microscopy (STEM) [250]. A larger number of inpainting methods has been proposed including relatively straight forward interpolation methods, methods inspired by CS theory and exemplar-based inpainting (EBI), or methods inspired by image editing. We evaluated several inpainting methods: (1) Natural neighbour interpolation [228], which is a simple interpolation method for scattered data based on Voronoi tessellation [181], (2) beta process factor analysis (BPFA) [281, 282] and (3) GOAL [114], which both are inpainting algorithms inspired by CS, and (4) our version of EBI.

Strategy 3.1: Natural Neighbour Interpolation

Given a sparsely sampled image, unknown pixels are inserted by weighting the areas of surrounding Voronoi cells. This is realised by inserting one Voronoi cell at a time for a new pixel, whose value is a weighted average of the Voronoi cells originally present in the area of the new cell. This kind of interpolation delivers smoother results than bilinear or bicubic interpolation.

Strategies 3.2 and 3.3: Compressed Sensing Inpainting

Beneath interpolation, there are more sophisticated methods for inpainting that are inspired by ideas from CS [185]. Binev et al. [27] have examined, which parts of CS may be useful for EM. The authors analysed different setups in EM where CS may have a useful impact. Stevens et al. [230] applied CS via Bayesian dictionary learning to high-resolution STEM images. The developed BPFA algorithm is a non-parametric Bayesian method that has been extended for image processing applications by Zhou et al. [282]. Given incomplete measurements, spatial inter-relationships are exploited using the Dirichlet process and probit-stick-breaking priors. This means that known pixels of the sparse image are used to infer an appropriate dictionary that is used for reconstruction of the missing pixels. It was shown that $5\,\%$ of the pixels of an image can suffice to recover the original image by applying inpainting. Anderson et al. [7] have demonstrated sparse sampling in an operational scanning electron microscope (SEM). They were able to speed up acquisition time by a factor of three while preserving an acceptable image quality. A randomly selected subset of pixels was used to reconstruct the original image by applying a Split Bregman formulation of regularised basis pursuit (BP) that leveraged block discrete cosine transform as a sparsifying basis, which is a CS method. GOAL [114] can also be used to recover sparsely sampled data via inpainting.

Strategy 3.4: Exemplar-based Inpainting

A further approach for the reconstruction of sparse data is EBI [58], which is a class of inpainting algorithms known from image processing as described in Section 4.2 and Chapter 5. There, the method is used to restore damaged regions of images or to remove objects from images by inserting information from the surroundings. Missing values are modified, so that the inserted parts match the surroundings visually. Adapting this idea, we developed a three-dimensional (3D) inpainting method [243] that inserts voxels with the help of prior knowledge. Instead of using the surroundings of missing voxels, a dictionary is trained based on uncorrupted data. This dictionary contains a large number of image patches with a predefined size from comparable images to the microscope data. Missing voxels are reconstructed iteratively by inserting whole patches. The procedure starts by identifying a region in the sparse image where a patch should be inserted. The scanned voxels in that patch are then used to find a patch in the dictionary that fits best. A cost function is used to determine this patch, for example the L_2 norm of known voxels. Missing voxels are inserted into the sparse image until the whole image has been reconstructed.

Orthogonal Strategy: Beam Current

Increasing the number of electrons per pixel without increasing the dwell time is an additional option, which can only be achieved by an increased beam current. This has the advantage of improving electron statistics without inducing additional acquisition time. However, the disadvantage is that the virtual spot size of the electron gun increases and the image appears blurred. The influence of increased beam current is evaluated in combination with each of the presented strategies.

7.2 Datasets for Evaluation

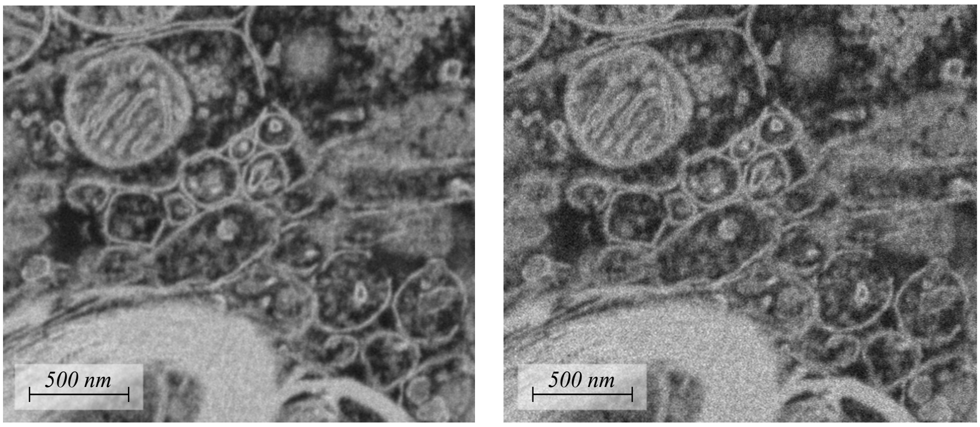
We acquired nano-scale cell biology images on a SEM, the FEI Helios 650 dual beam system. We used the optical ultra high resolution (UHR) mode at 4 mm working distance in high vacuum. The electron landing energy was 2kV using the through the lens detector (TLD) in the back scatter electron (BSE) detection mode. A block specimen of a mouse brain was selected for imaging. The same frame was scanned repeatedly at a resolution of $1,024 \times 884$ pixels using different dwell times of $10\,\mu$s and $30\,\mu$s and different beam currents of 0.1 nA, 0.2 nA, 0.4 nA, and 0.8 nA. After image registration we selected varying regions of interest within these scans with a size of 480×424 for further processing (Figure 7.1).

Based on those datasets, we simulated all acquisition schemes investigated in this study as follows. Combining the $10\,\mu$s and $30\,\mu$s images for each spot, we synthetically generated $40\,\mu$s images. That means the weighted average of three times the $30\,\mu$s image and one time the $10\,\mu$s image was computed. One spot with $40\,\mu$s dwell time and 0.1 nA beam current was chosen as ground truth for the evaluation. This ground truth was also slightly smoothed with Gaussian smoothing using $\sigma = 0.5$. This was done as it is quite common to post-process microscope acquisitions before using the images for further processing to remove small perturbations caused by noise.

Low resolution scans were simulated by downscaling the images by a factor of two. Random sparse scans were simulated by blanking all but a randomly selected set of pixels and setting the blanked pixel values to not a number (NaN). The different strategies and corresponding reconstruction methods were applied to images of the selected area of the sample using different beam currents, as previously mentioned, while keeping the average dwell time per pixel for each acquisition constant at $10\,\mu$s. This corresponds to an acquisition time of around two seconds per spot.

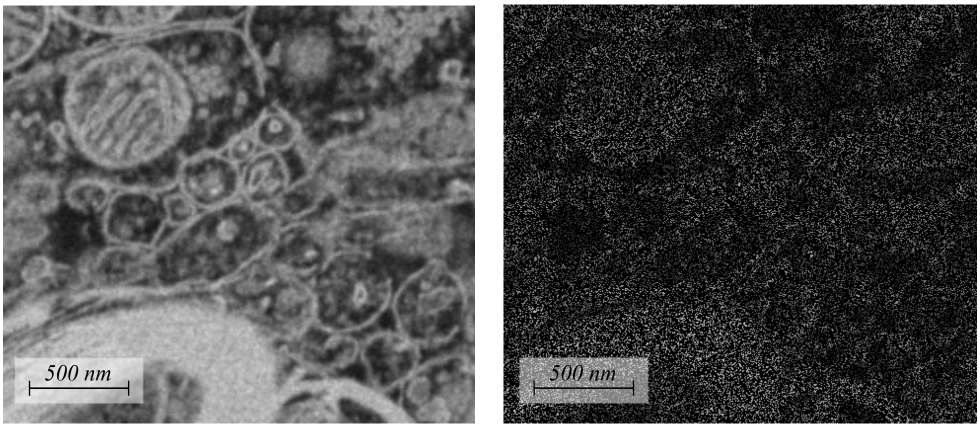
7.3 Algorithmic Setup

We investigated different approaches to acquire data from a SEM given a fixed acquisition time per frame. (1) Raster scans with a reduced dwell time per pixel of $10\,\mu$s (Figure 7.1b) and a pixel size of $5\,$nm were acquired. The unprocessed images are called "Unprocessed Raster", a version processed using GOAL for denoising is denoted "GOAL denoising". (2) Raster scans with half of the original resolution were simulated from the corresponding full frame images at $40\,\mu$s dwell time and $5\,$nm pixel size (Figure 7.1c), which corresponds to $10\,$nm pixel size. The resulting images were then upscaled using the described SR approach from [260] to increase the resolution to the original size. The corresponding dataset is denoted SR. (3) Sparse scans with $25\,\%$ randomly selected pixels were simulated from the $40\,\mu$s dwell time images at $5\,$nm pixel size (Figure 7.1d). The resulting images were reconstructed using different inpainting methods, which includes interpolation. Linear, bicubic, nearest neighbour, and natural neighbour interpolation were applied. As natural neighbour interpolation proved to be the best interpolation method for the data used in this study, only natural neighbour interpolation results are shown (NN interpolation). The compared inpainting methods were EBI, GOAL, and BPFA.



(a) Ground truth at $40\,\mu$s and a pixel size of 5 nm.

(b) Raster scan with a reduced dwell time per pixel of $10\,\mu$s and a pixel size of 5 nm.



(c) Raster scan with a dwell time per pixel of $40\,\mu$s and an increased pixel size of 10 nm (downscaled image contains only $25\,\%$ as many pixels as the ground truth).

(d) Sparse scan with a dwell time per pixel of $40\,\mu$s and a pixel size of 5 nm that only contains $25\,\%$ of randomly selected pixels.

Fig. 7.1.: Input for different strategies to save dwell time.

7.4 Results

Evaluation metrics on the whole image were misleading. Hence, in order to investigate how well the different acquisition methods preserve small structures, we manually selected 30 different locations in the dataset. A quantitative evaluation of the reconstruction quality is provided in Table 7.1.

Tab. 7.1.: Quantitative performance metrics applied to datasets obtained using different sampling strategies and reconstruction algorithms. 30 structures, like the ones in Figure 7.2, were evaluated and results are given as mean $\pm$ standard deviation σ. For each metric, the method with the highest value and all methods within $\pm\sigma$ are indicated in bold letters.

	PSNR	PSNR-HVS-M	SSIM	CW-SSIM
Unprocessed Raster	25.57 ± 0.60	$\mathbf{27.35 \pm 0.96}$	0.726 ± 0.061	$\mathbf{0.890 \pm 0.056}$
GOAL denoising	27.20 ± 0.91	23.81 ± 1.06	$\mathbf{0.888 \pm 0.038}$	$\mathbf{0.905 \pm 0.050}$
SR	25.48 ± 0.71	21.37 ± 0.91	0.814 ± 0.047	$\mathbf{0.886 \pm 0.056}$
NN Interpolation	27.38 ± 1.18	23.91 ± 1.36	$\mathbf{0.885 \pm 0.034}$	$\mathbf{0.891 \pm 0.046}$
GOAL Inpainting	27.37 ± 1.08	23.85 ± 1.19	$\mathbf{0.894 \pm 0.034}$	$\mathbf{0.894 \pm 0.050}$
EBI	25.86 ± 0.97	23.01 ± 1.24	0.789 ± 0.053	0.829 ± 0.064
BPFA Inpainting	$\mathbf{31.34 \pm 0.77}$	$\mathbf{29.22 \pm 1.42}$	$\mathbf{0.885 \pm 0.040}$	$\mathbf{0.891 \pm 0.044}$

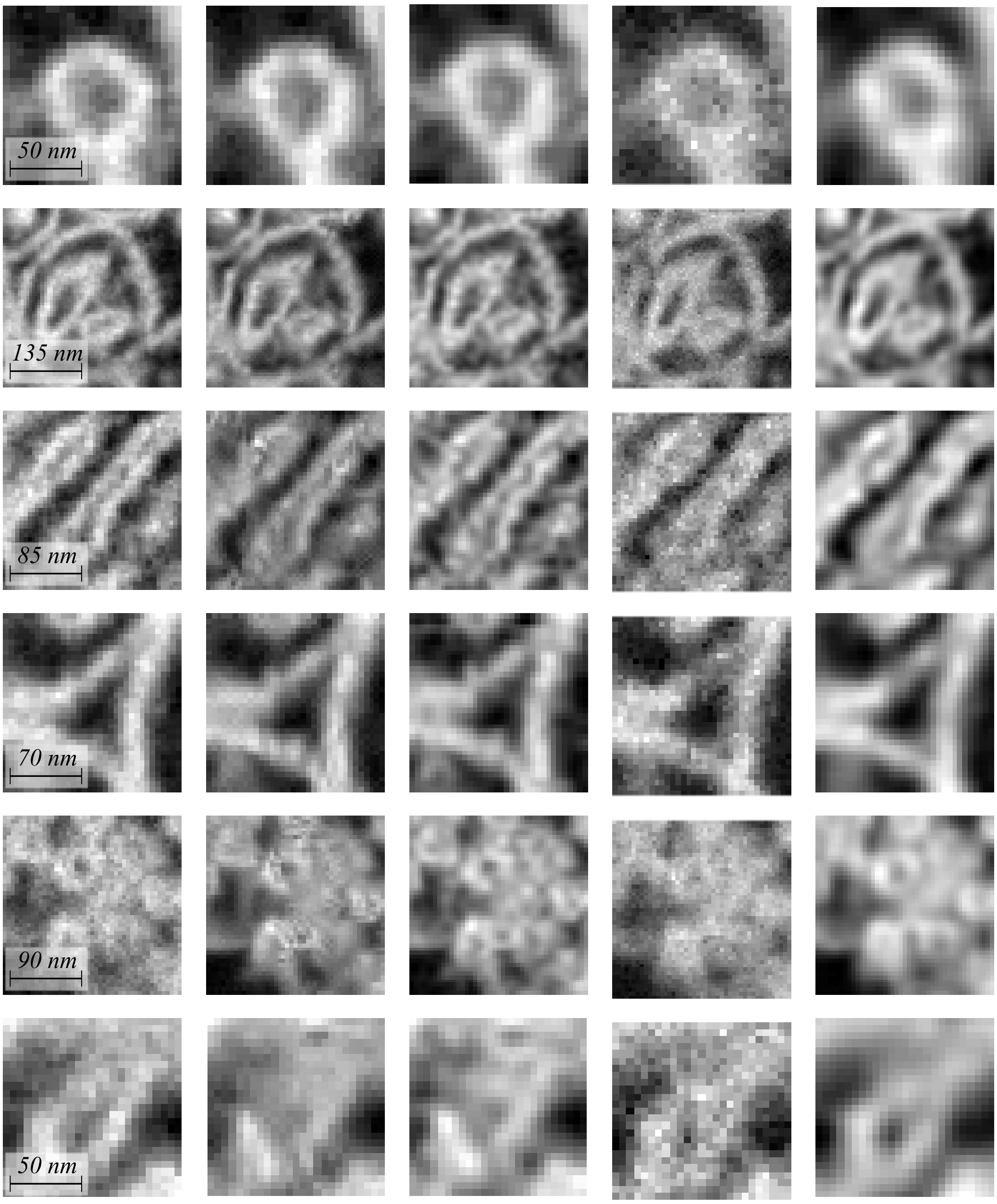
A conventional raster scan with a dwell time of $10\,\mu$s per pixel was used as a baseline for the results. This approach resulted in a peak signal-to-noise ratio (PSNR) of 25.57 ± 0.60, a masked peak signal-to-noise-ratio adapted to the human visual system (PSNR-HVS-M) of 27.35 ± 0.96, structural similarity index (SSIM) of 0.726 ± 0.061, and a complex wavelet structural similarity index (CW-SSIM) of 0.890 ± 0.056. Applying a state-of-the art denoising using the GOAL operator improved only some of the metrics (PSNR $=$ 27.20 ± 0.91) but reduced PSNR-HVS-M to 23.81 ± 1.06.

Processing a low resolution image of 10 nm pixel size with a state-of-the art SR algorithm gave inconclusive results as well. PSNR, SSIM and CW-SSIM were statistically the same, PSNR-HVS-M was worse compared to the unprocessed raster scan.

Better results were obtained by randomly selecting $25\,\%$ of the pixels at 5 nm pixel size and $40\,\mu$s pixel dwell time, followed by a reconstruction using inpainting algorithms. While nearest neighbour interpolation, GOAL inpainting and EBI showed slightly better performance than the unprocessed raster scan, all three methods were statistically indistinguishable from the smoothed version obtained by GOAL denoising. Best results were obtained using BPFA inpainting. The method resulted in the highest PSNR value of 31.34 ± 0.77, the highest PSNR-HVS-M of 29.22 ± 1.42 and was among the highest SSIM and CW-SSIM values.

We found that the different measures did not agree in all cases, nor did the quantitative measures always correspond to our subjective impression of image quality. Therefore, results for some representative structures are shown in Figure 7.2. As can be seen, the applied acquisition and reconstruction strategies generate visually different results. A particularly challenging structure is shown in Figure 7.2, last row. The image excerpt shows a closed, droplet-like structure with a clear interior. The results from GOAL denoising and SR show a deviating structure, where the interior is not identifiable, nor is the structure closed. In the results obtained with EBI, a closed structure is visible. However, the interior is not clearly identifiable as it could also be noise and it is smaller than in the ground truth. In the results obtained with BPFA inpainting, the structure is reconstructed more faithfully in terms of a closed structure and a clear interior, however, reconstructed images are smoother than the ground truth.

For a further visual comparison we computed line profiles through the structures of Figure 7.2, depicted in Figures 7.3 and 7.4.



(a) Ground truth (raster scan at 5 nm pixel size and 40 μs dwell time per pixel).

(b) Raster scan acquired at 10 μs dwell time per pixel, denoised using GOAL.

(c) Low resolution image acquired at 10 nm pixel size and 40 μs dwell time per pixel, processed using a SR algorithm.

(d) Random sparse scan of 25 % pixels at 5 nm pixel size and 40 μs dwell time per pixel, reconstructed using EBI.

(e) Acquisition as in d), but reconstructed using BPFA.

Fig. 7.2.: Impact of the different acquisition strategies and reconstruction algorithms shown on some representative structures. All methods b)-e) use the same total per frame acquisition time budget of 10 μs dwell time per pixel on average.

Influence of Beam Current. In addition to investigating the individual acquisition approaches and corresponding reconstruction methods in terms of dwell time, different beam currents were evaluated (Table 7.2). We incrementally increased the beam current from 0.1 nA to 0.2 nA, 0.4 nA, and 0.8 nA. The previously described reconstruction experiments were repeated for each setting. Results in terms of PSNR-HVS-M are shown in Table 7.2.

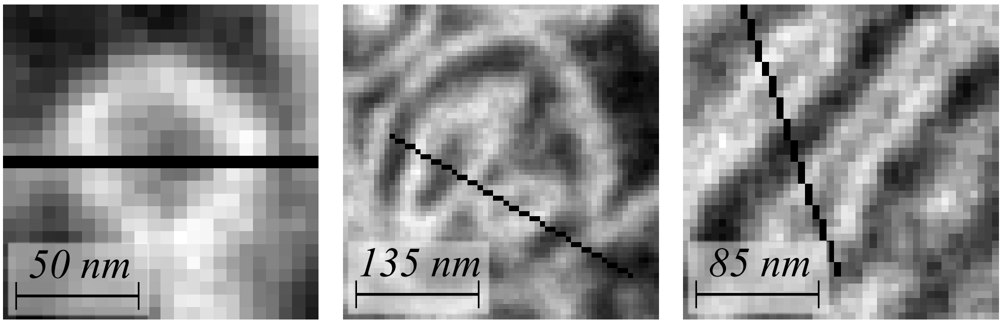
Tab. 7.2.: Influence of the beam current on the reconstruction quality, measured in terms of PSNR-HVS-M values. 30 structures, like the ones in Figure 7.2, were evaluated and results are given as mean $\pm$ standard deviation. For each strategy, the best beam current values and all within $\pm\sigma$ are indicated in bold letters

	0.1 nA	0.2 nA	0.4 nA	0.8 nA
Unprocessed Raster	$\mathbf{27.35 \pm 0.96}$	24.41 ± 1.02	22.10 ± 1.38	19.48 ± 1.46
GOAL denoising	$\mathbf{23.81 \pm 1.06}$	$\mathbf{22.82 \pm 1.86}$	$\mathbf{21.18 \pm 1.72}$	19.08 ± 1.54
SR	$\mathbf{21.37 \pm 0.91}$	$\mathbf{22.05 \pm 1.59}$	$\mathbf{21.16 \pm 1.56}$	$\mathbf{19.02 \pm 1.46}$
NN Interpolation	$\mathbf{23.91 \pm 1.36}$	$\mathbf{22.14 \pm 1.73}$	20.57 ± 1.75	18.67 ± 1.49
GOAL Inpainting	$\mathbf{23.85 \pm 1.19}$	$\mathbf{22.60 \pm 1.77}$	21.03 ± 1.85	18.92 ± 1.53
EBI	$\mathbf{23.01 \pm 1.24}$	$\mathbf{21.33 \pm 1.53}$	19.95 ± 1.54	18.30 ± 1.44
BPFA Inpainting	$\mathbf{29.22 \pm 1.42}$	24.36 ± 1.26	21.80 ± 1.54	19.26 ± 1.57

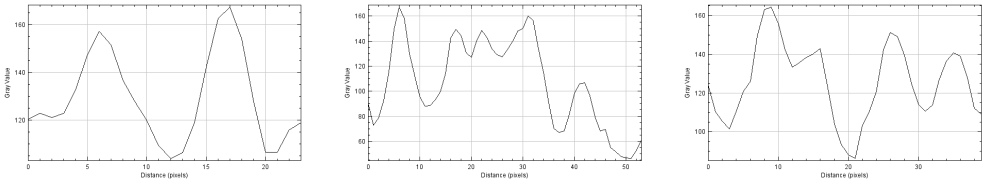
Optimal results were obtained for 0.1 nA for all reconstruction methods. Increasing the beam current either deteriorated results or gave the same quality within measurement precision.

Influence of Inpainting Dictionary. EBI is very sensitive to the dictionary used for the reconstruction. We therefore tested different dictionaries ranging from a very small dictionary as worst case scenario to a theoretically ideal dictionary built from ground truth data. This evaluation showed that an increasing dictionary quality lead to higher quality reconstructions. In the course of this study we achieved the following measure values as upper bounds on the impact of the dictionary on the reconstruction quality: PSNR $= 38.89 \pm 1.55$, PSNR-HVS-M $= 49.40 \pm 5.57$, SSIM $= 0.9823 \pm 0.0061$, and CW-SSIM $= 0.9976 \pm 0.022$. While an investigation into choosing an optimal dictionary is an interesting question, we will not go into further detail here, as the development of the reconstruction method itself is out of scope for this study.

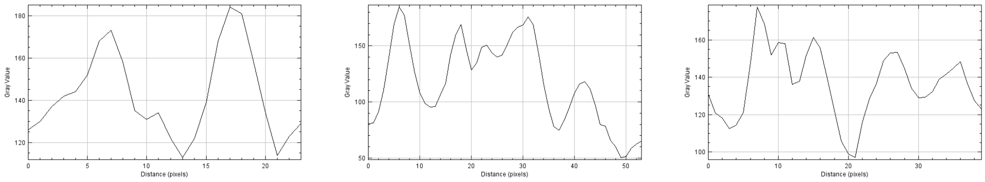
As opposed to EBI, GOAL is quite insensitive to the used dictionary. During tests of dictionaries learned on different datasets we found that for SEM reconstructions there were only small differences in terms of visual quality, which means that GOAL inpainting results did not depend heavily on the data used for dictionary learning. For BPFA the dictionary is always optimal, because this method learns the used dictionary directly



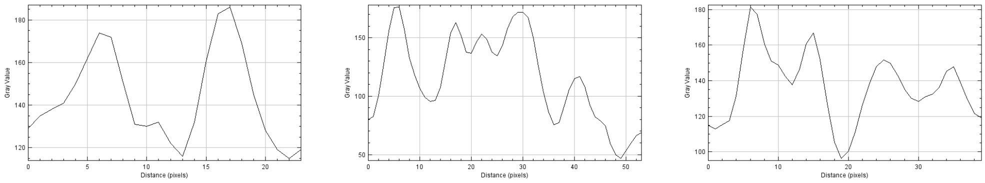
(a) Ground truth structures with selected line profile indicated.



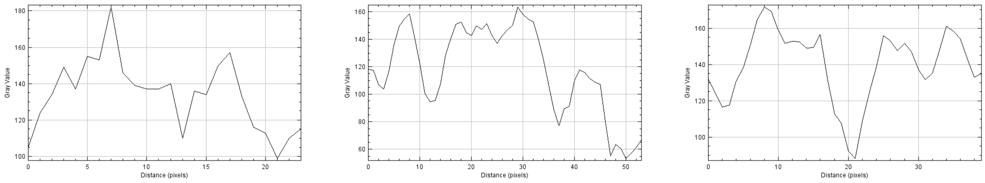
(b) Line profiles of ground truth structures.



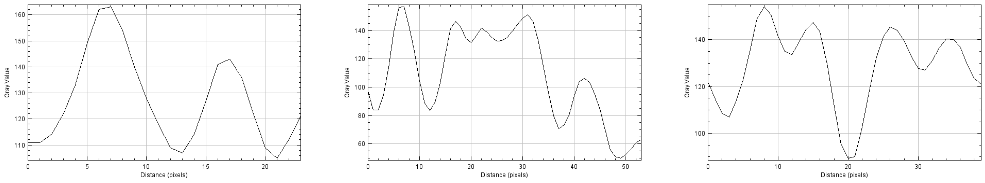
(c) Line profiles of structures from GOAL denoising.



(d) Line profiles of structures from SR.

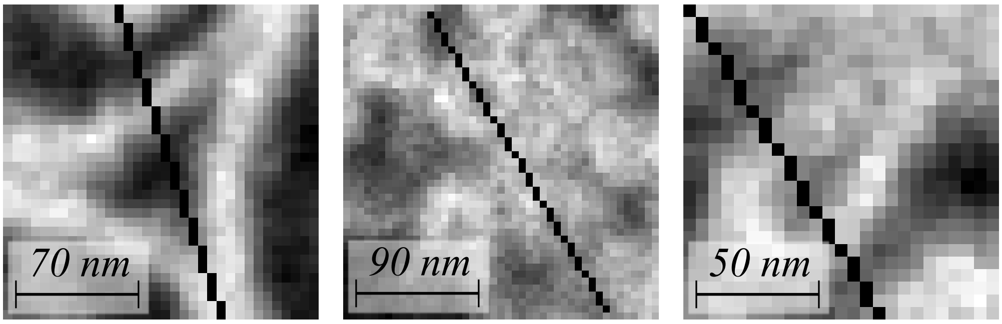


(e) Line profiles of structures from EBI.

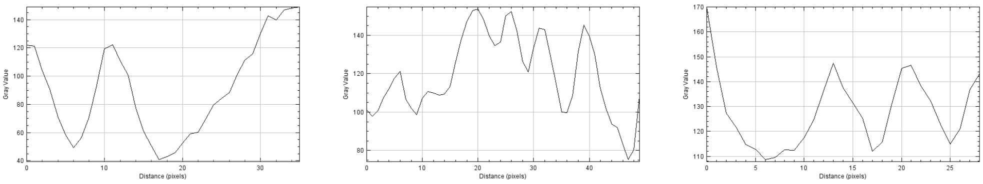


(f) Line profiles of structures from BPFA.

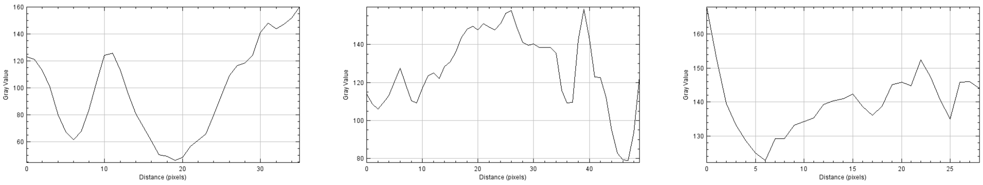
Fig. 7.3.: A line profile is depicted for each reconstruction method of the first three structures shown in Figure 7.2.



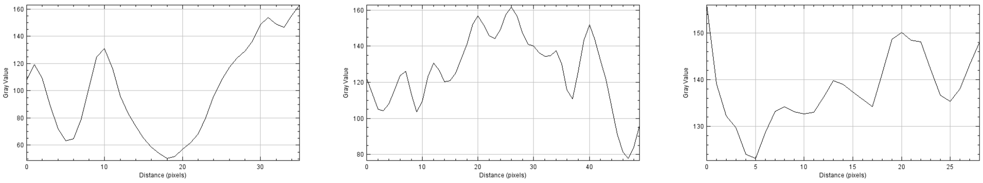
(a) Ground truth structures with selected line profile indicated.



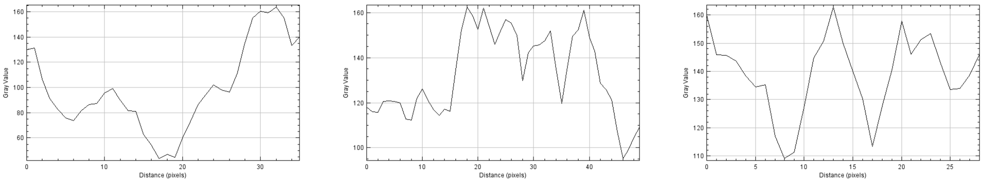
(b) Line profiles of ground truth structures.



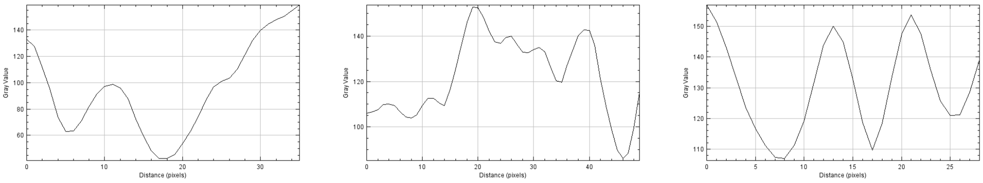
(c) Line profiles of structures from GOAL denoising.



(d) Line profiles of structures from SR.



(e) Line profiles of structures from EBI.



(f) Line profiles of structures from BPFA.

Fig. 7.4.: A line profile is depicted for each reconstruction method of the last three structures shown in Figure 7.2.

from the sparse data it is applied to for reconstruction, so there is only one result for BPFA inpainting.

7.5 Discussion

Looking at the quantitative measures and at many different image structures, it became evident that increasing the beam current for quality enhancement was counter-productive for all approaches investigated in this study. The reason for this is that at higher beam currents the virtual spot size increases, which leads to a defocused beam and thus to smoother and blurrier acquisitions. We found that in our setup, this blurring is not compensated for by the reduced shot noise. Furthermore, as a higher beam current also means a higher electron dose, increasing the beam current may damage a biological sample, which in turn may lead to distortions in the acquisition. Consequently, we excluded the higher beam currents from further evaluations and only took the 0.1 nA beam current results for further assessment. However, while the increase of the virtual spot size at higher beam currents is a fundamental principle, the strength of the effect depends on technical parameters such as the construction of the electron gun, the column, and the acceleration voltage. Therefore, the optimal dwell time that balances beam blurring against shot noise might differ among microscope setups.

Concerning the optimal acquisition strategy, we found that a raster scan at reduced pixel dwell time can be improved by means of GOAL denoising. However, the effect was less pronounced than we expected and did not show consistently across all quantitative measures for image quality.

Comparable results were obtained by using a raster scan at increased pixel size, followed by the application of a SR algorithm. In general, we found that quantitative ways to determine image quality can vary between measures. As visually different reconstructions can have almost the same values in many cases, we must be careful when interpreting small differences in these numbers. Keeping this in mind, we consider most results for random sparse scanning followed by inpainting to be inconclusive as well. The reconstructions using natural neighbour interpolation, EBI and GOAL inpainting show similar results as the GOAL denoising method in the sense that they are superior to the unprocessed raster scan in some error metrics but not in others, and seem visually superior or not depending on which structure is inspected.

A clearly different result is obtained for a random sparse scan followed by a reconstruction using BPFA inpainting, which gives the best results among the compared methods. This finding is supported by both the evaluation measures and the visual inspection of the structures.

One important consideration for dictionary based inpainting algorithms such as BPFA, GOAL and EBI is the impact the dictionary has on the reconstruction quality. BPFA performs on-the-fly learning of a dictionary from the sparse dataset, and thus is independent of the provided dictionary. GOAL and EBI, on the other hand, rely on additional prior knowledge that is provided in the form of a previously acquired dictionary. We found that GOAL is rather stable when using different dictionaries while EBI is highly sensitive to the used prior for dictionary learning. The optimal structure for EBI dictionaries for inpainting sparse scanning data is not well understood today, so there may be a large potential for improvement in this direction.

All investigated inpainting methods resulted in superior results compared to a fast raster scan with reduced dwell time, and best outcomes were obtained using BPFA. This leads to the conclusion, that the conventional approach of image acquisition should be challenged. Sparse acquisition techniques should have a much larger role in future microscopy, particularly for situations such as very large field-of-view scanning and 3D acquisitions of samples, where the total acquisition time and electron dose are limiting factors.

8

Sparse Scanning Electron Microscopy

> *Any sufficiently advanced technology is indistinguishable from magic.*
>
> — **Arthur C. Clarke**
> (British Physicist and Author)

High-throughput scanning electron microscopy (SEM) aims to reduce dose for sensitive specimens as well as reducing acquisition times to be able to acquire large volumes in a meaningful time. Sparse sampling is one key to make such acquisitions possible [29]. We propose a new reconstruction technique for such sparsely sampled SEM data, which is based on exemplar-based inpainting (EBI) known from image processing [246].

In serial block face (SBF) SEM a image stack of a specimen is acquired layer by layer. This is achieved by scanning the surface of the sample voxel by voxel to obtain one two-dimensional (2D) image. The whole volume is acquired as three-dimensional (3D) data by consecutively removing thin layers from the surface using a focused ion beam or a mechanical cutter until the whole specimen has been scanned. Multi-energy electron acceleration can also be used to acquire images at different depths without cutting [71]. The combination of cutting and multi-energy beams enables even thinner slices in z-direction. The bottleneck for recording large 3D volumes voxel by voxel is sensor bandwidth and, hence, acquisition time. For a volume of $2,048 \times 2,048 \times 1,024$ voxels resolution at a per-voxel dwell time of $20\,\mu s$ almost one day is needed for solely the data acquisition. Speeding up this acquisition could be achieved by reducing the dwell time, which results in a degradation of signal quality and in noisy images with bad signal-to-noise ratio (SNR).

Alternative to conventional scanning techniques that scan the image row by row on a Cartesian grid, sparse scanning techniques that only scan a small percentage of the whole specimen have gained attention recently. Scanning just a small subset of possible voxel positions can increase the imaging throughput remarkably. The sparsely scanned data must then be processed further to obtain a full resolution image. Applying appropriate reconstruction techniques, a dense volume with minimum loss of information is recovered. The scanning positions for recording the sparse data can be selected randomly, as it is usually done for inpainting algorithms inspired by compressed sensing (CS). If additional knowledge on the structure of the specimen is available, adaptive sampling schemes can provide superior results [61, 62].

EBI is a class of inpainting algorithms that originates from the field of image processing. The main goal of these methods is removing objects in images or restoring damaged portions of an image by inserting information from the surroundings. More specifically, the values of the missing region are modified such that the inserted part is visually indistinguishable from its surroundings. Adapting this idea, we developed a 3D inpainting method [68, 242–244, 246, 248] that inserts missing voxels with the help of prior knowledge as described in Chapter 5.

8.1 Sparse Coding Evaluation

In this section a comparison of different approaches to reconstruct sparsely sampled data is shown. Based on sample image data we simulated the sparse acquisition of image data with a microscope. These sparse images were then reconstructed using each of the algorithms shortly described in the following. Dictionary learning was conducted on data corresponding to the sparse image domain.

8.1.1 Selected Methods and Dataset

We compared different algorithms and approaches to reconstruct sparse data. Corresponding source codes of each algorithm were taken from the original implementations and adapted to our example.

Joint statistical model (JSM) [277]
Local smoothness and non-local self-similarity of natural images were combined to build a joint statistical model (JSM) in an adaptive hybrid space-transform domain. Using this JSM a new form of minimisation functional was proposed to solve an image inverse problem. An efficient software for solving this underdetermined inverse problem was implemented using a Split Bregman based algorithm. The software offered by the authors was used for the experiments.

Geometric analysis operator learning (GOAL) [114]
geometric analysis operator learning (GOAL) is based on an analysis model. In such models a suitable operator is needed to transform a complete signal into a sparse signal. GOAL uses training data to learn such an analysis operator based on L_p norm minimisation. Geometrical constraints are used to apply a conjugate gradient method on matrix manifolds. GOAL is tailored to reconstruct sparse image patches by applying

a learned analysis operator. The Matlab implementation offered by the authors was slightly adapted to conduct the experiments.

Basis pursuit (BP) [51] implemented in the software package YALL [279]

YALL stands for Your Algorithms for L_1. It is a Matlab solver that can be applied to different L_1 minimisation models. We use YALL to apply basis pursuit (BP). BP is a method for decomposing signals into a superposition of elements that form a basis for the signal space. This is done by solving a convex minimisation problem via linear programming by interior point methods, so that the coefficients of the final basis have the smallest L_1 norm among all possible decompositions. This basis can then be used to reconstruct sparse signals

Beta process factor analysis (BPFA) [282] [281]

beta process factor analysis (BPFA) establishes a non-parametric Bayesian formulation to compute a dictionary consisting of image patches as basis for reconstructing sparsely sampled image data. A beta process as well as Dirichlet process model are employed as priors to construct a hierarchical model for learning the dictionary. BPFA can also learn sparse dictionaries in situ by using only the sparse data without further training data. The learned dictionary is used to reconstruct sparse image data. The authors' provided Matlab implementation of BPFA was used to conduct the experiments.

Group-based sparse restoration (GSR) [276]

Instead of using single patches as basic dictionary elements, group-based sparse restoration (GSR) proposes groups of non-local patches with similar structure as unit elements. Each group is represented as a matrix where each column contains a single vectorised patch. A Split Bregman iteration framework is used to solve the minimisation problem for learning the group dictionary. Matlab code for group dictionary learning and corresponding reconstruction of sparsely sampled data was offered by the authors and used for the experiments.

Convolutional sparse coding (CSC) [116]

convolutional sparse coding (CSC) does not work on image patches but on whole images at once. This makes it possible to capture correlations between local neighbourhoods. By splitting the objective function into a sum of simple, convex functions it is possible to solve the sparse coding problem efficiently. The authors provide a Matlab implementation for dictionary learning and reconstruction of sparse data, which was used for the experiments.

Dataset for Evaluation.

The sparse data was generated from a mouse brain dataset. This dataset consisted of 100 subsequent image slices with a resolution of $1,024 \times 1,024$ pixels per image. An example of such a mouse brain image is depicted in Figure 8.1. The first 50

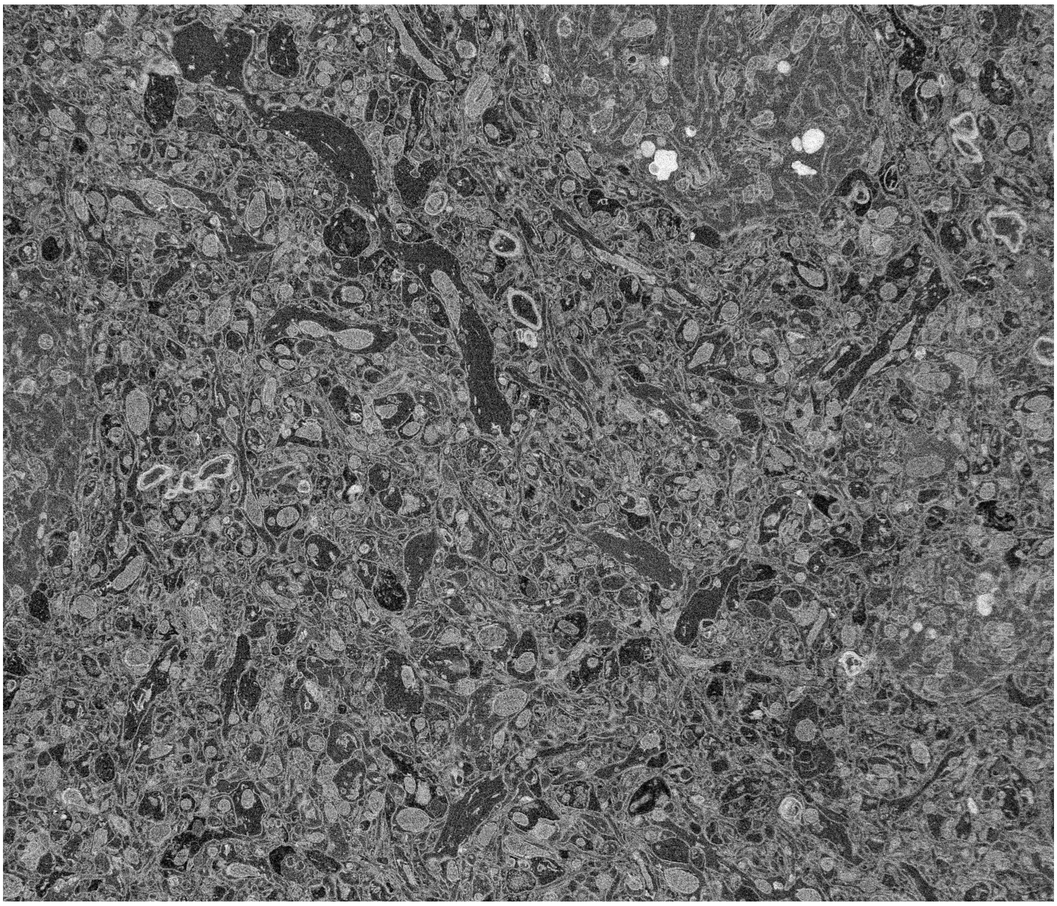


Fig. 8.1.: Example acquisition of a mouse brain with a scanning electron microscope.

slices were used for dictionary learning if required by an algorithm. The last 50 slices were used to simulate sparsely acquired images with a microscope by multiplying the original images with a binary mask following a Bernoulli distribution. The retained percentage of acquired image pixels was varied from $2\,\%$ to $50\,\%$ with percentage $\in \{2\,\%, 5\,\%, 10\,\%, 15\,\%, 20\,\%, 25\,\%, 30\,\%, 40\,\%, 50\,\%\}$ to investigate the influence of sparsity on these kinds of algorithms.

8.1.2 Influence of Sparsity on Reconstruction

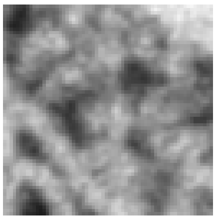
Presented are reconstruction results of single images, as all algorithms operate on 2D data. Evaluation metrics in Tables 8.1 and 8.2, as well as visual comparisons in Figures 8.2 - 8.7, show that all the algorithms tested behave differently both with respect to quality and runtime. With increasing amount of data used for the reconstruction, the quality of the reconstructions increased, which was to be expected. Nevertheless, there were clear differences between the algorithms. For example, at $50\,\%$ sparsity, peak signal-to-noise ratio (PSNR) ranged from 27.85 to 39.31, and structural similarity index (SSIM) ranged from 0.8539 to 0.9820. The drawback for a qualitatively better reconstruction is computational runtime. YALL took only 7 seconds whereas BPFA took 137 minutes for this sample. More details on the runtime are presented in the next section. Quality had highest priority, hence, we selected GOAL to more closely investigate possible gains from sparse coding, as GOAL was fastest among the best performing algorithms.

Tab. 8.1.: PSNR for evaluated algorithms and sparsity.

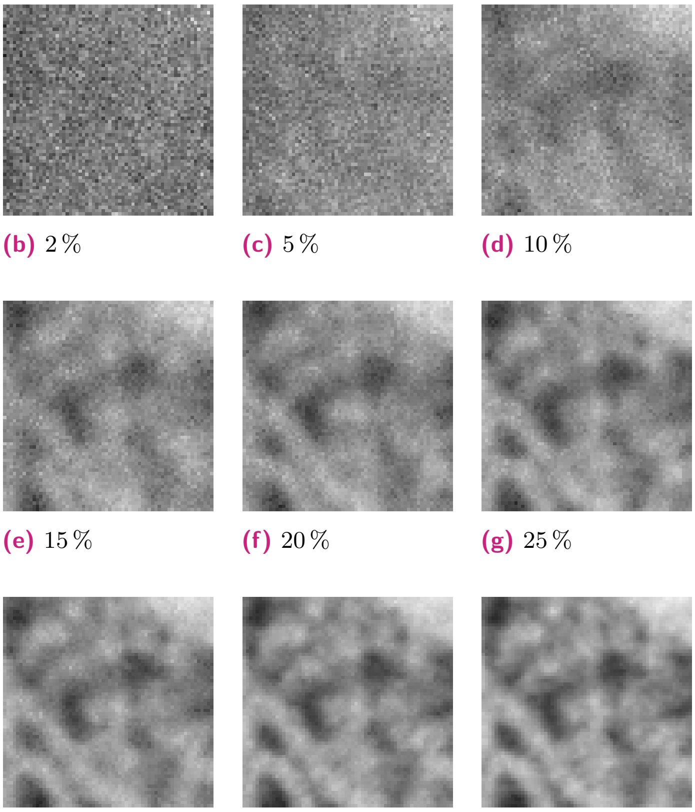
percentage	JSM	GOAL	BP (YALL)	BPFA	GSR	CSC
2 %	20.22	20.97	14.96	12.93	20.77	18.92
5 %	23.23	24.71	16.28	21.59	24.54	21.90
10 %	26.34	27.81	19.02	27.03	27.58	20.68
15 %	28.53	29.87	21.74	29.04	29.46	21.91
20 %	30.12	31.51	23.20	30.20	31.05	23.00
25 %	31.43	32.92	25.21	30.78	32.50	23.98
30 %	32.64	34.25	25.99	31.41	33.90	24.89
40 %	34.85	36.43	27.88	31.82	36.60	26.53
50 %	37.02	38.40	29.30	32.79	39.31	27.85

Tab. 8.2.: SSIM for evaluated algorithms and sparsity.

percentage	JSM	GOAL	BP (YALL)	BPFA	GSR	CSC
2 %	0.4484	0.5397	0.2063	0.3089	0.5068	0.1633
5 %	0.5810	0.7002	0.2224	0.6005	0.6735	0.2377
10 %	0.7230	0.8102	0.3543	0.7636	0.7928	0.3923
15 %	0.8079	0.8680	0.5181	0.8245	0.8526	0.5196
20 %	0.8580	0.9047	0.6200	0.8531	0.8915	0.6102
25 %	0.8920	0.9306	0.7107	0.8644	0.9200	0.6794
30 %	0.9163	0.9483	0.7686	0.8790	0.9404	0.7325
40 %	0.9487	0.9702	0.8593	0.8844	0.9670	0.8079
50 %	0.9687	0.9820	0.8968	0.9062	0.9819	0.8539



(a) Ground Truth



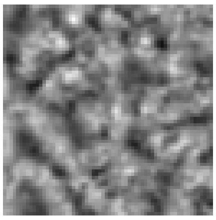
(b) 2 % **(c)** 5 % **(d)** 10 %

(e) 15 % **(f)** 20 % **(g)** 25 %

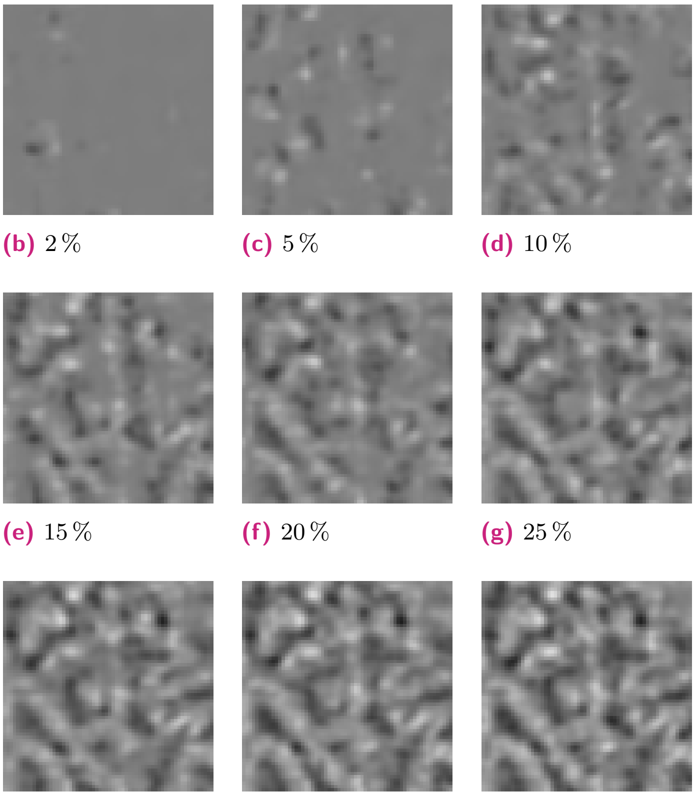
(h) 30 % **(i)** 40 % **(j)** 50 %

Fig. 8.2.: Reconstructed images from different sparsity percentages using YALL.

Figure 8.2 shows results reconstructed with YALL. Sparsity levels up to $10\,\%$ look like pure noise, which was supported by small PSNR values below 19 and SSIM values below 0.35. Increasing the percentage of pixels as input for the reconstruction further, the pure noise pattern vanished. At $25\,\%$ sparsity most of the structures could be identified. PSNR values ranged between 21.74 for $15\,\%$ and 29.3 for $50\,\%$ with corresponding SSIM values between 0.52 and 0.9.



(a) Transformed Ground Truth



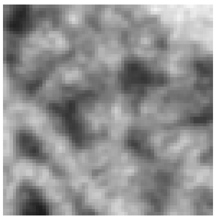
(b) $2\,\%$ **(c)** $5\,\%$ **(d)** $10\,\%$

(e) $15\,\%$ **(f)** $20\,\%$ **(g)** $25\,\%$

(h) $30\,\%$ **(i)** $40\,\%$ **(j)** $50\,\%$

Fig. 8.3.: Reconstructed images from different sparsity percentages using CSC.

Figure 8.3 shows results reconstructed with CSC. The available implementation of CSC transforms input images through a process akin to normalisation. It was not possible to invert the normalisation procedure. To compute the evaluation measures we performed the same transformation with the ground truth image. The transformed ground truth image was then taken as reference image in the calculations. Hence, the results may not be comparable to the other methods. Reconstructions of pixel levels below $25\,\%$ didn't capture the transformed ground truth. Even with percentages up to $50\,\%$ the reconstructions could not convince visually. PSNR values ranged between 23 for $20\,\%$ and 27.85 for $50\,\%$ with corresponding SSIM values between 0.61 and 0.85. It is not clear if the evaluation was negatively influenced by the preprocessing transform of CSC, nevertheless, we show the results for completeness.

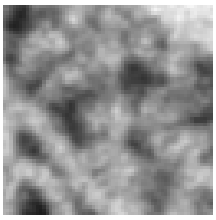


(a) Ground Truth

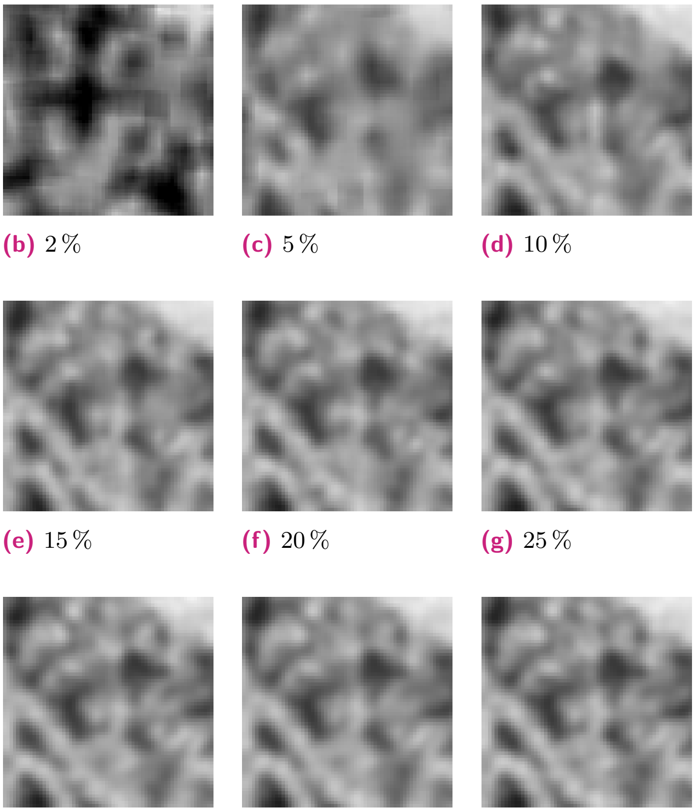
(b) 2 %
(c) 5 %
(d) 10 %
(e) 15 %
(f) 20 %
(g) 25 %
(h) 30 %
(i) 40 %
(j) 50 %

Fig. 8.4.: Reconstructed images from different sparsity percentages using JSM.

Figure 8.4 shows reconstructions obtained with JSM. Taking less than $10\,\%$ as input led to cartoon like reconstructions of which only the $10\,\%$ case may be used, as it looked like a segmentation of the original. From $15\,\%$ to $25\,\%$ the reconstructions already looked visually convincing with PSNR values from 28.53 to 31.43 and corresponding SSIM values between 0.81 and 0.97. However, small artefacts were visible in the reconstructions. Above $30\,\%$ these artefacts also vanished and PSNR values up to 37.02 with corresponding SSIM of 0.97 were reached.



(a) Ground Truth



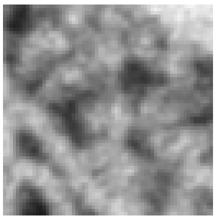
(b) $2\,\%$ **(c)** $5\,\%$ **(d)** $10\,\%$

(e) $15\,\%$ **(f)** $20\,\%$ **(g)** $25\,\%$

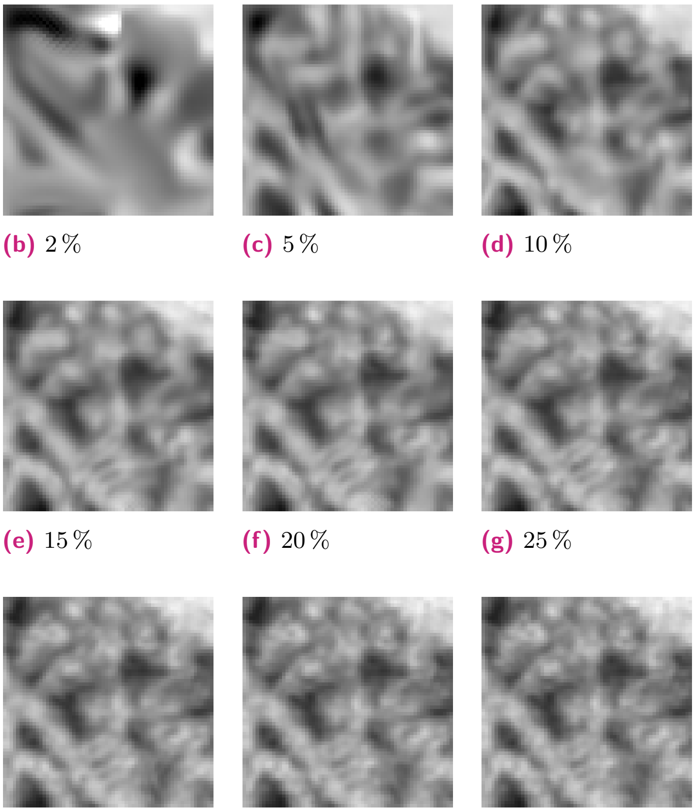
(h) $30\,\%$ **(i)** $40\,\%$ **(j)** $50\,\%$

Fig. 8.5.: Reconstructed images from different sparsity percentages using BPFA.

Figure 8.5 shows results reconstructed with BPFA. Reconstructions with less than $10\,\%$ missed some structure of the original. But already at $10\,\%$ the result resembled a denoised version of the original. This was also visible in the reconstructions with pixel levels up to $50\,\%$. Looking at evaluation measures, the PSNR at $10\,\%$ was 27.03 and the corresponding SSIM was 0.76. From $15\,\%$ to $50\,\%$ the PSNR values ranged from 29.04 to 32.79 and SSIM values were between 0.82 and 0.91. Due to the denoising like appearance, which led to smoother and blurrier results, the visual impression of the reconstructions is much better than the evaluation measures tell.



(a) Ground Truth



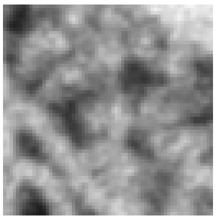
(b) $2\,\%$ (c) $5\,\%$ (d) $10\,\%$

(e) $15\,\%$ (f) $20\,\%$ (g) $25\,\%$

(h) $30\,\%$ (i) $40\,\%$ (j) $50\,\%$

Fig. 8.6.: Reconstructed images from different sparsity percentages using GSR.

Figure 8.6 shows reconstruction results from GSR. Already at a pixel level of $2\,\%$, some of the features of the original image were contained, even if not anywhere near the ground truth. Depending on the application, the $10\,\%$ reconstruction may already suffice, which had a PSNR of 27.58 and a SSIM of 0.79. Adding more pixels above $20\,\%$ had no significant visual influence, the reconstruction primarily gained more of the original noise like appearance. Looking at the evaluation measures starting at $20\,\%$, PSNR values ranged from 31.05 to 39.31 and SSIM values from 0.89 to 0.98.



(a) Ground Truth

(b) 2 %
(c) 5 %
(d) 10 %
(e) 15 %
(f) 20 %
(g) 25 %
(h) 30 %
(i) 40 %
(j) 50 %

Fig. 8.7.: Reconstructed images from different sparsity percentages using GOAL.

Figure 8.7 shows results for GOAL. Pixel levels below $10\,\%$ were very blurry and didn't capture the structure well, which was also reflected by PSNR values below 25 and SSIM values below 0.7. At $10\,\%$ the reconstructed image was still blurry, but the structures were clearer, comparable to a denoising filter applied to the original. PSNR around 28 and SSIM of 0.81 confirmed this enhancement. All pixel levels above $15\,\%$ were visually similar, so that the evaluation measures were needed to quantify the difference. PSNR values ranged from 29.87 at $15\,\%$ to $38.4\,\%$ at $50\,\%$, with corresponding SSIM values ranging from 0.87 to 0.98.

8.1.3 Runtime

For reconstruction algorithms in electron microscopy (EM) not only the quality plays an important role but also the runtime. When integrating sparse coding, respectively CS, into a microscope workflow, fast algorithms are needed. It is not meaningful to save seconds in one acquisition if the corresponding reconstruction takes minutes or even hours. Thus, we also compared the runtime for the different combinations of algorithm and sparsity depicted in Table 8.3. The runtime was measured based on the offered implementations from Section 8.1.1 on an Intel Core i7-6700K at 4GHz with 16GB memory.

Tab. 8.3.: Runtimes for evaluated algorithms and sparsity given in seconds.

percentage	JSM	GOAL	BP (YALL)	BPFA	GSR	CSC
$2\,\%$	2560	1780	4.66	960	5824	1080
$5\,\%$	2548	1700	8.31	1320	5077	1086
$10\,\%$	2584	1700	8.18	2100	4297	1161
$15\,\%$	2575	1800	7.71	2760	5209	1078
$20\,\%$	2597	1740	7.81	3360	5391	1081
$25\,\%$	2558	1740	7.66	3900	5390	1077
$30\,\%$	2678	1760	7.99	4740	5421	1073
$40\,\%$	2653	1760	7.58	6060	4561	1119
$50\,\%$	2643	1800	6.59	8220	4459	1258

8.1.4 Geometric Analysis Operator Learning on GPU

As GOAL performed best with respect to a runtime-quality trade-off, especially in the interesting sparsity range up to $20\,\%$, we conducted an evaluation study based on GOAL for a SEM data acquisition. Due to the slow runtime of GOAL we first re-implemented GOAL as a high-performance computing (HPC) version on graphics processing unit (GPU).

The first step before mapping GOAL to GPU was application profiling in Matlab, from which we identified the parts where most compute time was spent. GOAL was then re-implemented from scratch in C++ and CUDA [55] by porting most of the functionality to GPU. The CUDA kernels that consumed a substantial part of the total execution time were 2D convolutions and gradient computations. For the acceleration of these kernels we used the carefully tuned library implementations CuDNN [56] and CuBLAS [54] from Nvidia. We compared their performance against our highly-optimised manually written kernels, which showed that using the libraries helped to additionally save more than $80\,\%$ of runtime.

Mapping as much functionality as possible to GPU helped to reduce memory traffic between GPU and central processing unit (CPU). Although some of the functionality runs slower on GPU due to missing data parallelism, the effect on application throughput was positive, since it removed many of the time consuming memory transfers for intermediate results.

The HPC version of GOAL was then used to evaluate some parameters, such as data used for dictionary learning, number of atoms in the dictionary, iteration numbers, or learning rate during gradient descent. As results were highly dependent on the dataset, we do not cover the evaluation in detail. The main findings are the following. The data used for dictionary learning influenced the reconstruction result significantly, so that PSNR varied between 29.58 and 31.71. Increasing the number of dictionary elements, led to less iterations until the algorithm converged, however each iteration took exponentially more time. This means that it is preferable to choose the number of dictionary elements smaller, e.g. 192, while conducting more iterations, which results in much faster runtime and comparable quality. The primary reason may be that using more dictionary elements mainly adds redundancy. A general statement on how to select the number of iterations and the learning rate cannot be made, as this was highly dependent on the sparsity of the data as well as on the trade-off between quality and runtime. With help of the HPC version of GOAL we could reduce the runtime of a 512×512 image reconstruction with $20\,\%$ sparsity from 64.3 seconds to 2.0 seconds on an Nvidia Quadro P5000 with 16GB RAM, ignoring the initialisation times as these are negligible in a SEM setting.

One important fact we discovered concerns the best choice for the initialisation of GOAL. The default in the reference implementation was bilinear interpolation, however, we found that natural neighbour interpolation [183] is much more suitable, especially when dealing with noisy data. In our experiments the default initialisation alone had a PSNR of 16.32, whereas natural neighbour initialisation lead to a PSNR of 17.76 for a noisy input with low SNR.

8.1.5 Conclusions from Sparse Coding Evaluation

Comparing the results of this sparse coding evaluation and the results from Chapter 7, we obtained contradictory statements regarding the best sparse coding algorithm to use. In Chapter 7 BPFA performed best, whereas in the sparse coding comparison GOAL performed best. This shows that there is not the one best sparse coding algorithm. Further, the quality of a reconstruction highly depends on the input data and on properties of the algorithm.

The fast sparse coding algorithms we tested resulted in low quality reconstructions, both visually and quantitatively. The algorithms with best quality, however, were quite slow. As quality was given a higher priority than runtime, we selected GOAL for further evaluations. Additionally, we reimplemented GOAL on GPU resulting in a speed-up of up to $32\times$ compared to the Matlab reference implementation, which made experimenting with this algorithm much more comfortable.

Unfortunately, EM data in general is 3D. Applying one of the presented sparse coding algorithms, information in the third dimension is lost, which could otherwise help to enhance reconstruction quality. As sparse coding algorithms in general work on 2D data, and we also did not find any suitable 3D algorithm, it would be wasteful to reconstruct each 2D image individually while ignoring the 3D information. Hence, we decided to implement a 3D version of EBI to overcome this issue. The decision was supported by the result of Section 7.4 that EBI was able to outperform all other algorithms tested using an optimised dictionary. Domain-adapted dictionaries, which are tailored towards the data to be reconstructed, must be used for high quality reconstructions with EBI.

8.2 Application Oriented Evaluation for Quality Assessment

As already stated in Section 3.6 and observed in Sections 7.4 and 8.1, evaluation measures may be very misleading, especially if decoupled from an intended application. We propose a new direction to evaluate the capabilities of an algorithm tailored to the segmentation of mitochondria and membranes in mice brains [195]. As basis we take the sparse SEM workflow and evaluate its quality by comparing segmentations of ground truth full scans and corresponding reconstructed sparse scans.

8.2.1 Image Quality Evaluation

A non-trivial step in sparse imaging is the evaluation of image quality. In general, a full scan at low pixel dwell-times generates noisy images due to shot noise. In comparison, a sparse scan with the same average dwell time per pixel generates less noisy images. However, these images contain artefacts caused by the reconstruction algorithm. Comparing one to the other by manual observation does not provide consistent results as a preference for either type of image distortion depends to a large degree on the subjective tendency of the observer.

Seemingly more reliable are quantitative image similarity metrics as explained in Section 3.6. Despise advances in the definition of expressive metrics, image quality measures tend to correspond poorly to human impression of image quality [4]. Higher evaluation scores do not necessarily imply a more accurate result of a measurement in a subsequent image analysis task, for example a more accurate segmentation. We therefore propose that the result of the image analysis task, e.g. segmentation as proof of concept example, is a much more expressive measure for image quality. In the example, we use deep learning with convolutional neural networks to perform automatic segmentation of mitochondria and membranes on SEM images. The neural networks are trained specifically to deal with the distortions in the different images; noise in the case of full scans and reconstruction artefacts in the case of sparse imaging. Our image quality metric is the accuracy of the segmentation.

8.2.2 Sample Preparation and Data Acquisition

Our evaluation dataset was collected from a mouse brain sample that has been prepared and fixed according to an adapted protocol for SBF imaging as originally described in [72]. A Helios Nanolab 660 SEM was used to acquire the images. The following imaging conditions were used: HiVac mode, accelerating voltage $2\,\text{kV}$, beam current $0.2\,\text{nA}$, and working distance $3\,\text{mm}$. The through lens detector was used in back scattered electron mode to acquire the data. The image pixel resolution was $10\,\text{nm}$ for all images.

We acquired full frame images at $30\,\mu$s, $10\,\mu$s, $3\,\mu$s, $1\,\mu$s, and $300\,$ns. Sparse acquisitions were simulated by multiplying the full frame images with a random mask following a Bernoulli distribution, i.e. by discarding pixels. Using this procedure, we generated sparse datasets with $50\,\%$, $30\,\%$, $20\,\%$, and $10\,\%$ pixel coverage. The procedure was demonstrated in [29] to be a suitable simulation method for actual sparse image acquisitions with a well understood, limited effect on the image quality.

Every combination of dwell time and pixel coverage was reconstructed using our implementation of the GPU version of GOAL, as described in Section 8.1.4. The full frame images scanned at $30\,\mu$s dwell were used as ground truth.

8.2.3 Segmentation Procedure

We used a residual version of the convolutional neural network Unet [198], which is tailored for segmentation tasks. The training dataset was created from full frame images acquired at $1\,\mu$s and $3\,\mu$s pixel dwell time. By computationally reducing the pixel coverage to $50\,\%$, $30\,\%$, $20\,\%$, and $10\,\%$ followed by a GOAL reconstruction, the training dataset was increased by a factor of five. All images were manually labelled to segment cell membranes and mitochondria on a per pixel basis. We used standard data augmentations techniques including rotation, mirroring, scaling, affine and elastic deformation, Gaussian blur and Poisson noise addition. The majority of the images were from our own mouse brain specimen. A few additional images from the ISBI 2012 EM segmentation challenge [10] were added to expose the system to a richer variety of samples.

The proposed evaluation framework does not depend on the applied segmentation method, which is the reason we do not go into detail regarding the neural network learning. Nevertheless, we also state the setup and the parameters that were used for training the neural network for the sake of completeness. The segmentation network was implemented with TensorFlow [239]. ADAM optimisation was used with a fixed learning rate of 10^{-3} for $25,000$ iterations. ADAMs epsilon value was changed to $\epsilon = 10^{-4}$ and all other parameters were set to their TensorFlow defaults. This procedure was performed once for cell membranes, and once for mitochondria, resulting in one network for each class to be segmented. Sparse acquisitions were simulated as described above. Segmentation was performed using the networks trained to compensate GOAL specific artefacts.

8.2.4 Results

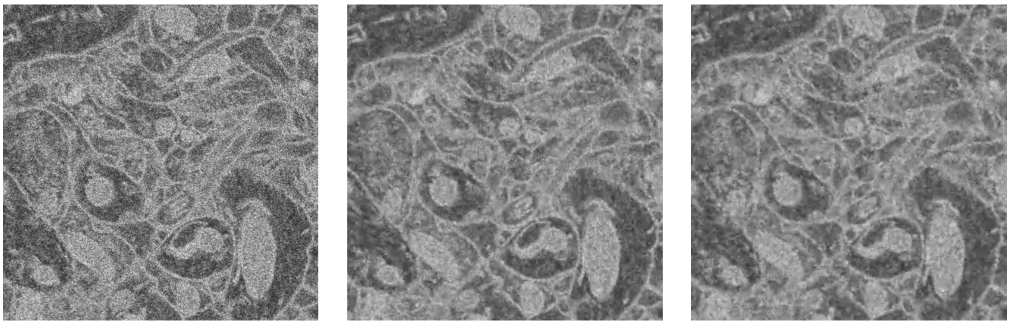
In the following, we distinguish the dwell time per pixel and the effective dwell time of a scan. The dwell time per pixel is the time the electron beam irradiates an individual pixel while the effective dwell time is the time required to illuminate the full frame, divided by the number of pixels. For a full scan, the two numbers are identical. For a sparse scan, the effective dwell time is the product of dwell time per pixel and pixel coverage. Acquiring images with a random sparse scan of $30\,\%$ and a dwell time per pixel of $3\,\mu$s results in approximately the same effective dwell time as a full scan acquired at $1\,\mu$s. However, the resulting images show a completely different structure. While the full scan images are quite noisy, the reconstructed sparse images have substantially less noise but suffer from the typical GOAL artefacts. At $20\,\%$ pixel coverage and $3\,\mu$s dwell time, the artefacts are even more pronounced, but we could reduce the effective dwell time by one third. The upper row of Figure 8.8 reflects these results.

Besides the new evaluation approach we also measured PSNR and SSIM. PSNR improved from 19.4 for a full scan at $1\,\mu$s to 25.0 for a sparse scan with a pixel coverage of $30\,\%$ and a dwell time per pixel of $3\,\mu$s. Similar results were obtained for SSIM, where the value improved from 0.36 for a full scan at $1\,\mu$s to 0.68 for a sparse scan with a pixel coverage of $30\,\%$ and a dwell time per pixel of $3\,\mu$s.

Tab. 8.4.: Impact of pixel coverage and dwell time per pixel on the segmentation quality. The quality is measured as percentage of correctly classified pixels for membranes and mitochondria (accuracy membranes / accuracy mitochondria). The reference full scan at $1\,\mu$s is highlighted in blue, all settings that use equal or less total dose compared to the reference are highlighted in grey. Optimal settings are depicted bold.

per pixel dwell time	$10\,\%$	$20\,\%$	$30\,\%$	$50\,\%$	$100\,\%$
300ns	0.76 / 0.94	0.79 / 0.95	0.82 / 0.96	0.84 / 0.98	0.88 / 0.97
$1\,\mu$s	0.82 / 0.99	0.86 / 0.98	0.87 / 0.97	0.91 / 0.98	0.93 / 0.96
$3\,\mu$s	0.87 / 0.99	**0.92 / 0.99**	**0.93 / 0.99**	0.95 / 0.98	0.96 / 0.97
$10\,\mu$s	0.90 / 0.99	0.95 / 0.99	0.96 / 0.99	0.97 / 0.99	0.98 / 0.98
$30\,\mu$s	0.92 / 1.00	0.96 / 1.00	0.97 / 1.00	0.99 / 0.99	1.00 / 0.99

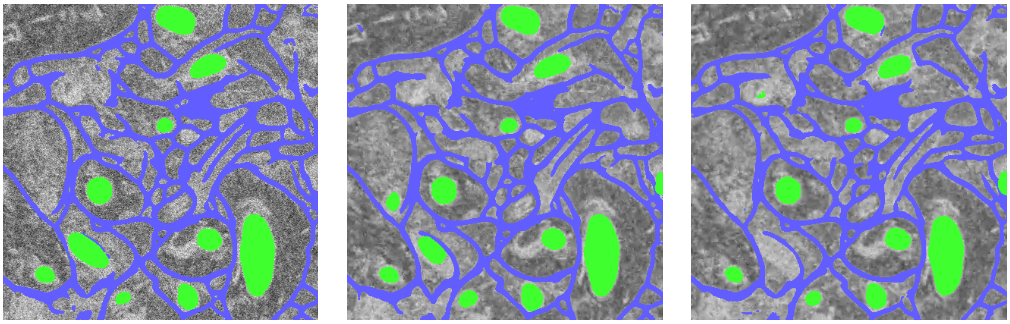
More important than the visual impression or quantitative quality metric, though, is the question of how well this scientific data can be used to answer a specific question. In the field of connectomics, it is often the question which pixels belong to certain parts



(a) Full scan with pixel dwell time of $1\,\mu$s.

(b) Inpainted sparse scan of $30\,\%$ of all pixels with dwell time of $3\,\mu$s.

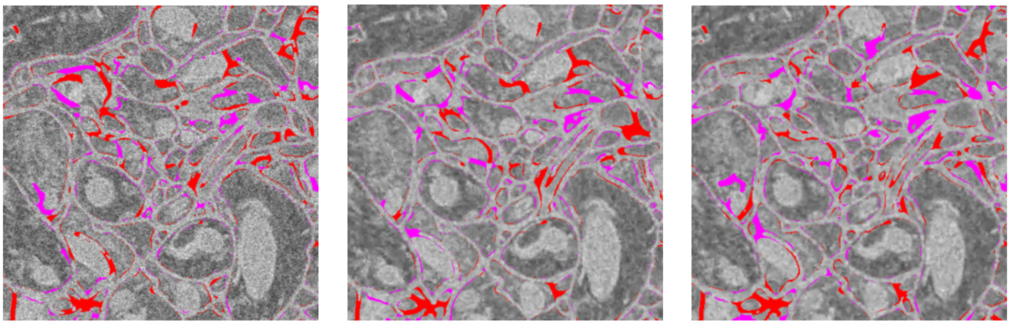
(c) Inpainted sparse scan of $20\,\%$ of all pixels with dwell time of $3\,\mu$s.



(d) Segmentation results with mitochondria in green and membranes in blue from (a).

(e) ... from (b)

(f) ... from (c)



(g) Errors in the segmentation. Red pixels are false positives, pink pixels are false negatives from (d).

(h) ... from (e).

(i) ... from (f).

Fig. 8.8.: Results of the segmentation procedure on reconstructions from sparsely sampled images.

of a cell, for example where are the mitochondria and membranes. If networks trained from perfect data (ground truth) are used to segment new data then neither of the reconstructions generates satisfactory results, because noise and artefacts cannot be compensated sufficiently. However, if the networks are trained from data that exhibits the same characteristics as the imperfect data, reasonable segmentation results can be acquired from either of the approaches (Figure 8.8 middle row). For the lowest total dwell time we tested neither of the approaches generated perfect segmentation results (Figure 8.8 lower row).

Obviously, the pixel dwell time is the primary factor for shot noise, and as such has high impact on the segmentation quality. We found that a pixel dwell time of $300\,$ns at the imaging conditions described earlier is insufficient independent of additional pixel coverage (Table 8.4). In order to achieve acceptable results, a minimum dwell time of $1\,\mu$s is required even for a full scan. Thus, we used the setting of a full scan at $1\,\mu$s dwell time per pixel as a reference and searched for imaging conditions that improve upon this reference. We found that random sparse scans of $30\,\%$ of the pixels, acquired at $3\,\mu$s per pixel, in general result in superior segmentation quality compared to the reference settings. For membranes, a segmentation accuracy of 0.93 could be reached which was identical to the reference settings result. For the mitochondria, the accuracy was 0.99 compared to 0.96 for the reference settings.

For the settings with $20\,\%$ pixel coverage at $3\,\mu$s per pixel, the accuracy was 0.92 for membranes (reference: 0.93), which is identical within measurement precision but the effective dwell time was reduced by $33\,\%$. For mitochondria, we reached an accuracy of 0.99 (reference: 0.96). Our results show that sparse scanning is favourable to a full scan at the same effective dwell time for a wide range, but not for all settings.

8.2.5 Discussion

The emergence of sparse scanning techniques in SEM poses the question of how images generated using sparse scanning can be compared to conventional raster scan images. Sparse scans suffer unavoidably from reconstruction artefacts, whereas the level of shot noise for full scans is much higher.

Our results show that in terms of quantitative error metrics such as PSNR or SSIM, sparse scans followed by dictionary-based inpainting result in a superior image quality compared to full scans performed at the same effective dwell time. However, the question remains how these results translate to actual advantages, for example in terms of improved segmentation performance.

We therefore propose that image quality should be measured by the performance of an analysis algorithm on the data. To demonstrate the principle, we trained two artificial neural networks to automatically segment mitochondria and membranes in the images. We found that performing automated segmentation based only on networks that were trained from ground truth data resulted in very poor segmentations and should be avoided. In order to achieve useful analysis results, the artificial neural networks need to be trained on imperfect data to learn to compensate for deteriorations such as noise or reconstruction artefacts.

Fast full scans as well as reconstructed sparse scans can lead to good segmentation results, with slight advantages for the reconstructed sparse scans. However, the advantages of sparse scans are not as strongly expressed as we would expect from the advantages in image quality indicated by quantitative image metrics such as PSNR and SSIM. We suspect that this stems from a not-yet optimal handling of the deteriorated data in the network models and speculate that further improvements can be achieved by more careful considerations of reconstruction artefacts in the machine learning models. Nevertheless, reconstructed sparse scans already lead to improved segmentation results compared to full scans at the same effective dose. This means that in order to achieve optimal segmentation results, sparse scans should be favoured over full scans, if simply increasing the effective dwell time is not an option.

8.3 Exemplar-based Inpainting in Sparse Scanning Workflow

In this section we describe the sparse SEM workflow as presented in [246]. The workflow consists of two parts that run in parallel. While the microscope is acquiring sparse data that must be processed afterwards, EBI is running on a server to reconstruct the sparsely sampled data (Figure 8.9).

```
Input  : dictionary data $D$, patch size $\Psi$, hyper-parameters for acceleration
         structure, sample for microscope
Output : reconstructed image
load $D$;
learn acceleration structure;
@microscope
while sample not fully acquired do
    acquire sparse scan (surface of sample);
    cut surface of sample;
end
@server
while sparse scan available do
    apply inpainting;
end
```

Fig. 8.9.: Workflow of accelerated sparse SEM.

8.3.1 Dictionary Assembly

As explained in Section 5.1, EBI relies on information of known image patches that are present in the image to be inpainted. For sparsely sampled data this prerequisite is not fulfilled. Hence, prior to applying EBI, we must gather data from the image domain of the inpainting target, e.g. reconstructing mouse brains requires full scans of mouse brains as dictionary. The consequence is that EBI is only applicable on sparsely sampled EM images with domain-adapted dictionaries acquired independently on fully scanned data. Such a dictionary represents prior knowledge that is used to reconstruct sparse acquisitions.

An ideal dictionary should contain each possible structure, respectively image patch, that could be contained in the image domain to be reconstructed. Consequently, the more complex an image domain is, the larger the dictionary should be for reasonable reconstructions. In the case of filled silica balls, compare Section 8.4.1, the underlying image content is not very complex and mainly consists of circular balls and background. With these properties, a smaller dictionary suffices to cover the possible patch space for such data adequately, which makes the filled silica balls dataset an ideal candidate for a proof of concept.

8.3.2 Recording Schemes

Acquiring sparse data in SEM requires a recording scheme to instruct the microscope which pixels should be scanned and which not. There are many different possibilities how to realise this. We evaluated three possible options [65].

Hereby, each recording scheme uses a fixed budget of dwell time, which is adjustable at the beginning of the acquisition. That means the maximal scanning time is predetermined, however, the budget can be distributed arbitrarily and in as many scan iterations as needed, as long as sufficient dwell time is remaining.

Random Sampling

The first approach is random sampling. Experiments as well as CS sampling theory have shown that random scan patterns following a Bernoulli distribution are good choices for the acquisition of sparse data [160]. Random sampling masks must be precomputed according to the available dwell time budget. In general a fixed percentage of pixels is scanned with a fixed dwell time per pixel, e.g. with a budget of one second and a dwell time per pixel of $10\,\mu s$ we can scan $100,000$ pixels.

Bernoulli sampling masks are generated by determining the percentage p of pixels that should be scanned based on a fixed budget. Sampling is then applied by drawing a random variable X uniformly for each pixel scan position. The $p\,\%$ pixels with the highest values for X are assigned a value of 1 (should be scanned) in the sampling mask, all others a value of 0 (should be skipped while scanning). Examples with different budget are shown in Figure 8.12 upper row.

This recording scheme is independent of the data to be scanned. One or several masks can be precomputed and used during sample acquisition. Further, applying random sampling only needs one iteration for each 2D image, which means each surface only has to be scanned once.

Feature Adaptive Sampling

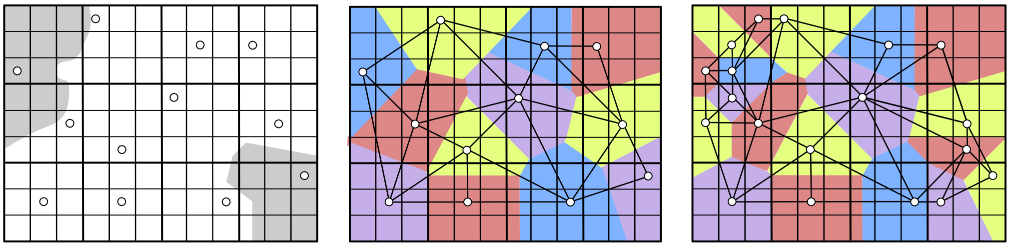
The second approach is called feature adaptive sampling. This recording scheme needs two acquisitions of the same image. In the first scanning round the whole image, i.e. each pixel position, is acquired with a low dwell time per pixel. The acquired image is used to detect edges in the image, as edges are considered very important features of an image. After applying an edge detection algorithm, e.g. canny detector [41], a thresholding is applied to convert the grey level gradient image into a binary image. In the second scanning round, all edge areas are scanned a second time with a high dwell time. The threshold is selected in such a way that the acquisition time of the initial scan and the acquisition time of the sparse scan sum up to the selected budget. These two images can then be used for reconstruction. For a detailed description of the whole procedure compare [61, 62]. Examples with different budget are shown in Figure 8.12 middle row.

Stratified Delaunay Sampling

The third approach we implemented and evaluated is primarily based on random stratified sampling [78] and Delaunay triangulation [73, 236]. Instead of scanning the whole specimen with a very low dwell time, or selecting pixels at random, the stratified Delaunay sampling mixes ideas from both. First, a fixed number of pixels are scanned randomly in a stratified way. That means prior to the random sampling, the surface to be scanned is pre-partitioned into n disjoint sets. The available number of pixels for the initialisation is also divided by n. Random pixels are scanned in each subset independently (Figure 8.10a). This procedure ensures that pixels are distributed uniformly over the whole image to prevent under-sampling in some regions and oversampling in others.

Afterwards, a fixed number k of iterations is selected. The number of remaining pixels to be scanned is divided by k, which results in a pixel number increment Δp. In each iteration Δp pixels are scanned based on the following importance scheme. The initial pixel pattern is used to compute a Voronoi diagram [181] and its dual representation, the Delaunay triangulation (Figure 8.10b). On the obtained graph the absolute differences of connected nodes, which represent the pixel values, are computed as edge weights. In homogeneous areas edge weights are near 0, which means there would be no high information gain if we were to sample additional pixels in between. In contrast, at transitions from edges to background the weights are larger. Edge detection algorithms via gradient computations work similarly. The greater a weight is, the more structure may be present between the connected pixels.

In order to determine the next pixels to be scanned, the weights are sorted in descending order and the top Δp edges are selected for further sampling. We put an additional node at the middle of each edge from the top Δp candidates and identify the corresponding pixel position in the image. These are scanned in the next iteration by the microscope. An iterative update of the Delaunay triangulation is calculated as well (Figure 8.10c). Continuing with the new graph, the procedure is repeated until the available pixel budget



(a) Scanning is initialised by random stratified sampling of a fixed amount of pixels.

(b) Neighbourhood relations are determined by computing a Voronoi diagram and performing a Delaunay triangulation.

(c) The diagram is refined iteratively by placing additional samples on edges that fulfil some gradient-based subdivision criterion.

Fig. 8.10.: Adaptive sparse sampling based on stratified Delaunay triangulation.

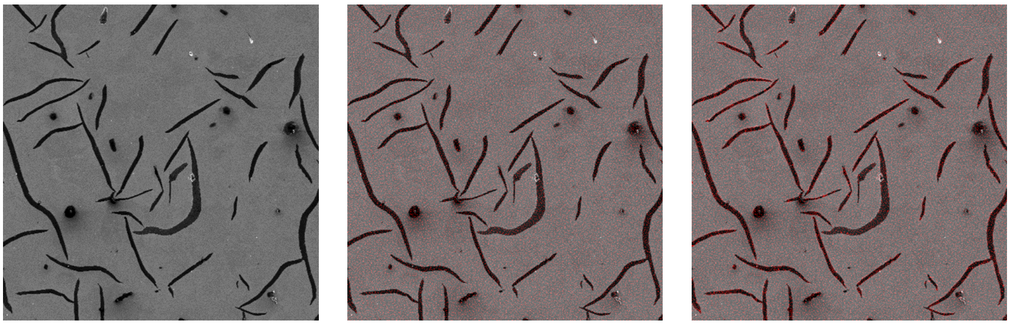
has been scanned. An example for this procedure is depicted in Figure 8.11. Examples with different budget are shown in Figure 8.12 lower row.

Comparison of Recording Schemes

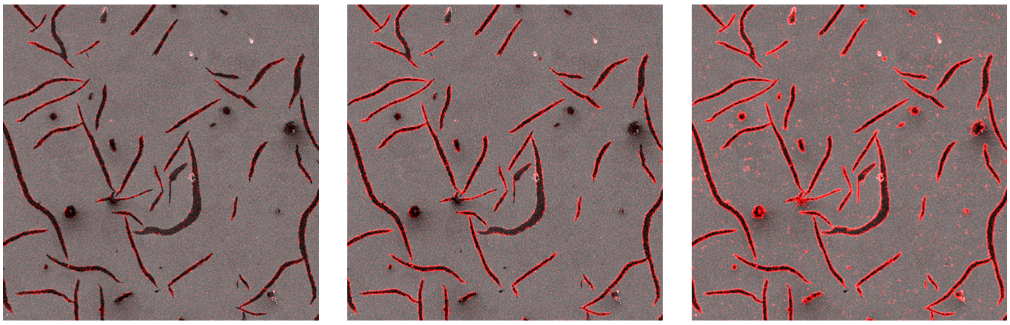
Figure 8.12 shows enlarged excerpts for each of the presented recording schemes with different pixel percentages. Judging only the scanning masks, the structure of the image is barely recognizable with random Bernoulli sampling. However, the pixels that should be scanned are distributed evenly over the whole image, which helps that reconstruction algorithms are not mislead due to local missing information. This is the main problem of feature adaptive sampling. The most prominent edges are scanned, but large areas of the image are not available. This can lead to wrong reconstructions if such areas are not very homogeneous. The Delaunay sampling combines both of the advantages of the aforementioned recording schemes. Both local pixel information and a higher density at structured regions like edges are contained in the sparsely acquired data.

8.3.3 Reconstruction

Having learned an acceleration structure for the available dictionary data the server is ready to perform reconstructions based on accelerated EBI. It is also possible to use the non-processed dictionary data as input when operating in brute force mode. While the microscope is scanning the sample that should be acquired slice by slice, the server reconstructs each sparsely sampled slice with some time delay. For an efficient workflow, the reconstruction procedure should not be much slower than the time required by the microscope to acquire the images, so that only the accelerated version of EBI is useful in a real acquisition scenario.

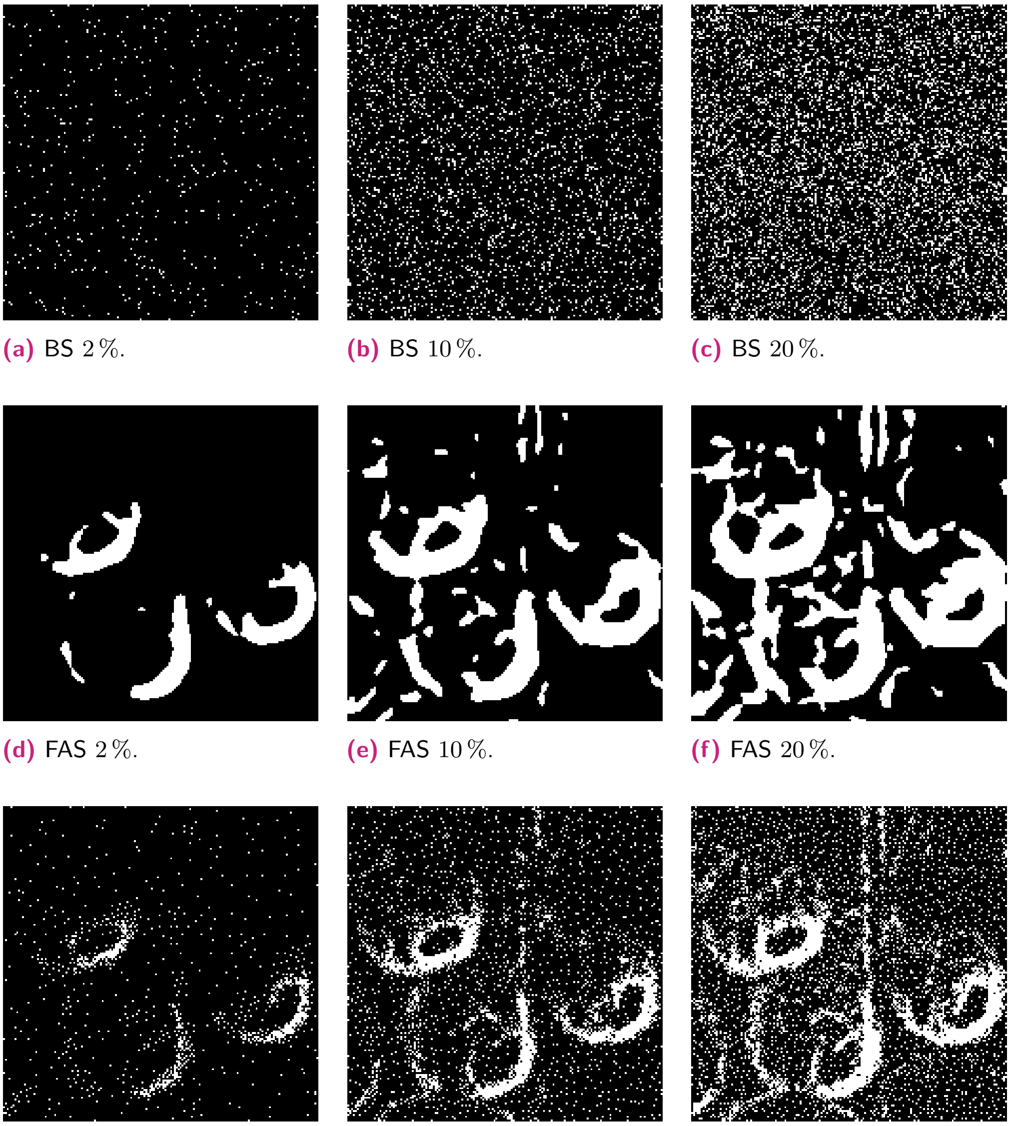


(a) Reference. **(b)** Initialisation $10\,\%$. **(c)** $11\,\%$.



(d) $12\,\%$. **(e)** $15\,\%$. **(f)** $20\,\%$.

Fig. 8.11.: Simulation for stratified Delaunay sampling. The selected pixels are shown in red. Initially, $10\,\%$ of the data has been picked randomly with stratified sampling and a sub-image size of 32×32. In each iteration $1\,\%$ additional pixels have been added by applying the presented importance scheme.


(a) BS $2\,\%$.

(b) BS $10\,\%$.

(c) BS $20\,\%$.

(d) FAS $2\,\%$.

(e) FAS $10\,\%$.

(f) FAS $20\,\%$.

(g) DS $2\,\%$.

(h) DS $10\,\%$.

(i) DS $20\,\%$.

Fig. 8.12.: Comparison of recording scheme masks with Bernoulli Sampling (BS), Feature Adaptive Sampling (FAS), and Delaunay Sampling (DS). Depicted are enlarged excerpts from simulated sampling masks with different pixel percentages extracted from Figure 8.13a.

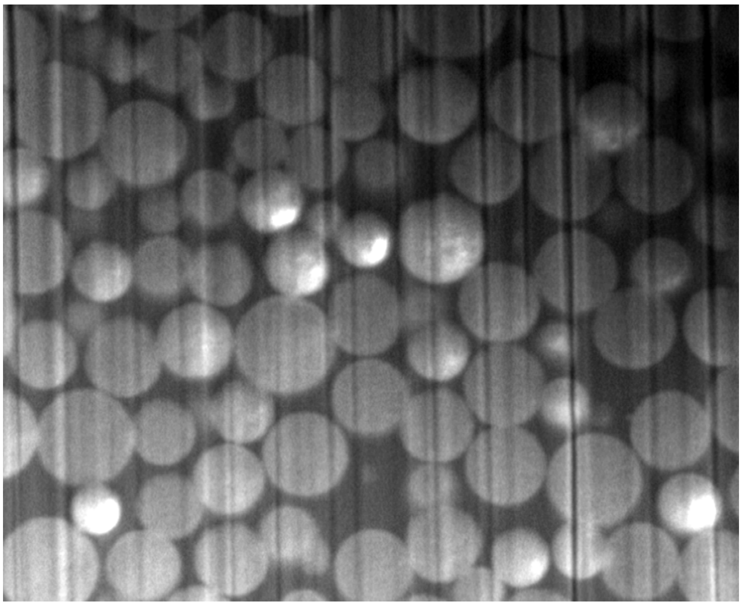
8.4 Proof of Concept for Sparse Scanning Workflow

We present some experiments for applying the sparse scanning workflow based on EBI presented in Section 8.3 as proof of concept for the method.

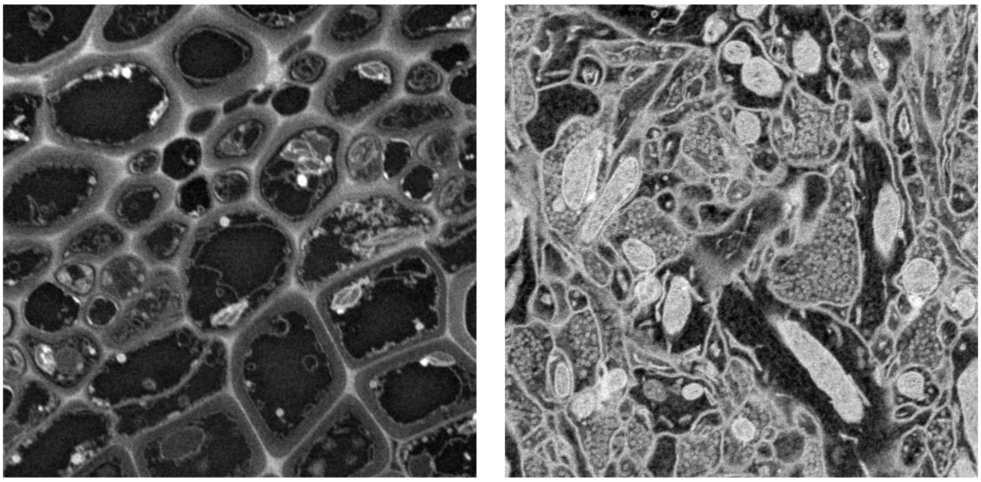
8.4.1 Datasets

We used several datasets for the evaluation of both the sampling schemes and EBI. The main dataset was an image stack of filled silica micro balls inside an accumulator. The silica balls dataset was scanned with a FIB-REM Strata 400 S microscope from FEI. The data was acquired with an ETD-SE detector and a dwell time per pixel of $30\,\mu s$. The images have been aligned with Fiji [215]. The workflow was then tested on this image stack by extracting 200 image slices as dictionary. One image that was not contained in the dictionary data was used to simulate the sparse data acquisition with a budget corresponding to $2\,\%$, $10\,\%$, and $20\,\%$ of the original data.

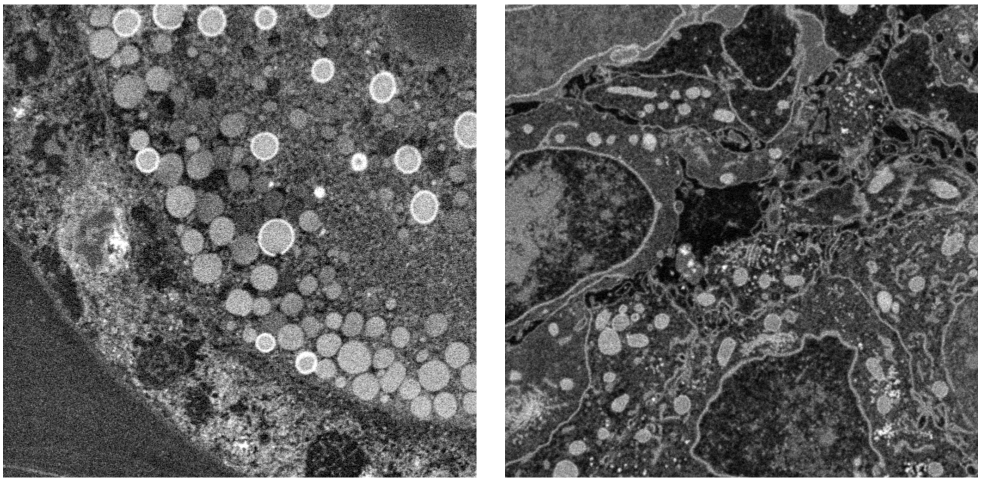
For further comparisons we used four different datasets that were acquired on a SEM. Dataset 1 (DS1) was a Arabidopsis thaliana sample with a resolution of $1,520 \times 828 \times 91$. Dataset 2 (DS2) was a mouse brain sample with a resolution of $3,277 \times 2,779 \times 100$. Dataset 3 (DS3) was an ovocyte sample with a resolution of $2,808 \times 3,099 \times 96$. Dataset 4 (DS4) was a cells sample with a resolution of $3,480 \times 2,172 \times 124$. For each dataset, the last image of the stack was cropped to $1,024 \times 1,024$, respectively 828×828 for DS1. These crops were used to simulate Bernoulli and Delaunay sampling with a $20\,\%$ pixel budget from the original data. As dictionary all but the last 10 images were used with the original resolution.



(a) Filled silica balls inside an accumulator.



(b) Arabidopsis thaliana sample (D1). **(c)** Mouse brain sample (D2).

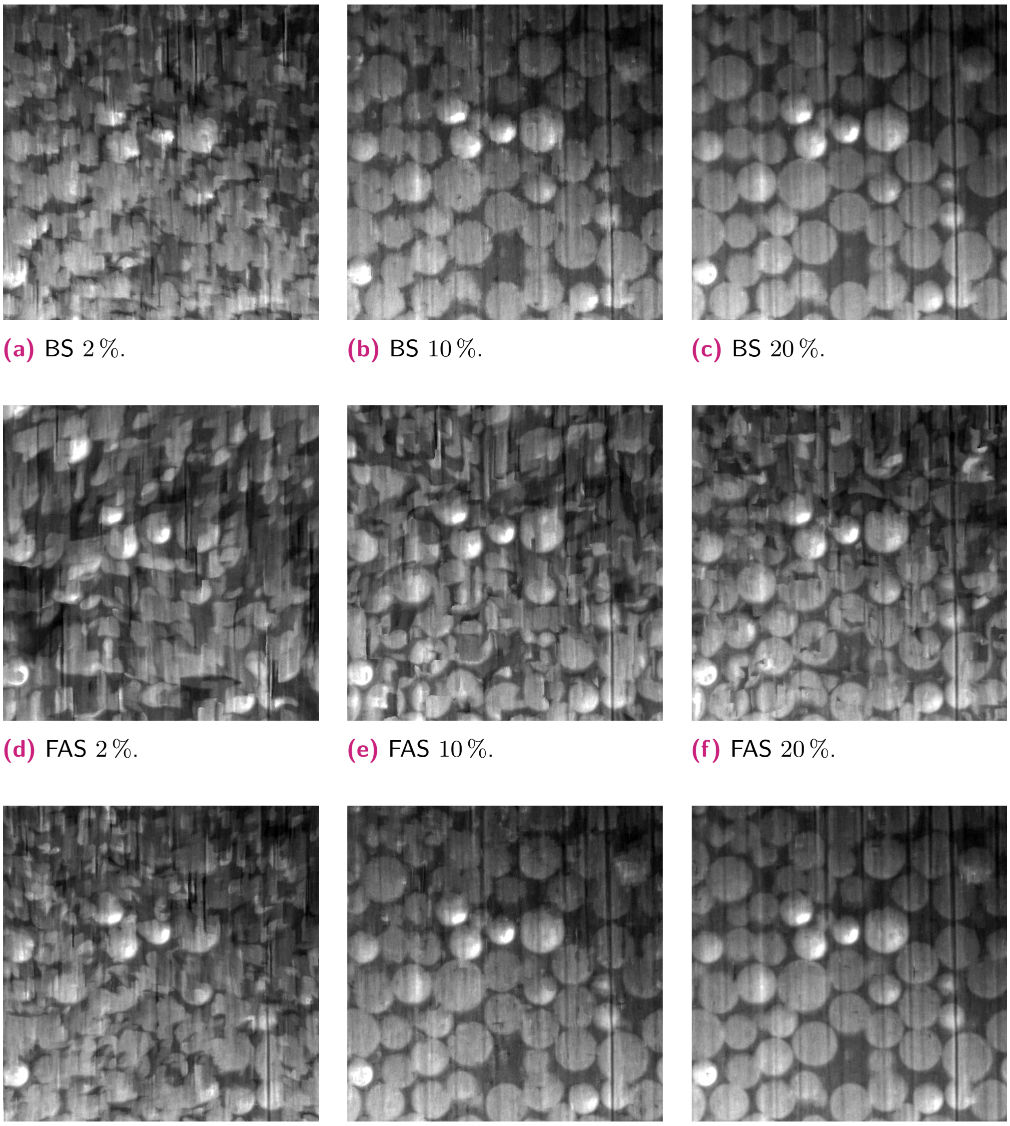


(d) Ovocyte sample (D3). **(e)** Cells sample (D4).

Fig. 8.13.: Example images from datasets for sparse SEM experiments.

8.4.2 Results

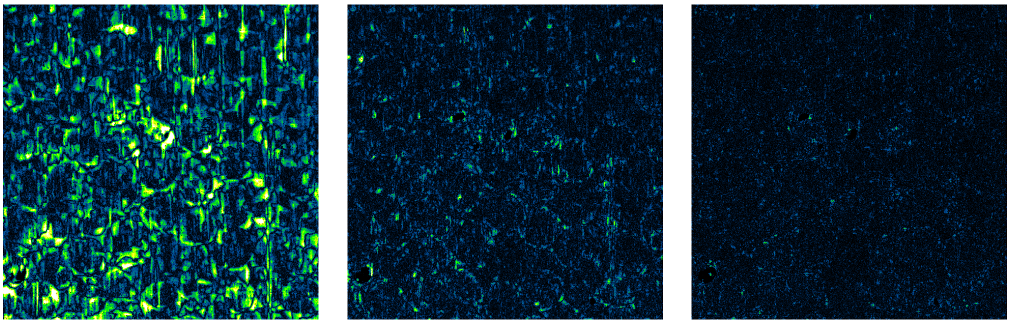
We evaluated the different recording schemes with the data specific dictionary for silica micro balls (Figure 8.13a). The different recording schemes were simulated based on full scans of the dataset. For a qualitative comparison we used the brute force version of EBI in a 2D setting, which should simulate the slice by slice reconstruction in a microscope. Reconstruction results are depicted in Figure 8.14.


(a) BS $2\,\%$. (b) BS $10\,\%$. (c) BS $20\,\%$.

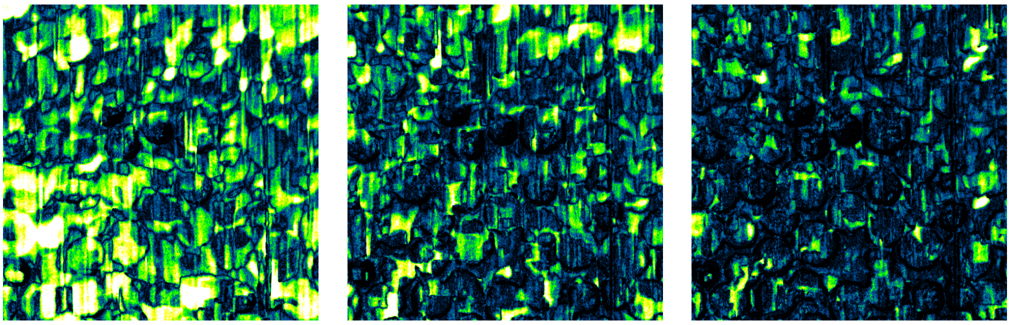
(d) FAS $2\,\%$. (e) FAS $10\,\%$. (f) FAS $20\,\%$.

(g) DS $2\,\%$. (h) DS $10\,\%$. (i) DS $20\,\%$.

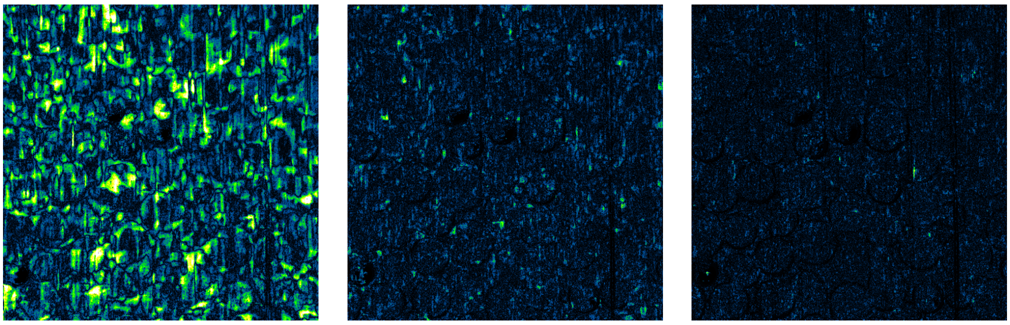
Fig. 8.14.: Comparison of reconstructions with different pixel budget based on Bernoulli Sampling (BS), Feature Adaptive Sampling (FAS), and Delaunay Sampling (DS).



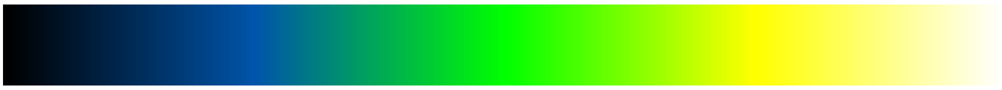
(a) BS 2 %. (b) BS 10 %. (c) BS 20 %.



(d) FAS 2 %. (e) FAS 10 %. (f) FAS 20 %.



(g) DS 2 %. (h) DS 10 %. (i) DS 20 %.



(j) Colour gradient, error increases from left to right.

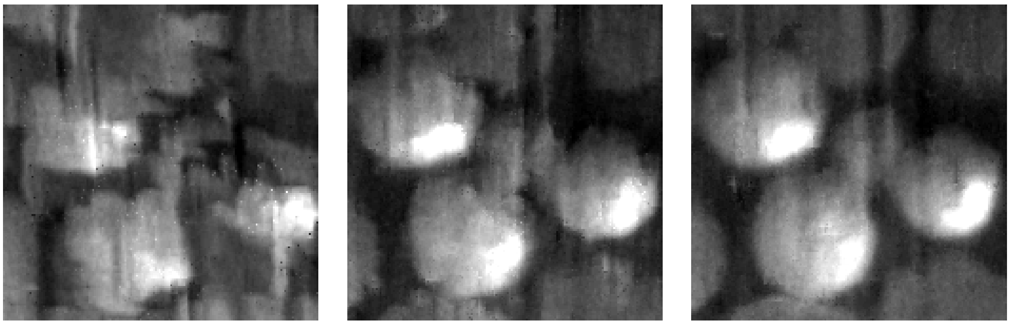
Fig. 8.15.: Corresponding error images for Figure 8.14 based on Bernoulli Sampling (BS), Feature Adaptive Sampling (FAS), and Delaunay Sampling (DS).

At the first glance it becomes obvious, that Feature Adaptive Sampling is not suited for very structured data. Looking at evaluation measures in Table 8.5 confirms this. Regarding the depicted evaluation measures and also the error images in Figure 8.15, Bernoulli Sampling and Delaunay Sampling reconstructions deliver similar results. However, as already argued in Sections 3.6, 7.4 and 8.2, evaluation measures may be misleading. Because of this it is mandatory to look closely at small important structures that are

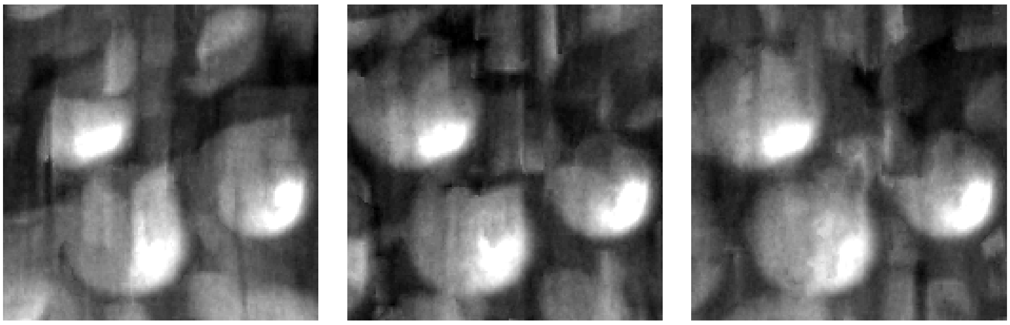
Tab. 8.5.: Evaluation measures for reconstructions of the silica balls dataset with different pixel budgets. We compared feature adaptive sampling (FAS), Bernoulli sampling, and Delaunay sampling. Depicted are PSNR / SSIM / max pixel error. Other figures of merit already used during this thesis are not depicted as their values behave similarly.

	FAS	Bernoulli Sampling	Delaunay Sampling
2 %	13.99 / 0.38 / 232	19.67 / 0.47 / 201	19.54 / 0.47 / 175
10 %	17.91 / 0.50 / 187	27.72 / 0.70 / 126	28.07 / 0.69 / 109
20 %	21.38 / 0.61 / 164	31.81 / 0.80 / 61	31.61 / 0.78 / 93

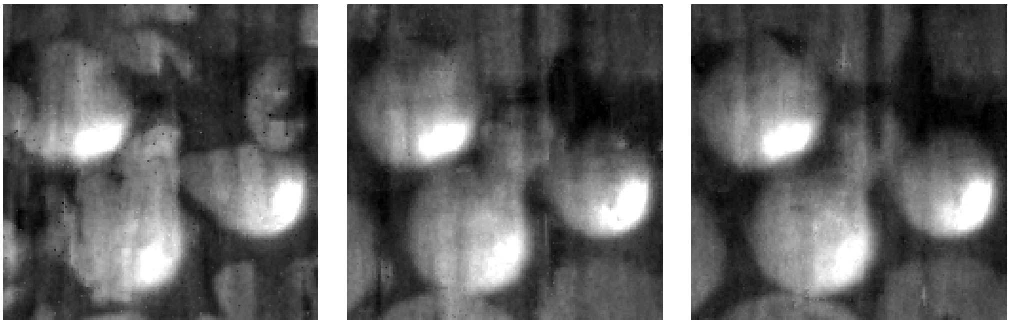
of scientific interest. For the silica balls that means we should look at individual micro balls and their image quality. Figure 8.16 serves this purpose.



(a) BS 2 %. **(b)** BS 10 %. **(c)** BS 20 %.



(d) FAS 2 %. **(e)** FAS 10 %. **(f)** FAS 20 %.



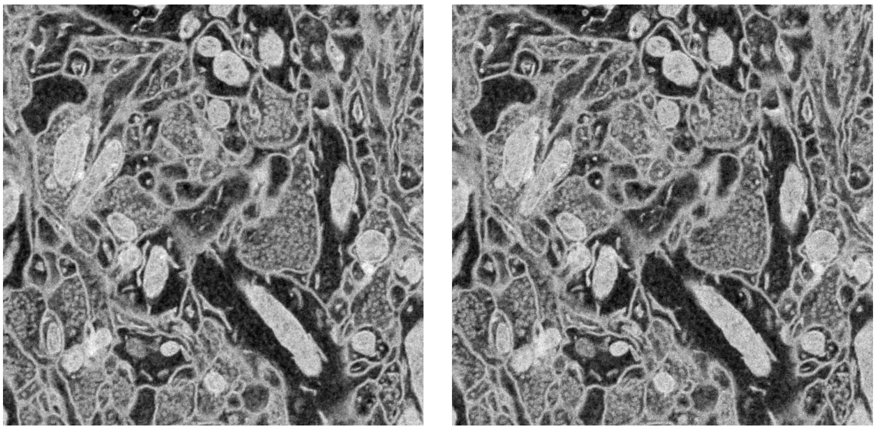
(g) DS 2 %. **(h)** DS 10 %. **(i)** DS 20 %.

Fig. 8.16.: Comparison of recording scheme reconstructions corresponding to Figure 8.12 based on Bernoulli Sampling (BS), Feature Adaptive Sampling (FAS), and Delaunay Sampling (DS) with different pixel percentages.

The Delaunay sampling based reconstructions in the lower row show least artefacts. This facilitates contour identification of the silica balls, which is closest to the real contours when acquiring the data via Delaunay sampling. Only for this sampling method, the silica balls have a round shape for the reconstructions when $10\,\%$ of the data had been scanned. For the reconstructions with the two other approaches, clear errors are visible and the silica balls shape is not round at some spots.

We performed an additional experiment to compare 2D and 3D reconstructions. A pixel budget of $20\,\%$ was simulated for DS2. The corresponding dictionary had a size of $1,280 \times 1,280 \times 90$. We selected a patch size of $\Psi = 9 \times 9 \times 5$. Bernoulli sampling reconstructions with the brute force GPU approach resulted in PSNR $= 24.32$, SSIM $= 0.66$, and a maximal pixel error of 100. For Delaunay sampling with the brute force GPU approach results were PSNR $= 23.63$, SSIM $= 0.61$, and a maximal pixel error of 98. Delaunay sampling also supports the z-curve acceleration structure, which resulted in PSNR $= 22.87$, SSIM $= 0.56$, and a maximal pixel error of 113.

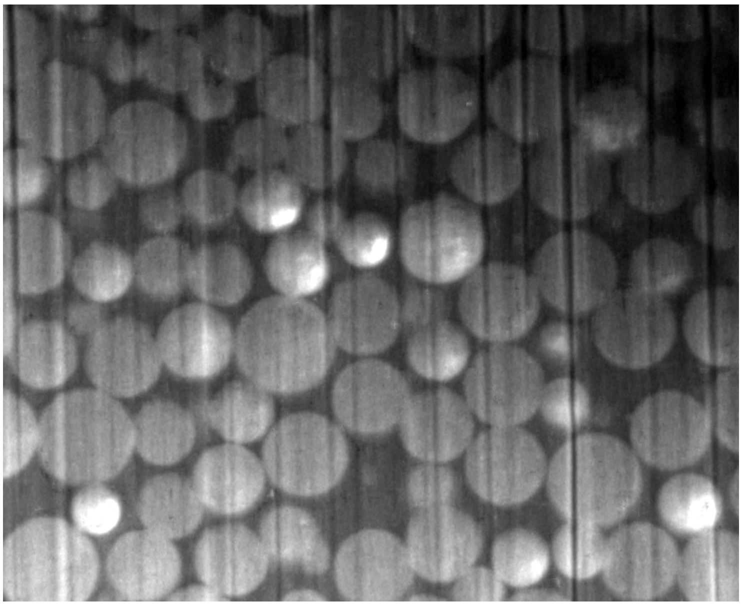
Further, we evaluated our approach in a 3D setting. The 3D experiments were run on an i7-6850K 6 core CPU with 3.6GHz with an Nvidia GeForce GTX 1080 Ti. Learning the dictionary took 30 minutes. The dictionary size on disk was 13.3GB, the required RAM for the reconstruction was 72.8GB. The pure reconstruction took 282 seconds. The brute force approach on GPU took $25,620$ seconds, so that the acceleration structure provided a speed-up of 90.9, which resulted in a reduction of 0.76 for the PSNR and a reduction of 0.05 for the SSIM. The maximal pixel error increased by 15. Figure 8.17 shows an example comparison between a brute force reconstruction and a z-curve based reconstruction.



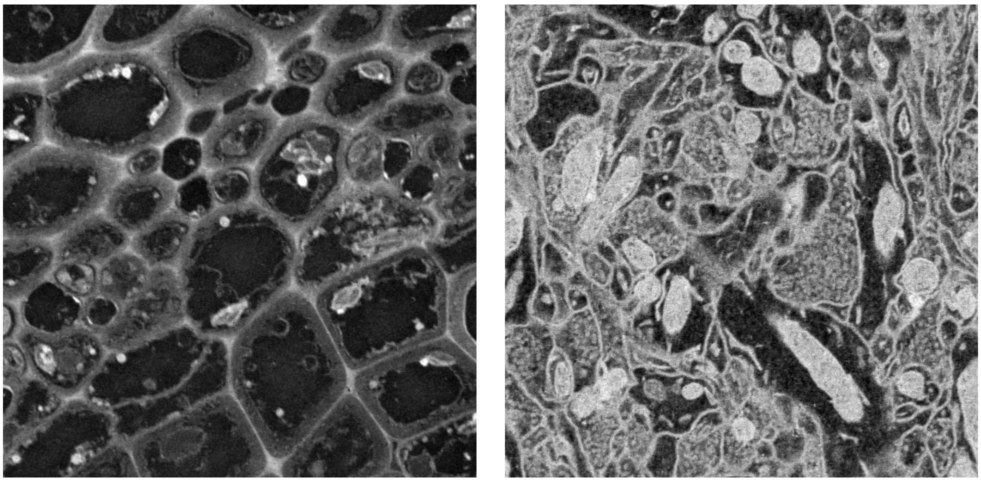
(a) Brute force inpainting. PSNR $= 23.63$, SSIM $= 0.61$, and maximal pixel error of 98.

(b) Z-curve based inpainting. PSNR $= 22.87$, SSIM $= 0.56$, and maximal pixel error of 113.

Fig. 8.17.: Comparison of 3D reconstruction of DS2 with and without acceleration structure based on Delaunay sampling. The patch size was $\Psi = 9 \times 9 \times 5$

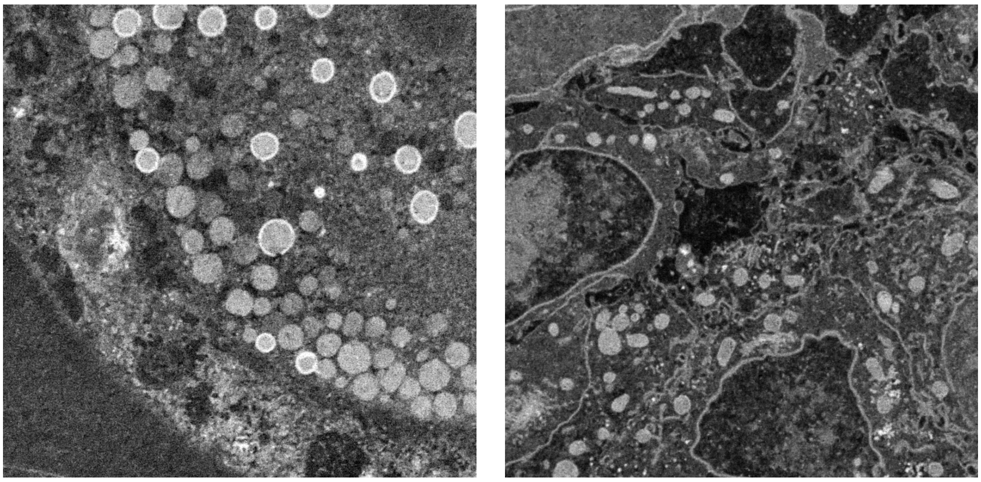


(a) Filled silica balls inside an accumulator.



(b) Arabidopsis thaliana sample.

(c) Mouse brain sample.



(d) Ovocyte sample.

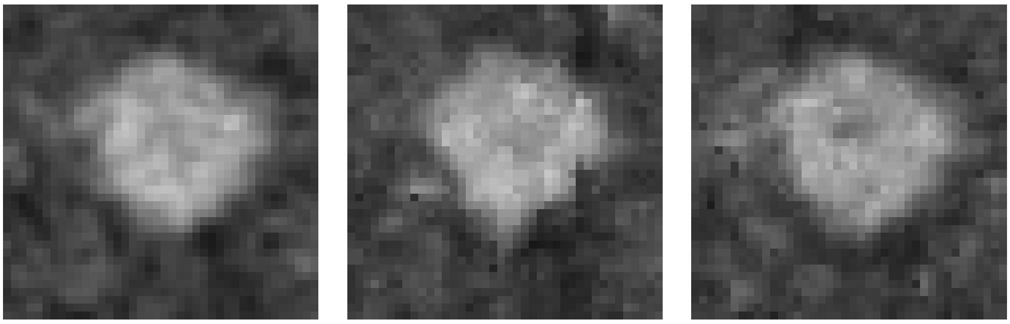
(e) Cells sample.

Fig. 8.18.: Reconstructed images using $20\,\%$ pixel budget corresponding to the examples from Figure 8.13. The sparse data was acquired using Delaunay sampling.

We applied only Bernoulli sampling and Delaunay sampling for further reconstruction comparisons, as we already excluded feature adaptive sampling based on the silica balls experiments. Table 8.6 depicts the evaluation measures corresponding to each of the datasets and the sampling methods. The measured values are quite similar for each comparison. There is no clear tendency, which sampling method performs better. Figure 8.18 illustrates the reconstructions based on Delaunay sampling. As we did in the silica balls experiment, we enlarged details of the various reconstructions to gain more insights into which sampling method may be better. We observed that Bernoulli sampling tended to deteriorate small structures, whereas Delaunay sampling was able to reconstruct contours (Figure 8.19). Based on these observations and the results from the silica balls experiment, Delaunay sampling should be preferred over Bernoulli sampling.

Tab. 8.6.: Evaluation measures for reconstructions of DS1 - DS4 (Figures 8.13b - 8.13e). Bernoulli and Delaunay sampling was simulated with a pixel budget of $20\,\%$. Depicted are PSNR / SSIM / max pixel error. Other figures of merit already used during this thesis are not depicted as their values behave similarly.

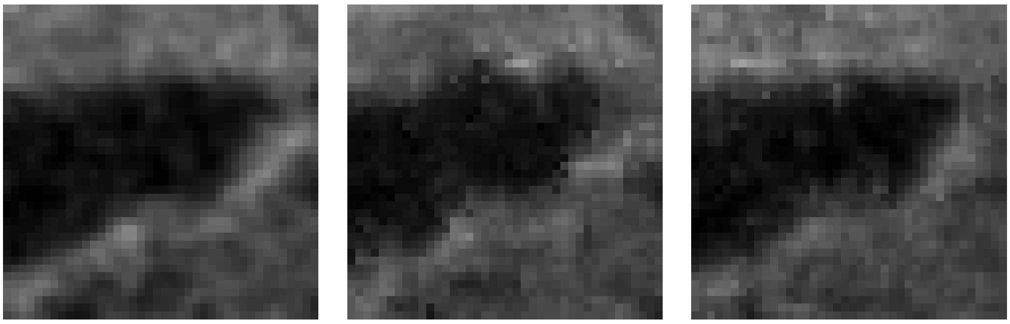
	Bernoulli Sampling	Delaunay Sampling
DS1	27.59 / 0.73 / 132	28.30 / 0.71 / 111
DS2	23.92 / 0.65 / 164	23.02 / 0.57 / 111
DS3	20.77 / 0.46 / 182	20.84 / 0.44 / 152
DS4	24.67 / 0.64 / 143	24.65 / 0.60 / 142



(a) GT excerpt 1 from DS4 (Figure 8.18e).

(b) Reconstruction of Figure 8.19a based on BS.

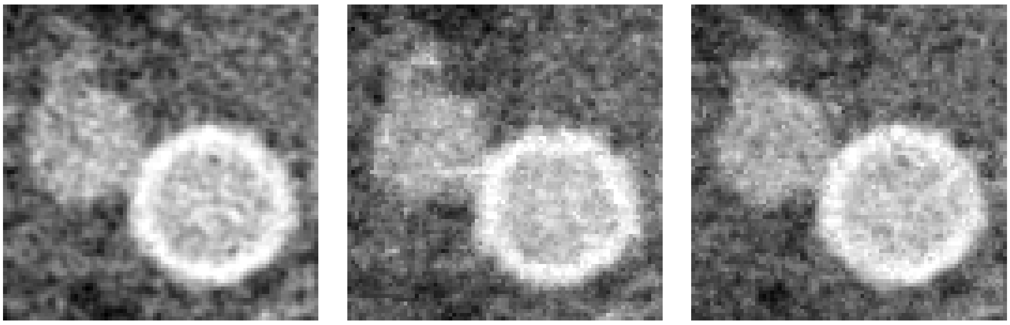
(c) Reconstruction of Figure 8.19a based on DS.



(d) GT excerpt 2 from DS4 (Figure 8.18e).

(e) Reconstruction of Figure 8.19d based on BS.

(f) Reconstruction of Figure 8.19d based on DS.



(g) GT excerpt from DS3 (Figure 8.18d).

(h) Reconstruction of Figure 8.19g based on BS.

(i) Reconstruction of Figure 8.19g based on DS.

(j) GT excerpt from DS1 (Figure 8.18b).

(k) Reconstruction of Figure 8.19j based on BS.

(l) Reconstruction of Figure 8.19j based on DS.

Fig. 8.19.: Comparison of enlarged details corresponding to Figure 8.18 between ground truth (GT), Bernoulli sampling (BS), and Delaunay sampling (DS).

8.4.3 Discussion

Depending on an appropriate recording scheme we were able to reconstruct images while preserving the visual quality. Feature adaptive sampling was not particularly well suited for very structured data. Instead it should be used for data that comprises many homogeneous areas as well as sparse gradients, so that the main features do not cover a large area of the image. For such data, applying EBI may be exaggerated as the proposed filtering method in [61] suffices. However, instead of only using the second scan, it could help to incorporate both acquired images when reconstructing sparse data with EBI. We experimented with this idea, but the other approaches worked much better for the sparse data acquisition and corresponding reconstructions, so we did not further investigate extending this recording scheme and excluded it from further experiments.

EBI worked quite well for the reconstruction of sparsely sampled data with both Bernoulli sampling and Delaunay sampling. Evaluation measures were similar for both methods in all of our experiments. However, investigating small structures of 40^2 to 100^2 pixels, visual comparisons brought up weaknesses of Bernoulli sampling. There were many examples where contours were not reconstructed adequately. These problems further lead to shape deformations, so that small structures may be misinterpreted after reconstruction. Delaunay sampling, due to its nature of sampling more information at spots with higher information density, had no such problems. This fact renders Delaunay sampling the best sampling method of the proposed ones and again showed the misleading character of evaluation metrics on their own.

Judging the results of the 3D comparison with and without acceleration structure according to the evaluation measures, the loss in quality is small compared to the reconstruction time speed-up. Nevertheless, if this is severe for an application or not depends on the subsequent processing of the images. If small structures are reconstructed erroneously, an analysis of these structures may not be an ideal application for our accelerated EBI. However, if the goal is to identify mitochondria and membranes, as performed in Section 8.2, small artefacts and pixel errors may not hurt.

9

Conclusions and Outlook

> *We cannot solve our problems with the same thinking we used when we created them.*
>
> — **Albert Einstein**
> Physicist, Winner of the Nobel Prize

As our main goal for this thesis we investigated and developed methods for the estimation of missing data in electron microscopy (EM). We explored sub-sampling strategies to acquire data differently than collecting each possible data point in Chapter 7, which saves time and electron dose. Subsequent reconstruction methods should lead to qualitatively similar data when compared to a full sampling of a specimen to facilitate further processing. We proposed to use exemplar-based inpainting (EBI) to generate the missing data, because EM data in general contains a high degree of structure, and EBI operators are most suited for texture-like data. Furthermore, we accelerated EBI using a high performance, parallel graphics processing unit (GPU) implementation as well as a newly proposed acceleration structure. The usefulness of EBI in EM was then shown on the basis of two very different applications underlying fundamentally different EM techniques. Thereby, we also recommended to use an application driven evaluation measure in addition to visual comparisons and common metrics to foster the assessment of newly developed methods. The meaningfulness for such an approach was demonstrated before in Sections 7.4 and 8.2. We showed the benefits of this approach in a sparse EM reconstruction experiment as proof of concept.

Before implementing and applying EBI to EM data, we examined the capabilities of compressed sensing (CS) and sparse coding. We explained in Section 3.4 that inpainting operators inspired by these methodologies may not be able to generate missing data as strong assumptions have to hold in CS. As shown in Section 8.1, violating these assumptions regularly led to reconstructed or generated data whose quality was not useful for further analysis and interpretation of the data. Present microscopes do not have the ability to acquire data in such a way that prerequisites of CS are met. Considering a three-dimensional (3D) setting, the applicability of CS inspired reconstruction methods seemed inconceivable. Inspired by the ideas to inpaint missing data and the suitability for texture generation, we implemented a 3D version of EBI for application in EM.

We accelerated EBI to facilitate practical viability in EM, as described in Section 5.3. Our acceleration approaches are applicable to any patch based EBI. We also proved the runtime of our fastest approach according to the current assumptions and implementation in Appendix A to be in $O(\log(n)^2)$. In a two-dimensional (2D) setting we were able to speed-up EBI up to a factor of 660 compared to the Matlab implementation of the original approach as shown in Section 5.3.4. In a 3D setup we could not run the Matlab implementation. Therefore, we only compared our GPU version against the acceleration structure, which gave a maximal speed-up of 90.9 with a very large dictionary of size $13.3\,$GB. In 2D settings this comparison led to a speed-up of factor 2 to 7.

These numbers show the significance of the impact, especially in 3D, and make EBI a feasible option for the reconstruction of missing data in EM. The main contribution hereby lies in the fact that the acceleration approaches are independent from most model-based aspects of the inpainting algorithm. Thus, it can be combined with a large number of existing and future EBI algorithms. The acceleration approaches make the generalisation of EBI to 3D datasets such as video or volumetric data viable.

However, the current accelerated version of EBI is not yet suited for near real time application. Single beam scanning electron microscopes (SEMs) need around 1 second for the full acquisition of a $1,024 \times 1,024$ image with a dwell time per pixel of $1\,\mu$s. Subsequently, reconstructing while scanning with sufficient quality is not possible at the current state, respectively with the hardware we used in our experiments. Nevertheless, non-optimal quality reconstructions are already possible when strictly constraining the size of a dictionary, which can serve as a preview. Reconstructing the whole 3D stack can be performed afterwards and scaled with help of more powerful hardware than was used in this work. This procedure also ensures that the 3D data is completely utilised for reconstructions compared to a slice by slice 2D reconstruction.

Furthermore, dictionary quality is a mandatory requirement. It is obvious that sufficient data from the domain to be inpainted must be available for reconstructions based on a dictionary. Dictionaries trained on mouse brains, for example, are completely useless for silica ball reconstructions. Also it is not possible to recover new knowledge with our EBI approach. The algorithm can only reconstruct what it has already seen in available data. In the field of connectomics, for example, one goal is to acquire many healthy mouse brains to investigate neuron structures of the brain. Provided that all possible image patches that can be contained in a mouse brain are available in the dictionary, EBI is perfectly suited to reconstruct sparsely sampled data from hundreds of mouse brains. Detecting tumours, which in general are very heterogeneous, is not a plausible application for EBI, as each tumour may look different. That means, even if we had sampled hundreds of tumours, we cannot guarantee that the next tumour is a subset of the prior knowledge we have.

Also we must keep in mind that most experiments were performed based on simulated data. In real microscope settings there are further issues that might cause problems. One important issue for scanning electron microscopy (SEM) is the misalignment of image slices, e.g. due to drift or charging of the sample caused by electrons. For full scanning setups there are solutions to these problems, but for sparse scanning only ideas exist. Thus, our algorithms must be incorporated into a microscope workflow much more tightly, which involves much experimentation on a real microscope. Unfortunately, we had no such microscope available, so that an integration into a real workflow could not be evaluated.

Besides SEM, EBI was also applied to tilt-based EM datasets. For this application, only 3D EBI was applicable, as a 3D image stack must be extended in longitudinal direction. The artificial generation of non-available projections and subsequent tomographic reconstruction of inpainted stacks was explained and evaluated in Chapter 6. Inpainted and real projections complemented each other quite well. The heuristically generated projections helped to reduce the missing wedge in the Fourier domain, which also reduced elongation artefacts along the longitudinal direction of the 3D dataset in real space.

The impact was demonstrated on artificial as well as real data. Thereby, the limits of EBI for generating missing projections also became obvious. The inpainting operator only worked reasonably, when the tilt series angle step size was approximately equidistant with step sizes up to 1.5°. Without this condition, the data did not provide enough information for the inpainting, which means the source region may have been too small. Further, big tilt increments lead to gaps between projections that falsify neighbourhood relationships. The main concern of the method is the introduction of new artefacts. Hence, this drawback must be kept in mind when using EBI to inpaint a data stack. An additional comparison with original reconstructions may help to prevent wrong conclusions due to introduced artefacts. Considering the conducted experiments, EBI helps to enhance the quality of tomographic reconstructions, which can facilitate the interpretation of acquired data.

During many of our experiments it was mandatory to evaluate the quality of reconstructed images. Common evaluation measures did not always help to decide which of the compared methods was better. As explained in Section 3.6 and shown in Sections 7.4 and 8.4.2, common metrics like peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) can be very misleading. Visual comparisons can improve the differentiation between results, however, they may also be very subjective if the difference is not obvious. Based on these observations, we presented an application driven evaluation procedure in Section 8.2.

In connectomics, the segmentation of membranes and mitochondria is the main goal in post-processing the acquired images, which serves as foundation for further research. We took this use case as a proof of concept example. The application driven measure for this experiment was the segmentation quality of full scans compared to reconstructed sparse scans. Although PSNR and SSIM showed clear advantages for the sparse reconstructions, the new procedure revealed that sparse reconstructions are not as superior as expected by interpreting common metrics, with regard to the application the images were acquired for. Hence, without incorporating the intended purpose into an evaluation the results may be useless. We recommend to think about appropriate application driven evaluations when developing algorithms.

Summary. We have shown that CS inspired inpainting should be used with caution when reconstructing sparsely sampled data, which we already assumed to be problematic in Section 4.5.1. Inspired by optimisation based inpaintings in CS we proposed to use EBI, as it only needs texture-like data and does not underlie the strong assumptions required by CS. Furthermore, we have proposed ways to accelerate EBI to make its use feasible, hereby the acceleration approaches are independent from most model-based aspects. This makes our acceleration approaches applicable to any kind of patch based EBI.

We utilised EBI in two quite different EM applications and demonstrated its impact, once by reducing missing wedge and elongation artefacts in tilt-based microscopy and once by reconstructing sparsely sampled data. In the course of sparse data reconstructions, we also investigated appropriate sampling schemes and showed that a clever adaptive sampling can lead to superior results when compared to random sampling. Also, we illustrated that application driven evaluation procedures may be superior to classical metrics and pure visual comparisons. Our contributions complement existing approaches for missing data generation in EM. The promising results demonstrate that EBI can be beneficial for appropriate EM problems.

9.1 Future Work

Integrating EBI in EM workflows was demonstrated to be beneficial in selected applications. Nevertheless, while we were able to answer many questions, many new questions emerged. In the following we gather suggestions for additional research items that go beyond the scope of our contributions.

In this thesis we concentrated mainly on the EBI approach from Criminisi et al. [58]. There are many different approaches that, for example, use different strategies for the filling order. Evaluating ingredients from other approaches and combining them in a new EBI adapted to EM data, also individually tailored for specialised techniques, is an interesting direction. In the course of such an investigation, we could also validate that our acceleration structures work with basically any EBI approach.

Extending EBI with prior knowledge could help to guide the inpainting process. For projections of a tilt series, for example, particles are moving on sinusoidal curves in the sinogram of each image slice. Incorporating this knowledge may help to enhance the artificial generation of non-available projections. In a SEM setting, 3D information could be used as proxy for a ground truth. For a small resolution in z-direction subsequent image slices are very similar. Hence, having scanned the first image slice, this slice could be used as proxy ground truth for the computation of an optimal scanning mask for the next slice. Also it may be possible to use this proxy as baseline for a densification approach. Combining these two ideas could foster the mutual influence of adaptive sampling and corresponding reconstruction.

Regarding the already accomplished acceleration possibilities, there is still room for further improvements. Our developed acceleration structure may be optimised, so that learning dictionaries works faster while reducing the memory consumption. Improving the implementation of the acceleration structure further on the low level could lead to a better possible runtime.

Optimising the data that is stored in a dictionary is a different way. Dictionaries with a size over 13 GB use more than 73 GB of RAM during the inpainting process and are thus not practical for a broad usage. Hence, we could think about schemes that reduce the number of patches, for example by discarding very similar patches or never used patches. Investigating further into the influence of the dictionary and determining properties, for example with help of statistical measures of contained patches, may help to generate smaller and more robust dictionaries in terms of possible reconstruction quality.

In addition to enhancements of EBI, integrating our workflow into a real SEM should be one of the next steps. The interplay from the microscope and our reconstruction will very probably raise new problems and questions that could be addressed. One obvious problem is the alignment of sparse data, so that 3D reconstruction algorithms can operate on the sparsely scanned raw data. Introducing a quality measure and at the same time an online learning scheme for updating the dictionary towards higher quality reconstructions is a further possible option.

With the advent of deep learning many new possibilities have arisen. Incorporating machine learning approaches based on convolutional neural networks is a very interesting and challenging task to enhance or even replace EBI. The capabilities of neural networks seem to be astonishing. First experiments during the last weeks yielded very impressive and promising results for the reconstruction of sparsely sampled data. Furthermore, deep learning may help to develop adaptive sampling strategies that are superior to current state of the art strategies. The opportunities seem inexhaustible, so that investigating the capabilities and limitations in itself may be worth a doctoral thesis.

Appendices

A

Traversal Algorithm Runtime Proof

> *There is something about the mindset of a scientist that is different - an awareness of uncertainty, modeling, proof.*
>
> — **Joseph Stiglitz**
> (American economist)

Hypothesis

The traversal algorithm from Section 5.3.2 has runtime $O(\log(n)^2)$ where n is the size of the dataset.

Assumptions

We use the following assumptions:

(a) The points in the dataset are uniformly distributed in each dimension.
(b) The cost function f is a norm.

Proof Notation

The binary representation of numbers is annotated with subscript "binary", e.g. $_{\text{binary}}101 = 5$. Squared brackets are used to index bits in a binary number, e.g., $R[m]$ is the m^{th} bit of the binary representation of the number R. We use $_{binary}0..0_{\{n\}}$ to annotate a binary string consisting of n times the digit 0. All symbols used during the proof and their meaning are depicted in Table A.1.

Symbol	Semantic
n	Number of points in the data set.
D	Dimensionality of search space.
nbit	Number of bits required to represent the search space per dimension, i.e. the size of the search space is 2^{nbit} in each dimension.
j	Traversal depth.
R_j^d	Active search space at traversal depth j, R is an axis aligned hyper cuboid.
$R_{\text{lower},j}^d$	Lower bound of R at traversal depth j in dimension d.
$R_{\text{upper},j}^d$	Upper bound of R at traversal depth j in dimension d.
(min, max)	Interval of indices in the dataset that correspond to R, i.e. min is the index of the first data point inside R, max is the index of the last data point in R.
$z_{\text{lower},j}, z_{\text{upper},j}$	Binary representation of the z-space coordinate of the interval corresponding to the active search space at traversal depth j.
V_0	Volume of search space $V_0 = 2^{\text{nbit}\cdot D}$ units.
V_j	Volume of active search region at traversal depth j.
$V_{\lvert f \rvert}^D$	Volume of D-dimensional ball under norm f.
ρ	Point density of the dataset $\rho = \frac{n}{V_0}$.
r	Norm value, also D-ball radius.
μ	Minimal number of data points to split recursively, threshold.
q	Query object.

Tab. A.1.: Notation for Traversal Algorithm Runtime Proof

Proof

The proof consists of two steps.

(i) First, we consider the recursive subdivision scheme that splits the dataset into two parts in each traversal step. We show that this subdivision reaches a depth of $O(\log(n))$, which basically means showing that the subdivision is sufficiently balanced.

(ii) Second, we show that once the subdivision reaches the maximum recursion depth, the algorithm only visits $O(\log(n))$ many parts of the dataset. Each subdivision step includes a binary search on the dataset, so that the cost per step is $O(\log(n))$ and we obtain a total runtime in $O(\log(n)^2)$.

(i) The recursive subdivision reaches a depth of $O(\log(n))$.

At the beginning of the algorithm, i.e. after 0 traversal steps, the active search region R_0 is the entire search space. It has the form of a D-dimensional hypercube that starts in the origin and spans 2^{nbit} units in each positive direction. Therefore for each $d \in \{1..D\}$, where D is the dimension of the search space, it holds that

$$\begin{aligned} R^d_{\text{lower},0} &= & 0 &= {}_{\text{binary}}0..0_{\{\text{nbit}\}}, \text{ and} \\ R^d_{\text{upper},0} &= & 2^{\text{nbit}} - 1 &= {}_{\text{binary}}1..1_{\{\text{nbit}\}}. \end{aligned} \tag{A.1}$$

The z-space representation of the initial active search space is the interval $(z_{\text{lower},0}, z_{\text{upper},0})$, where z_{lower} is the Morton encoding of $R^d_{\text{lower},0}$ and $z_{\text{upper},0}$ is the Morton encoding $R^d_{upper,0}$. Following this, z_{lower} is obtained by interleaving the bits of $R^d_{\text{lower},0}$ for all d and $z_{\text{upper},0}$ is obtained by interleaving the bits of $R^d_{\text{upper},0}$ for all d. Therefore:

$$\begin{aligned} z_{\text{lower},0} &= {}_{\text{binary}}0..0_{\{\text{nbit}\cdot D\}}, \text{ and} \\ z_{\text{upper},0} &= {}_{\text{binary}}1..1_{\{\text{nbit}\cdot D\}}. \end{aligned} \tag{A.2}$$

Lemma A.0.1. At traversal depth j, when the active search region was split j times, the active search interval takes the form $z_{\text{lower},j} = {}_{\text{binary}}p_1..p_j\ 0..0_{\{(\text{nbit}\cdot D)-j\}}$, and $z_{\text{upper},j} = {}_{\text{binary}}p_1..p_j\ 1..1_{\{(\text{nbit}\cdot D)-j\}}$, where $p_1..p_j$ is an arbitrary prefix bit string of length j.

Proof of Lemma A.0.1

Inductive hypothesis:

The Lemma itself.

Base case $j = 0$:

Lemma A.0.1 follows directly from Equation (A.2) as there is no prefix.

Inductive step $j \to j + 1$:

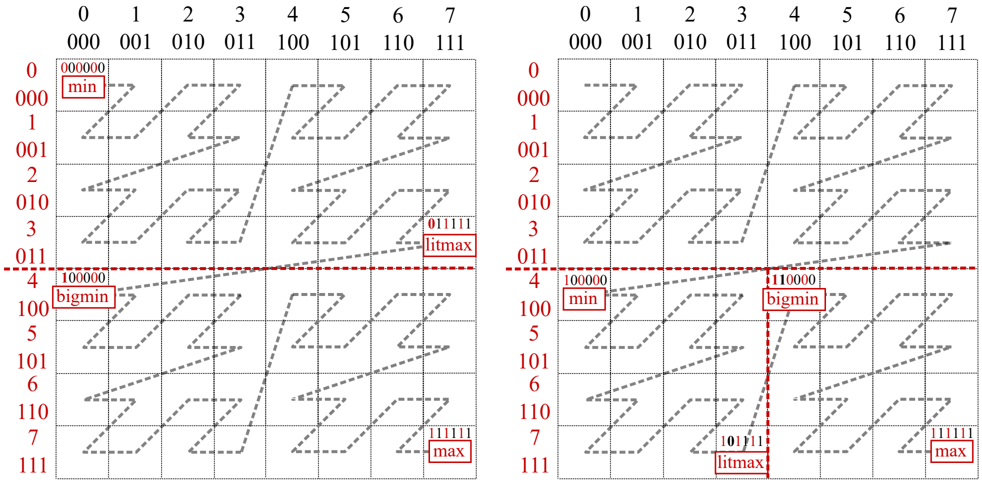
The litmax/bigmin algorithm chooses the split position to be the position where the most significant bits differ between $z_{\text{lower},j-1}$ and $z_{\text{upper},j-1}$ (compare Figure A.1). The value litmax is selected to be the highest value smaller than the split position:

$$\text{litmax}_j = {}_{\text{binary}}p_1..p_{j-1}01..1_{\{(\text{nbit}\cdot D)-j\}}.$$

The value bigmin is selected to be the smallest number higher than the split position:

$$\text{bigmin}_j = {}_{\text{binary}}p_1..p_{j-1}10..0_{\{(\text{nbit}\cdot D)-j\}}.$$

If the traversal algorithm goes to the left branch, then $z_{\text{lower},j+1} = z_{\text{lower},j}$ and $z_{\text{upper},j+1} = \text{litmax}_j$, so $p_j = 0$ and Lemma A.0.1 holds by the inductive hypothesis. If the traversal algorithm goes to the right branch, then $z_{\text{lower},j+1} = \text{bigmin}_j$ and $z_{\text{upper},j+1} = z_{\text{upper},j}$, so $p_j = 1$ and Lemma A.0.1 holds as well. ☐



(a) Splitting plane 1.

(b) Splitting plane 2.

Fig. A.1.: Two steps of the litmax/bigmin algorithm.

The size of the active search interval at recursion depth j is:

$$\begin{aligned} z_{\mathsf{upper},j} - z_{\mathsf{lower},j} &= \mathsf{binary} p_1..p_j 1..1_{\{\mathsf{nbit}\cdot D-j\ times\}} \\ &\quad - \mathsf{binary} p_1..p_j 0..0_{\{\mathsf{nbit}\cdot(D-j)\}} \\ &= \mathsf{binary} 1..1_{\{\mathsf{nbit}\cdot(D-j)\}} \\ &= 2^{\mathsf{nbit}\cdot(D-j)} - 1. \end{aligned} \tag{A.3}$$

In other words, every step splits the active search interval exactly in half. The positions of the data points in the search space are at random positions and distributed uniformly (assumption a). Thus, the number of data points in a segment of z-space addresses is proportional to the length of the segment. Let

$$\rho := \frac{n}{2^{\mathsf{nbit}\cdot D}} \tag{A.4}$$

be the average number of data points per unit volume in the set. Because the data points are distributed uniformly, it holds for any interval $(z_{\mathsf{lower},j}, z_{\mathsf{upper},j})$ that it contains an expected value of n_j many data points such that $n_j = (z_{\mathsf{upper},j} - z_{\mathsf{lower},j}) \cdot \rho$. Let $(z_{\mathsf{upper},j}, z_{\mathsf{lower},j})$ be the active search interval after step j, then it contains n_j data points with

$$\begin{aligned} n_j &= \rho \cdot (z_{\mathsf{upper},j} - z_{\mathsf{lower},j}) \\ &\overset{A.4}{=} \left(\frac{n}{2^{\mathsf{nbit}\cdot D}}\right) \cdot (z_{\mathsf{upper},j} - z_{\mathsf{lower},j}) \\ &\overset{A.3}{=} \left(\frac{n}{2^{\mathsf{nbit}\cdot D}}\right) \cdot (2^{(\mathsf{nbit}\cdot D)-j} - 1) \\ &\leq \left(\frac{n}{2^{\mathsf{nbit}\cdot D}}\right) \cdot (2^{(\mathsf{nbit}\cdot D)-j}) \\ &= \frac{n}{2^j} \end{aligned}$$

By definition, the algorithm continues the recursive subdivision as long as $n_j > \mu$, so we obtain that $\frac{n}{2^j} > \mu \Leftrightarrow j < \log_2(\frac{n}{\mu})$. As μ is constant the algorithm reaches a recursion depth of $O(\log\, n)$. For each level of the recursion, the algorithm performs a binary search on the presorted dataset, which takes $O(\log\, n)$ time, so that the total time for the recursive subdivision is $O(\log^2\, n)$.

(ii) Per step in the subdivision, the algorithm will only visit O(1) parts of the dataset.

Once $n_j < \mu$, the algorithm starts traversing, so the traversal depth stays constant and we know that the depth is

$$j < \log_2(n/\mu) \tag{A.5}$$

Now, the number of data points in the active search interval is below the threshold μ, the algorithm starts to compute the distance of the points in the active search region to the query object. We call this step filtering. We now consider the shape of the active search region during filtering.

Lemma A.0.2. If the algorithm is filtering, $\forall d \in \{1..D\}$ it holds that:

$$\frac{2^{\text{nbit}}}{\sqrt[D]{\frac{n}{\mu}}} \leq R^d_{\text{upper},j} - R^d_{\text{lower},j}. \tag{A.6}$$

Proof of Lemma A.0.2

In the following, we investigate the bit pattern of R^d_{lower}. Intuitively, our argument can be understood as follows: the Morton encoding of z_{lower} was created by interleaving the bits of R_{lower} of each dimension d. Thus, the most significant D bits of z_{lower} contain the most significant bits of R^d_{lower} of each dimension d and so on. Conversely, if z_{lower} is split into bit blocks of size D, then R_{lower} contains one bit from the first block, one bit from the second block and so on (Figure A.2). R_{upper} follows the same considerations. As the active search region is always split at the most significant bit, splitting j times means that R^d is split into $\frac{j}{D}$ sized parts in each dimension. Thus,

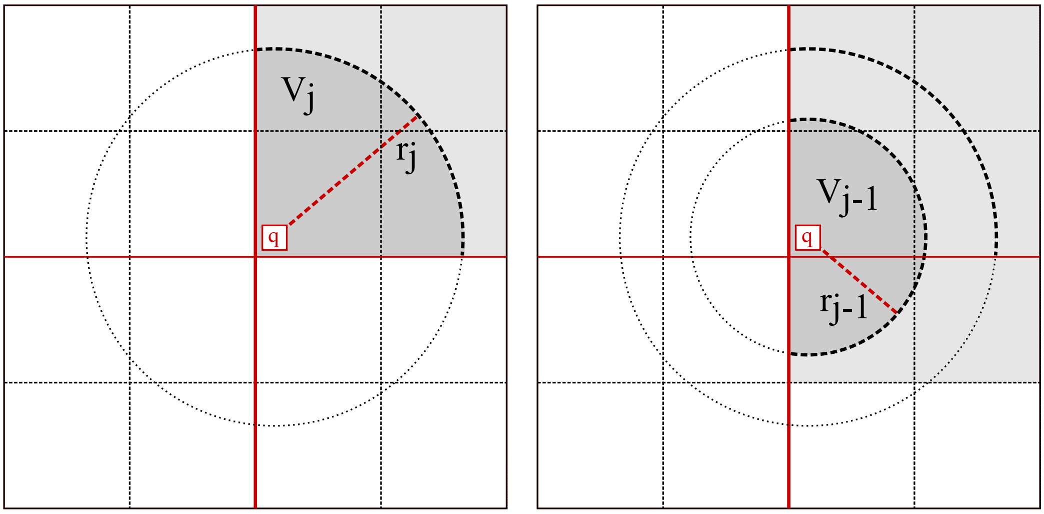
$$\begin{aligned}
\left|R^d_{\text{upper},j} - R^d_{\text{lower},j}\right| &= 2^{\text{nbit}-(\frac{j}{D})} \\
&\overset{A.5}{>} 2^{\text{nbit}-\frac{\log_2(\frac{n}{\mu})}{D}} \\
&= \frac{2^{\text{nbit}}}{2^{\frac{\log_2(\frac{n}{\mu})}{D}}} \\
&= \frac{2^{\text{nbit}}}{2^{\log_2(\frac{n}{\mu})\cdot\frac{1}{D}}} \\
&= \frac{2^{\text{nbit}}}{(\frac{n}{\mu})^{\frac{1}{D}}} \\
&= \frac{2^{\text{nbit}}}{\sqrt[D]{\frac{n}{\mu}}} \\
&=: l
\end{aligned} \tag{A.7}$$

□

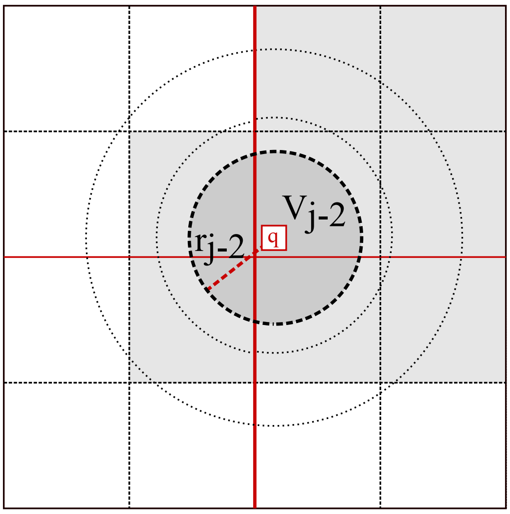
$${}_{\text{binary}}z_{\text{lower}} =_{\text{binary}} \overbrace{R^1_{\text{lower}}[1]\ R^2_{\text{lower}}[1]\ R^3_{\text{lower}}[1]\ ..}^{\text{first } D \text{ bits}}\ \overbrace{R^1_{\text{lower}}[2]\ R^2_{\text{lower}}[2]\ R^3_{\text{lower}}[2]\ ..}^{\text{next } D \text{ bits}}\ \ ..$$

Fig. A.2.: Bit pattern of z_{lower} is formed by interleaving the bits of R_{lower} for each dimension.

Combined with the observation that during the subdivision, the active search region is always split exactly in half, Lemma A.0.2 can be interpreted in such a way that if the algorithm traverses, the active search area takes the form of hypercubes that lie on a regular grid of edge length l. The active search region always covers one or more of these grid elements, so the number of traversal steps is smaller or equal to the number of visited grid elements. We now ask how many of the grid elements are visited.



(a) Traversal depth j **(b)** Traversal depth $j-1$



(c) Traversal depth $j-2$

Fig. A.3.: Example of the recursive subdivision scheme for $D=2$. For each traversal depth, the algorithm finds the k nearest neighbours, which corresponds to traversing a volume of V_k in each step, which in turn corresponds to hypersphere segments of successively smaller radii.

The algorithm visits the grid cells in a specific order defined by the subdivision scheme (Figure A.3). At each subdivision step, the algorithm first traverses the sub-space that is closer to the query object before visiting the sub-space on the far side of the clipping plane. We now describe the shape of the subspace visited by the algorithm at a subdivision depth j. By definition, the algorithm traverses the near side of the splitting plane until the k nearest neighbours *on this side of the plane* are found. As the data points are distributed uniformly in the search space ((Assumption (a) on page 135), the k nearest neighbours of the query object are contained in a subset of the search space that corresponds to a D-ball of volume

$$V_k = \frac{k}{\rho} = \frac{k}{n} \cdot 2^{\mathsf{nbit} \cdot D}. \tag{A.8}$$

However, at traversal depth j, the algorithm can only visit the part of the sub-space on the near side of the splitting plane. In the worst case, q lies exactly in the intersection of all spitting planes from $1..j$. In this case, the sub-space is split in half by each of the splitting planes. Consequently, only the $\frac{1}{2^j}$-th part of the volume can be visited (Figure A.3). The k closest points in this sub-space are contained within a ball of volume

$$V_j = 2^j \cdot V_k \overset{A.8}{=} 2^j \cdot \frac{k}{n} \cdot 2^{\mathsf{nbit} D}. \tag{A.9}$$

For the L_2 norm, such a ball is a D-sphere. We now need to relate the radius of the ball to the number of visited grid cells. To do so, we use a result from grid theory. For every ball of radius r in a D-dimensional Euclidean space, it holds that the number of grid points at integer positions $A_D(r)$ inside the ball equals the volume of the corresponding sphere plus an error in the order of the boundary [94, p. 5].

$$A_D(r) = V_D(r) + O\left(r^{\frac{(D-1)}{2}}\right) \tag{A.10}$$

In order to generalise the formula to other cost functions, we use that our cost function is a norm (Assumption (b) on page 135), and in a finite dimensional real space, every norm is equivalent to the L_2 norm. Thus, $\exists c, d$ such that $\forall x : c \cdot |x|_f \leq |x|_2 \leq d \cdot |x|_f$. We use the left half of the inequality ($\exists c$) to state that for a D-ball under arbitrary norm f, there exists a D-sphere under L_2 norm that fully contains the ball. Thus, $\exists c : \mid V^D_{|f|}(r) \leq V^D_{L_2}(c \cdot r)$.

$$\begin{aligned}
A_D(r) &= V^D_{|f|}(r) + O\left(r^{\frac{(D-1)}{2}}\right) \\
&\leq V^D_{|L_2|}(c \cdot r) + O\left((c \cdot r)^{\frac{(D-1)}{2}}\right) \\
&= c^D \cdot V^D_{|L_2|}(r) + O\left((c \cdot r)^{\frac{(D-1)}{2}}\right) \\
&= O(r^D) + O\left(c^{\frac{(D-1)}{2}} \cdot r^{\frac{(D-1)}{2}}\right) \\
&= O(r^D).
\end{aligned} \tag{A.11}$$

In other words, the number of grid points scales with the volume of the ball and the error can be neglected.

We now estimate how many grid points are visited per subdivision step. As Equation (A.11) assumes that the grid points are at integer coordinates, we need to scale the sphere by the edge length of the grid cells. Using Equations (A.7) and (A.9), we scale the volume by $\frac{1}{l}$ in any dimension and obtain the number of grid cells in the hypersphere.

$$\begin{aligned} A_{j,hypersphere} &= V_j \cdot l^{-D} \\ &\overset{A.7}{=} V_j \cdot \left(\frac{2^{\textsf{nbit}}}{\sqrt[D]{\frac{n}{\mu}}} \right)^{-D} \\ &= V_j \cdot \left(2^{\textsf{nbit}} \cdot (\frac{n}{\mu})^{-\frac{1}{D}} \right)^{-D} \\ &= V_j \cdot 2^{\textsf{nbit} \cdot (-D)} \cdot \frac{n}{\mu} \\ &\overset{A.9}{=} 2^j \cdot \frac{k}{n} \cdot 2^{\textsf{nbit} \cdot D} \cdot 2^{\textsf{nbit} \cdot (-D)} \cdot \frac{n}{\mu} \\ &= 2^j \cdot \frac{k}{\mu}. \end{aligned} \tag{A.12}$$

In our considered worst-case scenario, the query object lies exactly on all splitting planes from $1..j$, so half of the grid cells are on either side of the splitting plane. Thus, the algorithm will visit $2^{-j} \cdot A_j$ many grid cells, and we finally obtain that

$$\begin{aligned} A_{j,\textsf{visited}} &= 2^{-j} \cdot A_{j,\textsf{hypersphere}} \\ &\overset{A.12}{=} \frac{k}{\mu} = O(1). \end{aligned} \tag{A.13}$$

We have shown in the first section that the maximum recursion depth is in $O(\log(n))$, and the cost for the binary search is $O(\log\, n)$ per depth step, which gives a cost of $O(\log(n)^2)$ for the recursive subdivision. The total number of visited grid cells is in $O(\log\, n)$, and each cell takes constant time to traverse, as it contains a constant number of data points per definition. Thus, the total cost for the query operation is dominated by the recursive subdivision which is $O(\log(n)^2)$. ☐

Beyond the scope of this thesis ...

B

... further research was conducted. In the following, we give short summaries of our contributions.

We evaluated the use of spherically symmetric volume elements based on generalised Kaiser-Bessel window functions as basis function for discretisation instead of voxels in computed laminography [67, 247]. Band-limiting properties of blob functions were beneficial, particularly in the case of noisy projections and if only a limited number of projections was available, so that we were able to enhance resolution in computed laminography.

A comparative study of three widely used algorithms for the detection of fiducial markers in electron microscopy (EM) images was conducted [240, 241]. We introduced figures of merit and implemented all three algorithms in a unified code base to exclude software-specific differences in the evaluation. In conclusion, the marker detection algorithm has to be chosen carefully, because it highly depends on the properties of the dataset to be analysed.

We presented GeneTrail2, a web service allowing the integrated analysis of transcriptomic, miRNomic, genomic, and proteomic datasets [231, 245]. It offers multiple statistical tests, a large number of predefined reference sets, as well as a comprehensive collection of biological categories and enables direct comparisons between computed results. We used GeneTrail2 to explore pathogenic mechanisms of Wilms tumours. We not only succeeded in revealing signalling cascades that may contribute to the malignancy of blastemal subtype tumours, but also identified potential biomarkers for nephroblastoma with adverse prognosis. The presented use-case demonstrated that GeneTrail2 is well equipped for the integrative analysis of comprehensive -omics data and may help to shed light on complex pathogenic mechanisms in cancer and other diseases.

A tool for reasoning on causal Bayesian Network was implemented [232]. Causality allows to compute the effect of interventions, which are external changes to the system, caused by e.g. gene knockouts or an administered drug. Our tool CausalTrail includes multiple data import methods, a flexible query language for formulating hypotheses, as well as an intuitive graphical user interface, which was demonstrated based on a protein signaling network.

We performed hierarchical clustering of genetic markers using self organising maps [161]. Subsequent correlation analyses helped to identify potential drug targets for Wilms tumour patients. The study could be a foundation to conduct clinical trials for assessing the influence of preoperative chemotherapy for Wilms tumour treatments.

The spatial distribution of the HER2 receptor in the plasma membrane of SKBR3 and HCC1954 breast cancer cells was studied in [184]. We implemented an algorithm based on graph theory and performed a correlation analysis to show that HER2 receptors arrange in linear chains. A statistical analysis was performed to validate that the experimental finding were significant. Based on our findings we proposed that a mechanism between HER2 and actin filaments leads to linearly aligned oligomers. This points towards a fundamental principle behind the spatial-functional organisation of growth factor receptors in membrane ruffles in the plasma membrane via assembly into larger oligomeric structures.

C

Bibliography

[1] Michal Aharon and Michael Elad. „Sparse and redundant modeling of image content using an image-signature-dictionary“. In: *SIAM Journal on Imaging Sciences*, Vol. 1, No. 3 (2008), pp. 228–247 (cit. on p. 16).

[2] Michal Aharon, Michael Elad, and Alfred Bruckstein. „K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation“. In: *IEEE Transactions on Signal Processing*, Vol. 54, No. 11 (2006), pp. 4311–4322 (cit. on pp. 16, 31).

[3] Michal Aharon, Michael Elad, and Alfred M Bruckstein. „On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them“. In: *Linear algebra and its applications*, Vol. 416, No. 1 (2006), pp. 48–67 (cit. on p. 16).

[4] Yusra AY Al-Najjar, D Chen Soong, et al. „Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI“. In: *International Journal of Scientific & Engineering Research*, Vol. 3, No. 8 (2012), pp. 1–5 (cit. on p. 104).

[5] V. K. Alilou and F. Yaghmaee. „Introducing a new fast exemplar-based inpainting algorithm“. In: *2014 22nd Iranian Conference on Electrical Engineering (ICEE)*. 2014, pp. 874–878 (cit. on p. 25).

[6] John S Allen, Hanna Damasio, and Thomas J Grabowski. „Normal neuroanatomical variation in the human brain: An MRI-volumetric study“. In: *American Journal of Physical Anthropology: The Official Publication of the American Association of Physical Anthropologists*, Vol. 118, No. 4 (2002), pp. 341–358 (cit. on p. 2).

[7] Hyrum S. Anderson, Jovana Ilic-Helms, Brandon Rohrer, Jason Wheeler, and Kurt Larson. „Sparse imaging for fast electron microscopy“. In: (2013), pp. 86570C–86570C–12 (cit. on p. 77).

[8] James R Anderson, Bryan W Jones, Carl B Watt, Margaret V Shaw, Jia-Hui Yang, David DeMill, James S Lauritzen, et al. „Exploring the retinal connectome“. In: *Molecular vision*, Vol. 17 (2011), pp. 355–79 (cit. on p. 1).

[9] S. Andris, P. Peter, and J. Weickert. „A proof-of-concept framework for PDE-based video compression“. In: *2016 Picture Coding Symposium (PCS)*. 2016, pp. 1–5 (cit. on p. 23).

[10] Ignacio Arganda-Carreras, Srinivas C Turaga, Daniel R Berger, Dan Cireșan, Alessandro Giusti, Luca M Gambardella, Jürgen Schmidhuber, et al. „Crowdsourcing the creation of image segmentation algorithms for connectomics“. In: *Frontiers in neuroanatomy*, Vol. 9 (2015), pp. 1–13 (cit. on p. 105).

[11] Pablo Arias, Vicent Caselles, and Gabriele Facciolo. „Analysis of a variational framework for exemplar-based image inpainting“. In: *Multiscale Modeling & Simulation*, Vol. 10, No. 2 (2012), pp. 473–514 (cit. on p. 25).

[12] Pablo Arias, Gabriele Facciolo, Vicent Caselles, and Guillermo Sapiro. „A Variational Framework for Exemplar-Based Image Inpainting“. In: *International Journal of Computer Vision*, Vol. 93 (2011), pp. 319–347 (cit. on pp. 21, 24).

[13] Ilke Arslan, Jenna R Tong, and Paul A Midgley. „Reducing the missing wedge: High-resolution dual axis tomography of inorganic materials“. In: *Ultramicroscopy*, Vol. 106, No. 11 (2006), pp. 994–1000 (cit. on p. 33).

[14] Jean-François Aujol, Saïd Ladjal, and Simon Masnou. „Exemplar-based inpainting from a variational point of view“. In: *SIAM Journal on Mathematical Analysis*, Vol. 42, No. 3 (2010), pp. 1246–1285 (cit. on p. 24).

[15] Norio Baba and Eisaku Katayama. „A novel “ghost”-free tomographic image reconstruction method applicable to rotary-shadowed replica specimens“. In: *Ultramicroscopy*, Vol. 108, No. 3 (2008), pp. 239–255 (cit. on p. 33).

[16] Coloma Ballester, M. Bertalmio, V. Caselles, Guillermo Sapiro, and Joan Verdera. „Filling-in by Joint Interpolation of Vector Fields and Gray Levels“. In: *IEEE Transactions on Image Processing*, Vol. 10, No. 8 (2001), pp. 1200–1211 (cit. on p. 23).

[17] Richard. G. Baraniuk. „Compressive Sensing [Lecture Notes]“. In: *IEEE Signal Processing Magazine*, Vol. 24, No. 4 (2007), pp. 118–121 (cit. on p. 13).

[18] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. „PatchMatch: a randomized correspondence algorithm for structural image editing“. In: *ACM Transactions on Graphics (TOG)*, Vol. 28, No. 3 (2009), pp. 24–33 (cit. on pp. 25, 26).

[19] Connelly Barnes, Eli Shechtman, Dan B Goldman, and Adam Finkelstein. „The generalized patchmatch correspondence algorithm“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 6313. Springer. 2010, pp. 29–43 (cit. on p. 25).

[20] Kees Joost Batenburg and Jan Sijbers. „DART: a fast heuristic algebraic reconstruction algorithm for discrete tomography“. In: *IEEE International Conference on Image Processing (ICIP)*, Vol. 4 (2007), pp. IV–133–IV–136 (cit. on p. 33).

[21] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. „The R*-tree: an efficient and robust access method for points and rectangles“. In: *ACM Sigmod Record*. Vol. 19, No. 2. ACM. 1990, pp. 322–331 (cit. on p. 43).

[22] S Berchtold, DA Keim, and HP Kriegel. *The X-tree: An index structure for high-dimensional data*. 2001, pp. 451–462 (cit. on p. 43).

[23] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. „Image inpainting“. In: *Proceedings of the 27th annual conference on Computer Graphics and Interactive Techniques* (2000), pp. 417–424 (cit. on pp. 21, 24, 25).

[24] Sooraj Bhat. *Exemplar-based Inpainting according to Criminisi*. [Accessed: 06 December 2018]. 2015. URL: `https://github.com/1upon0/sura-2015/tree/master/AppGen/AppGen/inpainting%202` (cit. on p. 53).

[25] Sourabh Bhattacharya, Garrett Warnell, Rama Chellappa, and Tamer Basar. „Vision-guided feedback control of a mobile robot with compressive measurements and side information“. In: *American Control Conference (ACC)*. IEEE. 2014, pp. 97–102 (cit. on p. 28).

[26] Sujit Bhattacharya, Thomas Blumensath, Bernard Mulgrew, and Mike Davies. „Fast encoding of synthetic aperture radar raw data using compressed sensing“. In: *14th Workshop on Statistical Signal Processing (SSP)*. IEEE. 2007, pp. 448–452 (cit. on p. 29).

[27] Peter Binev, Wolfgang Dahmen, Ronald DeVore, Philipp Lamby, Daniel Savu, and Robert Sharpley. „Compressed sensing and electron microscopy“. In: *Modeling Nanoscale Imaging in Electron Microscopy*. Springer, 2012, pp. 73–126 (cit. on pp. 30, 77).

[28] Yochai Blau and Tomer Michaeli. „The perception-distortion tradeoff“. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 6228–6237 (cit. on p. 17).

[30] Karlheinz Brandenburg. „MP3 and AAC explained“. In: *Audio Engineering Society Conference: 17th International Conference: High-Quality Audio Coding*. Audio Engineering Society. 1999, pp. 1–12 (cit. on p. 13).

[31] Pierre Buyssens, Maxime Daisy, David Tschumperlé, and Olivier Lézoray. „Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions“. In: *IEEE Transactions on Image Processing*, Vol. 24, No. 6 (2015), pp. 1809–1824 (cit. on pp. 24, 26).

[32] Emmanuel J Candès. „Compressive sampling“. In: *Proceedings of the international congress of mathematicians*. Vol. 3. Madrid, Spain. 2006, pp. 1433–1452 (cit. on pp. 12, 13).

[33] Emmanuel J Candes. „The restricted isometry property and its implications for compressed sensing“. In: *Comptes rendus mathematique*, Vol. 346, No. 9-10 (2008), pp. 589–592 (cit. on p. 14).

[34] Emmanuel J Candes and David L Donoho. *Curvelets: A surprisingly effective nonadaptive representation for objects with edges*. Tech. rep. Stanford Univ Ca Dept of Statistics, 2000, pp. 1–16 (cit. on pp. 15, 76).

[35] Emmanuel J Candes, Yonina C Eldar, Deanna Needell, and Paige Randall. „Compressed sensing with coherent and redundant dictionaries“. In: *Applied and Computational Harmonic Analysis*, Vol. 31, No. 1 (2011), pp. 59–73 (cit. on p. 14).

[36] Emmanuel J Candes and Justin Romberg. „Quantitative robust uncertainty principles and optimally sparse decompositions“. In: *Foundations of Computational Mathematics*, Vol. 6, No. 2 (2006), pp. 227–254 (cit. on p. 13).

[37] Emmanuel J Candès, Justin Romberg, and Terence Tao. „Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information“. In: *IEEE Transactions on Information Theory*, Vol. 52, No. 2 (2006), pp. 489–509 (cit. on p. 13).

[38] Emmanuel J Candes, Justin K Romberg, and Terence Tao. „Stable signal recovery from incomplete and inaccurate measurements“. In: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, Vol. 59, No. 8 (2006), pp. 1207–1223 (cit. on p. 13).

[39] Emmanuel J Candes and Terence Tao. „Decoding by linear programming“. In: *IEEE Transactions on Information Theory*, Vol. 51, No. 12 (2005), pp. 4203–4215 (cit. on p. 13).

[40] Emmanuel J Candes and Terence Tao. „Near-optimal signal recovery from random projections: Universal encoding strategies?“ In: *IEEE Transactions on Information Theory*, Vol. 52, No. 12 (2006), pp. 5406–5425 (cit. on p. 13).

[41] John Canny. „A computational approach to edge detection“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, No. 6 (1986), pp. 679–698 (cit. on p. 112).

[42] Marco Cantoni and Lorenz Holzer. „Advances in 3D focused ion beam tomography“. In: *MRS Bulletin*, Vol. 39, No. 4 (2014), pp. 354–360 (cit. on p. 2).

[43] Frédéric Cao, Yann Gousseau, Simon Masnou, and Patrick Pérez. „Geometrically guided exemplar-based inpainting“. In: *SIAM Journal on Imaging Sciences*, Vol. 4, No. 4 (2011), pp. 1143–1179 (cit. on p. 24).

[44] David B Carlson and James E Evans. „Low-dose imaging techniques for transmission electron microscopy“. In: *The transmission electron microscope*. Intech, 2012, pp. 85–98 (cit. on p. 75).

[45] T Chan. „Local inpainting models and TV inpainting“. In: *SIAM J. Appl. Math.*, Vol. 62, No. 3 (2001), pp. 1019–1043 (cit. on p. 21).

[46] Tony F Chan and Jianhong Shen. „Non-texture inpainting by curvature-driven diffusions (CDD)“. In: *SIAM J. Appl. Math.*, Vol. 12, No. 4 (2001), pp. 436–449 (cit. on p. 21).

[47] Tony F Chan and Jianhong Shen. „Variational image inpainting“. In: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, Vol. 58, No. 5 (2005), pp. 579–619 (cit. on p. 21).

[48] Wai Lam Chan, Kriti Charan, Dharmpal Takhar, Kevin F Kelly, Richard G Baraniuk, and Daniel M Mittleman. „A single-pixel terahertz imaging system based on compressed sensing“. In: *Applied Physics Letters*, Vol. 93, No. 12 (2008), pp. 121105.1–121105.3 (cit. on p. 28).

[49] I-Cheng Chang, J Cloud Yu, and Chih-Chuan Chang. „A forgery detection algorithm for exemplar-based inpainting images using multi-region relation“. In: *Image and Vision Computing*, Vol. 31, No. 1 (2013), pp. 57–71 (cit. on p. 24).

[50] Delei Chen, Bart Goris, Folkert Bleichrodt, Hamed Heidari Mezerji, Sara Bals, Kees Joost Batenburg, Gijsbertus de With, and Heiner Friedrich. „The properties of SIRT, TVM, and DART for 3D imaging of tubular domains in nanocomposite thin-films and sections“. In: *Ultramicroscopy*, Vol. 147 (2014), pp. 137–148 (cit. on p. 2).

[51] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. „Atomic decomposition by basis pursuit“. In: *SIAM review*, Vol. 43, No. 1 (2001), pp. 129–159 (cit. on pp. 30, 91).

[52] Francois Chollet. *Deep learning with python*. Manning Publications Co., 2017 (cit. on p. 32).

[53] Ronald R Coifman, Yves Meyer, Steven Quake, and M Victor Wickerhauser. „Signal processing and compression with wavelet packets“. In: *Wavelets and their Applications*. Springer, 1994, pp. 363–379 (cit. on pp. 15, 76).

[54] Nvidia Corporation. *cuBLAS - Dense Linear Algebra on GPUs*. [Accessed: 06 December 2018]. URL: `https://developer.nvidia.com/cublas` (cit. on p. 101).

[55] Nvidia Corporation. *CUDA - Accelerated Computing on GPUs*. [Accessed: 06 December 2018]. URL: `https://developer.nvidia.com/cuda-zone` (cit. on p. 101).

[56] Nvidia Corporation. *cuDNN - GPU accelerated Deep Learning*. [Accessed: 06 December 2018]. URL: `https://developer.nvidia.com/cudnn` (cit. on p. 101).

[57] Antonio Criminisi, Patrick Perez, and Kentaro Toyama. „Object removal by exemplar-based inpainting“. In: *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2003, pp. II.1–II.8 (cit. on p. 24).

[58] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. „Region filling and object removal by exemplar-based image inpainting“. In: *IEEE Transactions on Image Processing*, Vol. 13, No. 9 (2004), pp. 1200–1212 (cit. on pp. 4, 7, 24, 35, 58, 77, 130).

[59] Antonio Criminisi, Patrick Perez, Kentaro Toyama, Michel Gangnet, and Andrew Blake. *Image region filling by exemplar-based inpainting*. US Patent 6,987,520. 2006 (cit. on p. 24).

[60] Tim Dahmen, Jean-Pierre Baudoin, Andrew R Lupini, Christian Kübel, Philipp Slusallek, and Niels De Jonge. „Combined scanning transmission electron microscopy tilt-and focal series“. In: *Microscopy and Microanalysis*, Vol. 20, No. 2 (2014), pp. 548–560 (cit. on pp. 19, 33).

[69] Maxime Daisy, Pierre Buyssens, David Tschumperlé, and Olivier Lézoray. „A smarter exemplar-based inpainting algorithm using local and global heuristics for more geometric coherence“. In: *IEEE International Conference on Image Processing (ICIP)*. IEEE. 2014, pp. 4622–4626 (cit. on p. 24).

[70] Mark A Davenport, Chinmay Hegde, Marco F Duarte, and Richard G Baraniuk. „Joint manifolds for data fusion“. In: *IEEE Transactions on Image Processing*, Vol. 19, No. 10 (2010), pp. 2580–2594 (cit. on p. 29).

[71] Michiel De Goede, Eric Johlin, Beniamino Sciacca, Faysal Boughorbel, and Erik C Garnett. „3D multi-energy deconvolution electron microscopy“. In: *Nanoscale*, Vol. 9, No. 2 (2017), pp. 684–689 (cit. on p. 89).

[72] T.J. Deerinck, Eric Bushong, A Thor, and Mark Ellisman. „NCMIR methods for 3D EM: A new protocol for preparation of biological specimens for serial block face scanning electron microscopy“. In: *Nat Center Microsc Imag Res* (2010), pp. 6–8 (cit. on p. 104).

[73] Boris Delaunay et al. „Sur la sphere vide“. In: *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, Vol. 7, No. 793-800 (1934), pp. 1–2 (cit. on p. 112).

[74] Minh N Do and Martin Vetterli. „Contourlets: a directional multiresolution image representation“. In: *International Conference on Image Processing (ICIP)*. Vol. 1. IEEE. 2002, pp. I.497–I.501 (cit. on p. 15).

[75] David L Donoho. „Compressed sensing“. In: *IEEE Transactions on Information Theory*, Vol. 52, No. 4 (2006), pp. 1289–1306 (cit. on pp. 12, 13).

[76] David L Donoho, Michael Elad, and Vladimir N Temlyakov. „Stable recovery of sparse overcomplete representations in the presence of noise“. In: *IEEE Transactions on Information Theory*, Vol. 52, No. 1 (2006), pp. 6–18 (cit. on p. 14).

[77] Richard Dosselmann and Xue Dong Yang. „A comprehensive assessment of the structural similarity index“. In: *Signal, Image and Video Processing*, Vol. 5, No. 1 (2011), pp. 81–91 (cit. on p. 18).

[78] Randal Douc and Olivier Cappé. „Comparison of resampling schemes for particle filtering“. In: *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*. IEEE. 2005, pp. 64–69 (cit. on p. 112).

[79] Marco F Duarte, Mark A Davenport, Dharmpal Takhar, Jason N Laska, Ting Sun, Kevin F Kelly, and Richard G Baraniuk. „Single-pixel imaging via compressive sampling“. In: *IEEE Signal Processing Magazine*, Vol. 25, No. 2 (2008), pp. 83–91 (cit. on p. 28).

[80] Michael F. Duarte, Shriram Sarvotham, Dror Baron, Michael B. Wakin, and Richard G. Baraniuk. „Distributed compressed sensing of jointly sparse signals“. In: *Asilomar Conference on Signals, Systems, and Computers*. Vol. 1. 2005, pp. 1537–1541 (cit. on p. 28).

[81] Jaroslav Dusek and Karel Roubík. „Testing of new models of the human visual system for image quality evaluation“. In: *Seventh International Symposium on Signal Processing and Its Applications*. Vol. 2. IEEE. 2003, pp. 621–622 (cit. on p. 17).

[82] Tsvi G Dvorkind, Yonina C Eldar, and Ewa Matusiak. „Nonlinear and nonideal sampling: theory and methods“. In: *IEEE Transactions on Signal Processing*, Vol. 56, No. 12 (2008), pp. 5874–5890 (cit. on p. 13).

[83] Alexei A Efros and William T Freeman. „Image quilting for texture synthesis and transfer“. In: *Proceedings of the 28th annual Conference on Computer Graphics and Interactive Techniques*. ACM. 2001, pp. 341–346 (cit. on p. 23).

[84] Alexei A Efros and Thomas K Leung. „Texture synthesis by non-parametric sampling“. In: *The Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV)*. Vol. 2. IEEE. 1999, pp. 1033–1038 (cit. on pp. 23, 24).

[85] Karen Egiazarian, Jaakko Astola, Nikolay Ponomarenko, Vladimir Lukin, Federica Battisti, and Marco Carli. „New full-reference quality metrics based on HVS“. In: *Proceedings of the Second International Workshop on Video Processing and Quality Metrics*. Vol. 4. 2006, pp. I.1–I.4 (cit. on p. 19).

[86] Michael Elad and Michal Aharon. „Image denoising via sparse and redundant representations over learned dictionaries“. In: *IEEE Transactions on Image processing*, Vol. 15, No. 12 (2006), pp. 3736–3745 (cit. on p. 21).

[87] Kjersti Engan, Sven Ole Aase, and J Hakon Husoy. „Method of optimal directions for frame design“. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Vol. 5. IEEE. 1999, pp. 2443–2446 (cit. on p. 16).

[88] Kjersti Engan, Sven Ole Aase, and John Håkon Husøy. „Multi-frame compression: Theory and design“. In: *Signal Processing*, Vol. 80, No. 10 (2000), pp. 2121–2140 (cit. on p. 16).

[89] Gabriele Facciolo, Pablo Arias, Vicent Caselles, and Guillermo Sapiro. „Exemplar-based interpolation of sparsely sampled images“. In: *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer. 2009, pp. 331–344 (cit. on p. 24).

[90] Qian Fan and Lifeng Zhang. „A novel patch matching algorithm for exemplar-based image inpainting“. In: *Multimedia Tools and Applications*, Vol. 77 (9 2018), pp. 1–15 (cit. on p. 24).

[91] Sina Farsiu, M Dirk Robinson, Michael Elad, and Peyman Milanfar. „Fast and robust multiframe super resolution“. In: *IEEE Transactions on Image Processing*, Vol. 13, No. 10 (2004), pp. 1327–1344 (cit. on p. 76).

[92] Jose-Jesus Fernandez. „Computational methods for electron tomography“. In: *Micron*, Vol. 43, No. 10 (2012), pp. 1010–1030 (cit. on p. 2).

[93] Joachim Frank. *Three-dimensional electron microscopy of macromolecular assemblies: visualization of biological molecules in their native state*. Oxford University Press, 2006 (cit. on p. 2).

[94] F. Fricker. *Einführung in die Gitterpunktlehre*. Birkhäuser Verlag, 1921 (cit. on p. 142).

[95] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. Springer Series in Statistics New York, 2001 (cit. on p. 39).

[96] Irena Galić, Joachim Weickert, Martin Welk, Andrés Bruhn, Alexander Belyaev, and Hans-Peter Seidel. „Image compression with anisotropic diffusion“. In: *Journal of Mathematical Imaging and Vision*, Vol. 31, No. 2-3 (2008), pp. 255–269 (cit. on p. 23).

[97] Rajat P Garg, Ilya A Sharapov, and Illya Sharapov. *Techniques for optimizing applications: high performance computing*. Sun Microsystems Press Palo Alto, 2002 (cit. on p. 52).

[98] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. „Image style transfer using convolutional neural networks“. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2414–2423 (cit. on p. 23).

[99] Dongdong Ge, Xiaoye Jiang, and Yinyu Ye. „A note on the complexity of Lp minimization“. In: *Mathematical Programming*, Vol. 129, No. 2 (2011), pp. 285–299 (cit. on p. 16).

[100] Mengshu Ge and Huolin L Xin. „Deep Learning Based Atom Segmentation and Noise and Missing-Wedge Reduction for Electron Tomography“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 504–505 (cit. on p. 32).

[101] Kfir Gedalyahu and Yonina C Eldar. „Time delay estimation from low rate samples: A union of subspaces approach“. In: *arXiv preprint arXiv:0905.2429* (2009) (cit. on p. 28).

[102] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. " O'Reilly Media, Inc.", 2017 (cit. on p. 32).

[103] Matthew L Gong, Su Jong Yoon, Raymond R Unocic, Hope Ishii, John P Bradley, Brandon D Miller, Daniel Masiel, et al. „Pioneering the Use of Neural Network Architectures and Feature Engineering for Real-Time Augmented Microscopy and Analysis“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 514–515 (cit. on p. 32).

[104] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*. Vol. 1. MIT press Cambridge, 2016 (cit. on p. 32).

[105] Bart Goris, Wouter Van den Broek, K Joost Batenburg, H Heidari Mezerji, and Sara Bals. „Electron tomography based on a total variation minimization reconstruction technique“. In: *Ultramicroscopy*, Vol. 113 (2012), pp. 120–130 (cit. on p. 34).

[106] Khronos Group. *Open CL*. [Accessed: 06 December 2018]. URL: `https://www.khronos.org/opencl/` (cit. on p. 53).

[107] Matthew Guay, Zeyad Emam, and Richard Leapman. „Problems and Progress in Automating Electron Microscopy Segmentation“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 508–509 (cit. on p. 32).

[108] Matthew D Guay, Wojciech Czaja, Maria A Aronova, and Richard D Leapman. „Compressed sensing electron tomography for determining biological structure“. In: *Scientific Reports*, Vol. 6 (2016), pp. 27614.1–27614.14 (cit. on p. 30).

[109] Onur G Guleryuz. „Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising-part I: theory“. In: *IEEE Transactions on Image Processing*, Vol. 15, No. 3 (2006), pp. 539–554 (cit. on p. 21).

[110] Onur G Guleryuz. „Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising-part II: adaptive algorithms“. In: *IEEE Transactions on Image Processing*, Vol. 15, No. 3 (2006), pp. 555–571 (cit. on p. 21).

[111] Doga Gursoy. „High-speed Fly-Scan Volumetric Imaging“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 498–499 (cit. on p. 29).

[112] Antonin Guttman. „R-trees: A dynamic index structure for spatial searching“. In: *Proceedings of the international Conference on Management of Data*, Vol. 14, No. 2 (1984), pp. 47–57 (cit. on p. 43).

[113] Jarvis Haupt, Waheed U Bajwa, Michael Rabbat, and Robert Nowak. „Compressed sensing for networked data“. In: *IEEE Signal Processing Magazine*, Vol. 25, No. 2 (2008), pp. 92–101 (cit. on p. 28).

[114] Simon Hawe, Martin Kleinsteuber, and Klaus Diepold. „Analysis operator learning and its application to image reconstruction“. In: *IEEE Transactions on Image Processing*, Vol. 22, No. 6 (2013), pp. 2138–2150 (cit. on pp. 30, 76, 77, 90).

[115] Kaiming He and Jian Sun. „Statistics of patch offsets for image completion“. In: *Computer Vision – ECCV 2012*. Vol. 7573. Springer, 2012, pp. 16–29 (cit. on p. 25).

[116] Felix Heide, Wolfgang Heidrich, and Gordon Wetzstein. „Fast and flexible convolutional sparse coding“. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 5135–5143 (cit. on pp. 30, 91).

[117] Stefan W Hell. „Far-field optical nanoscopy“. In: *Science*, Vol. 316, No. 5828 (2007), pp. 1153–1158 (cit. on p. 2).

[118] Maurice Herlihy and Nir Shavit. *The art of multiprocessor programming*. Morgan Kaufmann, 2011 (cit. on p. 53).

[119] David Hilbert. „Ueber die stetige Abbildung einer Line auf ein Flächenstück“. In: *Mathematische Annalen*, Vol. 38, No. 3 (1891), pp. 459–460 (cit. on pp. 45, 47).

[120] Laurent Hoeltgen, Markus Mainberger, Sebastian Hoffmann, Joachim Weickert, Ching Hoo Tang, Simon Setzer, Daniel Johannsen, et al. „Optimising spatial and tonal data for PDE-based inpainting“. In: *Variational Methods by M. Bergounioux, G. Peyré, C. Schnörr, J.-B. Caillau, and T. Haberkorn, eds*, No. 18 (2017), pp. 35–83 (cit. on pp. 23, 26).

[121] Sebastian Hoffmann, Markus Mainberger, Joachim Weickert, and Michael Puhl. „Compression of depth maps with segment-based homogeneous diffusion“. In: *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*. Vol. LNCS 7893. Springer. 2013, pp. 319–330 (cit. on p. 27).

[122] Gang Huang, Hong Jiang, Kim Matthews, and Paul Wilford. „Lensless imaging by compressive sensing“. In: *International Conference on Image Processing (ICIP)*. IEEE. 2013, pp. 2101–2105 (cit. on p. 31).

[123] Quan Huynh-Thu and Mohammed Ghanbari. „Scope of validity of PSNR in image/video quality assessment“. In: *Electronics Letters*, Vol. 44, No. 13 (2008), pp. 800–801 (cit. on p. 17).

[124] Francisco J Ibarrola and Ruben D Spies. „A two-step mixed inpainting method with curvature-based anisotropy and spatial adaptivity.“ In: *Inverse Problems & Imaging*, Vol. 11, No. 2 (2017), pp. 247–262 (cit. on p. 25).

[125] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. „Globally and locally consistent image completion“. In: *ACM Transactions on Graphics (TOG)*, Vol. 36, No. 4 (2017), pp. 107.1–107.14 (cit. on pp. 23, 32).

[126] Martin Jacob, Toby Sanders, Nicolas Bernier, Adeline Grenier, Rafael Bortolin Pinheiro, Frederic Mazen, Pascale Bayle-Guillemaud, and Zineb Saghi. „Multivariate Analysis and Compressed Sensing Methods for Spectroscopic Electron Tomography of Semiconductor Devices“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 500–501 (cit. on p. 30).

[127] Hosagrahar V Jagadish, Beng Chin Ooi, Kian-Lee Tan, Cui Yu, and Rui Zhang. „iDistance: An adaptive B+-tree based indexing method for nearest neighbor search“. In: *ACM Transactions on Database Systems (TODS)*, Vol. 30, No. 2 (2005), pp. 364–397 (cit. on pp. 43, 57).

[128] Herve Jegou, Matthijs Douze, and Cordelia Schmid. „Hamming embedding and weak geometric consistency for large scale image search“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 5302. Springer. 2008, pp. 304–317 (cit. on p. 53).

[129] Abdul J Jerri. „The Shannon sampling theorem—Its various extensions and applications: A tutorial review“. In: *Proceedings of the IEEE*, Vol. 65, No. 11 (1977), pp. 1565–1596 (cit. on p. 12).

[130] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. „Perceptual losses for real-time style transfer and super-resolution“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 9906. Springer. 2016, pp. 694–711 (cit. on p. 23).

[131] Lewys Jones and Clive Downing. „The MTF & DQE of Annular Dark Field STEM: Implications for Low-dose Imaging and Compressed Sensing“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 478–479 (cit. on p. 29).

[132] Lena Karos, Pinak Bheed, Pascal Peter, and Joachim Weickert. „Optimising Data for exemplar-based inpainting“. In: *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer. 2018, pp. 547–558 (cit. on p. 27).

[133] Mitsuro Kato, Noboru Kawase, Takeshi Kaneko, Shoichi Toh, Syo Matsumura, and Hiroshi Jinnai. „Maximum diameter of the rod-shaped specimen for transmission electron microtomography without the “missing wedge”“. In: *Ultramicroscopy*, Vol. 108, No. 3 (2008), pp. 221–229 (cit. on p. 33).

[134] Noboru Kawase, Mitsuro Kato, Hideo Nishioka, and Hiroshi Jinnai. „Transmission electron microtomography without the “missing wedge” for quantitative structural analysis“. In: *Ultramicroscopy*, Vol. 107, No. 1 (2007), pp. 8–15 (cit. on pp. 2, 33).

[135] Thomas F Kelly and Michael K Miller. „Atom probe tomography“. In: *Review of Scientific Instruments*, Vol. 78, No. 3 (2007), pp. 031101.1–031101.20 (cit. on p. 2).

[136] James R Kremer, David N Mastronarde, and J Richard McIntosh. „Computer visualization of three-dimensional image data using IMOD“. In: *Journal of Structural Biology*, Vol. 116, No. 1 (1996), pp. 71–76 (cit. on p. 63).

[137] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. „Imagenet classification with deep convolutional neural networks“. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105 (cit. on p. 31).

[138] Andreas Kupsch, Axel Lange, Manfred P Hentschel, Sebastian Lück, Volker Schmidt, Roman Grothausmann, André Hilger, and Ingo Manke. „Missing wedge computed tomography by iterative algorithm DIRECTT“. In: *Journal of Microscopy*, Vol. 261, No. 1 (2016), pp. 36–45 (cit. on p. 34).

[139] Andreas Kupsch, Axel Lange, Manfred P Hentschel, Sebastian Lück, Volker Schmidt, André Hilger, Francisco Garcia-Moreno, and Ingo Manke. „Reconstruction of limited view tomography data by DIRECTT“. In: *Proceedings 18th World Conference on Non-Destructive Testing*. 2012, pp. 1–9 (cit. on p. 34).

[140] Gitta Kutyniok and Demetrio Labate. „Introduction to shearlets“. In: *Shearlets*. Springer, 2012, pp. 1–38 (cit. on p. 15).

[141] Tsz-Ho Kwok, Hoi Sheung, and Charlie CL Wang. „Fast query for exemplar-based image completion“. In: *IEEE Transactions on Image Processing*, Vol. 19, No. 12 (2010), pp. 3106–3115 (cit. on p. 25).

[142] Tsz-Ho Kwok and Charlie CL Wang. „Interactive image inpainting using DCT based exemplar matching“. In: *International Symposium on Visual Computing*. Springer. 2009, pp. 709–718 (cit. on p. 24).

[143] Nouamane Laanait and Albina Y Borisevich. „Deep Convolutional Neural Network Approach as a Universal Tool for Determination of Local 3D Structure from ABF STEM Images of Perovskitesy“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 530–531 (cit. on p. 32).

[144] Will Landecker, Rick Chartrand, and Simon DeDeo. „Robust sparse coding and compressed sensing with the difference map“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 8691. Springer. 2014, pp. 315–329 (cit. on p. 31).

[145] Olivier Le Meur and Christine Guillemot. „Super-resolution-based inpainting“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 7577. Springer. 2012, pp. 554–567 (cit. on pp. 23, 24).

[146] Henri Lebesgue. „Sur les fonctions représentables analytiquement“. In: *Journal de mathematiques pures et appliquees*, Vol. 1 (1905), pp. 139–216 (cit. on p. 45).

[147] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. „Deep learning“. In: *Nature*, Vol. 521, No. 7553 (2015), pp. 436–444 (cit. on p. 31).

[148] Edwin Lee, Benjamin P Fahimian, Cristina V Iancu, Christian Suloway, Gavin E Murphy, Elizabeth R Wright, Daniel Castaño-Díez, et al. „Radiation dose reduction and image enhancement in biological imaging through equally-sloped tomography“. In: *Journal of Structural Biology*, Vol. 164, No. 2 (2008), pp. 221–227 (cit. on p. 32).

[149] Anat Levin, Assaf Zomet, and Yair Weiss. „Learning how to inpaint from global image statistics“. In: *International Conference on Computer Vision (ICCV)*, Vol. 2 (2003), pp. 305–313 (cit. on p. 21).

[150] Michael S Lewicki and Terrence J Sejnowski. „Learning overcomplete representations“. In: *Neural Computation*, Vol. 12, No. 2 (2000), pp. 337–365 (cit. on p. 16).

[151] Chuan Li and Michael Wand. „Precomputed real-time texture synthesis with markovian generative adversarial networks“. In: *European Conference on Computer Vision (ECCV)*. Vol. LNCS 9907. Springer. 2016, pp. 702–716 (cit. on pp. 23, 32).

[152] Shutao Li and Ming Zhao. „Image inpainting with salient structure completion and texture propagation“. In: *Pattern Recognition Letters*, Vol. 32, No. 9 (2011), pp. 1256–1266 (cit. on p. 24).

[153] Xin Li. „Image recovery via hybrid sparse representations: A deterministic annealing approach“. In: *IEEE Journal of Selected Topics in Signal Processing*, Vol. 5, No. 5 (2011), pp. 953–962 (cit. on p. 21).

[154] Yue Li, Karl A Hujsak, Vadim Backman, and Vinayak P Draivid. „Inpainting Assisted Controlled Rotation Tomography (CORT)“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 502–503 (cit. on p. 29).

[155] King Ip Lin, Hosagrahar V Jagadish, and Christos Faloutsos. „The TV-tree: An index structure for high-dimensional data“. In: *The International Journal on Very Large Data Bases (VLDB)*, Vol. 3, No. 4 (1994), pp. 517–542 (cit. on p. 43).

[156] Dong Liu, Xiaoyan Sun, Feng Wu, Shipeng Li, and Ya-Qin Zhang. „Image compression with edge-based inpainting“. In: *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 17, No. 10 (2007), pp. 1273–1287 (cit. on p. 24).

[157] Yunqiang Liu and Vicent Caselles. „Exemplar-based image inpainting using multiscale graph cuts“. In: *IEEE Transactions on Image Processing*, Vol. 22, No. 5 (2013), pp. 1699–1711 (cit. on p. 24).

[158] Artur Łoza, Lyudmila Mihaylova, David Bull, and Nishan Canagarajah. „Structural similarity-based object tracking in multimodality surveillance videos“. In: *Machine Vision and Applications*, Vol. 20, No. 2 (2009), pp. 71–83 (cit. on p. 18).

[159] Jiangbo Lu, Yu Li, Hongsheng Yang, Dongbo Min, Wei Yong Eng, and Minh N. Do. „PatchMatch Filter: Edge-Aware Filtering Meets Randomized Search for Visual Correspondence“. In: *Transaction on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 9 (2017), pp. 1866–1879 (cit. on p. 25).

[160] W. Lu, W. Li, K. Kpalma, and J. Ronsin. „Compressed Sensing Performance of Random Bernoulli Matrices with High Compression Ratio“. In: *Signal Processing Letters*, Vol. 22, No. 8 (2015), pp. 1074–1078 (cit. on p. 111).

[162] M Lustig, DL Donoho, and JM Pauly. „Rapid MR imaging with compressed sensing and randomly under-sampled 3DFT trajectories“. In: *14th Annual Meeting ISMRM*. Citeseer. 2006, pp. 695–695 (cit. on p. 27).

[163] Michael Lustig, David Donoho, and John M Pauly. „Sparse MRI: The application of compressed sensing for rapid MR imaging“. In: *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, Vol. 58, No. 6 (2007), pp. 1182–1195 (cit. on p. 27).

[164] Michael Lustig, Jin Hyung Lee, David L Donoho, and John M Pauly. „Faster imaging with randomly perturbed, under-sampled spirals and l1 reconstruction“. In: *Proceedings of the 13th annual meeting of ISMRM, Miami Beach*. Citeseer. 2005, pp. 685–685 (cit. on p. 27).

[165] Markus Mainberger, Andrés Bruhn, Joachim Weickert, and Søren Forchhammer. „Edge-based compression of cartoon-like images with homogeneous diffusion“. In: *Pattern Recognition*, Vol. 44, No. 9 (2011), pp. 1859–1873 (cit. on p. 23).

[166] Markus Mainberger, Sebastian Hoffmann, Joachim Weickert, Ching Hoo Tang, Daniel Johannsen, Frank Neumann, and Benjamin Doerr. „Optimising spatial and tonal data for homogeneous diffusion inpainting“. In: *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*. Vol. LNCS 6667. Springer. 2011, pp. 26–37 (cit. on p. 26).

[167] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. „Online learning for matrix factorization and sparse coding“. In: *Journal of Machine Learning Research*, Vol. 11, No. 3 (2010), pp. 19–60 (cit. on p. 16).

[168] Ruslana Makovetsky, Nicolas Piche, and Mike Marsh. „Dragonfly as a Platform for Easy Image-based Deep Learning Applications“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 532–533 (cit. on p. 32).

[169] Stéphane Mallat and Gabriel Peyré. „A review of bandlet methods for geometrical image representation“. In: *Numerical Algorithms*, Vol. 44, No. 3 (2007), pp. 205–234 (cit. on p. 15).

[170] Roummel F Marcia, Zachary T Harmany, and Rebecca M Willett. „Compressive coded aperture imaging“. In: *Computational Imaging VII*. Vol. 7246. International Society for Optics and Photonics. 2009, 72460G.1–72460G.13 (cit. on p. 28).

[171] Raúl Martínez-Noriega, Aline Roumy, and Gilles Blanchard. „Exemplar-based image inpainting: Fast priority and coherent nearest neighbor search“. In: *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE. 2012, pp. 1–6 (cit. on p. 24).

[172] Vivien Marx. „Brain mapping in high resolution“. In: *Nature*, Vol. 503, No. 7474 (2013), pp. 147–152 (cit. on p. 3).

[173] Simon Masnou and J-M Morel. „Level lines based disocclusion“. In: *International Conference on Image Processing (ICIP)*. Vol. 3. IEEE. 1998, pp. 259–263 (cit. on p. 21).

[174] MathWorks. *Matlab software.* [Accessed: 06 December 2018]. 2017b. URL: `https://de.mathworks.com/products/matlab.html` (cit. on p. 63).

[175] Werner Meyer-Ilse, Donna Hamamoto, Ajit Nair, SA Lelievre, Gregory Denbeaux, L Johnson, Angelic Lucero Pearson, et al. „High resolution protein localization using soft X-ray microscopy“. In: *Journal of Microscopy*, Vol. 201, No. 3 (2001), pp. 395–403 (cit. on p. 2).

[176] Moshe Mishali, Yonina C Eldar, Oleg Dounaevsky, and Eli Shoshan. „Xampling: Analog to digital at sub-Nyquist rates“. In: *Circuits, Devices & Systems*, Vol. 5, No. 1 (2011), pp. 8–20 (cit. on p. 28).

[177] Guy M Morton. *A computer oriented geodetic data base and a new technique in file sequencing*. International Business Machines Company New York, 1966 (cit. on pp. 45, 46).

[178] PD Nellist, GT Martinez, C O'Leary, H Yang, A Stevens, and ND Browning. „Focused-Probe STEM Ptychography: Reconstruction Methods, Transfer Functions and Signal-to-Noise“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 488–489 (cit. on p. 29).

[179] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2018 (cit. on p. 32).

[180] Takahiro Ogawa and Miki Haseyama. „Image inpainting based on sparse representations with a perceptual metric“. In: *EURASIP Journal on Advances in Signal Processing*, Vol. 2013, No. 1 (2013), pp. 179.1–179.26 (cit. on p. 24).

[181] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, and Sung Nok Chiu. *Spatial tessellations: concepts and applications of Voronoi diagrams*. Vol. 501. John Wiley & Sons, 2009 (cit. on pp. 76, 112).

[182] Bruno A Olshausen and David J Field. „Sparse coding with an overcomplete basis set: A strategy employed by V1?“ In: *Vision research*, Vol. 37, No. 23 (1997), pp. 3311–3325 (cit. on p. 15).

[183] Sung W Park, Lars Linsen, Oliver Kreylos, John D Owens, and Bernd Hamann. „Discrete Sibson Interpolation“. In: *IEEE Transactions on Visualization & Computer Graphics*, No. 2 (2006), pp. 243–253 (cit. on p. 102).

[185] Vishal M Patel and Rama Chellappa. *Sparse representations and compressive sensing for imaging and vision*. Springer Science & Business Media, 2013 (cit. on p. 77).

[186] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. „Context encoders: Feature learning by inpainting“. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2016, pp. 2536–2544 (cit. on pp. 23, 32).

[187] Yagyensh Chandra Pati, Ramin Rezaiifar, and Perinkulam Sambamurthy Krishnaprasad. „Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition“. In: *Asilomar Conference on Signals, Systems, and Computers*. Vol. 1. IEEE. 1993, pp. 40–44 (cit. on p. 15).

[188] Giuseppe Peano. „Sur une courbe, qui remplit toute une aire plane“. In: *Mathematische Annalen*, Vol. 36, No. 1 (1890), pp. 157–160 (cit. on p. 45).

[189] Pawel Penczek, Michael Marko, Karolyn Buttle, and Joachim Frank. „Double-tilt electron tomography“. In: *Ultramicroscopy*, Vol. 60, No. 3 (1995), pp. 393–410 (cit. on p. 33).

[190] Pascal Peter, Sebastian Hoffmann, Frank Nedwed, Laurent Hoeltgen, and Joachim Weickert. „Evaluating the true potential of diffusion-based inpainting in a compression context“. In: *Signal Processing: Image Communication*, Vol. 46 (2016), pp. 40–53 (cit. on p. 23).

[191] Pascal Peter and Joachim Weickert. „Compressing images with diffusion-and exemplar-based inpainting“. In: *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*. Vol. LNCS 9087. Springer. 2015, pp. 154–165 (cit. on p. 24).

[192] Mihai-Alexandru Petrovici, Cristian Damian, Cristian Udrea, Florin Garoi, and Daniela Coltuc. „Single Pixel Camera with Compressive Sensing by non-uniform sampling“. In: *International Conference on Communications (COMM)*. IEEE. 2016, pp. 443–448 (cit. on p. 28).

[193] Nikolay Ponomarenko, Vladimir Lukin, Alexander Zelensky, Karen Egiazarian, Marco Carli, and Federica Battisti. „TID2008 - a database for evaluation of full-reference visual quality assessment metrics“. In: *Advances of Modern Radioelectronics*, Vol. 10, No. 4 (2009), pp. 30–45 (cit. on p. 19).

[194] Nikolay Ponomarenko, Flavia Silvestri, Karen Egiazarian, Marco Carli, Jaakko Astola, and Vladimir Lukin. „On between-coefficient contrast masking of DCT basis functions“. In: *Proceedings of the 3^{rd} international workshop on Video Processing and Quality Metrics (VPQM)*. Vol. 4. 2007, pp. 1–4 (cit. on pp. 18, 19).

[196] Ekta Prashnani, Hong Cai, Yasamin Mostofi, and Pradeep Sen. „PieAPP: Perceptual Image-Error Assessment through Pairwise Preference“. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 1808–1817 (cit. on p. 17).

[197] Yongbo Qin and Feng Wang. „A curvature constraint Exemplar-based image inpainting“. In: *International Conference on Image Analysis and Signal Processing (IASP)*. IEEE. 2010, pp. 263–267 (cit. on p. 24).

[198] Tran Minh Quan, David GC Hildebrand, and Won-Ki Jeong. „Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics“. In: *arXiv preprint arXiv:1612.05360* (2016) (cit. on p. 105).

[199] Michael Radermacher. „Three-dimensional reconstruction of single particles from random and nonrandom tilt series“. In: *Microscopy Research and Technique*, Vol. 9, No. 4 (1988), pp. 359–394 (cit. on p. 2).

[200] Idan Ram, Michael Elad, and Israel Cohen. „Image processing using smooth ordering of its patches“. In: *IEEE Transactions on Image Processing*, Vol. 22, No. 7 (2013), pp. 2764–2774 (cit. on p. 26).

[201] Carlos Ramirez, Vladik Kreinovich, and Miguel Argaez. „Why l1 is a good approximation to l0: A geometric explanation“. In: *Journal of Uncertain Systems*, Vol. 7, No. 3 (2013), pp. 203–207 (cit. on p. 16).

[202] BW Reed, N Moghadam, RS Bloom, ST Park, and DJ Masiel. „Next Steps for Compressively Sensed Video in Transmission Electron Microscopy“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 494–495 (cit. on p. 29).

[203] Hongyu Ren, Diqi Chen, and Yizhou Wang. „RAN4IQA: Restorative Adversarial Nets for No-Reference Image Quality Assessment“. In: *arXiv preprint arXiv:1712.05444* (2017) (cit. on p. 17).

[204] Ryan Robucci, Jordan D Gray, Leung Kin Chiu, Justin Romberg, and Paul Hasler. „Compressive sensing on a CMOS separable-transform image sensor“. In: *Proceedings of the IEEE*, Vol. 98, No. 6 (2010), pp. 1089–1101 (cit. on p. 28).

[205] Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. „Dictionaries for sparse representation modeling“. In: *Proceedings of the IEEE*, Vol. 98, No. 6 (2010), pp. 1045–1057 (cit. on p. 15).

[206] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. „Artistic style transfer for videos and spherical images“. In: *International Journal of Computer Vision*, Vol. 126, No. 11 (2018), pp. 1–21 (cit. on p. 23).

[207] Tijana Ružić and Aleksandra Pizurica. „Context-aware image inpainting with application to virtual restoration of old paintings“. In: *IEICE Information and Communication Technology Forum (ICTF)*. 2013 (cit. on p. 25).

[208] Tijana Ružić and Aleksandra Pižurica. „Context-aware patch-based image inpainting using Markov random field modeling“. In: *IEEE Transactions on Image Processing*, Vol. 24, No. 1 (2015), pp. 444–456 (cit. on p. 25).

[209] Hans Sagan. *Space-filling curves*. Springer Science & Business Media, 2012 (cit. on p. 45).

[210] David Salomon. *Data compression: the complete reference*. Springer Science & Business Media, 2004 (cit. on pp. 16, 17).

[211] Mehul P Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K Markey. „Complex wavelet structural similarity: A new image similarity index“. In: *IEEE Transactions on Image Processing*, Vol. 18, No. 11 (2009), pp. 2385–2401 (cit. on p. 18).

[212] Mehul P Sampat, Zhou Wang, Mia K Markey, Gary J Whitman, Tanya W Stephens, and Alan C Bovik. „Measuring intra-and inter-observer agreement in identifying and localizing structures in medical images“. In: *International Conference on Image Processing (ICIP)*. IEEE. 2006, pp. 81–84 (cit. on p. 18).

[213] K Sangeetha, P Sengottuvelan, and E Balamurugan. „A Novel Exemplar based Image Inpainting Algorithm for Natural Scene Image Completion with Improved Patch Prioritizing“. In: *International Journal of Computer Applications*, Vol. 36, No. 4 (2011) (cit. on p. 25).

[214] Guy Satat, Gabriella Musarra, Ashley Lyons, Barmak Heshmat, Ramesh Raskar, and Daniele Faccio. „Compressive Ultrafast Single Pixel Camera“. In: *Computational Optical Sensing and Imaging*. Optical Society of America. 2018, CTu2E.1–CTu2E.2 (cit. on p. 28).

[215] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, et al. „Fiji: an open-source platform for biological-image analysis“. In: *Nature Methods*, Vol. 9, No. 7 (2012), pp. 676–682 (cit. on p. 116).

[216] Jakob Schiøtz, Jacob Madsen, Pei Liu, Ole Winther, Jens Kling, Jakob Birkedal Wagner, and Thomas Willum Hansen. „Identifying Atoms in High Resolution Transmission Electron Micrographs Using a Deep Convolutional Neural Net“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 512–513 (cit. on p. 32).

[217] Christian Schmaltz, Pascal Peter, Markus Mainberger, Franziska Ebel, Joachim Weickert, and Andrés Bruhn. „Understanding, optimising, and extending data compression with anisotropic diffusion“. In: *International Journal of Computer Vision*, Vol. 108, No. 3 (2014), pp. 222–240 (cit. on p. 23).

[218] Markus Schneider, Pascal Peter, Sebastian Hoffmann, Joachim Weickert, and Enric Meinhardt-Llopis. „Gradients versus Grey Values for Sparse Image Reconstruction and Inpainting-Based Compression“. In: *International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS)*. Vol. LNCS 10016. Springer. 2016, pp. 1–13 (cit. on p. 23).

[219] Eric Schwenker, Fatih Sen, Colin Ophus, Tadas Paulauskas, Jinglong Guo, Spencer Hills, Robert Klie, and Maria KY Chan. „An Autonomous Microscopy Workflow for Structure Determination from Atomic-Resolution Images“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 510–511 (cit. on p. 32).

[220] Thermo Fisher Scientific. *Tomography 4.0 software.* [Accessed: 06 December 2018]. URL: `https://www.fei.com/software/tomography-4` (cit. on p. 63).

[221] Thermo Fisher Scientific. *Xplore3D software.* [Accessed: 06 December 2018]. URL: `http://www.cnf.umcs.lublin.pl/pdfy/Xplore_3D_web_ds.pdf` (cit. on p. 63).

[222] Timos K. Sellis, Nick Roussopoulos, and Christos Faloutsos. „The R+-Tree: A Dynamic Index for Multi-Dimensional Objects“. In: 1987, pp. 507–518 (cit. on p. 43).

[223] Claude Elwood Shannon. „Communication in the presence of noise“. In: *Proceedings of the IRE*, Vol. 37, No. 1 (1949), pp. 10–21 (cit. on p. 12).

[224] H. R. Sheikh and A. C. Bovik. „Image Information and Visual Quality“. In: *Transaction on Image Processing*, Vol. 15, No. 2 (2006), pp. 430–444 (cit. on p. 17).

[225] Bin Shen, Wei Hu, Yimin Zhang, and Yu-Jin Zhang. „Image inpainting via sparse representation“. In: *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE. 2009, pp. 697–700 (cit. on p. 24).

[226] Jianhong Shen and Tony F Chan. „Mathematical models for local nontexture inpaintings“. In: *SIAM Journal on Applied Mathematics*, Vol. 62, No. 3 (2002), pp. 1019–1043 (cit. on p. 21).

[227] Lawrence A Shepp and Benjamin F Logan. „The Fourier reconstruction of a head section“. In: *IEEE Transactions on Nuclear Science*, Vol. 21, No. 3 (1974), pp. 21–43 (cit. on p. 62).

[228] Robin Sibson. „A brief description of natural neighbor interpolation“. In: *Interpreting Multivariate Data* (1981), pp. 21–36 (cit. on p. 76).

[229] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. „Summarizing visual data using bidirectional similarity“. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2008, pp. 1–8 (cit. on p. 24).

[230] Andrew Stevens, Hao Yang, Lawrence Carin, Ilke Arslan, and Nigel D Browning. „The potential for Bayesian compressive sensing to significantly reduce electron dose in high-resolution STEM images“. In: *Microscopy*, Vol. 63, No. 1 (2013), pp. 41–51 (cit. on pp. 23, 30, 77).

[233] John E Stone, David Gohara, and Guochun Shi. „OpenCL: A parallel programming standard for heterogeneous computing systems“. In: *Computing in Science & Engineering*, Vol. 12, No. 3 (2010), pp. 66–73 (cit. on p. 41).

[234] Mr. Student. „The probable error of a mean“. In: *Biometrika*, Vol. 6, No. 1 (1908), pp. 1–25 (cit. on p. 67).

[235] Christoph Studer, Patrick Kuppinger, Graeme Pope, and Helmut Bolcskei. „Recovery of sparsely corrupted signals“. In: *IEEE Transactions on Information Theory*, Vol. 58, No. 5 (2012), pp. 3115–3130 (cit. on p. 21).

[236] Peter Su and Robert L Scot Drysdale. „A comparison of sequential Delaunay triangulation algorithms“. In: *Computational Geometry*, Vol. 7, No. 5-6 (1997), pp. 361–385 (cit. on p. 112).

[237] Jian Sun, Lu Yuan, Jiaya Jia, and Heung-Yeung Shum. „Image completion with structure propagation“. In: *ACM Transactions on Graphics (ToG)*, Vol. 24, No. 3 (2005), pp. 861–868 (cit. on p. 24).

[238] Joshua A Taillon. „Compressive Sensing Reconstruction for EDS Analysis“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 486–487 (cit. on p. 29).

[239] Google Brain Team. *Tensorflow - An open source machine learning framework for everyone*. [Accessed: 06 December 2018]. URL: `https://www.tensorflow.org/` (cit. on p. 105).

[249] John Treichler, Mark Davenport, and Richard Baraniuk. „Application of compressive sensing to the design of wideband signal acquisition receivers“. In: *U.S. / Australia Joint Workshop on Defense Applications of Signal Processing (DASP)*, Vol. 5 (2009), pp. 1–10 (cit. on p. 28).

[250] Sylvain Trepout, Masih Nilchian, Cédric Messaoudi, Laurène Donati, Michael Unser, and Sergio Marco. „Random Beam Scanning Transmission Electron Microscopy and Compressive Sensing as Tools for Drastic Electron Dose Reduction in Electron Tomography“. In: *European Microscopy Congress*. Wiley Online Library. 2016, pp. 41–42 (cit. on p. 76).

[251] Herbert Tropf and H Herzog. „Multidimensional Range Search in Dynamically Balanced Trees“. In: *Angewandte Informatik*, No. 2 (1981), pp. 71–77 (cit. on p. 46).

[252] Joel A Tropp, Jason N Laska, Marco F Duarte, Justin K Romberg, and Richard G Baraniuk. „Beyond Nyquist: Efficient sampling of sparse bandlimited signals“. In: *IEEE Transactions on Information Theory*, Vol. 56, No. 1 (2010), pp. 520–544 (cit. on p. 28).

[253] Joel A Tropp and Stephen J Wright. „Computational methods for sparse solution of linear inverse problems“. In: *Proceedings of the IEEE*, Vol. 98, No. 6 (2010), pp. 948–958 (cit. on p. 13).

[254] Joshua Trzasko and Armando Manduca. „Highly Undersampled Magnetic Resonance Image Reconstruction via Homotopic L_0-Minimization“. In: *IEEE Transactions on Medical Imaging*, Vol. 28, No. 1 (2009), pp. 106–121 (cit. on p. 27).

[255] Yi-Hsuan Tsai and Ming-Hsuan Yang. „Locality preserving hashing“. In: *International Conference on Image Processing (ICIP)*. IEEE. 2014, pp. 2988–2992 (cit. on pp. 43, 47).

[256] Ronen Tur, Yonina C Eldar, and Zvi Friedman. „Innovation rate sampling of pulse streams with application to ultrasound imaging“. In: *IEEE Transactions on Signal Processing*, Vol. 59, No. 4 (2011), pp. 1827–1842 (cit. on p. 28).

[257] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. „Deep image prior“. In: *arXiv preprint arXiv:1711.10925* (2017) (cit. on pp. 23, 32).

[258] Michael Unser and Nicolas Chenouard. „A unifying parametric framework for 2D steerable wavelet transforms“. In: *SIAM Journal on Imaging Sciences*, Vol. 6, No. 1 (2013), pp. 102–135 (cit. on p. 15).

[259] Shreyas S Vasanawala, Marcus T Alley, Brian A Hargreaves, Richard A Barth, John M Pauly, and Michael Lustig. „Improved pediatric MR imaging with compressed sensing“. In: *Radiology*, Vol. 256, No. 2 (2010), pp. 607–616 (cit. on p. 27).

[260] Salvador Villena, Miguel Vega, S Derin Babacan, Rafael Molina, and Aggelos K Katsaggelos. „Using the Kullback-Leibler divergence to combine image priors in super-resolution image reconstruction“. In: *International Conference on Image Processing (ICIP)*. IEEE. 2010, pp. 893–896 (cit. on pp. 76, 79).

[261] Gregory K Wallace. „The JPEG still picture compression standard“. In: *Communications of the ACM*, Vol. 34, No. 4 (1991), pp. 30–44 (cit. on p. 13).

[262] Chao Wang, Xiangchen Qian, Yong Yan, Feng Dong, and Huaxiang Wang. „An evaluation method for reconstructed images in electrical tomography“. In: *Instrumentation and Measurement Technology Conference Proceedings (IMTC)*. IEEE. 2008, pp. 692–696 (cit. on p. 17).

[263] Zhou Wang and Alan C Bovik. „Mean squared error: Love it or leave it? A new look at signal fidelity measures“. In: *IEEE Signal Processing Magazine*, Vol. 26, No. 1 (2009), pp. 98–117 (cit. on pp. 18, 19).

[264] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. „Image quality assessment: from error visibility to structural similarity“. In: *IEEE Transactions on Image Processing*, Vol. 13, No. 4 (2004), pp. 600–612 (cit. on pp. 17, 18).

[265] Zhou Wang, Qiang Li, and Xinli Shang. „Perceptual image coding based on a maximum of minimal structural similarity criterion“. In: *International Conference on Image Processing (ICIP)*. Vol. 2. IEEE. 2007, pp. II.121–II.124 (cit. on p. 18).

[266] Li-Yi Wei and Marc Levoy. „Fast texture synthesis using tree-structured vector quantization“. In: *Proceedings of the* 27^{th} *annual Conference on Computer Graphics and Interactive Techniques*. ACM Press/Addison-Wesley Publishing Co. 2000, pp. 479–488 (cit. on p. 26).

[267] Lloyd Welch. „Lower bounds on the maximum cross correlation of signals (corresp.)“ In: *IEEE Transactions on Information Theory*, Vol. 20, No. 3 (1974), pp. 397–399 (cit. on p. 14).

[268] David A White and Ramesh Jain. „Similarity indexing with the SS-tree“. In: *International Conference on Data Engineering*. IEEE. 1996, pp. 516–523 (cit. on p. 43).

[269] D Wolf, A Lubk, and H Lichte. „Weighted simultaneous iterative reconstruction technique for single-axis tomography“. In: *Ultramicroscopy*, Vol. 136 (2014), pp. 15–25 (cit. on p. 34).

[270] Alexander Wong and Jeff Orchard. „A nonlocal-means approach to exemplar-based inpainting“. In: *International Conference on Image Processing (ICIP)*. IEEE. 2008, pp. 2600–2603 (cit. on p. 24).

[271] Jiying Wu and Qiuqi Ruan. „Object removal by cross isophotes exemplar-based inpainting“. In: *International Conference on Pattern Recognition (ICPR)*. Vol. 3. IEEE. 2006, pp. 810–813 (cit. on p. 24).

[272] Zheng Xu, Michael Wilber, Chen Fang, Aaron Hertzmann, and Hailin Jin. „Beyond Textures: Learning from Multi-domain Artistic Images for Arbitrary Style Transfer“. In: *arXiv preprint arXiv:1805.09987* (2018) (cit. on p. 23).

[273] Zongben Xu and Jian Sun. „Image inpainting by patch propagation using patch sparsity“. In: *IEEE Transactions on Image Processing*, Vol. 19, No. 5 (2010), pp. 1153–1165 (cit. on p. 24).

[274] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. „Generative image inpainting with contextual attention“. In: *arXiv preprint* (2018) (cit. on pp. 23, 32).

[275] Xin Yuan and Yunchen Pu. „Deep Learning for Lensless Compressive Imaging“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 506–507 (cit. on p. 32).

[276] Jian Zhang, Debin Zhao, and Wen Gao. „Group-based sparse representation for image restoration“. In: *IEEE Transactions on Image Processing*, Vol. 23, No. 8 (2014), pp. 3336–3351 (cit. on pp. 30, 91).

[277] Jian Zhang, Debin Zhao, Ruiqin Xiong, Siwei Ma, and Wen Gao. „Image restoration using joint statistical modeling in a space-transform domain“. In: *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 24, No. 6 (2014), pp. 915–928 (cit. on pp. 30, 90).

[278] Qing Zhang and Jiajun Lin. „Exemplar-based image inpainting using color distribution analysis“. In: *Journal of Information Science and Engineering*, Vol. 28, No. 4 (2012), pp. 641–654 (cit. on p. 24).

[279] Yin Zhang. „User's guide for yall1: Your algorithms for l1 optimization“. In: *Technical Report* (2009), pp. 09–17 (cit. on pp. 30, 91).

[280] S-R Zhao and H Halling. „A new Fourier method for fan beam reconstruction“. In: *Nuclear Science Symposium and Medical Imaging Conference Record*. Vol. 2. IEEE. 1995, pp. 1287–1291 (cit. on p. 2).

[281] Mingyuan Zhou. „Nonparametric bayesian dictionary learning and count and mixture modeling“. PhD thesis. Duke University, 2013 (cit. on pp. 30, 76, 91).

[282] Mingyuan Zhou, Haojun Chen, John Paisley, Lu Ren, Lingbo Li, Zhengming Xing, David Dunson, et al. „Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images“. In: *IEEE Transactions on Image Processing*, Vol. 21, No. 1 (2012), pp. 130–144 (cit. on pp. 30, 76, 77, 91).

[283] Yatong Zhou, Lin Li, and Kewen Xia. „Research on weighted priority of exemplar-based image inpainting“. In: *Journal of Electronics (China)*, Vol. 29, No. 1-2 (2012), pp. 166–170 (cit. on p. 24).

[284] Chunbo Zhu, Xiaoyan Sun, Feng Wu, and Houqiang Li. „Video coding with spatio-temporal texture synthesis and edge-based inpainting“. In: *International Conference on Multimedia and Expo*. IEEE. 2008, pp. 813–816 (cit. on p. 24).

[285] Andreas Zürner, Markus Döblinger, Valentina Cauda, Ruoshan Wei, and Thomas Bein. „Discrete tomography of demanding samples based on a modified SIRT algorithm“. In: *Ultramicroscopy*, Vol. 115 (2012), pp. 41–49 (cit. on p. 33).

D

Own Publications

[29] Faysal Boughorbel, Pavel Potocek, Milos Hovorka, Libor Strakos, John Mitchels, Tomas Vystavel, Patrick Trampert, et al. „High-Throughput Large Volume SEM Workflow using Sparse Scanning and In-painting Algorithms Inspired by Compressive Sensing“. In: *Microscopy and Microanalysis*, Vol. 23, No. S1 (2017), pp. 150–151 (cit. on pp. 30, 89, 105).

[61] Tim Dahmen, Michael Engstler, Christoph Pauly, Patrick Trampert, Niels de Jonge, Frank Mücklich, and Philipp Slusallek. „Feature Adaptive Sampling for Scanning Electron Microscopy“. In: *Scientific Reports*, Vol. 6 (2016), pp. 25350.1–25350.11 (cit. on pp. 90, 112, 126).

[62] Tim Dahmen, Niels de Jonge, Patrick Trampert, Michael Engstler, Christoph Pauly, Frank Mücklich, and Philipp Slusallek. „“Smart Microscopy”: Feature Based Adaptive Sampling for Focused Ion Beam Scanning Electron Microscopy“. In: *Microscopy and Microanalysis*, Vol. 22, No. S3 (2016), pp. 632–633 (cit. on pp. 90, 112).

[63] Tim Dahmen, Holger Kohr, Andrew R. Lupini, Jean-Pierre Baudoin, Christian Kübel, Patrick Trampert, Philipp Slusallek, and Niels de Jonge. „Combined Tilt- and Focal-Series Tomography for HAADF-STEM“. In: *Microscopy Today*, Vol. 24, No. 03 (2016), pp. 26–31 (cit. on p. 33).

[64] Tim Dahmen, Lukas Marsalek, Nico Marniok, Beata Turoňová, Sviatoslav Bogachev, Patrick Trampert, Stefan Nickels, and Philipp Slusallek. „The Ettention software package“. In: *Ultramicroscopy*, Vol. 161 (2016), pp. 110–118 (cit. on pp. 4, 41).

[65] Tim Dahmen and Patrick Trampert. „Sparse and Adaptive Sampling in Scanning Electron Microscopy“. In: *Microscopy and Microanalysis*, Vol. 25, No. S1 (2019), pp. 29–30 (cit. on p. 111).

[66] Tim Dahmen, Patrick Trampert, Niels de Jonge, and Philipp Slusallek. „Advanced recording schemes for electron tomography“. In: *MRS Bulletin*, Vol. 41, No. 07 (2016), pp. 537–541 (cit. on p. 33).

[67] Tim Dahmen, Patrick Trampert, and Philipp Slusallek. „Blob-based Algebraic Reconstruction Technique for Computed Laminography“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 994–995 (cit. on p. 145).

[68] Tim Dahmen*, Patrick Trampert*, Joachim Weickert, and Philipp Slusallek. „Space-Filling Curve Indices as Acceleration Structure for Exemplar-based Inpainting“. In: *arXiv preprint arXiv:1712.06326* (2017). *Patrick Trampert and Tim Dahmen contributed equally. (Cit. on pp. 41, 59, 72, 90).

[161] Nicole Ludwig, Tamara V Werner, Christina Backes, Patrick Trampert, Manfred Gessler, Andreas Keller, Hans-Peter Lenhof, et al. „Combining miRNA and mRNA Expression Profiles in Wilms Tumor Subtypes“. In: *International Journal of Molecular Sciences*, Vol. 17, No. 4 (2016), pp. 475.1–475.17 (cit. on p. 146).

[184] Kelly Parker*, Patrick Trampert*, Verena Tinnemann, Diana Peckys, Tim Dahmen, and Niels de Jonge. „Linear Chains of HER2 Receptors Found in the Plasma Membrane Using Liquid-Phase Electron Microscopy“. In: *Biophysical Journal*, Vol. 115, No. 3 (2018), pp. 503–513. *Patrick Trampert and Kelly Parker contributed equally. (Cit. on p. 146).

[195] Pavel Potocek, Remco Schoenmakers, Patrick Trampert, Tim Dahmen, and Maurice Peemen. „Sparse Scanning Electron Microscopy for Imaging and Segmentation in Connectomics“. In: *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2018, pp. 2461–2465 (cit. on p. 104).

[231] Daniel Stöckel, Tim Kehl, Patrick Trampert, Lara Schneider, Christina Backes, Nicole Ludwig, Andreas Gerasch, et al. „Multi-omics enrichment analysis using the GeneTrail2 web service“. In: *Bioinformatics (Oxford, England)*, Vol. 32, No. 10 (2016), pp. 1502–1508 (cit. on p. 145).

[232] Daniel Stöckel, Florian Schmidt, Patrick Trampert, and Hans-Peter Lenhof. „CausalTrail: Testing hypothesis using causal Bayesian networks“. In: *F1000Research*, Vol. 4 (2015) (cit. on p. 145).

[240] Patrick Trampert, Sviatoslav Bogachev, Tim Dahmen, and Philipp Slusallek. „A Comparative Study of Three Marker Detection Algorithms in Electron Tomography“. In: *Microscopy and Microanalysis*, Vol. 22, No. S3 (2016), pp. 1044–1045 (cit. on p. 145).

[241] Patrick Trampert, Sviatoslav Bogachev, Nico Marniok, Tim Dahmen, and Philipp Slusallek. „Marker Detection in Electron Tomography: A Comparative Study“. In: *Microscopy and Microanalysis*, Vol. 21, No. 06 (2015), pp. 1591–1601 (cit. on p. 145).

[242] Patrick Trampert, Faysal Bourghorbel, Pavel Potocek, Maurice Peemen, Christian Schlinkmann, Tim Dahmen, and Philipp Slusallek. „How should a fixed budget of dwell time be spent in scanning electron microscopy to optimize image quality?“ In: *Ultramicroscopy*, Vol. 191 (2018), pp. 11–17 (cit. on pp. 75, 90).

[243] Patrick Trampert, Delei Chen, Sviatoslav Bogachev, Tim Dahmen, and Philipp Slusallek. „Dictionary-based Filling of the Missing Wedge in Electron Tomography“. In: *Microscopy and Microanalysis*, Vol. 22, No. S3 (2016), pp. 554–555 (cit. on pp. 23, 77, 90).

[244] Patrick Trampert, Tim Dahmen, and Philipp Slusallek. „Fact or Fiction: Maximal Image Quality with Minimal Dwell Time“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 480–481 (cit. on pp. 75, 90).

[245] Patrick Trampert, Tim Kehl, Daniel Stöckel, Christina Backes, Andreas Keller, and Hans-Peter Lenhof. „GeneTrail2: a web server for the statistical analysis of molecular signatures“. In: *European Conference on Computational Biology*. 2014 (cit. on p. 145).

[246] Patrick Trampert, Sabine Schlabach, Tim Dahmen, and Philipp Slusallek. „Exemplar-Based Inpainting Based on Dictionary Learning for Sparse Scanning Electron Microscopy“. In: *Microscopy and Microanalysis*, Vol. 24, No. S1 (2018), pp. 700–701. Awarded with a Presidential Scholar Award at Microscopy and Microanalysis 2018. (Cit. on pp. 55, 89, 90, 110).

[247] Patrick Trampert, Jonas Vogelgesang, Christian Schorr, Michael Maisl, Sviatoslav Bogachev, Nico Marniok, Alfred Louis, et al. „Spherically symmetric volume elements as basis functions for image reconstructions in computed laminography“. In: *Journal of X-ray science and technology*, Vol. 25, No. 4 (2017), pp. 533–546 (cit. on p. 145).

[248] Patrick Trampert, Wu Wang, Delei Chen, Raimond BG Ravelli, Tim Dahmen, Peter J Peters, Christian Kübel, and Philipp Slusallek. „Exemplar-based inpainting as a solution to the missing wedge problem in electron tomography“. In: *Ultramicroscopy*, Vol. 191 (2018), pp. 1–10 (cit. on p. 90).

E

List of abbreviations

AE	acceleration error	55
BF-TEM	bright-field transmission electron microscopy	62
BP	basis pursuit	91
BPFA	beta process factor analysis	174
BSE	back scatter electron	78
CNN	convolutional neural network	32
CPU	central processing unit	102
CS	compressed sensing	127
CSC	convolutional sparse coding	91
CW-SSIM	complex wavelet structural similarity index	81
EBI	exemplar-based inpainting	173
EDS	energy dispersive spectroscopy	29
EM	electron microscopy	145
ET	electron tomography	61
FIB-SEM	focussed ion beam scanning electron microscopy	2
FFT	fast Fourier transform	29
FWHM	full-width at half-maximum	67
GOAL	geometric analysis operator learning	174
GPU	graphics processing unit	127
GSR	group-based sparse restoration	91
HAADF	high-angle annular dark-field	62
HPC	high-performance computing	101
HVS	human visual system	18
JSM	joint statistical model	90
knn	k-nearest neighbour	173
MR	magnetic resonance	27
MSE	mean squared error	26
NaN	not a number	78
PCA	principal component analysis	42
PDE	partial differential equation	21
PSNR	peak signal-to-noise ratio	175
PSNR-HVS-M	masked peak signal-to-noise-ratio adapted to the human visual system	81
RIP	restricted isometry property	14
RP	random projections	43
RBF	radial basis function	39

RMSE	root mean squared error	65
SBF	serial block face	89
SEM	scanning electron microscopy/microscope	174
SFC	space-filling curve	42
SIRT	simultaneous iterative reconstruction technique	64
SNR	signal-to-noise ratio	89
SR	super-resolution	75
SSD	sum of squared differences	36
SSIM	structural similarity index	175
STEM	scanning transmission electron microscopy	76
TEM	transmission electron microscopy	61
TLD	through the lens detector	78
TVM	total variation minimisation	27
UHR	ultra high resolution	78
3D	three-dimensional	173
2D	two-dimensional	128

List of Figures

F

List of Tables

G

Colophon

This thesis was typeset with LaTeX 2$_{\varepsilon}$. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at `http://cleanthesis.der-ric.de/`.