Martin Zach

Generative Models as Regularizers for Inverse Problems in Imaging

DOCTORAL THESIS

to achieve the university degree of
Doktor der technischen Wissenschaften

submitted to
Graz University of Technology

Supervisor

–
–

External examiner

–
–

–

Abstract

The variational approach to inverse problems in imaging involves minimizing an objective function composed of a data fidelity term and a regularization term. Classical variational regularizers have a rich history, are well understood, and often come with recovery guarantees. However, these approaches are frequently outperformed by various data-driven methods developed in the recent years. At the same time, the improvement in performance is typically accompanied by a loss of interpretability and robustness.

This thesis develops a rigorous framework that combines the strengths of variational and data-driven approaches. We adopt a strict Bayesian interpretation of the inverse problem, revealing that the regularizer is related to the prior distribution of the underlying signal. Consequently, we propose learning a parametric model of the prior distribution using generative learning techniques. Once learned, these parametric models serve as plug-and-play substitutes for classical variational penalties for inverse problems.

We apply this idea in two different domains: For inverse problems in magnetic resonance imaging (MRI), we propose a deep neural regularizer that encodes high-level domain statistics learned from reference data. Paired with a fast nonlinear inversion algorithm, we achieve state-of-the-art results for parallel MRI. The reconstruction algorithm accounts for different frequency selection patters via an appropriate data likelihood. Additionally, we provide uncertainty estimates for any reconstruction by exploiting the posterior distribution.

For inverse problems on natural images, we revisit classical Markov random field modeling techniques, combining them with modern ideas from diffusion models. This approach yields a family of models that serve as mean squared error (MSE) optimal denoisers for Gaussian noise of arbitrary variance. By linking MSE optimal denoisers to density estimation, we demonstrate that these models can also solve more general inverse problems. The principled construction leads to high performance with minimal parameters.

In summary, we demonstrate how generative learning techniques can be used to learn regularizers for inverse problems. The Bayesian framework is extremely versatile and leads to great performance.

Kurzfassung

Der variationsbasierte Ansatz zur Lösung inverser Probleme in der Bildgebung besteht darin, eine Zielfunktion zu minimieren, die aus einem Datenanpassungsterm und einem Regularisierungsterm besteht. Klassische Regularisierer haben eine umfangreiche Historie, sind gut verstanden, und bieten häufig Wiederherstellungsgarantien. Diese Ansätze werden jedoch oft von verschiedenen datengesteuerten Methoden, die in den letzten Jahren entwickelt wurden, übertroffen. Gleichzeitig geht die Leistungssteigerung typischerweise mit einem Verlust an Interpretierbarkeit und Robustheit einher.

In dieser Arbeit entwickeln wir einen rigorosen Rahmen, der die Vorzüge von Variationsmethoden mit datengesteuerten Ansätzen kombiniert. Wir nehmen eine strikte bayesianische Interpretation des inversen Problems an und stellen fest, dass der Regularisierer mit der a priori Verteilung des zugrunde liegenden Signals zusammenhängt. Daher schlagen wir vor, ein parametrisches Modell der a priori Verteilung mithilfe generativer Lerntechniken zu erlernen. Die erlernten parametrischen Modelle dienen als Plug-and-Play-Ersatz für klassische Regularisierer bei inversen Problemen.

Diese Idee wenden wir in zwei verschiedenen Bereichen an: Für inverse Probleme in der Magnetresonanztomographie (MRT) schlagen wir einen tiefen neuronalen Regularisierer vor, der nach dem Lernen an Referenzdaten hochrangige Domänenstatistiken kodiert. In Kombination mit einem schnellen gemeinsamen nichtlinearen Inversionsalgorithmus erzielen wir gerausragende Ergebnisse für paralleles MRT. Insbesondere kann der Rekonstruktionsalgorithmus unterschiedliche Frequenzauswahlmuster über eine geeignete Datenwahrscheinlichkeit berücksichtigen. Darüber hinaus können wir jede Rekonstruktion mit Unsicherheitsschätzungen versehen, indem wir die a posteriori Verteilung ausnutzen.

Für inverse Probleme bei natürlichen Bildern kombiniere wir klassische Techniken der Markov-Random-Feld-Modellierung mit modernen Ideen aus Diffusionsmodellen. Insbesondere erhalten wir eine Familie von Modellen, die als MSE-optimale Denoiser für Gaußsches Rauschen beliebiger Varianz dienen können. Durch die Verknüpfung MSE-optimaler Denoiser mit der Dichteschätzung zeigen wir, dass diese Modelle auch zur Lösung allgemeinerer inverser Probleme verwendet werden können. Die prinzipielle Konstruktion führt zu guter Leistung bei sehr wenigen Parametern.

Zusammenfassend zeigen wir, wie generative Lerntechniken verwendet werden können, um Regularisierer in inversen Problemen zu erlernen. Der Bayes'sche Rahmen ist äußerst vielseitig und führt zu hervorragender Leistung.

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

Date	Signature

Acknowledgments

_

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents:

Many phenomena can only be observed through indirect measurements, while the underlying causes remain obscure. Inferring these underlying causes from the measurements involves solving an *inverse problem*. This task is notably challenging in most interesting cases for several reasons.

First, we require an accurate model of the relationship between the measurements and the cause, known as the *forward problem*. The complexity of the forward problem stems from the various physical effects that influence the measurements. Even when armed with an accurate model of the measurement process, inferring the cause remains challenging when only limited or corrupted measurements are available. In such situations, many different causes might explain the same set of measurements. Therefore, we aim to identify the "most reasonable" of all possible causes. But how do we determine what makes a cause reasonable?

Traditionally, this determination was based on the cause's "regularity"; simple assumptions, such as the boundedness of the norm or the variation of the signal. However, classical regularity assumptions are often overly simplistic and fail to capture complex structure inherent in the causes. A significant part of this thesis is dedicated to *learning* these complex structures from reference data.

In the following sections, we introduce the particular domain of inverse problems that we study in this thesis: imaging. We showcase prototypical examples and review classical inference techniques based on regularity assumptions. After discussing their limitations, we explore how modern data-driven approaches address these challenges. Finally, we conclude this chapter by summarizing the contributions of this thesis and outline its structure.

1.1 Inverse problems in imaging

Inverse problems are ubiquitous in the imaging sciences. Applications in computer vision include denoising [207], optical flow estimation [121], segmentation [48], and object detection [187]. In medical imaging, which this thesis particularly focuses on, examples include positron emission spectroscopy, X-ray computed tomography (CT), and magnetic resonance imaging (MRI). These inverse problems are typically very challenging because the incomplete measurements can be explained by many

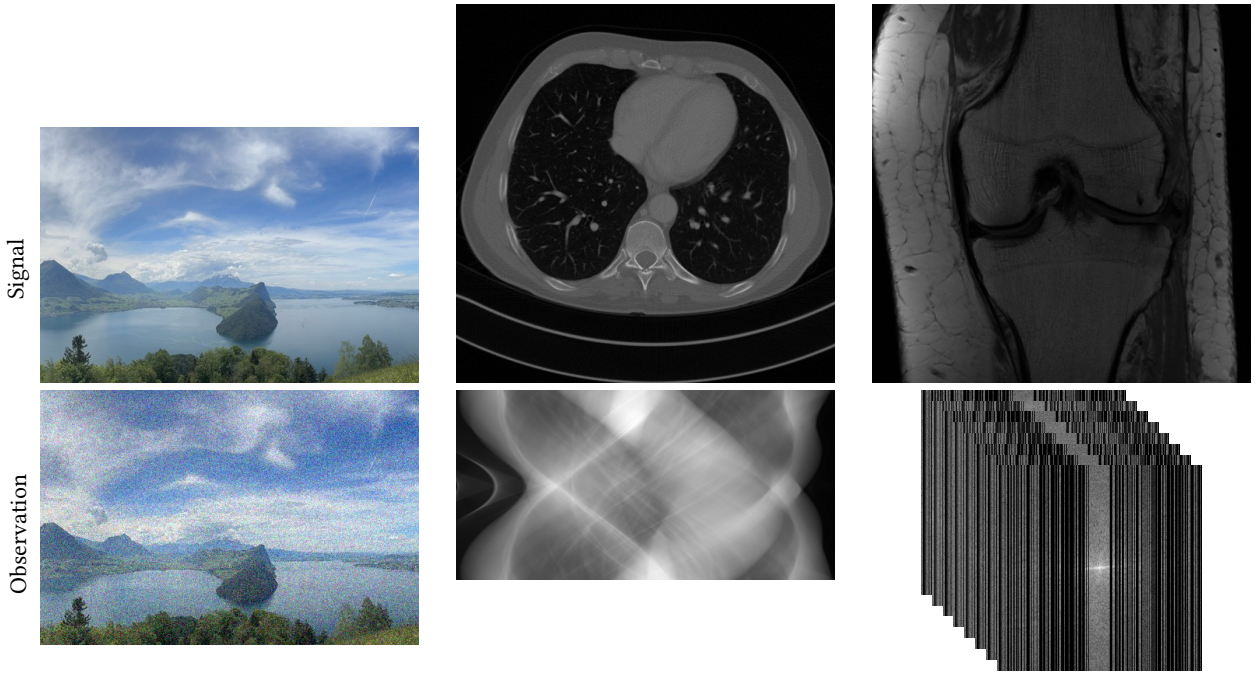

Figure 1.1: Examples of inverse problems in imaging: In the top row, the first column shows a natural scene, the second column shows an axial cross section of the human thorax from an X-ray CT scan, and the third column shows a coronal cross section of the human knee from an MRI scan. The bottom row shows the corresponding noisy observations.

causes and modeling, numerical, or measurement errors can result in heavily deteriorated reconstructions of the causes. This makes inverse problems generally *ill-posed* as defined by Hadamard [105]. In particular, a problem is ill-posed in the sense of Hadamard if it violates any of the following conditions:

- A solution to the problem exists.
- The solution is unique.
- The solution is stable with respect to the measurements.

Figure 1.1 illustrates three inverse problems in imaging that we consider in this thesis: Image denoising, image reconstruction from X-ray CT data, and image reconstruction from MRI data. In each case, the top row shows the underlying cause and the bottom row shows the corresponding observation. The goal of solving the inverse problem is to recover the causes in the top row from the observations in the bottom row. We formalize this in the following paragraphs.

The first column in fig. 1.1 shows an image of a natural scene (top row) and its *noisy* observation (bottom row).[1] Each channel of each pixel in the observation is corrupted with additive Gaussian noise. Formally, the underlying cause is a signal with square pixels that have three color channels each. Every channel can take one a real value and we say that the signal, which we endow with its own symbol, x, is an element of the vector space $\mathbb{R}^{m\times n\times 3}$. Then, each entry in the observation, which we endow with its own symbol, y, is given by

$$y_{i,j,k} = x_{i,j,k} + \eta_{i,j,k} \tag{1.1}$$

1: This is the Vierwaldstättersee in Switzerland; the image was taken by the author.

where $\eta_{i,j,k}$ is normally distributed with mean zero. More compactly,

$$y = x + \eta, \tag{1.2}$$

and hence the space of signal coincides with that of the observation: $y \in \mathbb{R}^{m \times n \times 3}$. The goal of solving the inverse problem is to recover the unknown signal x from the observation y.

Images of natural scenes are noisy when the sensors capture too few photons, which is common in low-light situations, when capturing fast-moving objects, or with small sensors in hand-held devices. The denoising problem is also significant in a broader context due to its close relationship to density estimation via Tweedie's identity, discussed in chapter 6. In fact, denoising algorithms are currently the foundation of state-of-the-art image generation algorithms (in the form of diffusion models, see e.g. [128]) as well as image reconstruction algorithms (in the form of plug-and-play methods, see e.g. [122]).

The second column in fig. 1.1 shows an axial cross-section of the human thorax obtained from an X-ray CT scan (top row) and its observation (bottom row). Each pixel in the signal represents the linear X-ray attenuation coefficient of the tissue, while each entry in the observation is an *area integral* of this coefficient over the cone spanned by the X-ray source and a detector element.

In this scenario, the signal and the observation are in different spaces. When the X-ray camera has n_d detectors and data are acquired at n_a angles during a half-circle rotation of the source around the anatomy, the observation consists of $n_d \times n_a$ real numbers. Visualizing these observations in rectilinear coordinates, i.e. as an image $y \in \mathbb{R}^{n_d \times n_a}$ yields the *sinogram* shown in the bottom row. There, the data are also noisy due to thermal noise in the measurement channels formally described by

$$y = Rx + \eta. \tag{1.3}$$

Here, $R \colon \mathbb{R}^{m \times n} \to \mathbb{R}^{n_d \times n_a}$ is a *linear* operator that models $n_d \times n_a$ area integrals according to the measurement geometry.[2] The multiplicative Poisson noise that is typically encountered in X-ray imaging is well approximated by heteroscedastic additive Gaussian noise η [226].

2: We chose the symbol R here to emphasize the close relation to the *Radon transform*.

X-ray CT images provide excellent hard-tissue contrast and are critical in clinical practice. However, X-ray radiation can have adverse health effects, making it essential to minimize radiation exposure while maintaining diagnostic value of the images. This can be achieved by reducing the X-ray tube current, which results in a decrease in signal-to-noise ratio. Alternatively, SparseCT approaches [44, 137] block the X-rays from entering the tissue via multi-slit collimators (effectively reducing the number of detectors) and angular undersampling approaches [45] via a shutter (effectively reducing the number of angles). In any case, missing or noisy data makes the reconstruction problem more challenging and necessitates robust algorithms that can adapt to the measurement setup.

The third column in fig. 1.1 shows a coronal cross-section of the human knee obtained from an MRI scan (top row) and its observation (bottom row). In MRI, the scanner's receiver coil measures the changes in the magnetism of nuclei excited by radio-frequency pulses. By encoding different spatial frequencies via excitation

coils, each entry in the observation correspond to a Fourier coefficient of the underlying signal.

Modern MRI systems often utilize multiple receiver coils to achieve faster acquisition, as pioneered by Roemer et al. [202]. There, the data depend on the spatially varying coil sensitivities of the receiver coils and the measurement process can be formalized as

$$\begin{aligned} y_1 &= MF(x \odot \sigma_1) + \eta_1, \\ y_2 &= MF(x \odot \sigma_2) + \eta_2, \\ &\vdots \\ y_c &= MF(x \odot \sigma_c) + \eta_c. \end{aligned} \tag{1.4}$$

Here, $y_1, y_2, \dots, y_c \in \mathbb{C}^{n_f}$ are the data from the $c \in \mathbb{N}$ receiver coils. The data of the i-th coil are related to the signal $x \in \mathbb{C}^{m \times n}$ and the coil sensitivity $\sigma_i \in \mathbb{C}^{m \times n}$ through the two-dimensional discrete Fourier transform $F \colon \mathbb{C}^{m \times n} \to \mathbb{C}^{m \times n}$. The coil sensitivities are usually unknown and depend on the dielectric properties of the imaged anatomy, necessitating their estimation along with the signal [134]. This results in a *nonlinear* relationship between the data and the estimation variables.

MRI images provide excellent soft-tissue contrast and are invaluable in clinical practice. However, long acquisition times limit patient throughput, leading to extensive research dedicated to minimizing the acquisition time while maintaining diagnostic value of the images. One way to reduce the acquisition time that does not necessitate changes to the hardware is to acquire less data, which is modeled by the frequency selection operator $M \colon \mathbb{C}^{m \times n} \to \mathbb{C}^{n_f}$ in eq. (1.4). The observation in the third column in fig. 1.1 uses a classical Cartesian frequency selection with unacquired frequencies shown in black. It is evident that reducing available data makes retaining diagnostic value more challenging. In addition, to account for particular anatomical features the frequency selection is subject to change in clinical practice. This again emphasizes the need for robust and adaptable algorithms.

The discussion above suggests a general formulation of a (possibly nonlinear) inverse problem:

$$y = A(x) + \eta. \tag{1.5}$$

Here, y belongs to some space $\mathscr{Y}$, x belongs to some space $\mathscr{X}$, A maps from $\mathscr{X}$ to $\mathscr{Y}$ encoding the measurement process, and $\eta \in \mathscr{Y}$ models measurement noise. In this thesis, $\mathscr{X}$ and $\mathscr{Y}$ are finite-dimensional vector spaces, and we assume that A is known exactly. The next section discusses classical approaches to recover the underlying cause x from an observation y. The limitations of classical approaches lead to the consideration of modern data-driven approaches.

1.2 Variational methods and Bayes theorem

In this section, we provide a brief historical perspective on classical approaches to solving inverse problems. This perspective is without an numerical example; in chapter 4, we discuss the benefits and drawbacks of these approaches through a prototypical MRI reconstruction problem. We focus on classical variational meth-

ods, regularization techniques, and the accompanying probabilistic interpretation, highlighting the subtleties and the necessity for proper probabilistic modeling.

A common approach to recovering the underlying signal from observations is to minimize the discrepancy to the observation, as measured through the forward operator. However, this problem is often ill-posed because many signals might explain the observation equally well. Therefore, *regularization* techniques have been developed. These techniques introduce and additional *regularization* term, which acts solely on the reconstruction. Among all signals that fit the observation, the regularization term favors those with desirable properties. For example, the property of a *bounded norm* of the signal leads to the famous variational problem

$$\arg\min_{x \in \mathscr{X}} \frac{1}{2}\|A(x) - y\|^2 + \frac{\lambda}{2}\|x\|^2. \tag{1.6}$$

This form of regularization is usually attributed to Tikhonov [229] and Phillips [186]. Here, the first term measures the squared deviation of the signal from the observation under the forward operator, and the second term ensures that the norm of the signal remains bounded. The *regularization parameter* $\lambda > 0$ controls the influence of the penalization of the norm.

In inverse problems in imaging, magnitude penalization results in dark images and is rarely useful.[3] Instead, an extremely popular approach is to penalize the *total variation* in the image. This technique penalizes the difference between neighboring pixels absolutely, rather than the magnitude quadratically. Introduced by Rudin, Osher, and Fatemi [207] in 1992, this idea has inspired extensive literature on designing regularizers [17, 29, 206] and optimization algorithms [36].

3: See the MRI example in chapter 4.

Generally, the variational approach to inverse problems amounts to solving the optimization problem

$$\arg\min_{x \in \mathscr{X}} D(x, y) + R(x). \tag{1.7}$$

The objective consists of two terms: The *data fidelity term* $D\colon \mathscr{X} \times \mathscr{Y} \to \mathbb{R}$, which penalizes deviations of the signal from the observation by utilizing the forward operator, and the *regularization term* $R\colon \mathscr{X} \to \mathbb{R}$, which penalizes undesired characteristics of the signal.

While the variational approach is versatile, choosing the appropriate regularizer can be challenging. From a statistical perspective, the regularizer is related to the *prior distribution* of the underlying signal in maximum a-posteriori (MAP) inference. These concepts are treated more rigorously in chapter 2, and a more formal treatment of inverse problems in the Bayesian perspective can be found in [223]. According to Bayes theorem, the *posterior density* of a signal given the data y is

$$p_{X|Y}(x, y) = \frac{p_{Y|X}(x, y)p_X(x)}{\int_{\mathscr{X}} p_{Y|X}(\xi, y)p_X(\xi)\,\mathrm{d}\xi}. \tag{1.8}$$

Here, $p_{Y|X}$ is the *data likelihood*, which quantifies the agreement between the signal and the data. In inverse problems in imaging, the data likelihood is typically determined by the forward model and noise statistics. Additionally, p_X is the *prior*,

which quantifies the likelihood of the signal based on characteristics of reference signals.

The posterior density quantifies the likelihood of a signal given an observation y. Commonly, we aim to find the signal that best explains the observation by maximizing the posterior density:

$$\arg\max_{x \in \mathscr{X}} p_{X|Y}(x, y). \tag{1.9}$$

Equivalently, by taking the negative logarithm and observing that the denominator in eq. (1.8) is constant with respect to the signal, we get

$$\arg\min_{x \in \mathscr{X}} -\log p_{Y|X}(x, y) - \log p_X(x). \tag{1.10}$$

Comparing eq. (1.7) and eq. (1.10), we relate the data fidelity term D to the negative data log-likelihood $-\log p_{Y|Z}$ and the regularization term R to the negative log-prior $-\log p_X$.

The probabilistic interpretation is the basis of many modern data-driven approaches to inverse problems and provides a rigorous framework for other probabilistic concepts such as uncertainty quantification. In this thesis, we adopt this interpretation and strictly separate the likelihood and prior. Consequently, finding an effective regularizer amounts to fitting a parametric model to the prior density of reference signals. However, this comes with subtleties that we uncover by invoking decision theory.

1.3 Bayes estimators

The equivalence between the classical variational approach to inverse problems and MAP estimation is evident "algebraically" by comparing eq. (1.7) and eq. (1.10). However, this does not imply that a regularizer that leads to good reconstructions in eq. (1.7) is a necessarily a good model of the true negative log-prior. The standard metric for comparing reconstructions to the reference is the mean squared error (MSE) (definition 2.5.1).[4] Given data y and the corresponding posterior $p_{X|Y}(\,\cdot\,, y)$, finding a Bayes estimator with the MSE loss involves solving

4: equivalently, peak signal-to-noise ratio (PSNR) (definition 2.5.3)

$$\arg\min_{x \in \mathscr{X}} \int_{\mathscr{X}} \|x - x'\|^2 p_{X|Y}(x', y)\, \mathrm{d}x', \tag{1.11}$$

which is known (see e.g. [127, page 172]) to be the posterior expectation

$$\int_{\mathscr{X}} x\, p_{X|Y}(x, y)\, \mathrm{d}x. \tag{1.12}$$

In contrast, the variational formulation seeks the MAP estimate. Thus, there is a mismatch between the evaluation metric and the inference procedure: a regularizer that leads to a small MSE via MAP inference is not necessarily a good model of the negative log-prior. Conversely, a good model of the negative log-prior does not necessarily lead to small MSE via MAP inference.

Generally, although the variational approach *can* be interpreted probabilistically, the regularizer does not have to be derived from statistics of reference data. Often, a regularizer can be useful for downstream tasks without any specific relation to reference statistics. In such cases, estimators of the form eq. (1.7) are sometimes referred to as *maximum penalized likelihood estimators* [24, 94]. In addition, Gribonval [94] has pointed out that for some class of regularizers, solutions to eq. (1.7) are minimum mean-squared-error (MMSE) estimates with respect to *some* prior. However, in general the regularizer is not the negative log of this prior.

In summary, when pursuing strict separation of likelihood and prior and adopting the *natural* probabilistic interpretation of the regularizer, evaluation (with the standard metrics) should be done on the MMSE estimate. We follow this principle throughout this thesis: In chapter 5, we learn a deep neural regularizer and compute MMSE estimates by sampling the posterior with Markov chain Monte Carlo (MCMC) methods. In chapter 6, the principled construction of the regularizer allows us to compute MMSE estimates for denoising problems with one gradient evaluation.

1.4 Data-driven regularizers

When adopting the probabilistic interpretation, finding a good regularizer amounts to learning the reference density. In the context of inverse problems in imaging, this was pursued in the foundational works by Zhu and Mumford in their series of papers [255, 256, 257] that lead to the celebrated filters, random fields, and maximum entropy (FRAME) model. However, their approach relied on hand-selected filters and is problematic in the context of gradient-based optimization due to the use of piecewise constant functions. Welling, Hinton, and Osindero [242] proposed learning the distribution of image patches via a product of one-dimensional experts, where the experts act on filter responses, and both the filters and the parameters of the experts[5] are learned. However, their learning algorithm does not scale to large images. The fields-of-experts (FoE) model by Roth and Black [206] extends this model to large images by replacing the Gibbs sampler relying on matrix inversions with a gradient-based MCMC sampling algorithm. Crucially, they obtain a *translation invariant* prior by sharing the same filters and experts across *all* patches in the image.

5: they chose Student-t experts

The FoE approach shares a lot of similarities with the learned regularizers we discuss in this thesis. In particular, the learning setup of the deep neural regularizer we propose in chapter 5 is largely the same; there we also resort to a gradient-based MCMC sampling algorithm to fit an intractable model to the reference density. However, instead of the product-of-experts structure, we design an expressive deep neural regularizer that is explicitly *not* translation invariant, to model the statistics of MRI scans of the human knee. In contrast, in chapter 6 we revisit the translation invariant FoE structure but use more efficient learning algorithms based on score matching [123] and additionally incorporate ideas from diffusion models [218].

1.5 Contributions and outline

In recent years, data-driven methods have become state-of-the-art in many inverse problems in imaging. The survey paper [10] of Arridge, Maass, Öktem, and Schönlieb provides a comprehensive overview of recent advances. This can largely be attributed to the availability of large datasets, the increase in parallel processing power of graphics processing units, and the development and free availability of automatic differentiation frameworks such as PyTorch [183]. Concurrently, there has been a shift towards *discriminative* signal recovery approaches, where the separation between data likelihood and prior is largely lost, making the models harder to interpret. The difference between *generative* and *discriminative* signal recovery approaches is discussed in detail in chapter 3. In the context of inverse problems in imaging, likelihoods are frequently subject to change, making discriminative point estimators problematic—an issue illustrated with an example in fig. 5.3.

In this thesis, we aim to leverage modern data-driven approaches within the framework of classical variational approaches to inverse problems. In particular, in light of the discussion in section 1.2, finding a good regularizer amounts to fitting a parametric model to the prior density of reference signals—*generative modeling*. For inverse problems in MRI, we design a deep neural regularizer that models the negative log-prior distribution of MRI images of human knees. We demonstrate that the learned model encodes high-level statistics by synthesizing realistic-looking images *without data*. We use the learned regularizer alongside a fast algorithm for solving the *nonlinear* inversion problem encountered in parallel MRI and achieve state-of-the-art reconstruction results. For inverse problems with natural images, we revisit classical translation-invariant priors and propose replacing computationally intensive maximum likelihood training with denoising score matching. By adopting ideas from diffusion models and carefully selecting filters and parametrization of the experts, we develop a model that serves as an MSE-optimal denoiser for Gaussian noise with arbitrary variance.

This thesis is intended to be largely self-contained. To this end, chapter 2 provides mathematical preliminaries of functional analysis, probability theory, stochastic differential equations, optimization, and the representation of digital images. These preliminaries are covered with (possibly excessive) rigor. Familiar readers can skip this chapter entirely, although forward references are provided to where the concepts are used in the remainder of the thesis. In chapter 3, we review basic concepts from machine learning and neural networks. In particular, the chapter introduces the neural network terminology used throughout this thesis and emphasizes the difference between generative and discriminative learning. Chapter 4 provides a historical overview of the development of regularizers in imaging, using MRI reconstruction as a running example. There, we also superficially introduce our proposed models and compare their performance to classical regularizers.

The previous chapters serve as an introduction to our contributions, which are discussed in chapter 5 and chapter 6. In particular, in chapter 5 we design a deep neural regularizer for MRI reconstruction, exploiting the characteristics of images in the application domain. Due to the alignment of the patient in the scanner,

features in the images consistently appear in distinct locations, and the anatomy exhibits subtle non-local dependencies that our regularizer can capture. In this chapter, we also revisit joint nonlinear inversion for parallel MRI reconstruction and propose a fast algorithm that eliminates the need for calibration scans for coil sensitivity estimation. In chapter 6 we revisit the structure of classical translation invariant regularizers and show how to efficiently learn a FoE type model with score matching. By leveraging ideas from diffusion models we retrieve a model that can act as an MSE optimal denoiser for Gaussian noise with arbitrary variance. Finally, we conclude the thesis and suggest future research directions in chapter 7.

Chapter 2

Preliminaries

In this chapter, we recall important mathematical concepts that are used in the remainder of this thesis. In particular, we recall standard results from functional analysis, probability theory, stochastic differential equations, and optimization and define our notion of images as signals defined on a finite Cartesian grid. This chapter serves primarily as a reference, with no new contributions. Subsequent chapters will refer back to the definitions, theorems, and algorithms discussed here as needed. However, section 2.3 on stochastic differential equations and section 2.4 on optimization discuss some peculiarities of how the concepts are used in this thesis.

Contents:

2.1 Functional analysis

In this section, we recall definitions and results from analysis, topology, and measure theory used in the subsequent sections. This overview is adapted from [7, 15, 132]. In this thesis, the field $\mathbb{K}$ are either the real numbers $\mathbb{R}$ or the complex numbers $\mathbb{C}$. For any $x \in \mathbb{K}$, we denote the absolute value by

$$|x| \coloneqq \sqrt{x\overline{x}} \text{ with } \overline{x} = \begin{cases} \mathfrak{Re}(x) - \mathfrak{Im}(x) & \text{if } \mathbb{K} = \mathbb{C}, \\ x & \text{if } \mathbb{K} = \mathbb{R}. \end{cases} \tag{2.1}$$

2.1.1 Vector spaces

Definition 2.1.1 (Vector space)**.** *A* vector space *over a field $\mathbb{K}$ is a nonempty set $\mathbb{V}$ endowed with a binary function $(\,\cdot +_{\mathbb{V}} \cdot\,)\colon \mathbb{V}\times\mathbb{V} \to \mathbb{V}$ ("vector addition") and a binary function $(\,\cdot \cdot_{\mathbb{V}} \cdot\,)\colon \mathbb{K}\times\mathbb{V} \to \mathbb{V}$ ("scalar multiplication") such that the following holds:*

1. *Let x, y, and z be arbitrary elements of $\mathbb{V}$. The vector addition satisfies*
   (a) $x +_{\mathbb{V}} (y +_{\mathbb{V}} z) = (x +_{\mathbb{V}} y) +_{\mathbb{V}} z$.
   (b) $x +_{\mathbb{V}} y = y +_{\mathbb{V}} x$.

   (c) *There exists a unique element* $0_{\mathbb{V}} \in \mathbb{V}$ *such that* $x +_{\mathbb{V}} 0_{\mathbb{V}} = x$.
   (d) *There exists an element* $-x \in \mathbb{V}$ *such that* $x +_{\mathbb{V}} (-x) = 0$.
2. *Let* a *and* b *be arbitrary elements of the field* $\mathbb{K}$ *and let* x *be an arbitrary element of* $\mathbb{V}$. *The scalar multiplication satisfies*
   (a) $a \cdot_{\mathbb{V}} (b \cdot_{\mathbb{V}} x) = (a \cdot_{\mathbb{V}} b) \cdot_{\mathbb{V}} x$.
   (b) *Let* $1_{\mathbb{K}}$ *denote the multiplicative identity in* $\mathbb{K}$. *Then* $1_{\mathbb{K}} \cdot_{\mathbb{V}} x = x$.
3. *The scalar multiplication distributes with respect to vector addition and field addition:*
   (a) $a \cdot_{\mathbb{V}} (x +_{\mathbb{V}} y) = a \cdot_{\mathbb{V}} x +_{\mathbb{V}} a \cdot_{\mathbb{V}} y$.
   (b) $(a +_{\mathbb{K}} b) \cdot_{\mathbb{V}} x = a \cdot_{\mathbb{V}} x +_{\mathbb{V}} b \cdot_{\mathbb{V}} x$.

The most prominent example of a vector space is $\mathbb{R}^n$. Its base field are the real numbers $\mathbb{R}$, the set $\mathbb{V}$ are all n-tuples with real components and vector addition and scalar multiplication are defined as

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} +_{\mathbb{R}^n} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 +_{\mathbb{R}} y_1 \\ x_2 +_{\mathbb{R}} y_2 \\ \vdots \\ x_n +_{\mathbb{R}} y_n \end{pmatrix}, \quad a \cdot_{\mathbb{R}^n} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} a \cdot_{\mathbb{R}} x_1 \\ a \cdot_{\mathbb{R}} x_2 \\ \vdots \\ a \cdot_{\mathbb{R}} x_n \end{pmatrix}. \tag{2.2}$$

Here $x = (x_1, x_2, \dots, x_n)^\top$, $y = (y_1, y_2, \dots, y_n)^\top \in \mathbb{R}^n$ and $a \in \mathbb{R}$. In this thesis, it is not necessary to distinguish between the set of a vector space and the vector space itself (the (set, addition, multiplication) triple). In other words, $\mathbb{R}^n$ refers to the *vector space* and implicitly carries over the operations as defined above.

2.1.2 Inner products

It is well known that two vectors in $\mathbb{R}^n$ are *orthogonal*, if their *dot product* is zero. As a generalization of the dot product, inner products can be used to impose geometry onto a vector space by mapping a pair of vectors to a scalar in the underlying field.

Definition 2.1.2 (Inner product)**.** *Let* $\mathbb{V}$ *be a vector space over a field* $\mathbb{K}$. *An* inner product $\langle \cdot, \cdot \rangle_{\mathbb{V}} \colon \mathbb{V} \times \mathbb{V} \to \mathbb{K}$ *satisfies*

1. $\langle x, y \rangle_{\mathbb{V}} = \overline{\langle y, x \rangle_{\mathbb{V}}}$ *for any* x, y *in* $\mathbb{V}$.
2. $\langle a_1 x_1 + a_2 x_2, y \rangle_{\mathbb{V}} = a_1 \langle x_1, y \rangle_{\mathbb{V}} + a_2 \langle x_2, y \rangle_{\mathbb{V}}$ *for any* $a_1, a_2 \in \mathbb{K}$ *and any* $x, y \in \mathbb{V}$.
3. $\langle x, x \rangle_{\mathbb{V}} \geq 0$ *for any* $x \in \mathbb{V}$ *and* $\langle x, x \rangle_{\mathbb{V}}$ *if and only if* $x = 0_{\mathbb{V}}$.

We call the ordered pair $(\mathbb{V}, \langle \cdot, \cdot \rangle_{\mathbb{V}})$ an *inner product space*. The aforementioned dot product on $\mathbb{R}^n$ is the map

$$\langle x, y \rangle_{\mathbb{R}^n} = \sum_{i=1}^{n} x_i y_i. \tag{2.3}$$

2.1.3 Norms

We use norms to rigorously define notions of distance from the origin in a vector space. In particular, any sensible distance function should absolutely commute with scaling, obey the triangle inequality, and assign zero only to the zero vector. This motivates the following definition:

Definition 2.1.3 (Norm). *Let $\mathbb{V}$ be a vector space over a field $\mathbb{K}$. A norm $\|\cdot\|_{\mathbb{V}} \colon \mathbb{V} \to \mathbb{R}$ satisfies*

1. $\|ax\|_{\mathbb{V}} = |a|\,\|x\|_{\mathbb{V}}$ *for any* $x \in \mathbb{V}$ *and* $a \in \mathbb{K}$.
2. $\|x + y\|_{\mathbb{V}} \leq \|x\|_{\mathbb{V}} + \|y\|_{\mathbb{V}}$ *for all* $x, y \in \mathbb{V}$.
3. $\|x\|_{\mathbb{V}} = 0 \implies x = 0_{\mathbb{V}}$.

Note that item 1 implies $\|0_{\mathbb{V}}\|_{\mathbb{V}} = 0$ and that item 1 and item 2 imply nonnegativity: $\|x\|_{\mathbb{V}} \geq 0$, another sensible property of a distance function.

A function that satisfies item 1 and item 2 but maps some nonzero vectors to zero is called a *seminorm*. The ordered pair $(\mathbb{V}, \|\cdot\|_{\mathbb{V}})$ is called a *normed vector space*. Any inner product on a vector space $\mathbb{V}$ induces a norm through $\|\cdot\|_{\mathbb{V}} = \sqrt{\langle \cdot, \cdot \rangle_{\mathbb{V}}}$.

In the previous definitions, we endowed the operations of vector addition, scalar multiplication, inner product, and norms with a subscript to emphasize the vector space these operations occur in. However, this notation is overly verbose and usually the space is evident from the context. Thus, in the remainder of this thesis we omit this subscript unless needed.

An important family of norms are the ℓ^p norms on $\mathbb{K}^n$ defined as

$$\|x\|_p := \begin{cases} \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} & \text{if } 1 \leq p < \infty, \\ \max_{i=1,\dots,n} |x_i| & \text{if } p = \infty. \end{cases} \tag{2.4}$$

Figure 2.1 visualizes the unit spheres of different ℓ^p norms in $\mathbb{R}^2$: The unit-one-norm-sphere is a rhombus, the unit-two-norm-sphere is a circle and as p approaches infinity, the unit-p-norm-sphere approaches a square.

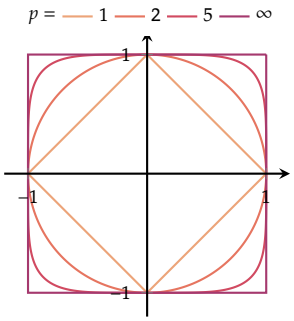

Figure 2.1: Unit spheres of different p-norms in $\mathbb{R}^2$.

2.1.4 Topology

In this section, we recall topological concepts that are required for understanding the optimization algorithms used in this thesis. To proceed, we first define norm balls: Let $(\mathbb{V}, \|\cdot\|)$ be a normed vector space. We denote with

$$\mathscr{B}_{\|\cdot\|}(c, r) := \{x \in \mathbb{V} \mid \|x - c\| < r\} \tag{2.5}$$

the open norm ball with respect to the norm $\|\cdot\|$ and radius $r > 0$, centered around $x \in \mathbb{V}$.

A set $S \subseteq \mathbb{V}$ is

- *bounded* if there exists a radius $r > 0$ such that $S \subset \mathscr{B}_{\|\cdot\|}(0_{\mathbb{V}}, r)$.

- *open* if for all $x \in S$ there exists an $\epsilon > 0$ such that $\mathscr{B}_{\|\cdot\|}(x, \epsilon) \subset S$.
- *closed* if $\mathbb{V} \setminus S$ is open.
- *compact* if every open cover of S has a finite subcover.

The last definition is somewhat technical and overkill for our purposes — thus we recall the simpler definition of compactness in the Euclidean case:

Theorem 2.1.1 (Heine-Borel Theorem). *Let $S \subset \mathbb{K}^n$. Then, the following statements are equivalent:*

- *S is bounded and closed.*
- *Every open cover of S has a finite subcover.*

Other basic notions that we will need for the rest of this section are the interior and the closure of a set:

Definition 2.1.4 (Interior, closure, boundary). *Let $(\mathbb{V}, \|\cdot\|)$ be a normed vector space and let $S \subseteq \mathbb{V}$.*

- *A point $x \in S$ is an* interior point *of S if there exists an open set $O \subseteq S$ such that $x \in O$. The set of all interior points forms the* interior *of S, denoted by* $\operatorname{int} S$.
- *A point $x \in \mathbb{V}$ is a* point of closure *of S if for every open set $O \subseteq S$ that contains x there exists an $s \in S$ such that $s \in O$. The set of all points of closure of S is the* closure *of S, denoted by $\overline{S}$.*
- *The* boundary *of S is the set $\partial S := \overline{S} \setminus \operatorname{int} S$.*

Finally, we define the neighborhood of a point in a normed space:

Definition 2.1.5 (Neighborhood). *Let $(\mathbb{V}, \|\cdot\|)$ be a normed vector space. A set $\mathscr{S}$ is a* neighborhood *of a point $x \in \mathbb{V}$ if there exists a radius $r > 0$ such that $\mathscr{B}_{\|\cdot\|}(x, r)$ is contained in $\mathscr{S}$.*

2.1.5 Convergence, completeness, and continuity

In this section, we introduce the concepts of convergence of sequences, which relates to completeness of spaces and finally continuity of functions. In this journey, we also rigorously define Hilbert spaces. To start, let $(\mathbb{V}, \|\cdot\|)$ be a normed vector space. We call a map from the natural numbers to $\mathbb{V}$ a *sequence* and use the shorthand notation $x^n := x(n)$ for a sequence x.

Definition 2.1.6 (Limit and convergence of a sequence). *A point $x_0 \in \mathbb{V}$ is the* limit *of a sequence x if for all $\epsilon > 0$ there exists and $N \in \mathbb{N}$ such that for every $n \geq N$, $\|x^n - x_0\| < \epsilon$. If such a point exists, we say that x* converges *to*

x_0 and write $x^n \to x_0$ as $n \to \infty$ or

$$\lim_{n\to\infty} x^n = x_0.$$

To define the completeness of a space, we require the slightly relaxed notion of a Cauchy sequence:

Definition 2.1.7 (Cauchy sequence)**.** *A sequence x is* Cauchy *if for every $\epsilon > 0$ there exists an $N \in \mathbb{N}$ such that for all $i, j > N$*

$$\|x^i - x^j\| < \epsilon.$$

In words, a sequence is Cauchy if the distance between iterates becomes arbitrarily small for large enough arguments. However, a Cauchy sequence is not necessarily convergent—we will show some examples later. Finally, we can characterize complete spaces:

Definition 2.1.8 (Completeness)**.** *The normed vector space $(\mathbb{V}, \|\cdot\|)$ is* complete *if every Cauchy sequence converges to an element of $\mathbb{V}$. A* Hilbert space *is an inner product space which is complete with respect to the induced norm $\|\cdot\| = \sqrt{\langle x, x\rangle}$. A* Banach space *is a normed space that is complete w.r.t. its norm.*

Spaces that are not complete are easy to construct. A famous example are the rational numbers endowed with the absolute value, $(\mathbb{Q}, |\cdot|)$: The sequence of rationals $n \mapsto \left(1 + n^{-1}\right)^n$ is Cauchy but does not converge in $(\mathbb{Q}, |\cdot|)$. In $(\mathbb{R}, |\cdot|)$, it famously converges to Euler's constant. The next theorem relates continuity of a function with the convergence of the image a sequence under the function. To state it, we first define continuity. For the following, we let $(\mathbb{V}, \|\cdot\|_{\mathbb{V}})$ and $(\mathbb{W}, \|\cdot\|_{\mathbb{W}})$ be two normed vector spaces.

Definition 2.1.9 (Continuity)**.** *A function $f\colon \mathbb{V} \to \mathbb{W}$ is* continuous at $x_0 \in \mathbb{V}$ *if for all $\epsilon > 0$ there exists a $\delta > 0$ such that $\|x - x_0\|_{\mathbb{V}} < \delta$ implies that $\|f(x) - f(x_0)\|_{\mathbb{W}} < \epsilon$.*

Let $S \subseteq V$. We call f *continuous on* S if it is continuous at every point in S. We call f *continuous* if it continuous on $\mathbb{V}$.

Theorem 2.1.2. *Let $f\colon \mathbb{V} \to \mathbb{W}$. The following statements are equivalent:*

- *f is continuous.*
- *For every set $S \subseteq \mathbb{W}$ open in $\mathbb{W}$, the preimage of S under f is open in $\mathbb{V}$.*
- *For every $x_0 \in \mathbb{V}$ and every sequence x on $\mathbb{V}$ such that $x^n \to x_0$ in $\mathbb{V}$ the sequence $f(x^n) \to f(x_0)$ in $\mathbb{W}$ as $n \to \infty$.*

Proof. See [7, section 2.17]. □

The optimization algorithms we use in this thesis often have slightly stronger assumptions on functions. In particular, they often require that the distance of the images of two points can be upper bounded by their distance, up to a multiplicative constant. This is captured in the notion of Lipschitz continuity:

Definition 2.1.10 (Lipschitz continuity). *A function* $f\colon \mathbb{V} \to \mathbb{W}$ *is called* Lipschitz continuous *with* Lipschitz constant $L > 0$ *if for all* $x, y \in \mathbb{V}$

$$\|f(x) - f(y)\|_{\mathbb{W}} \leq L\|x - y\|_{\mathbb{V}}.$$

Lipschitz continuity is illustrated for a map from $\mathbb{R}$ to $\mathbb{R}$ in fig. 2.2. There, the dark green function always stays outside of the double cone spanned by the two black lines. Classical examples of functions which are not Lipschitz continuous are the exponential function and the quadratic $x \mapsto x^2$ over $\mathbb{R}$, and $\sqrt{\cdot}$ over $\mathbb{R}_+$. The former two examples become infinitely steep as their arguments explode, the latter becomes infinitely steep as its argument approaches zero as illustrated in fig. 2.3. These examples demonstrate that Lipschitz continuity is rather strong; often it suffices that a function satisfies Lipschitz continuity *locally*:

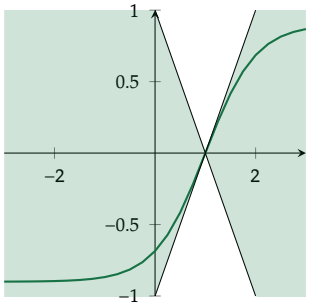

Figure 2.2: Geometric interpretation of Lipschitz continuity for a function over $\mathbb{R}$.

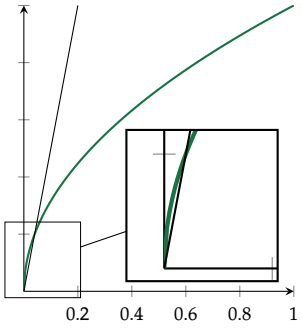

Figure 2.3: $\sqrt{\cdot}$ over $[0, 1]$ is not Lipschitz continuous.

Definition 2.1.11 (Local Lipschitz continuity). *A function* $f\colon \mathbb{V} \to \mathbb{W}$ *is* locally Lipschitz continuous *if for all* $x \in \mathbb{V}$ *there exists a neighborhood (definition 2.1.5)* $N(x)$ *such that the restriction of* f *to* $N(x)$ *is Lipschitz continuous.*

2.1.6 Measure theory

In this section we briefly introduce concepts from measure theory. It is mostly based on [30] and is by no means a complete overview of the field—its purpose is to introduce Lebesgue spaces and lay the foundations for probability theory.

Throughout the following discussion, we denote with $\mathscr{S}$ an arbitrary non-empty set. We denote with $2^{\mathscr{S}} := \{\mathscr{A} \mid \mathscr{A} \subseteq \mathscr{S}\}$ the power set of $\mathscr{S}$. First, we classify sets according to closedness properties.

Definition 2.1.12 (Closedness properties). *Let* $\mathfrak{F} \subseteq 2^{\mathscr{S}}$ *be a set of subsets, let* $\mathscr{A}, \mathscr{A}_i, \mathscr{B}, i \in \mathbb{N}$ *be arbitrary elements of* $\mathfrak{F}$*, and let* $\mathscr{I} \subseteq \mathbb{N}$ *be a finite or infinite subset of the natural numbers. The set of sets* $\mathfrak{F}$ *is*

- closed under intersections *if* $\mathscr{A} \cap \mathscr{B} \in \mathfrak{F}$.
- closed under countable intersections *if* $\bigcap_{i \in \mathscr{I}} \mathscr{A}_i \in \mathfrak{F}$.
- closed under unions *if* $\mathscr{A} \cup \mathscr{B} \in \mathfrak{F}$.
- closed under countable unions *if* $\bigcup_{i \in \mathscr{I}} \mathscr{A}_i \in \mathfrak{F}$.
- closed under differences *if* $\mathscr{A} \setminus \mathscr{B} \in \mathfrak{F}$.
- closed under complements *if* $\mathscr{S} \setminus \mathscr{A} \in \mathfrak{F}$.

The subsequent discussion of probability theory heavily relies on σ-algebras, which we now define and *measurable spaces*, which we now define.

Definition 2.1.13 (σ-algebra). *A set of subsets $\mathfrak{F} \subseteq 2^{\mathscr{S}}$ is a* σ-algebra on $\mathscr{S}$ *if*

- $\mathscr{S} \in \mathfrak{F}$.
- $\mathfrak{F}$ *is closed under complements.*
- $\mathfrak{F}$ *is closed under countable unions.*

Definition 2.1.14 (Measurable space and measurable sets). *A pair $(\mathscr{S}, \mathfrak{F})$, where $\mathscr{S}$ is an arbitrary nonempty set and $\mathfrak{F}$ is a σ-algebra over $\mathscr{S}$, is a* measurable space. *The sets in $\mathfrak{F}$ are called* measurable.

The smallest σ-algebra that contains a set of subsets $\mathfrak{E}$ is called the σ-algebra *induced* (or *generated*) by $\mathfrak{E}$, and we write $\sigma(\mathfrak{E})$. For a normed vector space $(\mathbb{V}, \|\cdot\|)$ the σ-algebra generated by all open sets is the *Borel σ-algebra* $\mathfrak{B}(\mathscr{S})$ on $\mathscr{S}$ and its elements are *Borel measurable*.

Later, we will need the notion of σ-algebras that are generated by (possibly more than) one map.

Definition 2.1.15 (Generated σ-algebra [132, Definition 1.79]). *Let $\mathscr{S}$ be a nonempty set and let I be an arbitrary index set. For any $i \in I$ let $(\mathscr{S}_i, \mathfrak{F}_i)$ be a measurable space and let $X_i \colon \mathscr{S} \to \mathscr{S}_i$ be an arbitrary map. Then*

$$\sigma(X_i, i \in I) \coloneqq \sigma\Big(\bigcup_{i\in I} \sigma(X_i)\Big) \coloneqq \sigma\Big(\bigcup_{i\in I} X_i^{-1}(\mathscr{S}_i)\Big) \tag{2.6}$$

is the σ-algebra on $\mathscr{S}$ that is generated *by $(X_i, i \in I)$. This is the smallest σ-algebra with respect to which all X_i are measurable.*

Now we develop *measures* to meaningfully quantify the "size" of sets. First, we recall properties of functions on sets and then define the notion of a *measure*.

Definition 2.1.16 (Properties of functions on sets). *Let $\mathfrak{F} \subseteq 2^{\mathscr{S}}$ and let $\mu \colon \mathfrak{F} \to [0, \infty]$ be a function on sets. μ is*

- monotone *if $\mu(\mathscr{A}) \leq \mu(\mathscr{B})$ for any $\mathscr{A}, \mathscr{B} \in \mathfrak{F}$ such that $\mathscr{A} \subseteq \mathscr{B}$.*
- additive *if $\mu\big(\bigcup_{i=1}^n \mathscr{A}_i\big) = \sum_{i=1}^n \mu(\mathscr{A}_i)$ for any mutually disjoint $\mathscr{A}_1, \mathscr{A}_2, \dots, \mathscr{A}_n \in \mathfrak{F}$ such that $\bigcup_{i=1}^n \mathscr{A}_i \in \mathfrak{F}$.*
- σ-additive *if $\mu\big(\bigcup_{i\in\mathbb{N}} \mathscr{A}_i\big) = \sum_{i\in\mathbb{N}} \mu(\mathscr{A}_i)$ for any mutually disjoint $\mathscr{A}_1, \mathscr{A}_2, \dots \in \mathfrak{F}$ such that $\bigcup_{i\in\mathbb{N}} \mathscr{A}_i \in \mathfrak{F}$.*
- subadditive *if $\mu\big(\bigcup_{i=1}^n \mathscr{A}_i\big) \leq \sum_{i=1}^n \mu(\mathscr{A}_i)$ for any $\mathscr{A}_1, \mathscr{A}_2, \dots, \mathscr{A}_n \in \mathfrak{F}$ such that $\bigcup_{i=1}^n \mathscr{A}_i \in \mathfrak{F}$.*
- σ-subadditive *if $\mu\big(\bigcup_{i=1}^n \mathscr{A}_i\big) \leq \sum_{i\in\mathbb{N}} \mu(\mathscr{A}_i)$ for any $\mathscr{A}_1, \mathscr{A}_2, \dots \in \mathfrak{F}$ such that $\bigcup_{i\in\mathbb{N}} \mathscr{A}_i \in \mathfrak{F}$.*

Definition 2.1.17 (Measure). *Let $(\mathscr{S}, \mathfrak{F})$ be a measurable space. A function $\mu\colon \mathfrak{F} \to [0,\infty]$ is a* measure *on $(\mathscr{S}, \mathfrak{F})$ if*

- $\mu(\varnothing) = 0$ *and*
- μ *is subadditive.*

The following enumerates "intuitive" measures that also find applications in this thesis:

1. *Counting measure:* Let $(\mathscr{S}, 2^{\mathscr{S}})$ with $\mathscr{S}$ nonempty be a measurable space. The *counting measure* on $(\mathscr{S}, 2^{\mathscr{S}})$

$$\mu(\mathscr{A}) = \begin{cases} \text{the number of elements in } \mathscr{A} & \text{if } \mathscr{A} \text{ is finite,} \\ \infty & \text{else,} \end{cases} \tag{2.7}$$

   assigns to any set the number of contained elements.
2. *Dirac measure:* Let $\mathscr{S} \subseteq \mathbb{R}^n$ be nonempty and $(\mathscr{S}, \mathfrak{B}(\mathscr{S}))$ be a measurable space. The *Dirac measure* on $(\mathscr{S}, \mathfrak{B}(\mathscr{S}))$

$$\delta_x(\mathscr{A}) = \begin{cases} 1 & \text{if } x \in \mathscr{A}, \\ 0 & \text{else,} \end{cases} \tag{2.8}$$

   assigns 1 to any set containing the point x, and 0 otherwise.
3. *Lebesgue measure:* Let $\operatorname{cube}(a,b) \coloneqq \{x \in \mathbb{R}^n \mid a_i \le x_i \le b_i \text{ for } i = 1, \dots, n\} \in \mathfrak{B}(\mathbb{R}^n)$ with $a, b \in \mathbb{R}^n$ and $a_i < b_i$ for all $i = 1, \dots, n$ denote half-open cuboids. Define

$$\hat{\mathfrak{L}}^n\big(\operatorname{cube}(a,b)\big) \coloneqq \prod_{i=1}^{n} (b_i - a_i) \tag{2.9}$$

   to be the volume of the cuboid $\operatorname{cube}(a,b)$. By the extension theorem of measures [132], there exists a unique extension of this measure to a measure on the measurable space $(\mathbb{R}^n, \mathfrak{B}(\mathbb{R}^n))$. This is the *$n$-dimensional Lebesgue measure*, which captures the intuitive idea of the (hyper-) volume of an n-dimensional set. To convey the importance, the Lebesgue measure is defined more rigorously in the following definition.

Definition 2.1.18 (Lebesgue measure). *The* n-dimensional Lebesgue measure *$\mathfrak{L}^n$ is the unique measure on $(\mathbb{R}^n, \mathfrak{B}(\mathbb{R}^n))$ such that*

$$\mathfrak{L}^n\big(\operatorname{cube}(a,b)\big) = \prod_{i=1}^{n} (b_i - a_i) \tag{2.10}$$

for all $a, b \in \mathbb{R}^n$ with $a_i < b_i$ for all $i = 1, \dots, n$.

A subclass of measures are *(σ-) finite measures* and *probability measures* that are defined as follows:

Definition 2.1.19 (Finite measures and probability measures). *Let μ be a measure over a measurable space $(\mathscr{S}, \mathfrak{F})$. If $\mu(\mathscr{S}) < \infty$, then μ is* finite. *If there exists a sequence of sets $\mathscr{S}_n$ in $\mathfrak{F}$ such that $\bigcup_{i\in\mathbb{N}} \mathscr{S}_i = \mathscr{S}$ for which $\mu(\mathscr{S}_i) < \infty$ for all $i \in \mathbb{N}$, then μ is* σ-finite. *In the special case that $\mu(\mathscr{S}) = 1$, μ is a* probability measure.

Finally, we can define a *measure space*.

Definition 2.1.20 (Measure space). *Let $\mathfrak{F}$ be a σ-algebra over $\mathscr{S}$ and let μ be a measure on $(\mathscr{S}, \mathfrak{F})$. The triple $(\mathscr{S}, \mathfrak{F}, \mu)$ is a* measure space.

Definition 2.1.21 (Null sets, almost everywhere, almost surely). *Let $(\mathscr{S}, \mathfrak{F}, \mu)$ be a measure space.*

- *A set $\mathscr{N} \subseteq \mathscr{S}$ is a* μ-null set *if there exists some $\mathscr{A} \in \mathfrak{F}$ with $\mathscr{N} \subseteq \mathscr{A}$ and $\mu(\mathscr{A}) = 0$.*
- *Let P be a statement which is true for all elements in $\mathscr{A}$. If $\mathscr{S} \setminus \mathscr{A}$ is a μ-null set, P holds* μ-almost everywhere (a.e.) on $\mathscr{S}$. *If μ is a probability measure, we say that P holds* μ-almost surely on $\mathscr{S}$.

Definition 2.1.22 (Completion of a measure space, μ-measurability). *Let $(\mathscr{S}, \mathfrak{F}, \mu)$ be a measure space. The σ-algebra $\mathfrak{F}_\mu$ defined by*

$$\mathscr{A} \in \mathfrak{F}_\mu \iff \mathscr{A} = \mathscr{B} \cup \mathscr{N} \text{ where } \mathscr{B} \in \mathfrak{F} \text{ and } \mathscr{N} \text{ is } \mu\text{-null} \tag{2.11}$$

is the completion *of $\mathfrak{F}$ with respect to μ. The elements of $\mathfrak{F}_\mu$ are* μ-measurable. *The measure μ can be extended to a measure μ^* on $(\mathscr{S}, \mathfrak{F}_\mu)$ via $\mu^*(\mathscr{A}) = \mu(\mathscr{B})$ with $\mathscr{A}, \mathscr{B}$ as defined above.*

To develop the notion of a Lebesgue integral, we follow [132] more closely. First, we define *measurable functions*, with the help of which we can define an *image measure*.

Definition 2.1.23 (Pre-image, measurable functions). *Let $(\mathscr{S}, \mathfrak{F})$ and $(\mathscr{S}', \mathfrak{F}')$ be measurable spaces. A function $f\colon \mathscr{S} \to \mathscr{S}'$ is* measurable *if for any $\mathscr{A}' \in \mathfrak{F}'$, the* pre-image *of $\mathscr{A}'$ under f is contained in $\mathfrak{F}$. In mathematical notation, f is measurable if*

$$f^{-1}(\mathscr{A}') := \{x \in \mathscr{S} \mid f(x) \in \mathscr{A}'\} \in \mathfrak{F} \tag{2.12}$$

for any $\mathscr{A}' \in \mathfrak{F}'$. We also use the notation $f^{-1}(\mathfrak{F}') := \left\{f^{-1}(\mathscr{A}') \,\middle|\, \mathscr{A}' \in \mathfrak{F}'\right\} \subseteq \mathfrak{F}$.

Definition 2.1.24 (Image measure). *Let $(\mathscr{S}, \mathfrak{S})$ and $(\mathscr{S}', \mathfrak{S}')$ be measurable spaces and let μ be a measure on $(\mathscr{S}, \mathfrak{S})$. Further, let $f\colon \mathscr{S} \to \mathscr{S}'$ be measurable. The* image measure *of μ under f is the measure $\left(\mu \circ f^{-1}\right)$ on $(\mathscr{S}', \mathfrak{S}')$ defined as*

$$\mu \circ f^{-1}\colon \mathfrak{S}' \to [0,\infty] : \mathscr{A}' \mapsto \left(\mu \circ f^{-1}\right)(\mathscr{A}'). \tag{2.13}$$

To define the Lebesgue integral, we introduce the notion of *simple functions* as a weighted sum of *characteristic functions*.

Definition 2.1.25 (Characteristic function). *The* characteristic function $\chi_{\mathscr{A}}\colon \mathscr{S} \to \{0,1\}$ *of an arbitrary set $\mathscr{A} \in 2^{\mathscr{S}}$ is the map*

$$\chi_{\mathscr{A}}(x) \coloneqq \begin{cases} 1 & \text{if } x \in \mathscr{A}, \\ 0 & \text{else.} \end{cases} \tag{2.14}$$

With this definition, we can formalize the notion of *simple functions*.

Definition 2.1.26 (Simple function). *Let $(\mathscr{S}, \mathfrak{S})$ be a measurable space. $f\colon \mathscr{S} \to \mathbb{R}^n$ is* simple *if there exists an $m \in \mathbb{N}$ and mutually disjoint measurable sets $\mathscr{A}_1, \dots, \mathscr{A}_m \in \mathfrak{S}$ as well as vectors $\alpha_1, \dots, \alpha_m \in \mathbb{R}^n$ such that*

$$f = \sum_{i=1}^{m} \alpha_i \chi_{\mathscr{A}_i}. \tag{2.15}$$

For functions $f, g\colon \mathscr{S} \to \mathbb{R}^m$ such that $(g(x))_i \leq (f(x))_i$ for all $i = 1, \dots, m$ and all $x \in \mathscr{S}$, we write $g \leq f$. Let $\mathbb{S}$ be the vector space of simple functions on $(\mathscr{S}, \mathfrak{S})$ and let $\mathbb{S}_+ = \{f \in \mathbb{S} \mid f \geq 0\}$. To construct the Lebesgue integral, we define the map

$$\begin{aligned} I\colon \mathbb{S}_+ &\to [0,\infty]^m : \\ \sum_{i=1}^{l} \alpha_i \chi_{\mathscr{A}_i} &\mapsto \sum_{i=1}^{l} \alpha_i \mu(\mathscr{A}_i) \end{aligned} \tag{2.16}$$

Armed with this object we can define the Lebesgue integral of nonnegative functions:

Definition 2.1.27 (Lebesgue integral). *Let $f\colon \mathscr{S} \to [0,\infty]^m$ be measurable. Its* Lebesgue integral *with respect to the measure μ is*

$$\int_{\mathscr{S}} f \,\mathrm{d}\mu \coloneqq \sup_{\{g \in \mathbb{S}_+ \mid g \leq f\}} I(g). \tag{2.17}$$

In this thesis, the "default" measure is the n-dimensional Lebesgue Measure $\mathfrak{L}^n$. If we integrate w.r.t. $\mathfrak{L}^n$, we sometimes omit it, for a measurable $f\colon \mathbb{R}^n \to [0,\infty]$ i.e. we write

$$\int_{\mathscr{S}} f \coloneqq \int_{\mathscr{S}} f \,\mathrm{d}\mathfrak{L}^n. \tag{2.18}$$

In the next definition, we define integrability for a broader set of functions.

Definition 2.1.28 (Integrability). *Let $(\mathscr{S}, \mathfrak{F}, \mu)$ be a measure space and $(\mathscr{S}', \|\cdot\|)$ be a Banach space where $\mathscr{S}' \subseteq \mathbb{R}^m$. A measurable function $f\colon \mathscr{S} \to \mathscr{S}'$ is μ-p-*integrable *if*

$$\int_{\mathscr{S}} \|f(x)\|^p \mathrm{d}\mu(x) < \infty. \tag{2.19}$$

*When this holds for $p = 1$, we call f simply μ-*integrable.

The space of all such functions is the *Lebesgue space*.

Definition 2.1.29 (Lebesgue space). *Let $(\mathscr{S}, \mathfrak{F}, \mu)$ be a measure space and let $(\mathscr{S}, \|\cdot\|)$ be a Banach space where $\mathscr{S}' \subseteq \mathbb{R}^m$. For $p \in [1, \infty]$*

$$L^p(\mathscr{S}, \mathfrak{F}, \mu) \coloneqq \left\{ f\colon \mathscr{S} \to \mathscr{S}' \;\middle|\; f \text{ } \mu\text{-measurable and } \|f\|_{L^p(\mathscr{S},\mathfrak{F},\mu)} < \infty \right\}, \tag{2.20}$$

is the Lebesgue space *with the norm defined as*

$$\|f\|_{L^p(\mathscr{S},\mathfrak{F},\mu)} \coloneqq \left(\int_{\mathscr{S}} \|f(x)\|^p \mathrm{d}\mu(x) \right)^{p^{-1}} \tag{2.21}$$

when p is finite and

$$\|f\|_{L^\infty(\mathscr{S},\mathfrak{F},\mu)} \coloneqq \inf_{N : N \subseteq \mathscr{S}, \mu(N)=0} \left(\sup_{x \in \mathscr{S} \setminus N} \|f(x)\| \right). \tag{2.22}$$

When μ is the Lebesgue measure and $\mathscr{S}'$ is $\mathbb{R}$ or $\mathbb{C}$, we write just $L^p(\mathscr{S})$.

To state the Radon-Nikodym derivative which will later link the cumulative distribution function with the density of a random variable, we first need the concept ob absolute continuity.

Definition 2.1.30 (Absolute continuity). *Let μ and ν be measures on a measurable space $(\mathscr{S}, \mathfrak{F})$. ν is* absolutely continuous *with respect to μ if for all $\mathscr{A} \in \mathfrak{F}$*

$$\mu(\mathscr{A}) = 0 \implies \nu(\mathscr{A}) = 0, \tag{2.23}$$

and we write $\nu \ll \mu$.

Theorem 2.1.3 (Radon-Nikodym). *Let μ and ν be σ-finite measures on a measurable space $(\mathscr{S}, \mathfrak{F})$. Then*

$$\nu \text{ has a density w.r.t. } \mu \iff \nu \ll \mu. \tag{2.24}$$

Then, the Radon-Nikodym derivative $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}$ *is $\mathfrak{F}$-measurable and finite μ-a.e.*

Proof. See [132, theorem 7.34] □

2.1.7 Convolutions

In this section, we introduce the *convolution*, a fundamental tool in image processing. We begin by defining the convolution of continuous signals, where we point out the translation equivariance property, and advance by discussing natural discretizations.

Definition 2.1.31 (Convolution). *Let $f, g\colon \mathbb{R}^d \to \mathbb{R}$ be measurable. The* convolution *of f with g is defined as*

$$(f * g)(x) = \int_{\mathbb{R}^d} f(y)g(x-y)\,\mathrm{d}y. \tag{2.25}$$

The convolution is obviously linear in both arguments. The translation equivariance is immediate: Let $t_y\colon \mathbb{R}^d \to \mathbb{R}^d : x \mapsto x+y$, then $(f*g)\circ t_y = (f*(g\circ t_y)) = ((f\circ t_y)*g)$. The extension of the definition of the convolution to measures is natural: Let μ be a measure on $\mathbb{R}^d$ and let $f\colon \mathbb{R}^d \to \mathbb{R}$ be measurable. Then, we define

$$(\mu * f)(x) = \int_{\mathbb{R}^d} f(x-y)\,\mathrm{d}\mu(y). \tag{2.26}$$

To derive the standard discretization of convolutions, we resort to piecewise constant interpolation. Since reasonable interpolation schemes are separable, in the following it suffices to consider one-dimensional signals. In detail, let $U\colon \mathbb{Z} \to \mathbb{R}$ be a discrete signal defined over all integers. From this, we construct a continuous signal $u\colon \mathbb{R} \to \mathbb{R}$ via

$$u(x) = \sum_{j=1}^{n} U_j \chi_{[-\frac{1}{2},\frac{1}{2})}(x-j), \tag{2.27}$$

and we use the shorthand $\phi \coloneqq \chi_{[-\frac{1}{2},\frac{1}{2})}$. With this, the definition of the convolution of continuous images becomes

$$\begin{aligned}(u * h)(k) &= \int_{\mathbb{R}} u(y)h(k-y)\,\mathrm{d}y \\ &= \int_{\mathbb{R}} \sum_{l\in\mathbb{Z}} U_l\phi(y-l) \sum_{m\in\mathbb{Z}} H_m\phi(k-y-m)\,\mathrm{d}y \\ &= \sum_{l\in\mathbb{Z}}\sum_{m\in\mathbb{Z}} U_l H_m \int_{\mathbb{R}} \phi(x)\phi(k-l-m-x)\,\mathrm{d}x.\end{aligned} \tag{2.28}$$

The integral has a nice form: We have that

$$\int_{\mathbb{R}} \phi(x)\phi(k-l-m-x)\,\mathrm{d}x = \begin{cases}1 & \text{if } m = k-l,\\ 0 & \text{else.}\end{cases} \tag{2.29}$$

Thus, we arrive at the concise expression

$$(u * h)(k) = \sum_{l\in\mathbb{Z}} U_l H_{k-l} \tag{2.30}$$

and therefore it is natural to define

$$(U * H)_k = \sum_{l \in \mathbb{Z}} U_l H_{k-l}. \tag{2.31}$$

Real signals, in particular images, have finite support and with the above definition we have to evaluate them at indices where they are not defined. To remedy this, a finite image $\tilde{U}\colon \{1, 2, \dots, n\} \to \mathbb{R}$ is extended to a signal $U\colon \mathbb{Z} \to \mathbb{R}$ over the integers. Many extensions are used in the literature—some of them have nice properties in particular applications. For instance, the *periodic* extension

$$U_i = \tilde{U}_{(i+(n-1) \bmod n)+1} \tag{2.32}$$

has the property that the convolution turns into a point-wise multiplication of the discrete Fourier transform of the inputs. Another widely used extension is constant padding where

$$U_i = \begin{cases} \tilde{U}_i & \text{if } 1 \le i \le n, \\ c & \text{else.} \end{cases} \tag{2.33}$$

2.1.8 The Fourier transform

In this thesis, we only use the two-dimensional discrete Fourier transform, in particular also for the two-dimensional discrete convolution theorem. Nevertheless, we start by introducing the Fourier transform in the Lebesgue space $L^1(\mathbb{R}^d)$.

Definition 2.1.32 (Fourier transform). *Let $u \in L^1(\mathbb{R}^d)$. The map*

$$F : u \mapsto \hat{u} = (2\pi)^{-\frac{d}{2}} \int_{\mathbb{R}^d} u(x) \exp(-\langle \mathrm{i}x, \cdot \rangle)\, \mathrm{d}x \tag{2.34}$$

is the Fourier transform. *$\hat{u}$ is the* Fourier transform of *u.*

From this definition, it follows immediately that the Fourier transform of a real-valued function obeys Hermitian symmetry. That is, when $u \in L^1(\mathbb{R}^d)$ is real-valued, $\overline{(Fu)(\xi)} = (Fu)(-\xi)$. Thus, for a real-valued signal, it suffices to store only half of the frequency plane.

2.1.9 The wavelet transform

In what follows, we briefly discuss the main concepts of the discrete wavelet transform needed for our purposes. For the sake of simplicity, we stick to the one-dimensional case but note that the extension to two dimensions is straight forward, see e.g. [30, chapter 4.4]. The following is largely adapted from [30], we refer the reader to this and [158, 237] for information on the extension to two-dimensional signals as well as efficient implementations using the fast wavelet transform. Let $\omega \in L^2(\mathbb{R})$ be a wavelet satisfying the admissibility condition

$$0 < \int_0^\infty \frac{|(F\omega)(\zeta)|^2}{\zeta}\, \mathrm{d}\zeta < \infty. \tag{2.35}$$

The set of functions

$$\left\{\omega_{j,k} = 2^{-j/2}\omega(2^{-j}\cdot - k) \,\middle|\, j,k \in \mathbb{Z}\right\} \tag{2.36}$$

forms an orthonormal basis of $L^2(\mathbb{R})$ under certain conditions that we now recall. Let $(V_j)_{j\in\mathbb{Z}}$ be a multiscale analysis with *generator* or *scaling* function $\phi \in V_0$, i.e. $\{T_k\phi \mid k \in \mathbb{Z}\}$ form an orthonormal basis of V_0 (T_k is a translation operator $(T_k\phi)(x) = \phi(x+k)$). The scaling property

$$u \in V_j \iff D_{1/2}u \in V_{j+1} \quad ((D_s u)(x) = u(sx)) \tag{2.37}$$

of the multiscale analysis $(V_j)_{j\in\mathbb{Z}}$ implies that the functions $\phi_{j,k} = 2^{-j/2}\phi(2^{-j}\cdot - k)$, $k \in \mathbb{Z}$ form an orthonormal basis of V_j. Further, the scaling property implies that $\phi \in V_{-1}$ and since $\phi_{-1,k}$ form an orthonormal basis of V_{-1}, we have that

$$\phi(x) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_k\phi(2x-k) \tag{2.38}$$

with $h_k = \langle\phi, \phi_{-1,k}\rangle_{L^2(\mathbb{R})}$. We define the *detail* or *wavelet spaces* W_j as the orthogonal complements of the *approximation spaces* V_j in V_{j-1}, i.e.

$$V_{j-1} = V_j \oplus W_j, \quad V_j \perp W_j. \tag{2.39}$$

From this follows that $V_j = \bigoplus_{m\geq j+1} W_m$ and due to the completeness of V_j, that $\bigoplus_{m\in\mathbb{Z}} W_m = L^2(\mathbb{R})$. By the orthogonality, we have that $\mathrm{proj}_{V_{j-1}} = \mathrm{proj}_{V_j} + \mathrm{proj}_{W_j}$ and hence $\mathrm{proj}_{W_j} = \mathrm{proj}_{V_{j-1}} - \mathrm{proj}_{V_j}$. Thus, any $u \in L^2(\mathbb{R})$ can be represented as

$$u = \sum_{j\in\mathbb{Z}} \mathrm{proj}_{W_j} u = \mathrm{proj}_{V_m} u + \sum_{j\leq m} \mathrm{proj}_{W_j} u \tag{2.40}$$

justifying the name multiscale analysis. Then (see [30, Theorem 4.67] for details) $\omega \in V_{-1}$ defined by

$$\omega(x) = \sqrt{2}\sum_{k\in\mathbb{Z}} (-1)^k h_{1-k}\phi(2x-k) \tag{2.41}$$

is a wavelet, $\{\omega_{j,k} \mid k \in \mathbb{Z}\}$ is an orthonormal basis of W_j and in particular the construction (2.36) is an orthonormal basis of $L^2(\mathbb{R})$.

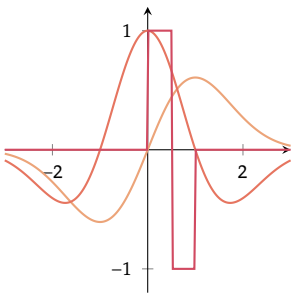

Figure 2.4: The derivative of Gaussian —, the Mexican hat —, and the Haar wavelet —.

2.1.10 The shearlet transform

We briefly describe our setup but refer the interested reader to [141, 146] for more details. We construct a digital shearlet system, specified by the positive scaling integer $j = 0, 1, \ldots, J$, and shearings $|k| \leq \lceil 2^{\lfloor \frac{J}{2} \rfloor} \rceil$. The system is constructed by a one-dimensional low-pass filter h_1 and a two-dimensional directional filter P. Given one-dimensional filters $h_{J-j/2}$ and g_{J-j} derived from h_1 in a wavelet multiresolution

analysis, let $W_j = g_{J-j} \otimes h_{J-j/2}$ and let p_j be the Fourier coefficients of P at scaling level j. Then, the system is constructed by

$$\gamma_{j,k} = \left[\left(S_k\left((p_j * W_j)_{\uparrow_{2^{j/2}}} *_1 h_{j/2}\right)\right) *_1 \overleftarrow{h}_{j/2}\right]_{\downarrow_{2^{j/2}}}. \tag{2.42}$$

Here, $\uparrow_a$ and $\downarrow_a$ are a-fold up- and down-sampling operators, $\overleftarrow{(\cdot)}$ indicates sequence reversal $\overleftarrow{(\cdot)}(n) = (\cdot)(-n)$, and S_k is a shearing operator. The digital shearlet transform of an image $x \in \mathbb{R}^{n\times n}$ is then given by

$$\lambda_{j,k}\overline{\gamma_{j,k}} * x \tag{2.43}$$

where $\lambda_{j,k} > 0$ are learnable weights that reflect the importance of the respective scale and shear level.

2.2 Probability theory

We use the measure theoretic concepts discussed in the previous section to rigorously define concepts in probability theory. This section is a summary from [132] and by no means exhaustive; it only serves to recall definitions that we need in the remainder of the thesis.

Modern probability theory relies on *probability spaces* $(\mathscr{S}, \mathfrak{F}, \mathbb{P})$—measure spaces equipped with a *probability measure* $\mathbb{P}$. In this context, we call the sets in the σ-algebra $\mathfrak{F}$ *events*. Typically, these events are not observable but encoded as a map from $\mathscr{S}$ to a space of outcomes. This measurable map is called the *random variable*, denoted as X in this section. The probabilities of the observations will be described by the *distribution* of the random variable, which is the image measure of $\mathbb{P}$ under X.

We begin by recalling the definition of a *probability measure*, which we already introduced in definition 2.1.19.

Definition 2.2.1 (Probability measure)**.** *Let $(\mathscr{S}, \mathfrak{F})$ be a measurable space and let $\mu\colon \mathfrak{F} \to [0, \infty]$ by a measure. μ is a* probability measure *if $\mu(\mathscr{S}) = 1$.*

As already hinted at in the previous paragraph, we use the symbol $\mathbb{P}$ to denote a probability measure. A *probability space* is a measurable space equipped with a probability measure.

Definition 2.2.2 (Probability space)**.** *Let $(\mathscr{S}, \mathfrak{F})$ be a measurable space and let $\mathbb{P}$ be a probability measure. The triple $(\mathscr{S}, \mathfrak{F}, \mathbb{P})$ is a* probability space*.*

We introduce *random variables* as measurable maps between arbitrary sets.

Definition 2.2.3 (Random variable)**.** *Let $(\mathscr{S}, \mathfrak{F})$ and $(\mathscr{S}', \mathfrak{F}')$ be measurable*

spaces and let $X\colon \mathscr{S} \to \mathscr{S}'$ *be measurable. Then,* X *is called a* random variable *with values in* $(\mathscr{S}', \mathfrak{F}')$.

For any $\mathscr{A}' \in \mathfrak{F}'$, we use the notation $\{X \in \mathscr{A}'\} \coloneqq X^{-1}(\mathscr{A}')$ to denote the pre-image of $\mathscr{A}'$ under X. In particular we also use the shorthand $\{X \geq a\} \coloneqq X^{-1}([a, \infty))$ or $\{X \leq b\} \coloneqq X^{-1}((-\infty, b])$. Thus, the probability that X takes on a value in the event $\mathscr{A}'$ is

$$\mathbb{P}\Big(\{X \in \mathscr{A}'\}\Big) = \int_{X^{-1}(\mathscr{A}')} \mathrm{d}\mathbb{P}. \tag{2.44}$$

A random variable defines a *distribution* on its image space as formalized in the next definition.

Definition 2.2.4 (Distribution)**.** *Let* X *be a random variable. The probability measure* $\mathbb{P}_X \coloneqq \mathbb{P} \circ X^{-1}$ *is called the* distribution *of* X*. We write* $X \sim \mathbb{P}_X$ *and say* X *has distribution* $\mathbb{P}_X$.

In particular, we also use the last notation if there is some measure $\mu = \mathbb{P}_X$, i.e. we write $X \sim \mu$ in this case. Thus, the distribution of a random variable is the *image measure* (definition 2.1.24) of $\mathbb{P}$ under X. Now, we develop the well-known notion of a *cumulative distribution function* from which we can derive the *density function* of a random variable.

Definition 2.2.5 (Cumulative distribution function)**.** *Let* X *be a random variable. The map* $x \mapsto \mathbb{P}\Big(\{X \leq x\}\Big)$ *from* $\mathscr{S}$ *to* $[0,1]$ *is the* cumulative distribution function *of* X.

The pre-image of the cumulative distribution function can be used to define quantiles:

Definition 2.2.6 (Quantile)**.** *Let* $F_X\colon \mathscr{S} \to [0,1]$ *be the distribution function of a random variable* X*. For* $q \in (0,1)$*, the* q-th quantile *of* X *is*

$$F_X{}^{-1}(q) \coloneqq \min\{x \in \mathscr{S} \mid F_X(x) \leq q\}. \tag{2.45}$$

Finally, we arrive at the familiar notion of a *density function*. In particular, if the cumulative distribution function can be represented by the Lebesgue integral (definition 2.1.27) with respect to some measure μ, the integrand is the *density function*. Thus, a *density function* is always defined with respect to a measure.

Definition 2.2.7 (Density function)**.** *Let* X *be a random variable with cumulative distribution function* $F_X\colon \mathbb{R}^n \to [0,1]$*. If* F_X *can be written as*

$$\begin{aligned} F_X(x) &= \int_{(-\infty, x]} p_X \,\mathrm{d}\mu \\ &\coloneqq \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_n} p_X(x_1, \dots, x_n) \,\mathrm{d}\mu(x_1, \dots, x_n) \end{aligned} \tag{2.46}$$

for all $x \in \mathbb{R}^n$ and some μ-integrable function $p_X \colon \mathbb{R}^n \to [0, \infty)$, then p_X is the density function *of the distribution function of X with respect to the measure μ.*

If a distribution $\mathbb{P}_X$ is absolutely continuous with respect to some measure μ, by the Radon-Nikodym theorem (theorem 2.1.3) there exists a density

$$p_X = \frac{\mathrm{d}\mathbb{P}_X}{\mathrm{d}\mu}. \tag{2.47}$$

In particular, when $\mathbb{P}_X \ll \mathfrak{L}^n$ we express the probability that the random variable X takes on a value in the event $\mathscr{A}'$ as

$$\mathbb{P}\big(\{X \in \mathscr{A}'\}\big) = \int_{X^{-1}(\mathscr{A}')} \mathrm{d}\mathbb{P} = \int_{\mathscr{A}'} p_X \tag{2.48}$$

where $p_X = \frac{\mathrm{d}\mathbb{P}_X}{\mathrm{d}\mathfrak{L}^n}$.

The *normal distribution* is ubiquitous in the natural sciences and used heavily throughout this thesis. The ubiquity stems from closure properties under addition and multiplication of random variables that are normally distributed. In addition, by the central limit theorem (CLT) [20, theorem 27.2] the distribution of the sum of n (properly normalized) random variables approaches the normal distribution as n approaches infinity.

Remark 1. *The class of distributions with similar closure properties are called* stable- *[20, 178, page 377] or* Pareto-Lévy *distributions [159] and there exists a generalization of the CLT to stable distributions [178, theorem 1.4]. The normal distribution is particularly interesting because it is the only stable distribution with defined and finite moments of any order.*

Definition 2.2.8 (Normal distribution)**.** *Let $m \in \mathbb{R}^n$ and Σ be a symmetric positive definite $n \times n$ matrix. Let X be a random variable with values in $\mathbb{R}^n$ satisfying*

$$\begin{aligned} &\mathbb{P}\big(\{X \leq x\}\big) \\ &= \det(2\pi\Sigma)^{-\frac{1}{2}} \int_{(-\infty, x]} \exp\Big(-\langle t - m, \Sigma^{-1}(t-m)\rangle/2\Big)\, \mathrm{d}\mathfrak{L}^n(t), \end{aligned} \tag{2.49}$$

*for all $x \in \mathbb{R}^n$. Then, $\mathbb{P}_X =: \mathscr{N}_{m,\Sigma}$ is the n-*dimensional normal distribution *with* mean *m and* covariance matrix *Σ. We say that X is* normally distributed*. The distribution $\mathscr{N}_{0, I_{\mathbb{R}^n}}$ is the* standard normal distribution*.*

By definition 2.2.7, it immediately follows that the density with respect to the Lebesgue measure of a normally distributed random variable is

$$p_X(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\Big(-\langle x - m, \Sigma^{-1}(x-m)\rangle/2\Big). \tag{2.50}$$

The *mean* and *covariance* of the normal distribution are its first and second *moment* respectively—see section 2.2.2. The density with respect to the Lebesgue measure of a normally distributed random variable on the real line with mean 1 and covariance 1 is shown in fig. 2.5.

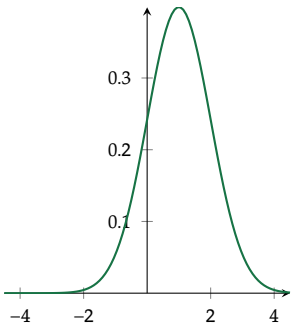

Figure 2.5: The density with respect to the Lebesgue measure of a one-dimensional normally distributed random variable with mean 1 and covariance 1.

2.2.1 Independence of random variables

A major part of this thesis is concerned about *learning* density functions from empirical distributions. These empirical distributions are, in essence, a collection of data points. Usually we assume that these data points are independent and identically distributed (i.i.d.), which leads to a beneficial structure of the learning problem. Without this assumption, the dependencies between the data points becomes complicated and the learning problem intractable. In this section, we introduce the concepts of independence of events and random variables. Throughout, $(\mathscr{S}, \mathfrak{F}, \mathbb{P})$ denotes a probability space and the elements of $\mathfrak{F}$ are called events. Often, the concrete probability space is unimportant and we do not explicitly state it. Then, the symbol $\mathbb{P}$ is "a" probability measure.

We start by defining the independence of a family of events. Intuitively, two events $\mathscr{A}_1'$ and $\mathscr{A}_2'$ are independent if the occurrence of $\mathscr{A}_1'$ does not change the probability of $\mathscr{A}_2'$ (and vice versa). Formally,

$$\mathscr{A}_1', \mathscr{A}_2' \text{ are independent} \iff \mathbb{P}(\mathscr{A}_1' \cap \mathscr{A}_2') = \mathbb{P}(\mathscr{A}_1')\mathbb{P}(\mathscr{A}_2'). \tag{2.51}$$

However, for three or more events $\mathscr{A}_1', \mathscr{A}_2', \mathscr{A}_3'$, pairwise independence is insufficient: It has to hold that

$$\mathbb{P}(\mathscr{A}_i' \cap \mathscr{A}_j') = \mathbb{P}(\mathscr{A}_i')\mathbb{P}(\mathscr{A}_j') \text{ for all } i, j = 1, 2, 3 \text{ and } i \neq j, \tag{2.52}$$

and that

$$\mathbb{P}(\mathscr{A}_1' \cap \mathscr{A}_2' \cap \mathscr{A}_3') = \mathbb{P}(\mathscr{A}_1')\mathbb{P}(\mathscr{A}_2')\mathbb{P}(\mathscr{A}_3'), \tag{2.53}$$

where neither implies the other. For a larger collection of events, analog conditions have to hold for all subcollections.

Definition 2.2.9 (Independence of events)**.** *Let $(\mathscr{A}_i')_{i \in \mathscr{I}}$ be an arbitrary collection of events with index set $\mathscr{I} \subset \mathbb{N}$. The collection is* independent *if for any finite subset $\mathscr{J} \subseteq \mathscr{I}$*

$$\mathbb{P}\Big(\bigcap_{j \in \mathscr{J}} \mathscr{A}_j'\Big) = \prod_{j \in \mathscr{J}} \mathbb{P}(\mathscr{A}_j'). \tag{2.54}$$

We will define independent random variables through the independence of their generated σ-algebras. Thus, we first define the independence between collections of events.

Definition 2.2.10 (Independence of classes of events)**.** *Let $(\mathfrak{E}_i)_{i \in \mathscr{I}}$ be an arbitrary collection of classes of events with index set $\mathscr{I} \subseteq \mathbb{N}$, where each $\mathfrak{E}_i \subseteq \mathfrak{F}$. The collection is* independent *if, for any finite subset $\mathscr{J} \subseteq \mathscr{I}$ and any choice $\mathscr{E}_j \in \mathfrak{E}_j$ where $j \in \mathscr{J}$*

$$\mathbb{P}\Big(\bigcap_{j \in \mathscr{J}} \mathscr{E}_j\Big) = \prod_{j \in \mathscr{J}} \mathbb{P}(\mathscr{E}_j). \tag{2.55}$$

To define independent random variables, let $\mathscr{I} \subseteq \mathbb{N}$ be an arbitrary index set and for each $i \in \mathscr{I}$ let $(\mathscr{S}_i, \mathfrak{F}_i)$ be a measurable space. In addition, we consider a collection of card $\mathscr{I}$ independent random variables $X_i \colon \mathscr{S} \to \mathscr{S}_i$.

Definition 2.2.11 (Independent random variables)**.** *The collection* $(X)_{i\in\mathscr{I}}$ *of random variables is* independent *if* $(X_i^{-1}(\mathfrak{F}_i))_{i\in\mathscr{I}}$ *is independent.*

This definition of independence is equivalent to the more familiar version: The collection $(X)_{i\in\mathscr{I}}$ is independent if and only if, for any finite $\mathscr{J} \subseteq \mathscr{I}$ and choice $\mathscr{A}_j' \in \mathfrak{F}_j$ where $j \in \mathscr{J}$,

$$\mathbb{P}\Big(\bigcap_{j\in\mathscr{J}}\{X \in \mathscr{A}_j'\}\Big) = \prod_{j\in\mathscr{J}} \mathbb{P}\big(\{X_j \in \mathscr{A}_j'\}\big). \tag{2.56}$$

We will give an additional characterization of the independence of random variables via their *joint distribution.*

Definition 2.2.12 (Joint distribution function, joint distribution)**.** *Let* $\mathscr{I} \subseteq \mathbb{N}$ *be an arbitrary finite index set and let* $(X_i)_{i\in\mathscr{I}}$ *be a collection of random variables. The map*

$$\begin{aligned} F_{\mathscr{I}} \coloneqq F_{(X_i)_{i\in\mathscr{I}}} \colon \bigtimes_{i\in\mathscr{I}} \mathscr{S}_i &\to [0,1] \\ x &\mapsto \mathbb{P}\Big(\bigcap_{i\in\mathscr{I}} X_i^{-1}\big((-\infty, x_j]\big)\Big) \end{aligned} \tag{2.57}$$

is the joint distribution function *of* $(X_i)_{i\in\mathscr{I}}$*. The probability measure* $\mathbb{P}_{(X_i)_{i\in\mathscr{I}}}$ *is the* joint distribution *of* $(X_i)_{i\in\mathscr{I}}$*.*

The following theorem states that the cumulative distribution function of independent random variables factorizes as the product of the individual distribution functions.

Theorem 2.2.1. *A collection* $(X_i)_{i\in\mathscr{I}}$ *of random variables is independent if and only if for every finite* $\mathscr{J} \subseteq \mathscr{I}$ *and every* $(x_j)_{j\in\mathscr{J}} \in \bigtimes_{j\in\mathscr{J}} \mathscr{S}_j$

$$F_{\mathscr{J}}(x) = \prod_{j\in\mathscr{J}} F_{X_j}(x_j). \tag{2.58}$$

Proof. See [132, Section 2.2, Theorem 2.21]. □

This theorem has the corollary that when the cumulative distribution function $F_{\mathscr{J}}$ has a continuous density $p_{\mathscr{J}} = p_{(X_j)_{j\in\mathscr{J}}}$, the *joint density*, then $(X_j)_{j\in\mathscr{J}}$ is independent if and only if

$$p_{\mathscr{J}}(x) = \prod_{j\in\mathscr{J}} p_X(x_j) \tag{2.59}$$

for all $(x_j)_{j\in\mathscr{J}} \in \bigtimes_{j\in\mathscr{J}} \mathscr{S}_j$. Finally, we introduce the notion of identically distributed random variables and combine it with independence to yield the concept of i.i.d. random variables.

Definition 2.2.13 (Identically distributed). *Let $\mathscr{I} \subseteq \mathbb{N}$ be an arbitrary index set and let $(X_i)_{i\in\mathscr{I}}$ be a collection of random variables. $(X_i)_{i\in\mathscr{I}}$ is called* identically distributed *if for all $i, j \in \mathscr{I}$*

$$\mathbb{P}_{X_i} = \mathbb{P}_{X_j}. \tag{2.60}$$

We say that a collection of random variables $(X_i)_{i\in\mathscr{I}}$ is *i.i.d.* if $(X_i)_{i\in\mathscr{I}}$ is independent and identically distributed.

2.2.2 Moments

In this subsection, $(\mathscr{S}, \mathfrak{F}, \mathbb{P})$ denotes a probability space.

Definition 2.2.14 (Expectation and mean). *Let X be a $\mathbb{P}$-integrable random variable. We call*

$$\mathbb{E}[X] := \int_{\mathscr{S}} X \,\mathrm{d}\mathbb{P} \tag{2.61}$$

the expectation *or* mean *of X. In addition, for an integrable function $f\colon \mathscr{S}' \to \mathscr{S}''$ we define*

$$\mathbb{E}_{X\sim\mathbb{P}_X}[f] := \mathbb{E}[f \circ X] = \int_{\mathscr{S}} f \circ X \,\mathrm{d}\mathbb{P}. \tag{2.62}$$

Note that if X has a density p_X with respect to the Lebesgue measure, then

$$\mathbb{E}[X] = \int_{\mathscr{S}} x p_X(x) \,\mathrm{d}\mathfrak{L}^n(x). \tag{2.63}$$

Another characteristic quantity of a random variable that we use in this thesis is the variance, which quantifies the deviation of a random variable from its expectation.

Definition 2.2.15 (Variance and standard deviation). *Let $X \in L^2(\mathscr{S}, \mathfrak{F}, \mathbb{P})$, then*

$$\mathrm{Var}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \tag{2.64}$$

is the variance *of X, and $\sqrt{\mathrm{Var}[X]}$ is the* standard deviation *of X.*

Finally, we define the *kurtosis* of a random variable, which (vaguely) relates to the tendency of its distribution to produce outliers.

Definition 2.2.16 (Kurtosis). *The* kurtosis *of a random variable X is defined as*

$$\frac{\mathbb{E}[(X - \mathbb{E}[X])^4]}{\mathbb{E}[(X - \mathbb{E}[X])^2]^2}. \tag{2.65}$$

The kurtosis of a normally distributed random variable is 3, regardless of its mean an variance. A classification of distributions based on kurtosis is usually done by the *excess kurtosis*—whether the kurtosis is larger or smaller than that of a normally distributed random variable:

Definition 2.2.17 (Excess kurtosis, platykurtic, mesokurtic, leptokurtic)**.** *The* excess kurtosis *of a random variable is three less than its kurtosis. A random variable is called*

- platykurtic *if its excess kurtosis is negative,*
- mesokurtic *if its excess kurtosis is zero, and*
- leptokurtic *if its excess kurtosis is positive.*

Figure 2.6 shows the densities of three distributions with increasing kurtosis: The uniform distribution over $[-1, 1]$ is highly platykurtic; it produces no "outliers". The normal distribution is mesokurtic by definition. The leptokurtic Laplace distribution with density

$$x \mapsto \frac{1}{2\sigma} \exp\Big(-\frac{|x-\mu|}{\sigma}\Big) \tag{2.66}$$

with $\sigma, \mu > 0$ has "fatter" tails; it has a tendency to produce more outliers than the normal distribution.

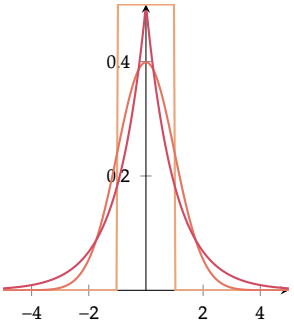

Figure 2.6: Three densities of distributions with increasing kurtosis: The uniform distribution —, the normal distribution —, and the Laplace distribution —.

2.2.3 Entropy and divergence measures

A significant portion of this thesis is concerned with *learning* distributions of random variables, i.e. (vaguely speaking) finding parameters of a parametrized distribution such that it is as close as possible to some reference. To make this meaningful, we require a measure of closeness of distributions. Here, we recall the measures that we use in this thesis; for a more general overview we refer to [148] for divergences based on the Shannon entropy (definition 2.2.18 below), to [82] for divergences based on the Rényi entropy, and to [66] for an overview of the usefulness of the general class of f-divergences in statistical inference. We begin by introducing the Shannon entropy.

Defining a general concept of entropy for all types of random variables is surprisingly difficult. For the sake of simplicity, we restrict ourselves here to continuous random variables that admit a density with respect to the Lebesgue measure.

Definition 2.2.18 (Differential Shannon entropy, [64, chapter 8])**.** *Let X be a random variable with values in $\mathscr{S}$ and density p_X. The* differential Shannon entropy *$H[p_X]$ is defined as*

$$H[p_X] = -\int_{\mathscr{S}} p_X(x) \log p_X(x)\, \mathrm{d}x. \tag{2.67}$$

The notation $H[p_X]$ reflects that the entropy of a random variable only depends on its density.

Based on the differential Shannon entropy, we can define the Kullback-Leibler divergence. We define the Kullback-Leibler divergence more generally on probability measures.

Definition 2.2.19 (Kullback-Leibler divergence, relative entropy, [64, section 8.5]). *Let $\mathbb{P}_X$ and $\mathbb{P}_Y$ be two probability measures (definition 2.2.1) on a measurable space (definition 2.1.14) $\mathscr{S}$, and let $\mathbb{P}_X$ be absolutely continuous (definition 2.1.30) with respect to $\mathbb{P}_Y$. Then, the* Kullback-Leibler divergence of $\mathbb{P}_X$ from $\mathbb{P}_Y$ *is*

$$(\mathbb{P}_X \,\|\, \mathbb{P}_Y)_{\mathrm{KL}} = \int_{\mathscr{S}} \log\Big(\frac{\mathbb{P}_X(\mathrm{d}x)}{\mathbb{P}_Y(\mathrm{d}x)}\Big)\mathbb{P}_X(\mathrm{d}x). \tag{2.68}$$

This is also called the relative entropy of $\mathbb{P}_X$ with respect to $\mathbb{P}_Y$.

Here, $\frac{\mathbb{P}_X(\mathrm{d}x)}{\mathbb{P}_Y(\mathrm{d}x)}$ is the Radon-Nikodym derivative (theorem 2.1.3) and we say "of $\mathbb{P}_X$ with respect to $\mathbb{P}_Y$" to emphasize the asymmetry.[1] Rewriting the definition as

$$(\mathbb{P}_X \,\|\, \mathbb{P}_Y)_{\mathrm{KL}} = \int_{\mathscr{S}} \frac{\mathbb{P}_X(\mathrm{d}x)}{\mathbb{P}_Y(\mathrm{d}x)} \log\Big(\frac{\mathbb{P}_X(\mathrm{d}x)}{\mathbb{P}_Y(\mathrm{d}x)}\Big)\mathbb{P}_Y(\mathrm{d}x) \tag{2.69}$$

makes the interpretation as a *relative entropy* more clear. In addition, it reveals that—in a measure theoretic context—the differential Shannon entropy can be seen as the relative entropy of a probability measure to the Lebesgue measure.[2] Now, letting μ be a measure on $\mathscr{S}$ with respect to which the distributions admit densities, i.e. $\mathbb{P}_X(\mathrm{d}x) = p_X(x)\mu(\mathrm{d}x)$ and $\mathbb{P}_Y(\mathrm{d}x) = p_Y(x)\mu(\mathrm{d}x)$, then

$$(\mathbb{P}_X \,\|\, \mathbb{P}_Y)_{\mathrm{KL}} = \int_{\mathscr{S}} p_X(x) \log\Big(\frac{p_X(x)}{p_Y(x)}\Big)\mu(\mathrm{d}x), \tag{2.70}$$

which reduces to the classical expression

$$(p_X \,\|\, p_Y)_{\mathrm{KL}} = \int_{\mathscr{S}} p_X(x) \log \frac{p_X(x)}{p_Y(x)}\,\mathrm{d}x \tag{2.71}$$

when μ is the Lebesgue measure (where we again directly insert the densities in the definition). Note that by absolute continuity, $p_Y(x) = 0$ implies that $p_X(x) = 0$ and motivated by continuity we set $0 \log \frac{0}{0} := 0$. We will see in section 5.3.2 that minimizing the Kullback-Leibler divergence is equivalent to maximizing the likelihood (of the data under the model).

Another divergence measure that has become popular recently is the Fisher divergence. For its definition, we again only consider random variables with values in $\mathscr{S}$ whose distributions admit densities with respect to the Lebesgue measure, although more general formulations are possible, see [180, equation 8].

Definition 2.2.20 (Fisher divergence). *Let X and Y bet two random variables with values in $\mathscr{S}$ and continuously differentiable densities p_X and p_Y. The* Fisher

1: The asymmetry is also why we speak about *divergence* measures, as opposed to distances or metrics. The Kullback-Leibler divergence is neither symmetric, nor does it obey the triangle inequality.

2: Clearly, the Lebesgue measure is *not* a probability measure. This is why we defined the differential Shannon entropy directly with respect to the density.

divergence of p_X from p_Y *is defined as*

$$(p_X \,\|\, p_Y)_{\mathrm{F}} = \int_{\mathscr{S}} p_X(x) \|\nabla \log p_X(x) - \nabla \log p_Y(x)\|^2 \,\mathrm{d}x. \tag{2.72}$$

This divergence has attracted significant attention in the deep learning literature in the recent years, since it has the very desirable property that it does not depend on the normalizing constant of p_Y. In other words, the definition remains valid when p_Y is only known up to a multiplicative constant.

The learning objective associated with the Fisher divergence is nowadays known as *score matching* and was introduced by Hyvärinen in 2005 [123]. Thus, the Fisher divergence is to score matching what the Kullback-Leibler divergence is to maximum likelihood.[3] Rewriting the Fisher divergence as

$$(p_X \,\|\, p_Y)_{\mathrm{F}} = \int_{\mathscr{S}} p_X(x) \left\| \nabla \log \frac{p_X(x)}{p_Y(x)} \right\|^2 \mathrm{d}x \tag{2.73}$$

reveals a similar form to the Kullback-Leibler divergence in eq. (2.71); for more connections between the two and a generalization of score matching we refer to [157]. We note that a variant of score matching called *denoising score matching*—introduced by Vincent in 2011 [238]—is extremely closely related to diffusion models: In the variance exploding formulation (see margin note 4), diffusion models learn the density at each diffusion time via a straight-forward application of denoising score matching.

3: And the Fisher information [143, chapter 2, eq. (5.10)] is to the Fisher divergence what the differential Shannon entropy is to the Kullback-Leibler divergence.

2.3 Differential equations and stochastic differential equations

In this section we first define differential equations and stochastic differential equations, which involve stochastic processes. We begin by discussing differential equations. The use of concepts from differential equations in this thesis is limited, and thus this overview is limited. We refer to [259] for a gentle introduction to differential equations.

Definition 2.3.1 (Differential equation)**.** *A* differential equation *is an equation containing the derivatives of an unknown function.*

In this thesis we consider differential equations that involve the derivative with respect to a one-dimensional parameter, which has the physical interpretation of "time". For the discussion of differential equations, it is helpful to explicitly name this parameter and we name it t. Consider some map $f : \mathbb{R}^n \times [0, \infty) \to [0, \infty)$. Here, the second argument would be explicitly named "t". An example of a differential equation is the *heat equation*, which using our notation reads

$$\frac{\partial f}{\partial t} = \Delta_1 f. \tag{2.74}$$

Here, the left hand side is the partial derivative w.r.t. the second argument (the explicitly named "time t"). The right hand side utilizes the Laplace operator which

maps a function to the sum of its second derivatives. The 1 in Δ_1 indicates the application to the first argument, i.e. $\Delta_1 f := \partial_1^2 f(\cdot, t') + \cdots + \partial_n^2 f(\cdot, t')$ is the Laplacian of the map $f(\cdot, t')\colon \mathbb{R}^n \to [0, \infty)$ for a fixed t'. The symbols ∂_i^2 for $i = 1, \dots, n$ denote the second partial derivative w.r.t. the i-th scalar variable in the first argument.

Definition 2.3.2 (Solution of a differential equation)**.** *A* solution *to a differential equation is a function, which when substituted into the differential equation reduces the equation to an identity. If a function is a solution to a differential equation, we also say it* obeys *the differential equation.*

2.3.1 Stochastic differential equations

Stochastic differential equations (SDEs) play a major role in this thesis: In chapter 5, we use a discretization of the Langevin diffusion (eq. (2.94)) to sample from a distribution whose density is given by a deep neural network. In chapter 6 we use a particular diffusion process (definition 2.3.11) that admits a connection between the density of the random variable representing the boundary condition and the density of the random variable undergoing the diffusion via Tweedie's formula (see section 6.2.2 and [195, 197]).

Informally, an SDE is a differential equation over random variables with values in $\mathbb{R}^n$, where at least one of the terms is a stochastic process. In more detail, random fluctuations are coded as an Itô integral with respect to a Brownian motion. To formalize this, we start by defining stochastic processes.

Definition 2.3.3 (Stochastic process)**.** *Let $I \subseteq \mathbb{R}$. A family of random variables $(X_t)_{t\in I}$ with values in $\mathscr{S}$ is called a* stochastic process *with index (or time) set I and range $\mathscr{S}$.*

The following definition gives a characterization of stochastic processes.

Definition 2.3.4 (Characterization of stochastic processes)**.** *A stochastic process $(X_t)_{t\in I}$ with range $\mathscr{S}$*

- *is* real-valued *if $\mathscr{S} = \mathbb{R}$.*
- *has* independent increments *if $(X_t)_{t\in I}$ is real-valued and for all $n \in \mathbb{N}$ and all $t_0, t_1, \dots, t_n \in I$ with $t_0 < t_1 < \dots < t_n$*

$$(X_{t_i} - X_{t_{i-1}})_{i=1,\dots,n} \text{ is independent.} \tag{2.75}$$

- *is a process with stationary increments if $(X_t)_{t\in I}$ is real-valued and the distribution of $(X_{s+t+r} - X_{t+r})$ is equal to the distribution of $(X_{s+r} - X_r)$ for all $r, s, t \in I$.*

In addition, an important subclass of stochastic processes are Markov processes (indeed all SDEs are Markov processes):

Definition 2.3.5 (Markov process [132, Definition 17.3])**.** *Let $I \subset [0,\infty)$ be an arbitrary index set that contains 0 and is closed under addition. Let $X = (X_t)_{t\in I}$ be a stochastic process with range $\mathscr{S}$. X is a* (time homogeneous) Markov process *with distributions $(\mathbb{P}_x)_{x\in\mathscr{S}}$ if:*

- *For every $x \in \mathscr{S}$, X is a stochastic process with $\mathbb{P}_x\Big(\{X_0 \in \{x\}\}\Big) = 1$.*
- *For every $\mathscr{A}' \in \mathfrak{B}(\mathscr{S})$, every $x \in \mathscr{S}$, and all $s, t \in I$*

$$\mathbb{P}_x\Big(\{X_{t+s} \in \mathscr{A}' \mid \sigma(X_r, r \le s)\}\Big) = \kappa_t(X_s, \mathscr{A}'). \tag{2.76}$$

Here the stochastic kernel or Markov kernel ([132, Definition 8.25]) $\kappa_t(X_s, \mathscr{A}') = \mathbb{P}_x\Big(\{X_t \in \mathscr{A}'\}\Big)$ are the transition probabilities.

For practical algorithms, usually only a discrete index set is interesting. In this case, without loss of generality we assume that the index set is $\{0, 1, ...\}$. Such Markov processes are called *Markov chains*.

Definition 2.3.6 (Markov chain)**.** *A Markov process with $I = \{0, 1, ...\}$ is called a* Markov chain*. Here, the existence of a Markov kernel $\kappa_1 \colon \mathscr{S} \times \mathfrak{B}(\mathscr{S}) \to [0, 1]$ implies the existence of a Markov chain [132, Theorem 17.11]: The n-*step transition probabilities *can be computed recursively via*

$$\kappa_n = \kappa_{n-1}\kappa_1 = \int_{\mathscr{S}} \kappa_{n-1}(\,\cdot\,, \mathrm{d}x)\kappa_1(x, \,\cdot\,). \tag{2.77}$$

Algorithms that approximate a distribution via a Markov chain are called MCMC algorithms (see for instance the unadjusted Langevin algorithm algorithm 1). An important characteristic of a Markov process are their *invariant distributions*. In essence, a distribution is invariant if it remains unchanged under application of the stochastic kernel of the Markov chain. It can be shown that (under mild conditions) the distribution of a Markov chain converges to an invariant distribution, see [132, chapter 18] and [166, Theorem 6.1]. The next definition provides more rigor for this concept:

Definition 2.3.7 (Invariant distribution [165, chapter 10, page 234])**.** *A measure μ on $\mathfrak{B}(\mathscr{S})$ with the property that*

$$\mu(\mathscr{S}') = \int_{\mathscr{S}} \kappa(x, \mathscr{S}')\, \mathrm{d}\mu(x) \tag{2.78}$$

is invariant *with respect to the stochastic kernel κ.*

In this thesis, we rely heavily on stochastic processes that can be described as SDEs that involve Brownian motion. Brownian motion is a stochastic process with certain characteristics, as shown in the following definition.

Definition 2.3.8 (Brownian motion). *A real-valued stochastic process* $B = (B_t)_{t\in[0,\infty)}$ *is called a* Brownian motion *if*

1. $B_0 = 0$.
2. B *has independent, stationary increments (definition 2.3.4).*
3. $B_t \sim \mathcal{N}_{0,t}$ *for all* $t > 0$.
4. *The map* $t \mapsto B_t$ *is* $\mathbb{P}$*-almost surely (definition 2.1.21) continuous.*

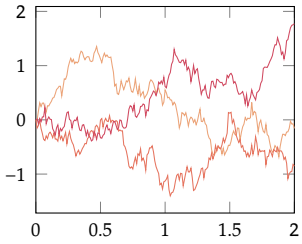

Figure 2.7: Three realizations of a Brownian motion.

We show three realizations of a Brownian motion in fig. 2.7.

The theory of stochastic differential equations heavily relies on *Itô integrals*. That is, a notion of an integral under which expressions like

$$I_t^B(H) = \int_0^t H_s \, \mathrm{d}B_s \tag{2.79}$$

where B is Brownian motion motion and $H_s \colon \Omega \to \mathbb{R}$ is a general integrand are well defined. Since a Brownian motion B is not the distribution function of a Lebesgue measure, such expressions are not well defined in the integration theory that we developed in this thesis. The idea behind the Itô construction is to first consider simple functions as integrands, for which the integral can be defined as a finite sum. Then, the integral can be extended to integrands that can are the limit of a certain sequence of simple integrands.

The following is a sketch of the Itô construction outlining the most important details. We refer to [132, chapter 25] for the technicalities.

Definition 2.3.9 (Predictable simple processes). *We denote with* $\mathscr{E}$ *the vector space of* predictable simple processes. *Elements of this space are maps* $H \colon \Omega \times [0,\infty) \to \mathbb{R}$ *of the form*

$$H(\omega, t) \coloneqq H_t(\omega) = \sum_{i=1}^{n} h_{i-1}(\omega)\chi_{(t_{i-1},t_i]} \tag{2.80}$$

where $n \in \mathbb{N}$, $0 = t_0 < t_1 < \ldots < t_n$ *and* h_{i-1} *is bounded and measurable for all* $i = 1, \ldots, n$. $\mathscr{E}$ *is endowed with the norm*

$$\|H\|_{\mathscr{E}}^2 = \sum_{i=1}^{n} \mathbb{E}\Big[h_{i-1}^2\Big](t_i - t_{i-1}) = \mathbb{E}\Big[\int_0^\infty H_s^2 \, \mathrm{d}s\Big]. \tag{2.81}$$

For these simple functions, we can define the integral immediately: For $H \in \mathscr{E}$ and $t \geq 0$, we define

$$I_t^W(H) \coloneqq \sum_{i=1}^{n} h_{i-1}(B_{\min(t_i,t)} - B_{\min(t_{i-1},t)}) \tag{2.82}$$

and

$$I_\infty^W(H) \coloneqq \sum_{i=1}^{n} h_{i-1}(B_{t_i} - B_{t_{i-1}}). \tag{2.83}$$

Now, we define a more general class of processes $\mathscr{E}_0$ and define the Itô integral as the continuous extension of the map I_∞^B to the closure of $\mathscr{E}$ in $\mathscr{E}_0$. In particular, we consider $\mathscr{E}$ as a subspace of

$$\mathscr{E}_0 := \left\{ H \,\middle|\, H \text{ progressively measurable and } \|H\|_{\mathscr{E}_0}^2 := \mathbb{E}\Big[\int_0^\infty H_t^2 \,\mathrm{d}t\Big] < \infty \right\} \tag{2.84}$$

and denote with $\overline{\mathscr{E}}$ the closure of $\mathscr{E}$ in $\mathscr{E}_0$. We do not give a precise definition of progressive measurability here—intuitively, it means that for a stochastic process $(X_t)_{t\in[0,\infty)}$, for every $t \geq 0$ the map $\Omega \times [0,t] \to E$, $(x,s) \mapsto X_s(x)$ is measureable with respect to the "standard" measures. For a more rigorous definition, see [132, definition 25.5 (ii)].

Definition 2.3.10 (Itô integral)**.** *For $H \in \overline{\mathscr{E}}$, we define the* Itô integral

$$\int_0^\infty H_t \,\mathrm{d}B_t \tag{2.85}$$

as the continuous extension of $I_\infty^B \colon \mathscr{E} \to L^2(\mathbb{P})$ from $\mathscr{E}$ to $\overline{\mathscr{E}}$.

Thus, if $(H^n)_{n\in\mathcal{N}}$ is a sequence in $\mathscr{E}$ with $\lim_{n\to\infty}\|H - H^n\|_{\mathscr{E}_0} \to 0$, then

$$\int_0^\infty H_t \,\mathrm{d}B_t = \lim_{n\to\infty} I_\infty^B(H^n). \tag{2.86}$$

Finally, we note that the condition $\mathbb{E}\big[\int_0^\infty H_t^2 \,\mathrm{d}t\big] < \infty$ can be readily weakened to $\mathbb{E}\big[\int_0^T H_t^2 \,\mathrm{d}t\big] < \infty$ for any $T > 0$.

2.3.2 Diffusion processes

The subclass of stochastic processes that can be expressed as Itô integrals with respect to a Brownian motion is called *diffusion processes*, defined more rigorously as follows:

Definition 2.3.11 (Diffusion process, proper diffusion)**.** *Let B be a Brownian motion and let f and g be progressively measurable stochastic processes such that $\int_0^t f_t^2 + |g_t| \,\mathrm{d}t < \infty$ almost surely for all $t \geq 0$. Then, the process X defined by*

$$X_t = \int_0^t f_s \,\mathrm{d}B_s + \int_0^t g_s \,\mathrm{d}s \tag{2.87}$$

for $t \geq 0$ is a generalized diffusion process *(or an* Itô process*) with* diffusion coefficient *f and* drift *g.*

If the maps f and g are of the form $f_t = \tilde{f}(X_t)$ and $g_t = \tilde{g}(X_t)$, for some

$\tilde{f}\colon \mathbb{R} \to [0,\infty)$ *and* $\tilde{g}\colon \mathbb{R} \to \mathbb{R}$, *then* X *is a* proper diffusion process.

The definition of Brownian motion in definition 2.3.8 and the subsequent discussion was restricted to real-valued stochastic processes. We now consider the extension to stochastic processes with values in $\mathbb{R}^n$ written in differential form as

$$\begin{aligned} X_0 &= Y \\ \mathrm{d}X_t &= f(X_t, t)\,\mathrm{d}B_t + g(X_t, t)\,\mathrm{d}t. \end{aligned} \tag{2.88}$$

Here, Y is an $\mathbb{R}^n$-valued random variable with distribution $\mathbb{P}_Y$, $f\colon \mathbb{R}^n \times [0,\infty) \to \mathbb{R}^{n\times m}$ is the matrix of diffusion coefficients, $B = (B^1, \dots, B^m)$ is an m-dimensional Brownian motion, and $g\colon \mathbb{R}^n \times [0,\infty) \to \mathbb{R}^n$ is the vector of drift coefficients. We assume that the entries in the matrix of diffusion coefficients and the vector of drift coefficients are measurable maps.

In analogy to definition 2.3.2, under a *solution* to eq. (2.88) we understand a stochastic process X with values in $\mathbb{R}^n$ such that eq. (2.88) reduces to an identity. Written in integral form, we seek a stochastic process X such that

$$X_t = Y + \int_0^t f(X_s, s)\,\mathrm{d}B_s + \int_0^t g(X_s, s)\,\mathrm{d}s. \tag{2.89}$$

This equality is understood component-wise and $\mathbb{P}$-almost surely for all $t \geq 0$. The following theorem ensures the existence of a unique (strong) solution of the stochastic differential equation eq. (2.88). For the classification of solutions as weak and strong, see [132, Definition 26.1]. Informally, a strong solution is measurable w.r.t. σ-algebras generated by Y and B (i.e. we do not have to extend the generated σ-algebras).

Theorem 2.3.1 (Unique strong solution)**.** *Let f and g be Lipschitz continuous in the first argument and assume that for all $x \in \mathbb{R}^n$ and $t \geq 0$*

$$\|f(x,t)\|^2 + \|g(x,t)\|^2 \leq K^2(1 + \|x\|^2). \tag{2.90}$$

Then, for every initial point $X_0 = x \in \mathbb{R}^n$ there exists a unique strong solution of eq. (2.88)*. This solution is a Markov process (definition 2.3.5).*

As an example (for the sake of simplicity for a real-valued stochastic process, i.e. $m = n = 1$), consider the stochastic differential equation

$$\begin{aligned} X_0 &= Y \\ \mathrm{d}X_t &= \alpha\,\mathrm{d}B_t + \beta X_t\,\mathrm{d}t. \end{aligned} \tag{2.91}$$

Here, both the diffusion coefficient and the drift coefficient are constants $f \equiv \alpha > 0$ and $g \equiv \beta \in \mathbb{R}$. The solution to this stochastic differential equation is the *Ornstein-Uhlenbeck process*

$$X_t := \exp(\beta t)Y + \alpha \int_0^t \exp((t-s)\beta)\,\mathrm{d}B_s \tag{2.92}$$

for $t \geq 0$. When $\beta = 0$, the stochastic process reduces to

$$X_t = Y + \alpha \int_0^t \mathrm{d}B_s. \tag{2.93}$$

Here, X_t for any $t > 0$ (informally) is the sum of the prescribed "boundary condition" Y and normally distributed perturbations. This stochastic process (and its associated SDE in eq. (2.91)) are interesting because the density of X_t inherits smoothness properties from the density of the normal distribution whose variance increases linearly with t.[4] We exploit this in chapter 6 where we learn the density of a random variable defined via eq. (2.93).

4: Hence, this stochastic differential equation is referred to as the *variance exploding SDE* in the literature on diffusion models [218, section 3.4].

Another diffusion process that plays a major role in this thesis is the *Langevin diffusion*

$$\begin{aligned} X_0 &= Y \\ \mathrm{d}X_t &= \mathrm{d}B_t + \frac{1}{2}\nabla \log p(X_t)\,\mathrm{d}t, \end{aligned} \tag{2.94}$$

where the diffusion coefficient is one, $f(x,t) = 1$, and the drift coefficient is the gradient of the logarithm of the density of a random variable Z, $g(x,t) = \frac{1}{2}\nabla \log p(x)$. In the seminal paper [198] Roberts and Tweedie show that the Langevin diffusion eq. (2.94) has invariant measure p and additionally the transition probabilities converge to p in the total variation norm. The following theorem is valid under a "standard" setup (random variables on $\mathbb{R}^n$ with Borel σ-algebra $\mathfrak{B}(\mathbb{R}^n)$ and densities with respect to the Lebesgue measure).

Theorem 2.3.2 (Convergence of Langevin diffusion)**.** *Let p be a positive density function such that $\log p$ is continuously differentiable. The Markov process defined by the Langevin diffusion eq. (2.94) has invariant distribution p and*

$$\sup_{\mathscr{S} \in \mathfrak{B}(\mathbb{R}^n)} \left| \mathbb{P}_x\big(\{X_t \in \mathscr{S}\}\big) - p(\mathscr{S}) \right| \to 0 \text{ as } t \to \infty \tag{2.95}$$

for all $x \in \mathbb{R}^n$.

Proof. See [198, theorem 2.1]. □

This construction is extremely useful as it allows sampling from distributions whose density is known up to a constant: If $p = \tilde{p}/c$ for a constant $c > 0$, then $\nabla \log p = \nabla \log \tilde{p}$. In this thesis, we exploit this to sample from a distribution whose unnormalized density is given by a neural network in chapter 5. However, for a practical implementation on a computer the Langevin diffusion eq. (2.94) needs to be discretized and we utilize the first-order Euler-Maruyama discretization. This scheme is simple to implement at the cost of a large discretization error; we refer to [91, chapter 6] for more elaborate and higher order schemes. Following [91, chapter 6], we denote with $\hat{X}$ the time-discrete approximation of eq. (2.94). Thus, we let $\hat{X}_0 = X_0$ and let $0 < t_0 < t_1 < \ldots$ and be arbitrary time points. The Euler-Maruyama method for solving SDEs is a straight-forward application of Euler's method for partial differential equations (PDEs) (see [106, chapter 1]) to the

stochastic case and has been popularized by Maruyama [162]. The approximation of the continuous time process reads

$$\hat{X}_{t_{i+1}} = \hat{X}_{t_i} + (t_{i+1} - t_i)\frac{1}{2}\nabla \log p(X_{t_i}) + \sqrt{t_{i+1} - t_i} Z_{i+1} \tag{2.96}$$

where $Z_1, Z_2, \ldots$ are independent and normally distributed random vectors on $\mathbb{R}^n$ with zero-mean and unit covariance. In a practical implementation, typically the time intervals are equispaced and we denote the spacing with $\tau > 0$. Then $t_i = 2i\tau$[5] and with the shorthand $\hat{X}_{i\tau} \coloneqq \hat{X}_i$ we can write the discrete-time Langevin diffusion as the Markov chain (definition 2.3.6)

5: The factor of 2 is arbitrary and just leads to a slightly nicer form in eq. (2.97).

$$\hat{X}_{i+1} = \hat{X}_i + \tau \nabla \log p(X_{t_i}) + \sqrt{2\tau} Z_{i+1}. \tag{2.97}$$

Thus, the Markov kernel (definition 2.3.6) of the discrete time Langevin diffusion is

$$\kappa_1(x, \cdot) = \mathcal{N}_{x+\tau\nabla \log p(x), 2\tau I} \tag{2.98}$$

The algorithm is compactly written in algorithm 1 in standard algorithmic notation, and is usually referred to as unadjusted Langevin algorithm (ULA). Here, *unadjusted* emphasizes that the algorithm is biased, which could be remedied by metropolization; see below.

Algorithm 1: Unadjusted Langevin algorithm

Input: Starting point $x^0 \in \mathbb{R}^n$ and step size sequence τ_k on $(0, \infty)$.

while *not converged* **do**
- $z_k \sim \mathcal{N}_{0,I}$
- $x^{k+1} = x^k - \tau^k \nabla \log p_X(x^k) + \sqrt{2\tau^k} z^k$
- $k = k + 1$

As with any general PDE or SDE, going from continuous time to discrete time introduces discretization errors. In particular, the Markov chain of the Langevin diffusion eq. (2.97) does not leave p invariant. This is easily demonstrated by the example taken from [198]: Let p be the density of a standard normal distribution on $\mathbb{R}$,

$$p(x) = \exp(-x^2/2)/\sqrt{2\pi}, \tag{2.99}$$

then $x - \nabla \log p(x) = 0$ for all $x \in \mathbb{R}$ and the Markov chain eq. (2.97) immediately converges[6] to a zero-mean normal distribution with variance 2τ. Thus, for almost all choices of τ the invariant distribution does not have density p.

6: Here, convergence is understood in the sense that all subsequent samples are from the same distribution.

In the example above, the immediate convergence of the chain (even to the correct distribution for $\tau = 1/2$) is due to the simplicity of the setup. Generally, a Markov chain starting from an arbitrary point requires many iterations to converge to the invariant distribution[7] *and* the invariant distribution does *not* have a density p. The iterations needed for a Markov chain to reach its invariant distribution are colloquially known as the *burn-in time*, see [31, section 1.11].

7: Here we assume that the Markov chain converges, but this is true under relatively mild conditions, see again [166, theorem 6.1].

The classical method to construct a Markov chain with an invariant distribution having density p is via *metropolization*. The Metropolis-Hastings algorithm,

popularized by Metropolis [164] and Hastings [110], endows any Markov chain with an accept or reject step. The acceptance rate is chosen to insure that the distribution with density p is left invariant. Metropolization a general tool applicable to arbitrary Markov chains (e.g. random walks). The metropolized version of the discrete Langevin diffusion is known as Metropolis adjusted Langevin algorithm (MALA) [198, section 1.4.2]. However, in practice, the correction often worsens the convergence properties (with respect to "wall time"), especially in high-dimensional settings, as empirically demonstrated in [79]. Therefore, metropolization is not used in this thesis.

A practical implementation also requires a finite stopping time. Recent years have seen extensive literature on the convergence properties of the discrete Langevin diffusion, both asymptotically and non-asymptotically. Dalalyan provided a first bound of the distribution obtained by eq. (2.97) in terms of the total variation (TV) distance, which Durmus and Moulines [78] improved and extended. They also provided bounds with respect to the Wasserstein distance of order 2 under the assumption of strong convexity of $-\log p$ [77], and Cheng and Bartlett prove bounds with respect to the Kullback-Leibler divergence (definition 2.2.19) under similar assumptions in [50].

In chapter 5 we use discrete Langevin diffusion on nondifferentiable densities where $-\log p$ is nonconvex. Pereyra [184] first extended discrete Langevin diffusion to nondifferentiable densities, later expanded by Durmus, Moulines, and Pereyra to composite problems [79], and by Luu, Fadili, and Cheseneau to nonconvex cases [156]. In these works, gradient steps on $-\log p$ in eq. (2.97) are replaced by proximal steps (definition 2.4.18) on $-\log p$, effectively sampling from a smoothed density. Habring, Holler and Pock [104] proposed replacing the gradient step with a subgradient step, but their analysis does not cover the nonconvex case. We utilize the discrete Langevin diffusion for a nondifferentiable and nonconvex potential. We are not aware of any results on convergence in this setting.

2.4 Optimization

Optimization is ubiquitous in the natural sciences. In physics, we often seek states that *minimize energy*. Similarly, machine learning problems aim to *minimize empirical risk*. Some argue that nature is governed by optimization processes: soap membranes form *minimal surfaces*, chemical reactions *minimize* the *energy* in atomic bonds. Consequently, *optimization* as a mathematical discipline has been extensively studied. This section reviews key optimization concepts used in this thesis.

Generally, optimization problems can be formalized as

$$\inf_{x \in C} f(x). \tag{2.100}$$

Here, the *constraint set* $C \subset \mathbb{V}$ includes all feasible solutions and $f \colon \mathbb{V} \to \overline{\mathbb{R}}$ is the *objective function*. The closure $\overline{\mathbb{R}} \coloneqq \mathbb{R} \cup \{-\infty, \infty\}$ of $\mathbb{R}$ (definition 2.1.4) allows functions to take the value ∞, useful for encoding constraints on the optimization

variable. Such functions are called *extended real-valued (e.r.v.)* functions (see [15, chapter 2]), and they are *proper* if they never take on the value $-\infty$. A classical example used in this thesis is the *indicator function* of a set.

Definition 2.4.1 (Indicator function)**.** *For any $C \subseteq \mathbb{V}$, the* indicator function *of C is the e.r.v. function*

$$\delta_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ \infty & \text{else.} \end{cases} \tag{2.101}$$

8: In finite dimensions, we can assume that $\mathbb{V} = \mathbb{R}^n$ for a positive integer n without loss of generality, since all finite dimensional vector spaces are isomorphic, see [12, theorem 3.70].

The vector space $\mathbb{V}$ represents any finite dimensional vector space, e.g. $\mathbb{R}^n$.[8] In this thesis, we focus on finding infima in optimization problems. Thus, $x^* \in C$ is a *globally optimal solution*, if it satisfies

$$f(x^*) \leq f(x) \text{ for all } x \in C. \tag{2.102}$$

We classify optimization problems based on characteristics of the constraint set and the objective function. An optimization problem is *unconstrained* if the constraint set equals the underlying space, $C = \mathbb{V}$. It is *constrained* if the constraint set is a proper subset of the underlying space, $C \subset \mathbb{V}$. The problem is *smooth* when the objective function is continuously differentiable. If the objective function is not continuously differentiable, the optimization problem is *nonsmooth*. This classification is useful because the optimal efficiency of numerical algorithms typically depends on the structure of the optimization problem.

In the remainder of this section, we develop the necessary optimization tools used in the main body of the thesis. We discuss optimality conditions, concepts from convex and nonconvex optimization, and stochastic optimization algorithms for large-sum problems.

2.4.1 Optimality conditions

In this section, the ordered pair $(\mathbb{V}, \|\cdot\|)$ denotes a Banach space (definition 2.1.8) over the field $\mathbb{R}$.

The optimization algorithms that we develop aim to find globally or locally optimal points. We begin by defining these concepts more rigorously, focusing on minimization problems. Definitions for maxima follow analogously by reversing inequalities.

Definition 2.4.2 (Global minimum)**.** *Let $C \subseteq \mathbb{V}$ and let $f\colon C \to \overline{\mathbb{R}}$. We call*

- *$x^* \in C$ a* global minimum *of f over C if $f(x^*) \leq f(x)$ for all $x \in C$.*
- *$x^* \in C$ a* strict global minimum *of f over C if $f(x^*) < f(x)$ for all $x \in C, x \neq x^*$.*

Analogously, when these inequalities only hold in a local neighborhood around the minimum:

Definition 2.4.3 (Local minimum). *Let $C \subseteq \mathbb{V}$ and let $f\colon C \to \overline{\mathbb{R}}$. We call*

- *$x^* \in C$ a* local minimum *of f over C if there exists an $\epsilon > 0$ such that $f(x^*) \leq f(x)$ for all $x \in C \cap \mathscr{B}_{\|\cdot\|}(x^*, \epsilon)$.*
- *$x^* \in C$ a* strict local minimum *of f over C if $f(x^*) < f(x)$ for all $x \in C \cap \mathscr{B}_{\|\cdot\|}(x^*, \epsilon), x \neq x^*$.*

To illustrate these concepts, consider the function from $\mathbb{R}$ to $\mathbb{R}$ defined by

$$x \mapsto \begin{cases} (x-0.5)^2 + 1 & \text{if } x \leq 0, \\ 1 & \text{if } x \in (0,1], \\ x & \text{if } x \in (1,2], \\ -4x + 8 & \text{if } x \in (2, 2.5], \\ 4x - 10 & \text{if } x > 2.5. \end{cases} \tag{2.103}$$

It has a global minimum at 2 and infinitely local minima located in the interval $[0.5, 1]$. This function along with its set of minimizers is plotted in fig. 2.8. By these definitions, any global minimum is also a local minimum, and multiple global minima are possible. For instance, the function from $\mathbb{R}$ to $\mathbb{R}$

$$x \mapsto \begin{cases} 0 & \text{if } x \in \{-1, 1\}, \\ 1 & \text{else,} \end{cases} \tag{2.104}$$

has two global minima at -1 and 1.

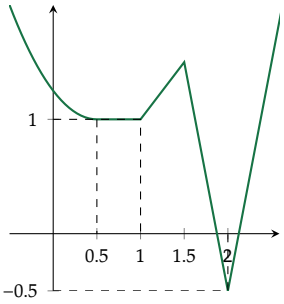

Figure 2.8: Illustration of local and global minima.

Most optimization problems in machine learning and imaging are solved using gradient-based methods. For completeness, we briefly mention alternative optimization paradigms:[9] Gradient-free optimization methods include those based on approximating the gradient (e.g. via finite differences), simulated annealing, genetic algorithms, grid search, multilevel coordinate search, and the Nelder-Mead algorithm. However, in our applications where the optimization variable has millions of dimensions, these methods are impractical and we focus in gradient-based methods.

To develop the optimization algorithms used in this thesis, we first define the gradient of a function. Let $f\colon C \to \overline{R}$ where C is a subset of a finite dimensional vector space. For simplicity and without loss of generality (see [12, theorem 3.7]) we assume this vector space is $\mathbb{R}^n$ for some positive integer n. The gradient of f, denoted by ∇f, is a map from C to $\overline{\mathbb{R}^n}$ given by

$$\nabla f = \begin{pmatrix} \partial_1 f \\ \vdots \\ \partial_n f \end{pmatrix} \tag{2.105}$$

where ∂_i is the partial derivative with respect to the i-th scalar argument.

Geometrically, the gradient $\nabla f(x)$ points in the direction of the largest increase of f at a point $x \in C$. Another interpretation, particularly in function approximations and Taylor series, is that it locally provides the best linear approximation of

9: Although classical definitions do not, we include subgradient-type methods as well as methods based on second derivatives in the class of *gradient-based methods.*

the original function. The first-order Taylor expansion of f around a point $a \in C$ is

$$x \mapsto f(a) + \langle \nabla f(a), x - a \rangle, \tag{2.106}$$

see also the illustration in fig. 2.12.

The gradient (and its generalizations, see section 2.4.3) is crucial in characterizing locally optimal points, as captured by Fermat's theorem.

Theorem 2.4.1 (Fermat's theorem on stationary points). *Let $f \colon C \to \overline{R}$ be proper, defined over a set $C \subseteq \mathbb{V}$, and differentiable. If a point $x^* \in \operatorname{int} C$ is locally optimal, then $\nabla f(x^*) = 0_{\mathbb{V}}$.*

Proof. See [15, theorem 3.72], where the role of the proper convex function is played by the indicator function (definition 2.4.1) of C. □

Figure 2.9 illustrates Fermat's theorem by demonstrating that the best linear approximation of function at a stationary point is a constant function.

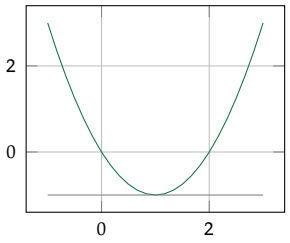

Figure 2.9: Graphical interpretation of Fermat's theorem on stationary points. At a stationary point (here, 1), the best linear approximation to the green function is a constant function.

2.4.2 Convex optimization

Convex optimization involves optimizing a convex objective function over a convex constraint set. This class of optimization problems allows for fast optimization algorithm with convergence guarantees to a global minimum. Here, we review concepts from convex optimization used in this thesis, some of which are also useful in nonconvex optimization.

First, we define convex sets, and identify convex functions through the convexity of the *epigraph*.

Definition 2.4.4 (Convex set). *A set $C \subseteq \mathbb{V}$ is* convex *if for any $x, y \in C$ and $\alpha \in [0,1]$*

$$\alpha x + (1 - \alpha) y \in C.$$

In other words, a set is convex if it contains all lines connecting any two of its elements, as illustrated in fig. 2.10 Examples of convex sets include hyperplanes $\{x \in \mathbb{V} \mid \langle a, x \rangle_{\mathbb{V}} = b\}$, halfspaces $\{x \in \mathbb{V} \mid \langle a, x \rangle_{\mathbb{V}} \leq b\}$, and norm balls $\mathscr{B}_{\|\cdot\|}(a, b)$, where a is an arbitrary element of the vector space $\mathbb{V}$ and b is a real number.

The second ingredient of convex optimization are *convex functions*.

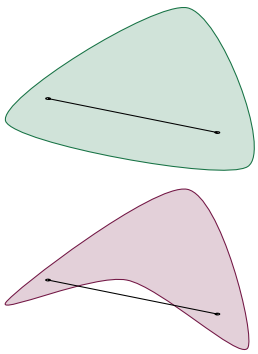

Figure 2.10: An example of a convex (top) and a nonconvex (bottom) subset of $\mathbb{R}^2$.

Definition 2.4.5 (Convex functions). *A function $f \colon C \to \overline{\mathbb{R}}$ defined over a convex set C is* convex *if for all $x, y \in C$ and $\alpha \in [0,1]$*

$$f(\alpha x + (1 - \alpha) y) \leq \alpha f(x) + (1 - \alpha) f(y).$$

In other words, a function is convex if it is always below the secant line connecting any two points on the graph. This definition is equivalent to a convex function having a convex epigraph.

Theorem 2.4.2 (Convexity of epigraph). *A function $f\colon C \to \overline{\mathbb{R}}$ defined over a set $C \subseteq \mathbb{V}$ is convex if and only if its epigraph* $\operatorname{epi} f := \{(x, y) \in \mathbb{V} \times \mathbb{R} \mid f(x) \leq y\}$ *is convex.*

A prototypical example of convex functions are norms, which play an important role in optimization as they induce a distance metric on a space.

Proposition 2.4.1. *Norms are convex.*

Proof. Let $\|\cdot\|$ be any norm. Then, for any $x, y \in \mathbb{V}$ and $\alpha \in [0, 1]$

$$\|\alpha x + (1-\alpha)y\| \leq \|\alpha x\| + \|(1-\alpha)y\| = \alpha\|x\| + (1-\alpha)\|y\|.$$

The first inequality is the triangle inequality applied to the points αx and $(1-\alpha)y$. The second equality follows from the absolute homogeneity of norms and the fact that α and $(1-\alpha)$ are greater than or equal to zero. □

We frequently require a linear combination of $n \in \mathbb{N}$ object with non-negative weights that sum to 1. This is ubiquitous in probabilistic contexts, as it allows to combine normalized densities such that the resulting object is also normalized. For this, we define the n-dimensional unit simplex.

Definition 2.4.6 (Unit simplex). *Let n be a natural number. The* n-dimensional unit simplex, *denoted by Δ^n, is the set*

$$\Delta^n := \left\{ x \in \mathbb{R}^n \;\middle|\; \sum_{i=1}^{n} x_i = 1, x_i \geq 0 \text{ for } i = 1, 2, \dots, n \right\}$$

The n-dimensional unit simplex is a $(n-1)$-dimensional manifold embedded in $\mathbb{R}^n$. Some authors also refer to this object as the $(n-1)$-dimensional simplex, or the *probability simplex* as it resembles the structure of a probability density.

Definition 2.4.5 states that a function is convex if the function value at the convex combination of two points is less than or equal to the convex combination of the function values at the two points. Jensen's inequality generalizes this to an arbitrary number of points.

Theorem 2.4.3 (Jensens inequality). *Let $f\colon C \to \overline{\mathbb{R}}$ convex and let $n \in \mathbb{N}$. Then, for any $x_1, \dots, x_m$ and $\alpha \in \Delta^n$*

$$f\left(\sum_{i=1}^{n} \alpha_i x_i\right) \leq \sum_{i=1}^{n} \alpha_i f(x_i).$$

Definition 2.4.5 requires no smoothness assumptions. The two following theorems establish convexity under stronger smoothness assumptions.

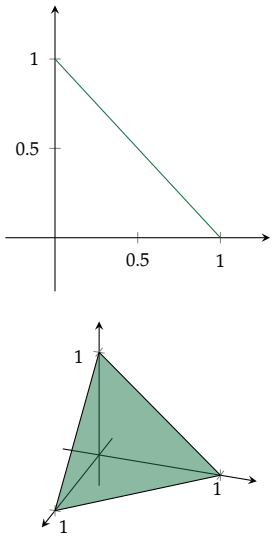

Figure 2.11: The unit simplex in two (top) and three (bottom) dimensions.

Theorem 2.4.4 (First order condition of convexity). *Let $C \subseteq \mathbb{V}$ be convex and let $f : C \to \overline{R}$ be continuously differentiable. Then, f is convex if and only if for all $x, y \in C$*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle.$$

We refer to the above inequality as the gradient inequality.

Proof. See [27, Section 3.1.3, Proof of first-order convexity condition]. □

In other words, the linearization of a differentiable convex function at any point provides a global lower bound. This is illustrated in fig. 2.12.

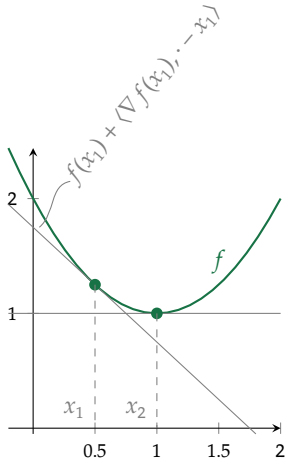

Figure 2.12: Convex functions are globally lower bound by their linearization around any point.

Theorem 2.4.5 (Second order condition of convexity). *Let $C \subseteq \mathbb{V}$ be open and convex and let $f : C \to \overline{R}$ be twice continuously differentiable. Then, f is convex if and only if for all $x, y \in C$*

$$\langle y, \nabla^2 f(x) y \rangle \geq 0.$$

Here, $\nabla^2 f$ denotes the Hessian of f.

2.4.3 Subgradients

To optimize non-smooth convex functions, we need to extend the concept of a gradient. The extension is based on the geometric interpretation of the first order condition of convexity theorem 2.4.4: Consider the absolute value function $|\cdot| : \mathbb{R} \to \mathbb{R} : x \mapsto \max(x, -x)$. At zero, there exists a family of tangent lines that provide a global lower bound—any of $x \mapsto \alpha x$ where $\alpha \in [-1, 1]$. We call $\alpha \in [0, 1]$ a *subgradient* of $|\cdot|$ at zero. This idea is formalized in the following definition.

Definition 2.4.7 (Subgradient). *Let $f : \mathbb{V} \to \overline{\mathbb{R}}$ be proper and let $x \in \operatorname{dom} f$. $g \in \mathbb{V}^*$ is a* subgradient *of f at x if for all $y \in \mathbb{V}$*

$$f(y) \geq f(x) + \langle g, y - x \rangle.$$

We refer to the above inequality as the subgradient inequality.

A subgradient g is an element of the *dual space* $\mathbb{V}^*$ of $\mathbb{V}$ (see definition 2.4.12), meaning it is a linear map. By Riesz' representation theorem, any linear functional f on $\mathbb{V}$ can be identified by a vector in $\mathbb{V}$, such that $f(x) = \langle g, x \rangle$, see [13, theorem 1.9]. With slight abuse of notation, we directly "identify" this linear functional via the vector g.

The only difference between $\mathbb{V}$ and $\mathbb{V}^*$ is the norm:

Definition 2.4.8 (Dual norm). *Let $\mathbb{V}$ be endowed with a norm $\|\cdot\|$. The dual*

space $\mathbb{V}^*$ of V is endowed with the norm

$$\|y\|_* = \max_{\|x\| \leq 1} \langle y, x \rangle. \tag{2.107}$$

For non-smooth convex functions, there may be infinitely many subgradients at a particular point. The set of all subgradients forms the subdifferential.

Definition 2.4.9 (Subdifferential)**.** *Let $f \colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper and let $x \in \operatorname{dom} f$. The set of all subgradients of f at x is the* subdifferential *of f at x. We denote the subdifferential of f at x with*

$$\partial f(x) \coloneqq \left\{ g \in \mathbb{V}^* \;\middle|\; f(y) \geq f(x) + \langle g, y - x \rangle \text{ for all } y \in \mathbb{V} \right\}.$$

A function f is subdifferentiable at x *if $\partial f(x)$ is nonempty. A function f is* subdifferentiable *if $\partial f(x)$ is nonempty for all $x \in \mathbb{V}$.*

In fig. 2.13 we illustrate this concept on the function f from $\mathbb{R}$ to $\mathbb{R}$ given by

$$x \mapsto \begin{cases} x^2 & \text{if } x \leq 0, \\ x & \text{else.} \end{cases} \tag{2.108}$$

All functions $x \mapsto \alpha x$ where $\alpha \in [0, 1]$ are global lower bounds and tangent at 0. Hence, $\partial f(0) = [0, 1]$.

The subdifferential generalizes the gradient for convex functions: if a proper convex function f is differentiable at a point x, then the subdifferential is the singleton containing the gradient, i.e. $\partial f(x) = \{\nabla f(x)\}$. However, a differentiable nonconvex function is not necessarily subdifferentiable; the Clarke subgradient definition 2.4.19 provides a generalization for nonconvex functions.

With the subdifferential, we can establish more general optimality condition for convex functions:

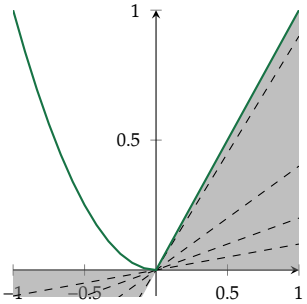

Figure 2.13: Illustration of the subdifferential. All black dashed functions are global lower bounds of and tangent to the green function at zero. The shaded area is the span of all such functions.

Theorem 2.4.6 (Optimality condition for convex optimization)**.** *Let $f \colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper and convex. Then*

$$x^* \in \arg\min_{x \in \mathbb{V}} f(x) \iff 0_{\mathbb{V}} \in \partial f(x^*).$$

Proof. The theorem follows immediately from the subgradient inequality under the choice $g = 0_{\mathbb{V}}$. □

2.4.4 Dual space

Definition 2.4.10 (Linear transform)**.** *Let $\mathbb{V}$ and $\mathbb{W}$ be two vector spaces over a field $\mathbb{K}$. A function $f \colon \mathbb{V} \to \mathbb{W}$ is a* linear transform *if for all $x, y \in \mathbb{V}$ and*

$\alpha, \beta \in \mathbb{K}$

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y).$$

Remark 2. *We frequently omit parenthesis between linear transforms and their argument. This is to emphasize that linear maps between finite dimensional vector spaces over the real numbers resemble matrix-vector multiplication, where this is standard notation: Since all n-dimensional vector spaces over $\mathbb{R}$ are isomorphic to $\mathbb{R}^n$ [12, theorem 3.70], any linear transform could be written as matrix-vector multiplication via an isomorphism.*

Definition 2.4.11 (Linear functional)**.** *A* linear functional *on a vector space $\mathbb{V}$ over a field $\mathbb{K}$ is a linear transform that maps from $\mathbb{V}$ to $\mathbb{K}$.*

Definition 2.4.12 (Dual space)**.** *The* dual space *is the set of all linear functionals on a vector space $\mathbb{V}$.*

We denote the dual space by an asterisk, i.e. the dual space of a vector space $\mathbb{V}$ is denoted by $\mathbb{V}^*$.

In this thesis, we frequently make use of the adjoint transform, that we now define.

Definition 2.4.13 (Adjoint transform)**.** *Let $(\mathbb{V}, \langle \cdot, \cdot \rangle_{\mathbb{V}})$ and $(\mathbb{W}, \langle \cdot, \cdot \rangle_{\mathbb{W}})$ be two Hilbert spaces and let $A \colon \mathbb{V} \to \mathbb{W}$ be a linear transform. The* adjoint linear transform *$A^* \colon \mathbb{W}^* \to \mathbb{V}^*$ is the unique linear transform satisfying*

$$\langle Ax, y \rangle_{\mathbb{W}} = \langle x, A^* y \rangle_{\mathbb{V}}$$

for all $x \in \mathbb{V}$ and $y \in \mathbb{W}^*$.

2.4.5 Convex conjugate

The convex conjugate plays an important role in finding efficient algorithms for convex optimization. The following provides a formal definition of the convex conjugate:

Definition 2.4.14 (Convex conjugate)**.** *Let $f \colon \mathbb{V} \to \overline{\mathbb{R}}$. The map $f^* \colon \mathbb{V}^* \to \overline{\mathbb{R}}$ defined as*

$$f^* = \sup_{x \in \mathbb{V}} \langle \cdot, x \rangle - f(x)$$

is the convex conjugate *of f.*

This definition does not make any assumptions on convexity or closedness. However, all functions constructed by this conjugation are closed and convex: It is defined as the supremum over affine functions, which are themselves closed and convex. Supremization preserves convexity:

Theorem 2.4.7 (Maximization preserves convexity)**.** *Let $f_1, \dots, f_n$ be convex functions over the convex set C. Then,*

$$f(x) = \max_{i=1,\dots,n} f_i(x)$$

is convex over C.

Proof.

$$\begin{aligned} f(\lambda x + (1-\lambda)y) &= \max_{i=1,\dots,n} f_i(\lambda x + (1-\lambda)y) \\ &\leq \max_{i=1,\dots,n} \{\lambda f_i(x) + (1-\lambda) f_i(y)\} \\ &\leq \lambda \max_{i=1,\dots,n} f_i(x) + (1-\lambda) \max_{i=1,\dots,n} f_i(y) \\ &= \lambda f(x) + (1-\lambda) f(y). \end{aligned}$$

□

Showing that the convex conjugate of any function is closed and convex is a straight-forward application of the above theorem:

Theorem 2.4.8 (Closednes and convexity of convex conjugate)**.** *The convex conjugate of any extended real-valued function is closed and convex.*

Proof. The convex conjugate is defined as the point-wise supremum over affine functions. Since affine functions are closed and convex and convexity is preserved under maximization, the convex conjugate is a closed and convex function. □

In addition, convex conjugation preserves properness.

Theorem 2.4.9 (Properness of convex conjugate)**.** *The convex conjugate of a proper function is proper.*

Proof. See [15, theorem 4.5] □

We conclude the discussion about convex conjugates with a geometric interpretation and its relation to the gradient: Assume that $f\colon \mathbb{V} \to \overline{\mathbb{R}}$ is continuously differentiable. At any point $y \in \mathbb{V}^*$, the convex conjugate of f is $f^*(y) = \sup_{x\in\mathbb{V}} \langle y, x\rangle - f(x)$. By employing first-order optimality conditions, we obtain

$$y - \nabla f(x^*) = 0 \implies y = \nabla f(x^*). \tag{2.109}$$

Thus, if f is differentiable, we are interested in finding the point $x^* \in \mathbb{V}$ such that the gradient of f at x^* equals y and where the difference $\langle y, x^*\rangle - f(x^*)$ is maximal. Further, by translating the plane $\langle y, x^*\rangle$ such that it is tangent to f at x^*, the (negative of the) value of the convex conjugate can be read off at zero. This is illustrated in fig. 2.14.

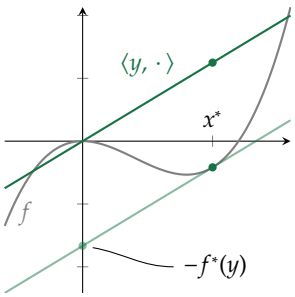

Figure 2.14: Illustration of the convex conjugate and its relation to the gradient of smooth functions.The point x^* is where the difference between the linear function $\langle y, \cdot\rangle$ and f is the largest.

Many optimization problems that arise in imaging applications include norms, as they are a natural choice to impose a bound on or sparsity in a vector. The next theorem establishes the convex conjugate of any norm.

Theorem 2.4.10 (Convex conjugate of a norm). *The convex conjugate of any norm is the indicator function of the closed unit dual norm ball:*

$$(\|\cdot\|)^*(x) = \delta_{\overline{\mathscr{B}_{\|\cdot\|_*}(0,1)}}(x) = \begin{cases} 0 & \text{if } \|\cdot\|_* \leq 1, \\ \infty & \text{else.} \end{cases} \tag{2.110}$$

Proof. See [15, section 4.4.12]. □

We show an example of this in fig. 2.15 using the one-norm on $\mathbb{R}^2$.

Many optimization algorithms rely on the fact that for proper, closed, and convex functions, conjugating twice yields the original function. This is formalized by the following definitions and theorems surrounding the biconjugate.

Definition 2.4.15 (Biconjugate). *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$. The* biconjugate $f^{**}\colon \mathbb{V} \to \overline{\mathbb{R}}$ *is defined as*

$$f^{**} = \sup_{y \in \mathbb{V}^*} \langle \cdot, y \rangle - f^*(y).$$

The biconjugate is closed an convex.

In addition, it provides a global lower bound on the associated function:

Lemma 2.4.1. *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$. Then, $f(x) \geq f^{**}(x)$ for all $x \in \mathbb{V}$.*

Proof. From the definition of the convex conjugate, we have for any $x \in \mathbb{V}$ and $y \in \mathbb{V}^*$ that $f^*(y) \geq \langle y, x \rangle - f(x)$ and thus $f(x) \geq \langle y, x \rangle - f^*(y)$. This further implies that $f(x) \geq \sup_{y \in \mathbb{V}^*} \langle y, x \rangle - f^*(y) = f^{**}(x)$. □

Finally, if we apply biconjugation to a proper, closed, and convex function, the resulting biconjugate is equal to the original function.

Theorem 2.4.11. *A proper, closed, and convex function is its own biconjugate.*

Proof. See [201, page 104]. □

The following theorem provides the convex conjugate of a scaled function which we need in later chapters.

Theorem 2.4.12 (Convex conjugate under positive scaling). *Let $g : \mathbb{V} \to (-\infty, \infty)$ be an extended real-valued function and let $a > 0$. Then, the convex conjugate of the function $f = ag$ is given by $f^*(y) = ag^*\left(\frac{y}{a}\right)$.*

Proof.

$$f^*(y) = \sup_{x \in \mathbb{V}} \langle x, y \rangle - ag(x) = a \sup_{x \in \mathbb{V}} \langle x, y/a \rangle - g(x) = ag^*\left(\frac{y}{a}\right). \tag{2.111}$$

□

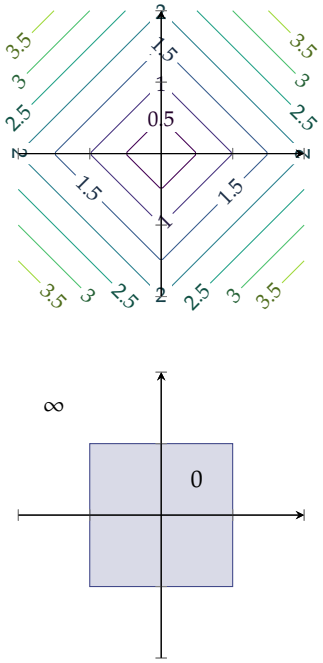

Figure 2.15: The one-norm $\|\cdot\|_1$ on $\mathbb{R}^2$ (top) and its convex conjugate $\delta_{\|\cdot\|_\infty \leq 1}$.

2.4.6 Proximal operator

A fundamental building block in both convex and nonconvex optimization are proximal operators, as they are an important tool in dealing with nonsmoothness. We introduce proximal maps via the concept of infimal convolutions:

Definition 2.4.16 (Infimal convolution). *Let $f, g\colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper. The* infimal convolution *of f and g is the map*

$$f \,\square\, g = \inf_{y \in \mathbb{V}} f(z) + g(\cdot - z).$$

The infimal convolution is *exact* if the infimum is attained. Although the infimal convolution is symmetric with respect to the functions, often one function is fixed and we call the second function the *kernel*. The infimal convolution preserves convexity in the following sense:

Theorem 2.4.13 (Infimal convolution preserves convexity). *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper and convex and let $g\colon \mathbb{V} \to \mathbb{R}$. Then, $f \,\square\, g$ is convex.*

Proof. See [15, theorem 2.19]. □

The next step towards the proximal map is the Moreau envelope, which arises when convolving with a (scaled) quadratic kernel. The following definition describes this in more detail.

Definition 2.4.17 (Moreau envelope). *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper, closed, and convex. The* Moreau envelope *of f with* smoothing parameter *$\lambda > 0$ is the map*

$$f_\lambda := f \,\square\, \frac{1}{2\lambda}\|\cdot\|^2.$$

Figure 2.16 illustrates the construction of the Moreau envelope using the prototypical example of smoothing the absolute value. The resulting function

$$\left(|\cdot| \,\square\, \tfrac{1}{2\lambda}|\cdot|^2\right)(x) = \begin{cases} x^2/2 & \text{if } |x| \le \lambda, \\ \lambda(|x| - \lambda/2) & \text{else,} \end{cases} \tag{2.112}$$

is known as the Huber function and plays an important role in robust statistics and optimization.

Finally, the proximal operator is defined as the minimizing argument of the Moreau envelope.

Definition 2.4.18 (Proximal operator). *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper closed an*

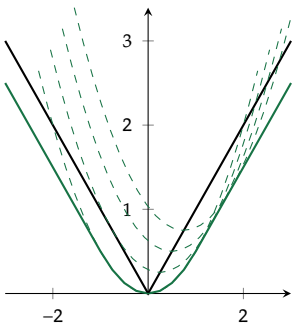

Figure 2.16: Illustration of the Moreau envelope of the absolute value, leading to the Huber function.

convex and let $\lambda > 0$. The proximal operator *of f is*

$$\operatorname{prox}_{\lambda f} := \arg\min_{y \in \mathbb{V}} f(y) + \frac{1}{2\lambda}\|y - \cdot\|^2.$$

Again, the prototypical example is the proximal map of the absolute value which is also known as the soft-shrinkage:

$$\operatorname{prox}_{\lambda|\cdot|}(x) = \max(|x| - \lambda, 0)\operatorname{sign}(x). \tag{2.113}$$

The soft-shrinkage operator is shown in fig. 2.17 for $\lambda = 1$. Existence and uniqueness of the proximal operator are ensured by the following theorem.

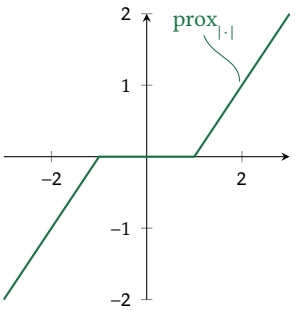

Figure 2.17: The proximal operator of the absolute value is the well-known soft shrinkage.

Theorem 2.4.14 (Existence and uniqueness of the proximal operator)**.** *Let $f\colon \mathbb{V} \to \overline{\mathbb{R}}$ be proper, closed, and convex. Then, for any $x \in \mathbb{V}$, $\operatorname{prox}_f(x)$ exists and is unique.*

Proof. For any $x \in \mathbb{V}$, the function $f + \frac{1}{2}\|\cdot - x\|^2$ as a sum of a proper, closed, and convex function f and a strongly convex function $\frac{1}{2}\|\cdot - x\|^2$, is proper, closed, and strongly convex. Thus, it has a unique minimizer. □

The orthogonal projection onto a convex set can be regarded as a special case of the proximal operator: For a nonempty, closed, and convex set C, its indicator function $\delta_C\colon \mathbb{V} \to \overline{\mathbb{R}}$ is proper, closed, and convex. Thus, $\operatorname{prox}_{\delta_C}$ exists and reduces to the projection

$$\operatorname{proj}_C = \operatorname{prox}_{\delta_C}, \tag{2.114}$$

where the orthogonal projection is defined as

$$\operatorname{proj}_C(x) = \arg\min_{y \in C} \frac{1}{2}\|y - x\|^2. \tag{2.115}$$

The projection is schematically illustrated in fig. 2.18.

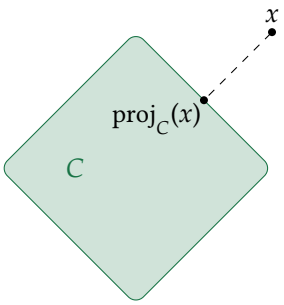

Figure 2.18: Illustration of the orthogonal projection onto a closed and convex set.

2.4.7 Gradient methods

The first class of algorithms that we consider to solve the problem eq. (2.100) assume an unconstrained problem and that f is continuously differentiable with L-Lipschitz continuous gradient. The simplest algorithm for these types of problem is gradient descent, where the iterates are following the negative gradient of the objective function as outlined in algorithm 2. The step sizes can be selected in various ways; under our assumptions of a L-Lipschitz continuous gradient, the method converges for constant step sizes $\tau^k = \tau \in (0, 2/L)$ [172]. When f is convex, the method converges with rate[10] $\mathcal{O}\left(\frac{1}{k}\right)$ to the optimum [172].

To achieve the optimal rate of $\mathcal{O}\left(\frac{1}{k^2}\right)$, Nesterov extended the algorithm with an extrapolation step in 1983 [171]. As evident in algorithm 3, the complexity of the algorithm is essentially unchanged from gradient descent; it only requires memory to store the previous iterate but otherwise incurs almost no computational overhead. The step size can again be chosen constant $\tau^k = \tau \in (0, 1/L)$.

10: these rates are understood with respect to the difference of the function values of the iterates to the optimum

Algorithm 2: Gradient descent

Input: Let $k = 0$, choose $x^0 \in \mathbb{R}^n$ and a step size sequence τ.

while *not converged* **do**
- $x^{k+1} = x^k - \tau^k \nabla f(x^k)$
- $k = k + 1$

Algorithm 3: Nesterov's accelerated gradient method

Input: Set $k = 0$, $\alpha^0 = 1$, choose $x^0 \in \mathbb{R}^n$ and a step size sequence τ.

$x^{-1} = x^0$

while *not converged* **do**
- $\alpha^{k+1} = \left(1 + \sqrt{1 + 4(\alpha^k)^2}\right)/2$
- $\beta^k = \frac{\alpha^k - 1}{\alpha^{k+1}}$
- $\bar{x}^k = x^k + \beta^k\left(x^k - x^{k-1}\right)$
- $x^{k+1} = \bar{x}^k - \tau^k \nabla f(\bar{x}^k)$
- $k = k + 1$

2.4.8 Proximal methods

The main tool we use to deal with nondifferentiability are proximal methods. In this discussion, we assume that the objective function can be split into a function with Lipschitz continuous gradient and a function that admits efficient evaluations of its proximal map. Formally, we consider the optimization problems

$$\min_{x \in \mathbb{R}^n} \left\{ f(x) = (g + h)(x) \right\} \tag{2.116}$$

where $g\colon \mathbb{R}^n \to \mathbb{R}$ is such that ∇g is L-Lipschitz continuous and $h\colon \mathbb{R}^n \to \overline{\mathbb{R}}$ is such that $\operatorname{prox}_{L^{-1}h}$ can be efficiently evaluated. Many applications in imaging can be cast into this form. In particular, the class of problems of this form also encompasses constrained optimization, where h would be an indicator function. A classical algorithm to solve these problems is the proximal gradient algorithm [15, chapter 10], outlined in algorithm 4. This algorithm is known in the literature also under the names forward-backward splitting [150, 59] and iterative shrinkage and thresholding algorithm (ISTA) [69]. It extends gradient descent with a proximal step on the nonsmooth function. However, this algorithm does not achieve optimal convergence rates. To overcome this, Beck and Teboulle combined inertial and proximal methods in their influential paper [16]. The resulting fast iterative shrinkage and thresholding algorithm (FISTA) algorithm is outlined in algorithm 5. Despite its name, the algorithm can be used for any problem of the form eq. (2.116) and achieves the optimal convergence rate of first order methods.

Algorithm 4: Proximal gradient

Input: Set $k = 0$, $\alpha^0 = 1$, choose $x^0 \in \mathbb{R}^n$.

while *not converged* **do**

$x^{k+1} = \operatorname{prox}_{L^{-1}h}\left(x^k - L^{-1}\nabla g(x^k)\right)$

$k = k + 1$

Algorithm 5: Fast iterative shrinkage and thresholding algorithm [16]

Input: Set $k = 0$, $\alpha^0 = 1$, choose $x^0 \in \mathbb{R}^n$.

$x^{-1} = x^0$

while *not converged* **do**

$\alpha^{k+1} = \left(1 + \sqrt{1 + 4(\alpha^k)^2}\right)/2$

$\beta^k = \frac{\alpha^k - 1}{\alpha^{k+1}}$

$y^k = x^k + \beta^k\left(x^k - x^{k-1}\right)$

$x^{k+1} = \operatorname{prox}_{L^{-1}h}\left(y^k - L^{-1}\nabla g(y^k)\right)$

$k = k + 1$

2.4.9 Primal-dual methods

Proximal gradient methods, such as FISTA, are suitable for composite problem where the objective function can be split into a continuously differentiable function and a function with efficiently computable proximal map. If the objective function does not follow this structure, transforming the convex minimization problem into a convex-concave saddle point problem can be beneficial. Algorithms that solve these types of saddle point problem are called primal-dual methods [36, 201].

Formally, primal-dual methods apply to problems of the form

$$\min_{x \in \mathbb{R}^n} (f \circ K + g)(x), \tag{2.117}$$

where $K\colon \mathbb{R}^n \to \mathbb{R}^m$ is a linear operator and $f\colon \mathbb{R}^m \to \overline{\mathbb{R}}$ and $g\colon \mathbb{R}^n \to \overline{\mathbb{R}}$ are proper, closed, and convex. Notably, f and g do not need to be differentiable. Since f is proper, closed, and convex, it is its own biconjugate, i.e. $f^{**} = f$, allowing eq. (2.117) to be cast as the saddle point problem

$$\min_{x \in \mathbb{R}^n} \max_{y \in \mathbb{R}^m} \langle Kx, y \rangle - f^*(y) + g(x). \tag{2.118}$$

The primal-dual hybrid gradient (PDHG) algorithm [36], outlined in algorithm 6 is widely used to solve such problems. It alternates between a proximal gradient descent step on the primal variable and a proximal gradient ascent step on the dual variable, with the gradient evaluated after extrapolating the primal variable. This algorithm achieves an optimal rate $\mathcal{O}\left(\frac{1}{k}\right)$ for non-smooth convex problems when the step sizes are chosen such that $\tau\sigma\|K\|^2 < 1$ [36].

Algorithm 6: Primal dual hybrid gradient [36]

Input: Set $k = 0$, choose $x^0 \in \mathbb{R}^n$, $y^0 \in \mathbb{R}^m$ and step sizes $\tau, \sigma > 0$.

$x^{-1} = x^0$

while *not converged* **do**

  $x^{k+1} = \operatorname{prox}_{\tau g}\left(x^k - \tau K^* y^k\right)$

  $y^{k+1} = \operatorname{prox}_{\sigma f^*}\left(x^k - \sigma K\left(2x^{k+1} - x^k\right)\right)$

  $k = k + 1$

2.4.10 Nonconvex optimization

This thesis often deals with optimization problems involving nonconvex and possibly nonsmooth objectives. Specifically, in chapter 5 and chapter 6 we encode the distribution of reference data in (deep) neural networks and use the them to regularize inverse problems. Imposing convexity onto the neural networks would be too restrictive. Thus, we review the main tools and algorithms from nonconvex optimization.

Analyzing nonconvex optimization algorithms poses challenges: convergence to a global minimum cannot generally be guaranteed. Typically, convergence is ensured only in the sense of the first-order inclusion criterion $0 \in \partial f(x^*)$, which holds for any stationary point x^*. The analysis of optimization algorithms for nonconvex problem often relies on the Kurdyka-Łojasiewicz property [11, section 3.2]. Although functions encountered in optimization problems in imaging usually have this property, it is technical and sometimes hard to verify.

To solve nonconvex nonsmooth optimization problems, Ochs et al. [179] extended Polyak's heavy ball algorithm [189] to account for a nonsmooth function via a proximal map. In detail, Ochs et al. consider optimization problems of the form

$$\min_{x \in \mathbb{R}^n} \left\{ f(x) = (g + h)(x) \right\} \tag{2.119}$$

where $g\colon \mathbb{R}^n \to \mathbb{R}$ has a Lipschitz continuous (definition 2.1.10) gradient (but is possibly nonconvex) and $h\colon \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex (but possibly nonsmooth) and admits a fast procedure to compute its proximal map (definition 2.4.18). The resulting inertial proximal algorithm for nonconvex optimization (iPiano) algorithm is shown in algorithm 7; for $h \equiv 0$ it reduces to Polyak's heavy ball method and for $\alpha \equiv 0$ it reduces to the proximal gradient method.

Pock and Sabach [188] later extended the iPiano algorithm to account for multiple variable blocks. They consider optimization problems of the form[11]

$$\min_{(x_1, x_2) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}} f(x_1, x_2) = g(x_1, x_2) + h_1(x_1) + h_2(x_2). \tag{2.120}$$

11: As in the original paper, we write the algorithm for two variable blocks. However, the algorithm is analogous for more than two variable block.

The assumptions are that $g\colon \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \to \mathbb{R}$ has a Lipschitz continuous gradient in each variable block and that $h_1\colon \mathbb{R}^{n_1} \to \overline{\mathbb{R}}$ and $h_2\colon \mathbb{R}^{n_2} \to \overline{\mathbb{R}}$ admit an efficiently computable proximal map. Specifically, for a given $x_2 \in \mathbb{R}^{n_2}$ the map $\nabla g(\cdot, x_2)$ is Lipschitz continuous, and for a given $x_1 \in \mathbb{R}^{n_1}$ the map $\nabla g(x_1, \cdot)$ must be

Algorithm 7: Inertial proximal algorithm for nonconvex optimization [179]

Input: Choose starting point $x^0 \in \operatorname{dom} h$, choose the sequence α on $[0,1]$, and step size sequence τ on $(0,\infty)$.

$x^{-1} = x^0$
while *not converged* **do**
  $y^k = x^k + \alpha^k\left(x^k - x^{k-1}\right)$
  $x^{k+1} = \operatorname{prox}_{\tau^k h}\left(y^k - \tau^k \nabla g\left(x^k\right)\right)$
  $k = k+1$

Lipschitz continuous. In both cases, the Lipschitz constant can depend on the fixed variable. Then, the inertial proximal alternating linearized minimization (iPALM) algorithm outlined in algorithm 8 converges to a critical point of f. When the partial Lipschitz constants are unknown or too difficult to compute, backtracking schemes like the one proposed in [16] can be used.

Algorithm 8: Inertial proximal alternating linearized minimization [188]

Input: Choose starting point $(x_1^0, x_2^0) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$, sequences $\alpha_1, \beta_1, \alpha_2, \beta_2$ on $[0,1]$, and step size sequences τ_1, τ_2 on $(0,\infty)$.

$(x_1^{-1}, x_2^{-1}) = (x_1^0, x_2^0)$
while *not converged* **do**
  $y_1^k = x_1^k + \alpha_1^k\left(x_1^k - x_1^{k-1}\right)$
  $z_1^k = x_1^k + \beta_1^k\left(x_1^k - x_1^{k-1}\right)$
  $x_1^{k+1} = \operatorname{prox}_{\tau_1^k h_1}\left(y_1^k - \tau_1^k \nabla_1 g\left(z_1^k, x_2^k\right)\right)$
  $y_2^k = x_2^k + \alpha_2^k\left(x_2^k - x_2^{k-1}\right)$
  $z_2^k = x_2^k + \beta_2^k\left(x_2^k - x_2^{k-1}\right)$
  $x_2^{k+1} = \operatorname{prox}_{\tau_2^k h_2}\left(y_2^k - \tau_2^k \nabla_2 g\left(x_1^{k+1}, z_2^k\right)\right)$
  $k = k+1$

Finally, we discuss the Gauss-Newton Method to solve nonlinear least-squares problems; see also [70, section 1.5.1]. This algorithm is utilized in chapter 6 to optimize wavelets building blocks. Consider the nonlinear least-squares problem

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|f(x)\|^2 \tag{2.121}$$

where $f\colon \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable. At each iteration, the Gauss-Newton method minimizes the linearization of f around the current iterate in the least squares sense: Let $x^k \in \mathbb{R}^n$ be the current iterate, then

$$x^{k+1} = \operatorname*{arg\,min}_{x \in \mathbb{R}^n} \frac{1}{2} \|Jf(x^k)(x - x^k) + f(x^k)\|^2, \tag{2.122}$$

where Jf denotes the Jacobian of f. The solution is explicitly given by

$$x^{k+1} = x^k - \left(Jf(x^k)^* Jf(x^k)\right)^{-1} Jf(x^k)^* f(x^k). \tag{2.123}$$

In practice, when $\left(Jf(x^k)^* Jf(x^k)\right)$ is ill conditioned or singular, a small multiple of the identity can be added. In addition, the update step can be endowed with a step size, which leads to the Levenberg-Marquardt iterations summarized in algorithm 9.

Algorithm 9: Levenberg-Marquardt algorithm

Input: Starting point $x^0 \in \mathbb{R}^n$, step size sequence $\alpha > 0$, regularization sequence $\beta \geq 0$.

1 **while** *not converged* **do**

2 $\quad x^{k+1} = x^k - \alpha^k \left(Jf(x^k)^* Jf(x^k) + \beta^k I\right)^{-1} Jf(x^k)^* f(x^k)$

3 $\quad k = k + 1$

2.4.11 Clarke subdifferential

In the section on convex optimization section 2.4.2, we defined the subdifferential definition 2.4.9 as the generalization of the gradient to nondifferentiable functions *with respect to the gradient inequality* (theorem 2.4.4). However, this approach only applies to convex functions. Here, we introduce the Clarke subdifferential, which extends the concept to nonsmooth, nonconvex functions, based on notes by Boyd, Duchi, Pilanci and Vandenberghe.[12]

Consider the function f from $\mathbb{R}$ to R given by

$$x \mapsto \begin{cases} \text{eq. (2.108)} & \text{if } -1 \leq x < 1, \\ 2 - |x| & \text{else,} \end{cases} \tag{2.124}$$

illustrated in fig. 2.19. Recall that eq. (2.108) is defined as

$$x \mapsto \begin{cases} x^2 & x \leq 0, \\ x & x > 0. \end{cases} \tag{2.125}$$

Although subdifferentiable *everywhere*, embedding it in $x \mapsto 2 - |x|$ on $[-1, 1]$ makes the resulting map subdifferentiable *nowhere*, highlighting the restriction of the concept of the subdifferential to convex functions.

We define the Clarke subdifferential on locally Lipschitz continuous functions (definition 2.1.11). Based on a result from Rademacher, for such a function f, any neighborhood (definition 2.1.5) of a point $x \in \operatorname{dom} f$ contains a point y where f is differentiable. At nondifferentiable points, the Clarke subdifferential is the convex hull of admissible gradients in the neighborhood.

12: The notes can be found at https://web.stanford.edu/class/ee364b/lectures/subgradients_notes.pdf (accessed 2024-05-21).

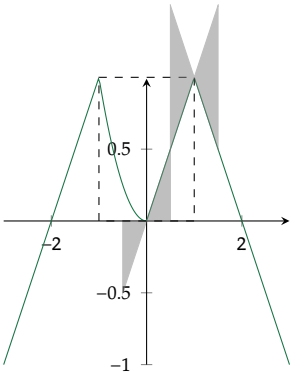

Figure 2.19: The Clarke subdifferential provides a generalization of the gradient to nondifferentiable functions. The dashed region is shown in fig. 2.13, the gray shaded areas show the span of linear functions constructed from the Clarke subdifferential at 0 and 1.

Definition 2.4.19 (Clarke subdifferential). *Let $f\colon \mathbb{V} \to \mathbb{R}$ be locally Lipschitz continuous (definition 2.1.11). The* Clarke subdifferential *of f at x is the set*

$$\operatorname{conv}\Big\{s \in \mathbb{V} \;\Big|\; \text{there exists a sequence } x^k \to x \text{ such that } \nabla f(x^k) \text{ exists, and } \nabla f(x^k) \to s\Big\}. \tag{2.126}$$

The elements of this set are the Clarke subgradients *of f at x.*

This definition applies to a broad class of functions, independent of global characteristics like convexity. The Clarke subdifferential comes with its own calculus, e.g. the chain rule, see [200, chapter 10]. For the purposes in this thesis, particularly computing a Clarke subgradient of a neural network composed of convolutions and rectified linear unit (ReLU) activations in chapter 5, it suffices to note that the chain rule works "as expected".

2.4.12 Large sum problems

Machine learning problems often involve minimizing some risk over a data set, where the objective function decomposes over the data points This structure can be represented as

$$\min_{x \in C} \left\{ f(x) = \sum_{i=1}^{m} f_i(x) \right\}, \tag{2.127}$$

with component functions $f_1, f_2, \dots, f_m\colon \mathbb{R}^n \to \mathbb{R}$ and the constraint set $C \subseteq \mathbb{R}^n$.

An example is supervised signal recovery: Let $c_1, c_2, \dots, c_m \in \mathbb{R}^l$ be m corrupted signals of length l with known reference signals $r_1, r_2, \dots, r_m \in \mathbb{R}^l$. The component functions might be $f_i(x) = \|\phi(c_i, x) - r_i\|^2$, where $\phi\colon \mathbb{R}^l \times \mathbb{R}^n \to \mathbb{R}^l$ is a parametrized function with parameters explicit in the second argument. Denoting the optimal parameters as x^*, the reference signal of a previously unseen corrupted signal can then be recovered using $\phi(\,\cdot\,, x^*)$. This approach gained popularity with data-driven signal recovery methods, see the extensive discussion in later chapters.

For large m, evaluating ∇f becomes costly, common in modern machine learning applications[13]. This necessitates methods that consider the component functions *incrementally*. A straightforward algorithm is the projected incremental gradient algorithm, which updates iteratively with a projected gradient step with respect to some selected component functions. The component selection function has form $\mathscr{C}\colon \mathbb{N} \to \{\text{all subsets of size } b \text{ of } \{1, 2, \dots, m\}\}$. Here, the *batch size* b is typically much smaller than m. Classical component selection functions include batched cyclic permutation or random selections with uniform probability. This algorithm is summarized in algorithm 10; in machine learning, the step size τ^k is referred to as the *learning rate*. For a detailed discussion, see Bertsekas' survey [19].

13: This is evident with large-scale dataset like LAION-5B [212], containing almost 5 850 000 000 image-text pairs.

A common algorithm for problems large sum problems like eq. (2.127) is adaptive moments (Adam) [130]. As shown in algorithm 11, it combines ideas from momentum-based methods with gradient preconditioning. It tracks first and second order moments of the gradient via an exponential moving average with decay rates $\beta_1, \beta_2 \in [0, 1)$ respectively. The gradient update uses the bias-corrected

first order moment, scaled by the bias corrected second order moment. We have included a projection in the update of the optimization variable, not present in the original publication. Adam is the standard for optimizing the parameters of deep neural networks, despite convergence issues outlined by Reddi et al. [196]. They showed that algorithms utilizing gradient updates scaled by square roots of exponential moving averages of squared past gradients fail to converge in general.

Algorithm 10: Projected incremental gradient [19]

Input: Set $k = 0$, choose $x_0 \in \mathbb{R}^n$.

while *not converged* **do**

$x^{k+1} = \operatorname{proj}_C\left(x^k - \tau^k \sum_{i \in \mathscr{C}(k)} \nabla f_i(x^k)\right)$

$k = k + 1$

Adam has been extended by AdaBelief [258], which adjusts the second-order moment using the difference between the observed gradient and the first-order moment. This approach adapts more effectively to gradient changes, leading to improved performance. The AdaBelief algorithm is outlined in algorithm 12.

Algorithm 11: Projected adaptive moments [131]

Input: Set $k = 0$, $m_0 = 0_{\mathbb{R}^n}$, $v_0 = 0_{\mathbb{R}^n}$. Choose $x_0 \in \mathbb{R}^n$, $\epsilon > 0$, $\tau > 0$ and $\beta_1, \beta_2 \in [0, 1)$.

while *not converged* **do**

$g^k = \sum_{i \in \mathscr{C}(k)} \nabla f_i(x^k)$

$m^k = \beta_1 m^{k-1} + (1 - \beta_1) g^k$

$v^k = \beta_2 v^{k-1} + (1 - \beta_2)\left(g^k \odot g^k\right)$

$\hat{m}^k = \frac{m^k}{1 - \beta_2^k}$

$\hat{v}^k = \frac{v^k}{1 - \beta_2^k}$

$x^{k+1} = \operatorname{proj}_C^{\operatorname{diag} \sqrt{\hat{v}^k}}\left(x^k - \tau^k \frac{\hat{m}^k}{\sqrt{\hat{v}^k} + \epsilon}\right)$

$k = k + 1$

2.5 Representation of images

In this section, we define the notion of images as used throughout this thesis. We start with the general idea of an image as a function and progress to the matrix representation of two-dimensional discrete images with square pixels.

An image u is a function mapping from a d-dimensional *image domain* Ω to a color space F:

$$u \colon \Omega \to F. \tag{2.128}$$

Algorithm 12: Projected adaptive beliefs [258]

Input: Set $k = 0$, $m_0 = 0_{\mathbb{R}^n}$, $s_0 = 0_{\mathbb{R}^n}$. Choose $x_0 \in \mathbb{R}^n$, $\epsilon > 0$, $\tau > 0$ and $\beta_1, \beta_2 \in [0, 1)$.

while *not converged* **do**

$g^k = \nabla f_{c(k)}(x^k)$

$m^k = \beta_1 m^{k-1} + (1 - \beta_1) g^k$

$s^k = \beta_2 s^{k-1} + (1 - \beta_2)\Big((g^k - m^k) \odot (g^k - m^k)\Big) + \epsilon$

$\hat{m}^k = \frac{m^k}{1-\beta_2^k}$

$\hat{s}^k = \frac{s^k}{1-\beta_2^k}$

$x^{k+1} = \operatorname{proj}_C^{\operatorname{diag}\sqrt{\hat{s}^k}}\left(x^k - \tau \frac{\hat{m}^k}{\sqrt{\hat{s}^k}+\epsilon}\right)$

$k = k + 1$

Both Ω and F can be discrete or continuous spaces. In this thesis, we focus on two-dimensional discrete and finite domains,

$$\Omega = \{1, 2, \dots, m\} \times \{1, 2, \dots, n\}, \ m, n \in \mathbb{N}. \tag{2.129}$$

The domain indexes equispaced locations $(ih, jh)^\top \in \mathbb{R}^2$ where $(i, j) \in \Omega$. Here, $m, n \in \mathbb{N}$ are the number of vertical and horizontal *square pixels* of size $h > 0$. We consider images that map to the real line, i.e. continuous gray-scale images, so $F = \mathbb{R}$. Unless stated otherwise, we do *not* assume that pixel values are within any closed interval (e.g. $[0, 1]$).[14]

14: This is important in evaluation. We do not clip any output images to any interval prior to evaluation.

For convenience, instead of viewing images as functions, we use the shorthand

$$x_{i,j} \coloneqq u(i, j) \in \mathbb{R} \text{ for all } (i, j) \in \Omega, \tag{2.130}$$

and represent images as matrices $x \in \mathbb{R}^{m \times n}$ as illustrated in fig. 2.20.

2.5.1 Quality metrics

As much of this thesis is concerned with recovering images from incomplete data, we need quantitative methods to measure the success of an algorithm. The most widespread quality metric[15] in signal processing is the MSE:

15: The name quality metric might imply that "larger is better". This is not necessarily the case in this thesis.

Definition 2.5.1 (Mean squared error). *Let $x, \hat{x} \in \mathbb{R}^{m \times n}$ be two images. The* mean squared error *between x and $\hat{x}$ is*

$$\frac{1}{mn} \sum_{i,j=1}^{m,n} (x_{i,j} - \hat{x}_{i,j})^2 = \frac{1}{mn} \|x - \hat{x}\|_2^2. \tag{2.131}$$

To account for the scale of the image[16], the MSE can be normalized by the "power" of the reference signal, leading to the definition of the normalized mean-squared error (NMSE).

16: For instance, $F = [0, 1]$ versus $F = [0, 255]$

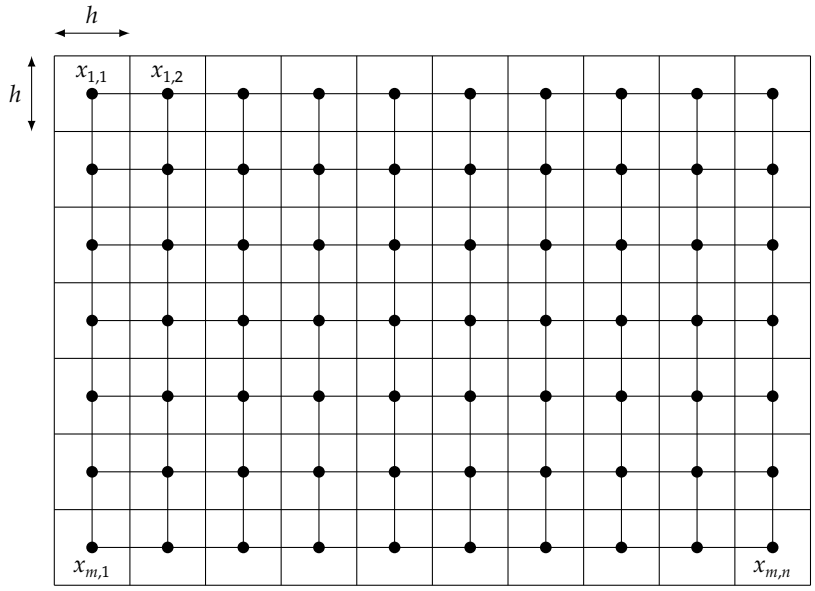

Figure 2.20: Images are represented as a regular grid of square pixels with length d. We represent an image as a matrix $x \in \mathbb{R}^{m \times n}$.

Definition 2.5.2 (Normalized mean squared error)**.** *Let $x \in \mathbb{R}^{m \times n}$ be a reference signal that we aim to recover and let $\hat{x} \in \mathbb{R}^{m \times n}$ be an estimation. The* normalized mean-squared error *between x and $\hat{x}$ is*

$$\frac{||x - \hat{x}||_2^2}{||x||_2^2}. \tag{2.132}$$

While the MSE is symmetric in the arguments, the NMSE is not. In the latter, the MSE is normalized by dividing by the power of the *reference signal*, as otherwise we could make the error arbitrarily small by increasing the norm of the estimation (for instance via multiplication by a scalar).

Another related metric is the PSNR which is usually shown logarithmically (in decibel (dB)). It represents the ratio between the highest possible power in the signal and the MSE.

Definition 2.5.3 (Peak signal to noise ratio)**.** *Let $x, \hat{x} \in \mathbb{R}^{m \times n}$ be two images and $a \in \mathbb{R}$ be the highest possible pixel intensity. The* peak signal-to-noise ratio *between x and $\hat{x}$ is*

$$10 \log_{10}\left(\frac{mna^2}{||x - \hat{x}||_2^2}\right). \tag{2.133}$$

The highest possible pixel intensity may or may not be well defined. For instance, in MRI reconstruction with the fastMRI [251] knee dataset, it depends on scanner-specific details and ranges over multiple orders of magnitude. Thus, in chapter 5, we define the highest possible pixel intensity *per image* as $\max_{i,j} x_{i,j}$. For natural

images in chapter 6, we use $a = 1$ for all images, irrespective of whether this value is achieved in a particular image or not.

MSE, NMSE, and PSNR all measure individual pixel differences and without any invariance properties.[17] These metrics do not align well with the human visual system. To demonstrate, we construct five corrupted images $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_5 \in \mathbb{R}^{256\times256}$ with the same MSE to the reference image $x \in [0,1]^{256\times256}$,[18] i.e. they are on the hypersphere $\{y \in \mathbb{R}^{256\times256} \mid \|x-y\|_2^2/256^2 = r\}$ with $r = 2.2 \times 10^{-3}$. These corruptions are

1. Adding a constant offset: $\hat{x}_1 = x + o$ for some $o > 0$. We used $o = 0.047$.
2. Contrast enhancement: $\hat{x}_2 = c(x-\mu)+\mu$ for some $c > 0$, where $\mu = \sum_{i,j=1}^{m,n} x_{i,j}$. We used $c = 1.287$.
3. Salt and pepper noise: $(\hat{x}_3)_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in \mathscr{S}, \\ 0 & \text{if } (i,j) \in \mathscr{P} \setminus \mathscr{S}, \\ x_{i,j} & \text{if } (i,j) \notin \mathscr{S} \cup \mathscr{P}, \end{cases}$ where $\mathscr{S}$ and $\mathscr{P}$ are sets of indices[19].
4. Blur: $\hat{x}_4 = b * x$ where b is a 4×4 uniform filter.
5. JPEG compression: $\hat{x}_5$ is a JPEG compression of x.[20]

The results in fig. 2.21 reveal that, despite having the same MSE, these images appear vastly different to a human observer. Adding a constant offset has minimal perceptual impact, whereas severe JPEG compression impairs visual quality significantly. This discrepancy poses a problem if the output of an algorithm is presented to a human, for instance, for diagnosis or entertainment.

Numerous quality metrics have been proposed to address this issue and we refer to [163] for a small overview and a study of how well quality metrics predict radiologists' performance. In this thesis, we use the popular structural similarity (SSIM). The definition of the SSIM is understood locally on small patches; the image-level metric is the mean over all overlapping patches, with a filter to avoid blocking artifacts.

Definition 2.5.4 (Structural similarity)**.** *Let $x, \hat{x} \in \mathbb{R}^{d\times d}$ be two image patches and let $\mu_x, \mu_{\hat{x}} \in \mathbb{R}$ and $\sigma_x, \sigma_{\hat{x}} > 0$ be their mean and standard deviation respectively, e.g. $\mu_x = \frac{1}{d^2}\sum_{i,j=1}^{d,d} x_{i,j}$ and $\sigma_x^2 = \frac{1}{d^2-1}\sum_{i,j=1}^{d,d}(x_{i,j}-\mu_x)^2$. In addition, let $\operatorname{cov}_{x\hat{x}} = \frac{1}{d^2-1}\sum_{i,j=1}^{d,d}(x_{i,j}-\mu_x)(\hat{x}_{i,j}-\mu_{\hat{x}})$ and let $K_1 > 0$ and $K_2 > 0$. The* structural similarity *between x and $\hat{x}$ is*

$$\frac{(2\mu_x\mu_{\hat{x}} + C_1)(2\operatorname{cov}_{x\hat{x}} + C_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + C_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + C_2)} \tag{2.134}$$

where $C_1 = (K_1 m)^2$ and $C_2 = (K_2 m)^2$ with $a \in \mathbb{R}$ defined as in definition 2.5.3.

SSIM can be decomposed into components comparing "luminance", "contrast", and "structure", for details see the original publication [241]. In this thesis, we use a 7×7 uniform filter[21] and standard choices $K_1 = 0.01$ and $K_2 = 0.03$. As for the PSNR, when the highest pixel intensity is not well defined, we take the

17: For example, invariance to scalar multiplication, since "brightness" can be adjusted in image viewers.

18: Since the reference image is the same, they consequently also have the same NMSE.

19: All possible indices are in $\mathscr{S}$ or $\mathscr{P}$ with a chance of 3.8×10^{-3}

20: We used imagemagick's `convert reference.png -quality 6 jpeg.jpg`

21: This means that, in the above definition, $d = 7$ and the image patches are essentially taken directly (without filtering).

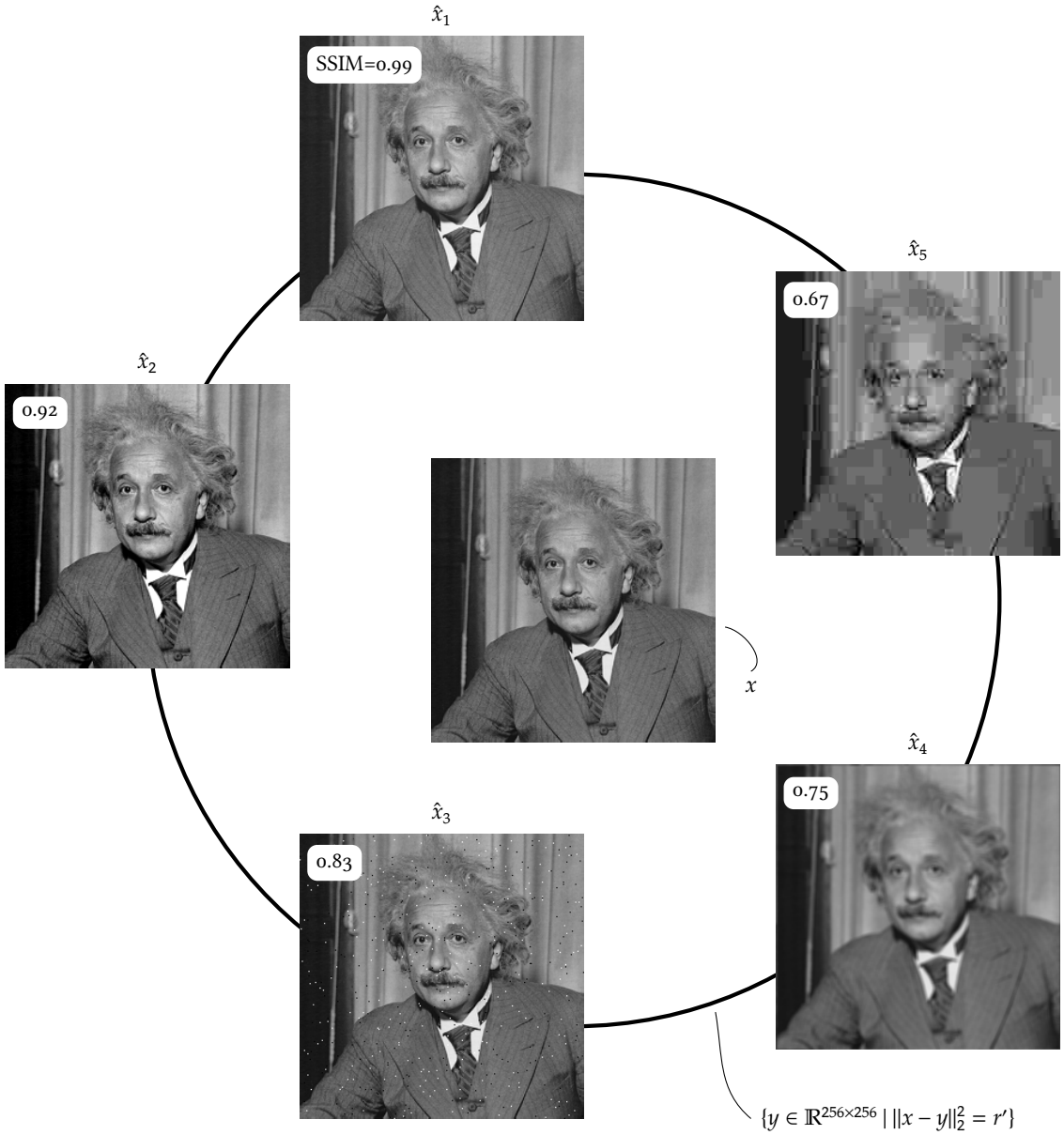

Figure 2.21: The corrupted images $\hat{x}_1, \ldots, \hat{x}_5$ all have the same MSE with respect to the reference image x. However, to a human they appear vastly different. The image x_1, where the same constant offset is added to each pixel, is "the same" as the original image x to the human observer. On the other hand, the image x_5, which has undergone severe JPEG compression, has lost many details and looks much worse to the human. The inlays show the SSIM value (the images are ordered in decreasing SSIM counterclockwise from twelve o'clock), which align much closer with human assessment.

highest pixel intensity in the reference image. Figure 2.21 shows SSIM as inlays; with images ordered in decreasing SSIM (counterclockwise from twelve o'clock), demonstrating that SSIM provides a better alignment with human assessment.

Chapter 3

Machine learning

In this chapter we define neural networks (NNs) and their building blocks, including convolutional layers and popular nonlinearities, and discuss the implications of their nonsmoothness. We then outline our notions of supervised and unsupervised learning as well as generative versus discriminative learning. This chapter is intended to set the stage for the subsequent chapters and is not a comprehensive introduction to these concepts. For a broader overview, refer to [21].

Contents:

3.1 Neural networks

In this thesis, we adopt a broad interpretation of the term "NN". We consider all functions with the following characteristics as NNs: First, a neural network is a *parametric* function, with parameters that we *learn* from data. Second, these functions follow a particular structure. The term neural network implies that the architecture of these functions is (loosely) inspired by the structure of the brain, where electrical signals from spikes of neighboring neurons are weighted and summed. When the electric potential in the neuron exceeds a threshold, it releases a spike.[1] Thus, the inputs to the neuron are combined *linearly*, and the output of the neuron is a *nonlinear activation function*.[2] Then, the next layer of neurons downstream of the output proceeds similarly, creating a *layered* structure of linear weightings and a non-linear functions acting on individual neurons.

In this analogy, the learnable parameters are the weights of the inputs, and possible parameters of the activation function. In most contemporary neural network literature, the activation functions are chosen a-priori and seen as fixed.[3] However, the models in chapter 6 have *learnable* activation functions when viewed in this framework. Specifically, these models can be regarded as one-layer neural networks with trainable activation functions that are the negative logarithm of a one-dimensional Gaussian mixure model (GMM).

We formalize the layered structure of NNs as follows: Let $\mathscr{X}$ be the input space and Θ the set of admissible parameters. In this thesis, the input space is always (at least isomorphic to) $\mathbb{R}^n$. For example, grayscale images of size $m \times n$ make $\mathscr{X} = \mathbb{R}^{m \times n}$. The set of admissible parameters Θ encodes the space of all learnable parameters, with possible constraints. For example, learning $o \in \mathbb{N}$ convolution

1: We are not considering spiking neural networks in this thesis. This interpretation is just to motivate the following discussion.

2: In this chapter, we borrow the terminology from the neural network community. In chapter 6, we borrow the terminology from the Markov random field community. There, these activation functions are called *potentials*, whose derivatives are the *activations*.

3: Arguably, this changes with the recent introduction of Kolmogorov-Arnold networks [151].

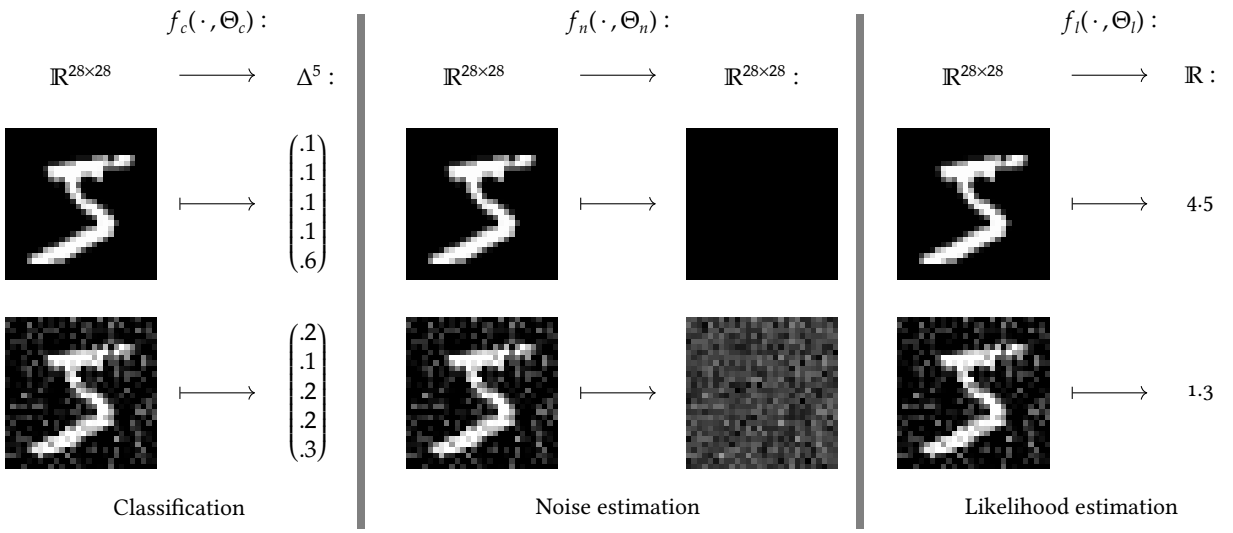

Figure 3.1: Examples of neural networks: In the five-class classification problem on the left, the neural network $f_c(\cdot, \Theta_c)$ maps from the input space, images of size 28×28 with real pixels, to the five-dimensional unit simplex. The entries of a vector which is an element of the five-dimensional unit simplex can be interpreted as probabilities. The goal here is (loosely speaking) that the correct class (here class "5") has the highest probability. In the noise estimation problem in the middle, the neural network $f_n(\cdot, \Theta_n)$ maps from the input space to itself; An estimate of a noise-free image could be given by subtracting the output of the neural network from the input. In the likelihood estimation problem on the right, the neural network $f_l(\cdot, \Theta_l)$ maps from the input space to the real line. Here, the real number that is assigned to an input reflects its likelihood under the learned model. Although the noise- and likelihood estimation problem seem unrelated, there is an extremely close connection via Tweedie's identity. We discuss this in more detail in chapter 6.

filters of size $b \times b$ gives $\Theta = \mathbb{R}^{b\times b\times o}$. In imaging applications, invariance with respect to radiometric shift is often desired and translates to a zero-mean constraint on the filters, then $\Theta = \{x \in \mathbb{R}^{b\times b\times o} \mid \sum_{i,j=1}^{b,b} x_{i,j,k} = 0 \text{ for all } k = 1, \dots, o\}$.

Irrespective of input and parameter spaces, a NN is a map f from $\mathscr{X} \times \Theta$ to the output space $\mathscr{Y}$:

$$f \colon \mathscr{X} \times \Theta \to \mathscr{Y}. \tag{3.1}$$

The output space depends on the task: It could be the k-dimensional unit simplex Δ^k (definition 2.4.6) for k-class classification [181], or the input space $\mathscr{X}$ when the NN models the gradient of an unknown real-valued function of the input [53, 219].[4] In the applications we focus on, the output space is $\mathbb{R}$, meaning the NN maps its input to a scalar. This scalar indicates whether the input to the NN is "likely" under some reference distribution learned by NN. Figure 3.1 shows examples of how neural networks are used for different tasks.

4: This real-valued function could be related to the density of some reference distribution, as is the case in chapter 5 and chapter 6 in this thesis.

We now explore the *architecture* of NNs, recalling some concepts needed later. A NN f typically consists of a cascade of composed functions. Specifically, an *L-layer network* is structured as

$$f = f_L \circ f_{L-1} \circ \cdots \circ f_2 \circ f_1. \tag{3.2}$$

Each layer $f_i \colon \mathscr{X}_{i-1} \times \Theta_i \to \mathscr{X}_i$ for $i = 1, \dots, L$ can be decomposed into a linear map followed by a point-wise non-linearity, and typically has its own parameters in the parameter space $\Theta_i \subseteq \Theta$.[5] An L-layer network is often referred to as having a *depth* of L, with all layers between the input and the output considered *hidden*.

5: We say subset of or equal to because the NN may only have one parametrized layer.

In the next section, we provide more detail on the linear and non-linear elements that constitute a layer. Specifically, within the context of image processing, we explore *convolutional layers* and some of the most popular nonlinearities. The architectures and building blocks we use in this thesis are relatively straightforward. For a more comprehensive review, see Cong and Zhou's survey [62].

3.1.1 Linear layers

The linear sub-layers in NNs are general affine maps. Formally, let $\mathscr{X}_l$ and $\mathscr{X}_{l+1}$ represent the input and output spaces of layer f_l. The linear sub-layer within f_l is the affine map

$$\mathscr{X}_l \times \Theta_l \ni (x, \theta) \mapsto K_l x + b_l \tag{3.3}$$

where $K_l \colon \mathscr{X}_l \to \mathscr{X}_{l+1}$ is an arbitrary linear operator, and $b_l \in \mathscr{X}_{l+1}$ is the affine offset, commonly referred to as *bias* in the NN literature. The parameters θ_l could be all weights in K_l and entries in b_l. However, often K_l and b_l are derived from a much lower dimensional object. For example, the operator K_l can model the convolution of the input with the convolution kernels encoded in the entries of θ.

This thesis considers two primary types of linear sub-layers: dense and fully-learned operators, and convolution operators. These are particularly relevant in imaging applications. Dense linear operators are useful for "inverting" imaging transforms such as the Fourier or the Radon transform. Conversely, convolutions are crucial due to their inherent translation equivariance which is often desired in imaging tasks.

Sub-layers employing dense, fully-learnable operators are typically called *fully connected*. Here, the learnable parameters are *all* weights in K_l and entries in b. In imaging applications, this can be extremely memory intensive: Consider an image x of size $m \times n$ with three color channels, i.e., $x \in \mathbb{R}^{m \times n \times 3}$ and assume we want to learn a linear operator that maps from this space to itself. This linear operator has $(m \times n \times 3) \times (m \times n \times 3)$ weights. For a $m = n = 256$ square image, this amounts to 38 654 705 664 weights, requiring more than 154 GB of storage as 32-bit floating-point numbers. We use a fully connected layer as the last layer in our deep neural regularizer discussed in chapter 5. There, the output space is a scalar and consequently the number of learnable parameters in the fully connected layer is equal to the dimensionality of its input space.

In imaging applications, *convolution layers* are especially important. Unlike fully connected layers, not all weights in the linear operator are learnable. Instead, the operator's weights are derived from a learnable *kernel* and the operator is constructed to encode a convolution. Let $x \in \mathscr{X}_{l-1} = \mathbb{R}^{c_{l-1} \times m_{l-1} \times n_{l-1}}$ be the input the l-th layer of the NN. In image processing, $c_{l-1} \in \mathbb{N}$ are *features* or *channels* of the $(l-1)$-th layer. If the l-th layer has c_l features, the linear operator is a map $K_l \colon \mathbb{R}^{c_{l-1} \times m_{l-1} \times n_{l-1}} \to \mathbb{R}^{c_l \times m_{l-1} \times n_{l-1}}$, i.e. the size of the features remains unchanged. The application of the linear operator is described by a total of $c_{l-1} c_l$ convolutions. Formally, let $k^l_{c,d} \in \mathbb{R}^{s \times s}$ be a kernel of size $s \times s$ where $1 \le c \le c_{l-1}$ and $1 \le d \le c_1$.

Then, the application of the linear operator K_l is described by

$$(K_l x)_{c,i,j} = \sum_{d=1}^{c_{l-1}} \sum_{a,b=1}^{s,s} (k^l_{c,d})_{a,b} \cdot x_{d,\mathrm{bdy}(i-a+\lfloor s/2 \rfloor, m_{l-1}),\mathrm{bdy}(j-b+\lfloor s/2 \rfloor, n_{l-1})}. \tag{3.4}$$

Here, the map $\mathrm{bdy}\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ can be used to realize different boundary conditions when $0 < i - a + \lfloor s/2 \rfloor \leq m_{l-1}$ or $0 < j - b + \lfloor s/2 \rfloor \leq n_{l-1}$ do not hold, see section 2.1.7. In the context of this thesis, the most important technique is *periodic* or *circular* boundary handling due to its relations with the Fourier transform. There, the signals are assumed to be (m_{l-1}, n_{l-1})-periodic, i.e. bdy is given by[6]

$$(a, b) \mapsto (a + (b-1) \bmod b) + 1. \tag{3.5}$$

In this case, we can make use of the *convolution theorem* to diagonalize the linear operator K_l using the Fourier transform, see chapter 6.

6: This is written in array-notation with one-based indexing. A more familiar form is simply $a \bmod b$ for sequence-notation with zero-based indexing.

To expand the receptive field of subsequent layers, neural networks often integrate *downsampling* layers. It suffices to describe the action of downsampling layers on one channel; the application to multiple channels is channel-wise. Applying a d-fold downsampling layer to an image of size $m \times n$ results in an output image of size $\frac{m}{d} \times \frac{n}{d}$ as illustrated in fig. 3.2. There, every d-th pixel is copied into the output and consequently downsampling layers have no learnable parameters. A practical implementation of downsampling involves merging convolution and downsampling operations into *strided convolutions*, minimizing unnecessary computations.

8	1	6	8
3	5	7	9
4	9	2	5
9	4	5	6

8	6
4	2

Figure 3.2: Visualization of a 2-fold downsampling operation.

3.1.2 Nonlinearities

Composing only linear layers would result in an overall linear function, which is insufficient for many tasks and famously fails to solve the "XOR" problem. Thus, in order for the composition eq. (3.2) to become any stronger than linear, the layers must incorporate nonlinearities.

Nonlinearities in NNs typically act point-wise. Formally, assume that the linear sub-layer in the l-th layer (which is not the output layer) maps to $\tilde{\mathscr{X}}_l = \mathbb{R}^{f_l \times m_l \times n_l}$. The nonlinear sub-layer is a function $\Phi_l\colon \tilde{\mathscr{X}}_l \to \tilde{\mathscr{X}}_l$ often structured such that it applies the same function $\phi_l\colon \mathbb{R} \to \mathbb{R}$ point-wise:

$$\big(\Phi_l(x)\big)_{i,j,k} = \phi_l(x_{i,j,k}) \tag{3.6}$$

and often the scalar function ϕ_l is also shared between all layers. The output space of the i-th nonlinear sub-layer is the input space of the $(i+1)$-th layer, i.e., $\mathscr{X}_{i+1} = \tilde{\mathscr{X}}_l$.

Numerous activation functions have been proposed in the literature; for a relatively recent and comprehensive overview see [76]. Popular examples include the logistic sigmoid $x \mapsto (1 + \exp(-x))^{-1}$ and the hyperbolic tangent $x \mapsto \frac{\exp(x) - \exp(-x)}{\exp(x) - \exp(-x)}$. However, these suffer from vanishing or exploding gradients. The rectified linear unit $\nearrow = (x \mapsto \max(0, x))$ has become a favored choice to mitigate this. While $\nearrow$ is not differentiable at 0, automatic differentiation frameworks such as PyTorch [183],

handle this by choosing an element of the subdifferential at 0, usually 0. However, an implicit differentiation framework that encompasses nonsmooth functions has been missing until relatively recently. In 2021, Bolte et al. introduced the notion of *path differentiability* in [25] which aims to resolve this problem.

The nondifferentiability of $\nearrow$ poses a challenge for classical optimization algorithms. We optimize our learned networks (w.r.t. their input) in chapter 5 with standard first-order optimization methods discussed in section 2.4.10. While optimization works well empirically, we can not guarantee convergence. Surrogates for $\nearrow$ such as the swish $x \mapsto \frac{x}{1+\exp(-x)}$ or the exponential linear unit

$$x \mapsto \begin{cases} x & \text{if } x > 0, \\ \exp(x) - 1 & \text{else,} \end{cases}$$

offer differentiable alternatives. If one wants to avoid the computational complexity of these functions during training but ensure convergence of optimization algorithms, it is possible to replace the rectified linear activation functions with a differentiable counterpart after training. However, in order for the learned parameters to be meaningful, the surrogate must approximate $\nearrow$ well, and the gradient of any reasonable approximation will necessarily have an exploding Lipschitz constant, effecting optimization speed.

In chapter 4 we use a *leaky* rectified linear activation given by

$$\nearrow = x \mapsto \max(\gamma x, x) \tag{3.7}$$

with the small leakage coefficient $\gamma > 0$. The discussion about nondifferentiability, its implications, and possible remedies carries over to this version. Figure 3.3 showcases a selection of popular nonlinearities.

The choice of nonlinearity in the output layer depends on the task and is not necessarily point-wise. For instance, in K-class classification problems, the soft-argmax[7]

$$\mathbb{R}^K \ni (x_1, \dots, x_K) \mapsto \begin{pmatrix} \frac{\exp(x_i)}{\sum_{i=1}^K \exp(x_i)} \\ \vdots \\ \frac{\exp(x_i)}{\sum_{i=1}^K \exp(x_i)} \end{pmatrix}. \tag{3.8}$$

is a popular choice. It maps any point onto the K-dimensional unit simplex Δ^K and hence its output values can be interpreted as probabilities. Depending on the application, the identity map $x \mapsto x$ may be suitable, making the output of the network the output of the last linear sub-layer.

The models we discuss in chapter 6 can be viewed as one-layer networks, with each of the $o \in \mathbb{N}$ convolution operators equipped with their own scalar-valued activation function. Thus, Φ has the block structure

$$\mathbb{R}^{m \times n \times o} \ni (x_1, \dots, x_o) \mapsto (\Phi_1(x_1), \dots, \Phi_o(x_o)) \tag{3.9}$$

where each $\Phi_i \colon \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ shares the same $\phi_i \colon \mathbb{R} \to \mathbb{R}$ across all $i = 1, \dots, o$. The nonlinearities are log-sum-exp functions, interpreted as the negative logarithm of a probability density function parametrized through a GMM. Thus, the nonlinearities are endowed with parameters that we *learn* from data.

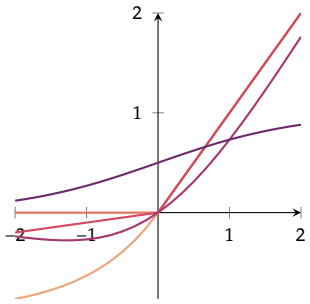

Figure 3.3: Common nonlinearities in neural networks: The exponential linear unit —, the rectified linear unit — and its leaky variant —, the swish —, and the sigmoid —.

7: This is sometimes given the misleading name "softmax". However, it smoothly approximates the argmax function, not the maximum.

3.2 Supervised and unsupervised learning

Machine learning approaches are classically divided into *supervised learning* and *unsupervised learning* [21]. In this section, we elaborate on the differences between these two approaches.

In *supervised learning*, we are given a reference dataset of (input, output) pairs, and the goal is to learn a parametric map that accurately reproduces the reference output for a given input. Formally, consider a pair (X, Y) of random variables, where Y is the input random variable with values in $\mathscr{Y}$, and X is the output random variable with values in $\mathscr{X}$. Here, we adopt the notation from the introduction and the rest of the thesis which is in contrast to the previous section; there the input and output spaces are flipped. We assume that the pair (X, Y) is distributed as $p_{X,Y}$.

The objective is to learn a parametric map

$$f\colon \mathscr{Y} \times \Theta \to \mathscr{X} \tag{3.10}$$

that maps an input $y \in \mathscr{Y}$ to $f(y, \theta) \in \mathscr{X}$ given parameters $\theta \in \Theta$. The set of admissible parameters Θ can encode constraints on the parameter vector θ, e.g. to enforce invariances. To find optimal parameters, we minimize a loss function $l\colon \mathscr{X} \times \mathscr{X} \to$ that compares two points in the output space $\mathscr{X}$:

$$\min_{\theta \in \Theta} \mathbb{E}_{(X,Y)\sim p_{X,Y}}\Big[l(f(Y, \theta), X)\Big]. \tag{3.11}$$

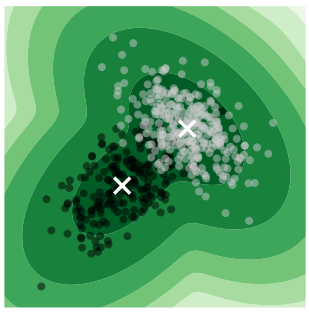

Figure 3.4: Density estimation of a point cloud with a Gaussian mixture model. The white crosses indicate the means of the two components and the shades of green indicate the density.

Supervised learning encompasses classification, where the output space is discrete, and regression, where the output space is continuous. Examples of the former include classification problems in computer vision, such as semantic segmentation or object detection, but also problems like spam detection or bot detection. Examples of the latter include polynomial regression, stock price prediction, and image reconstruction.

In *unsupervised learning*, we aim to find structure in the distribution of the *input* random variable Y. A prototypical example is *density estimation*, which aims to fit a parametric model to the density of the random variable. This is often formalized as maximum likelihood learning, which amounts to finding parameters θ of a parametric density $p\colon \mathscr{Y} \times \theta \to [0, \infty]$ via

$$\min_{\theta \in \Theta} \mathbb{E}_{Y\sim p_Y}\Big[-\log p(Y, \theta)\Big]. \tag{3.12}$$

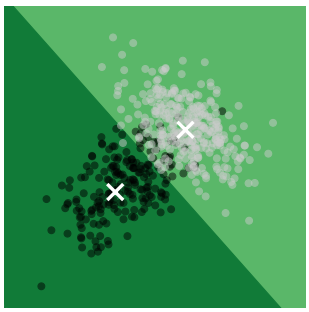

Figure 3.5: Clustering of a point cloud with K-means. The white crosses indicate the centroids of the two clusters and the shades of green indicate the cluster a point is assigned to.

Another classic unsupervised task is *clustering*, where the aim is to partition the input space into disjoint regions based on an unlabeled dataset. A query point can then be assigned to one of the clusters based on its region. Fixing the number of clusters to K and constructing the disjoint regions such that the Euclidean distance between points in the regions is minimized leads to the famous K-means algorithm [152]. Figure 3.4 and fig. 3.5 show examples of density estimation with a Gaussian mixture model and clustering with the K-means algorithm on a point cloud drawn from two Gaussians in two dimensions.

3.3 Generative and discriminative learning

This thesis focuses on using *generative learning* for inverse problems in imaging. Here, we distinguish between generative and discriminative learning by considering a supervised estimation problem. Consider a pair of random variables (X, Y) with joint distribution $p_{X,Y}$ and marginal distributions p_X and p_Y. As in the previous section, Y is the *input* random variable and X is the output random variable.

In the *discriminative approach*, the joint distribution is factorized as

$$p_{X,Y} = p_{X|Y} p_Y, \tag{3.13}$$

where $p_{X|Y}$ is the *posterior* of the output given the input and p_Y is the density of the input. Training parametric estimators of this type[8] amounts to finding

$$\min_{\theta_{X|Y}, \theta_Y} \mathbb{E}_{(X,Y)\sim p_{X,Y}}\Big[-\log \hat{p}_{X|Y}(X, Y, \theta_{X|Y})\Big] + \mathbb{E}_{Y\sim p_Y}\Big[-\log \hat{p}_Y(Y, \theta_Y)\Big], \tag{3.14}$$

where $\hat{p}_{X|Y}(\cdot, \cdot, \theta_{X|Y})$ and $\hat{p}_Y(\cdot, \theta_Y)$ are parametric estimators. Thus, the discriminative approach decouples into the posterior and the input density.

For practical inference problems, the parametric posterior suffices: Given an input $y \in \mathscr{Y}$, the associated estimation of the output can be found via[9]

$$\arg\max_{x\in\mathscr{X}} \hat{p}_{X|Y}(x, y, \theta_{X|Y}). \tag{3.15}$$

The discriminative approach has the advantage that the parametric posterior is typically simpler than the density of the output; see the example below. However, it is challenging to encode domain knowledge and it is typically impossible to adapt to changes in the relationship between the input and output random variables; we demonstrate this with an example in MRI in section 5.2.

In the *generative approach*, the joint distribution is factorized as

$$p_{X,Y} = p_{Y|X} p_X, \tag{3.16}$$

where $p_{Y|X}$ is the *likelihood* of an input given an output, and p_X is the density of the output. Training parametric estimators of this type amounts to finding

$$\min_{\theta_{Y|X}, \theta_X} \mathbb{E}_{(X,Y)\sim p_{X,Y}}\Big[-\log \hat{p}_{Y|X}(X, Y, \theta_{Y|X})\Big] + \mathbb{E}_{X\sim p_X}\Big[-\log \hat{p}_X(X, \theta_X)\Big], \tag{3.17}$$

where $\hat{p}_{Y|X}(\cdot, \cdot, \theta_{Y|X})$ and $\hat{p}_X(\cdot, \theta_X)$ are parametric estimators. When the parameters are learned, we can invoke Bayes theorem and write the posterior as

$$\hat{p}_{X|Y} = \frac{\hat{p}_{Y|X}(\cdot, \cdot, \theta_{Y|X})\hat{p}_X(\cdot, \theta_X)}{\hat{p}_Y} \tag{3.18}$$

where the denominator is irrelevant with respect to the inference

$$\arg\max_{x\in\mathscr{X}} \hat{p}_{Y|X}(x, y, \theta_{Y|X})\hat{p}_X(x, \theta_X). \tag{3.19}$$

Thus, the generative approach separates the input likelihood from the output prior and invokes Bayes theorem for the posterior.

8: in the standard maximum-likelihood framework; the conclusions also hold for more general formulations.

9: For the sake of simplicity, we only discuss MAP estimation

10: usually one: the noise variance

In this thesis, the likelihood is determined by the physical acquisition model and has extremely few parameters[10], identified through grid search on a validation dataset. Thus, the likelihood of Y given X is *known, easy to model, and subject to change*. Therefore, a generative learning approach is advantageous, as it allows for easy encoding of domain knowledge in the likelihood. In addition, changes to the likelihood only require retraining of the parametric likelihood $-\log \hat{p}_{Y|X}(\cdot, \cdot, \theta_{Y|X})$.

In this taxonomy, *supervised and unsupervised* as well as *generative and discriminative* are understood with respect to the downstream task. For example, in chapter 5, we learn a parametric form of the density of MRI images of the human knee. For image reconstruction, this learned density represents the density of the *output* random variable, making the approach *generative* because it separates the input likelihood from the output prior. However, it is *supervised* because we train on the density of the output random variable. For pathology detection[11], this approach is *unsupervised* (because there are no labels) but not generative: Generative learning would amount to learning the density of the output random variable (essentially a scalar representing the pathologies prevalence), as well as the class conditional densities of images with and without pathologies.

We emphasize that discriminative learning is typically easier than generative learning through a two-class classification problem: Assuming two normally distributed random variables with arbitrary mean and covariance, discriminative methods derive a parametric form of the posterior class density given a query point. This object can famously be represented[12] as (see, e.g. [21, section 4.2.1.])

$$x \mapsto \left(1 + \exp(\langle x, Ax\rangle + \langle b, x\rangle + c)\right)^{-1}, \tag{3.20}$$

where A, b, and c are a symmetric matrix, vector, and scalar, respectively. For determining the most likely class, this simplifies further to identifying the parameters defining the hyperbolic decision boundary (depicted as a white line in fig. 3.6). In contrast, generative modeling finds the two class-conditional distributions, both general normal distributions, along with the class priors. This requires estimating two symmetric positive definite matrices, two mean vectors and a scalar representing the class priors. This is illustrated in fig. 3.7, where level lines are related to the covariance matrices and the size of the cross indicating the means is related to the class prior. A new point is then classified according to its weighted likelihood.

In the above example, the challenge lied mostly in finding the likelihood densities. In this case, (for the purpose of pure classification) there is not merit to the generative approach. However, in the inverse problems in imaging considered in this thesis, the likelihood of an input given an output is fixed by the physical acquisition model and is parameter-free up to scaling. In addition, the physical acquisition model is subject to change: As an example, in chapter 5 we consider different frequency selections in a Fourier imaging setup. On the other hand, the prior distribution of images is an extremely complicated object. Thus, in this case the generative approach proves beneficial. Chapters 5 and 6 discuss two methods for learning a parametric density of image priors, highlighting the applicability of generative learning in such contexts.

11: With the learned density, pathology detection could for instance be implemented via simple likelihood evaluation.

12: For either class; the posterior density for the other is just one minus the object.

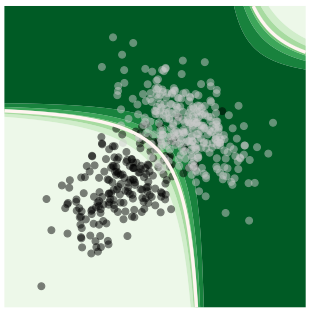

Figure 3.6: Discriminative approach to the two-class classification problem: Darker green indicates higher posterior values of belonging to the "white" class. The white line is the hyperbolic decision boundary.

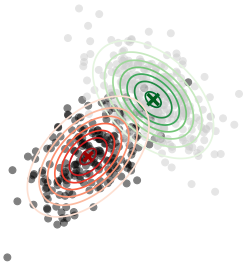

Figure 3.7: Generative approach to the two-class classification problem: The level lines are related to the covariance matrices, the size of the cross markers indicating the mean is related to the prior probability of a class. New points are classified according to the weighted likelihood; the decision boundary is shown in fig. 3.6.

Chapter 4

On the historical development of regularizers

In this chapter we outline the historical development of regularizers, starting with classical variational penalties such as magnitude or TV penalties. TV regularization finds motivation in the histograms of natural images responses on finite difference filters, where the absolute value provides the best convex fit to the negative log-histograms. This immediately suggests a generalization of this model: a suitable set of filters, which can be readily computed from a set of reference images, is given by the principal directions of all image patches extracted from the reference images. When sacrificing convexity, suitable potential functions are derived from the corresponding negative log-histograms.

However, despite their effectiveness in MAP inference, these models do not account for the nontrivial correlation of overlapping patches, making them a bad model for the reference density. In contrast, Markov random field (MRF) modeling accounts for this correlation. By explicitly imposing translation invariance via convolutions, such models become very parameter efficient at the cost of a much more complex learning problem. The product of Gaussian mixture diffusion model (PoGMDM) discussed in chapter 6 exemplifies this approach. Finally, the deep neural regularizer discussed in chapter 5 models the reference density through a general map provided by a deep neural network. Although sacrificing some interpretability and presenting an even more challenging learning problem, these models demonstrate excellent performance.

Contents:

4.1 A running example from MRI reconstruction

To illustrate the historical development of regularizers, we employ a running example from MRI reconstruction. However, the core focus of this chapter lies in examining *the effect of the choice of the regularizer*. For this purpose, we consider a simple reconstruction problem using synthetic data. In contrast, much of chapter 5 is dedicated to developing clinically relevant reconstruction algorithms for real data.

We aim to reconstruct an image from Fourier data $y \in \mathbb{C}^f$ constructed as

$$y = MF_{\mathbb{R}}x + \eta \coloneqq Ax + \eta, \tag{4.1}$$

where $x \in \mathbb{R}^{m\times n}$ represents the reference signal. The two-dimensional "real" Fourier transform $F_{\mathbb{R}} \colon \mathbb{R}^{m\times n} \to \mathbb{C}^{m\times(\lfloor n/2\rfloor+1)}$ exploits conjugate symmetry of the Fourier transform for real-valued signal by storing only half of the frequency plane. Additionally, the Fourier transform is *orthonormal*, i.e. that $F_{\mathbb{R}}F_{\mathbb{R}}^* = F_{\mathbb{R}}^*F_{\mathbb{R}} = I$. The frequency selection operator[1] $M \colon \mathbb{C}^{m\times(\lfloor n/2\rfloor+1)} \to \mathbb{C}^f$ is a binary diagonal operator that selects $f \in \mathbb{N}$ frequencies, simulating the *acceleration* in clinical MRI systems (details in chapter 5). A is a shorthand for $MF_{\mathbb{R}}$.

In this chapter, the reference signal x is the root-sum-of-squares reconstruction of an MRI scan in the fastMRI [251] data set.[2] The images in the fastMRI dataset are square with size $m = n = 320$. Figure 4.1 and fig. 4.2 depict the reference signal and the data respectively. In order to visualize the data y, we map it back to $\mathbb{C}^{m\times(\lfloor n/2\rfloor+1)}$ using M^*.

This chapter is dedicated to finding

$$\arg\min_{x\in\mathbb{R}^{m\times n}} \frac{1}{2}\|Ax - y\|_2^2 + R(x) \tag{4.2}$$

under varying choices of R. To underscore the significance of the choice of R, we initially set $R \equiv 0$. Consequently, the least squares problem $\arg\min_{x\in\mathbb{R}^{m\times n}}\{f(x) \coloneqq \frac{1}{2}\|Ax - y\|^2\}$ has the (not necessarily unique) solution A^*y. This can be seen by noting that Fermat's condition (theorem 2.4.1) at A^*y,

$$\nabla f(A^*y) = A^*(AA^*y - y) = 0, \tag{4.3}$$

is fulfilled since $AA^* = MF_{\mathbb{R}}F_{\mathbb{R}}^*M^* = I$ (on $\mathbb{C}^f$). This solution is commonly referred to as the *zero-filling* solution, as $A^*y = F_{\mathbb{R}}^*M^*y$ essentially fills the missing frequencies with 0 before inverting the Fourier transform. The reconstruction in fig. 4.3 shows the classical back-folding artifacts as well as high-frequency noise.

1: Other names for this object include *undersampling mask* and *subsampling-* or *downsampling operator*.

2: It is the 17-th slice in the file `file1000005.h5` folder `multicoil_train`.

Figure 4.1: The reference signal used throughout this chapter.

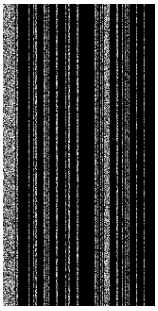

Figure 4.2: The logarithm of the absolute value of the zero filled data M^*y.

4.2 Classical variational penalties

As the first interesting choice for R, we consider the quadratic two-norm $\frac{\lambda}{2}\|\cdot\|^2$, where $\lambda > 0$. Under this choice of R, the first order optimality conditions of eq. (4.2) lead to the unique solution

$$x^* = (A^*A + \lambda I)^{-1}A^*y, \tag{4.4}$$

where the existence of the inverse of the operator $A^*A + \lambda I$ is assured by construction.

This regularization is extremely popular in many applications; for instance, in machine learning applications it is called *weight decay* [92, section 5.2.2, section 7.1.1]. In the Bayesian framework adopted in this thesis, it corresponds to the

assumption that the underlying signal is normally distributed with mean zero. In imaging applications, this is typically not useful as it biases the solution towards dark images. It enforces "regularity" solely on the magnitude of pixel intensities, irrespective of their neighbourhood, making in invariant with respect to arbitrary permutations of the pixels. This characteristic is undesirable for regularizers of images, which have inherent spatial ordering.

For our forward operator, the inverse of $A^*A + \lambda I$ can be calculated quickly by noting that $(F_{\mathbb{R}}^* M^* M F_{\mathbb{R}} + \lambda I)^{-1} = (F_{\mathbb{R}}^* M^* M F_{\mathbb{R}} + \lambda F_{\mathbb{R}}^* F_{\mathbb{R}})^{-1} = (F_{\mathbb{R}}^*(M^*M + \lambda I)F_{\mathbb{R}})^{-1} = F_{\mathbb{R}}^*(M^*M + \lambda I)^{-1} F_{\mathbb{R}}$ where M^*M is a diagonal operator containing only ones and zeros. Therefore, we have

$$\begin{aligned} x^* &= F_{\mathbb{R}}^*(M^*M + \lambda I)^{-1} F_{\mathbb{R}} F_{\mathbb{R}}^* M^* y \\ &= F_{\mathbb{R}}^*(M^*M + \lambda I)^{-1} M^* y \\ &= (1+\lambda)^{-1} F_{\mathbb{R}}^* M^* y, \end{aligned} \tag{4.5}$$

Figure 4.3: Zero-filling solution $F_{\mathbb{R}}^* M^* y$.

where the last equality holds since M^*y is only non-zero in entries whose corresponding position in the diagonal operator $M^*M + \lambda I$ are $1 + \lambda$. In other words, the reconstruction shown in fig. 4.4 is just the zero-filling solution multiplied by $(1+\lambda)^{-1}$.

Instead of penalizing the magnitude of pixel intensities, a fruitful approach is to penalize[3] responses to linear filters. Regularizers of this type can in general be represented as

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sum_{k=1}^{o} \phi_k\Big((K_k x)_{i,j}\Big). \tag{4.6}$$

Figure 4.4: Reconstruction using quadratic pixel magnitude penalization.

Here, $K_1, K_2, \dots, K_o \colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ are convolution operators (section 2.1.7) and $\phi_1, \phi_2, \dots, \phi_o \colon \mathbb{R} \to \mathbb{R}$ are scalar functions.

In this thesis, we adopt the nomenclature from the MRF literature, see e.g. [88, 206, 256, 257]. Each operator K_k is endowed with its own *potential* ϕ_k, which is applied to the *filter responses*. In terms of the statistical model where the regularizer is interpreted as the negative log-prior, the regularizer encodes a *Gibbs distribution* [257, theorem 1 and the two preceding definitions] with density

$$x \mapsto Z(K_1, \psi_1, K_2, \psi_2, \dots, K_o, \psi_o)^{-1} \prod_{i,j=1}^{m,n} \prod_{k=1}^{o} \psi_k\Big((K_k x)_{i,j}\Big). \tag{4.7}$$

Here, analogous to the products-of-experts model from Hinton [116], the one-dimensional functions $\psi_1, \psi_2, \dots, \psi_o \colon \mathbb{R} \to \mathbb{R}$ are termed *experts*. Thus, the potential ϕ_k is related to the expert ψ via $\phi_k = -\log \psi_k$. The gradient of the potential function is called the *activation*.[4] Z is the *partition function* such that the density is normalized.

A simple choice for the convolution operators is $K_1 = D_h, K_2 = D_v \colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$, where D_h and D_v are discretizations of horizontal and vertical image gradients, respectively. A standard choice [33, 36, 37] for defining these operators is

3: The regularizers we consider in these sections generally penalize responses. Later in section 4.4 we discuss regularizers where this is not necessarily true.

4: This is borrowed from the neural network literature.

through forward finite differences with Neumann boundary conditions,

$$(D_v x)_{i,j} = \begin{cases} x_{i+1,j} - x_{i,j} & \text{if } 1 \leq i < m, \\ 0 & \text{else,} \end{cases}$$
$$(D_h x)_{i,j} = \begin{cases} x_{i,j+1} - x_{i,j} & \text{if } 1 \leq j < n, \\ 0 & \text{else.} \end{cases} \tag{4.8}$$

This choice is popular due to its simplicity but there exist many other discretizations. However, the choice of discretization rarely matters in applications [34] and we adopt the forward finite differences scheme. For ease of notation, we define a linear operator $D\colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n\times 2}$ that summarizes the application of D_v and D_h via

$$(Dx)_{i,j,1} = (D_v x)_{i,j} \text{ and } (Dx)_{i,j,2} = (D_h x)_{i,j}. \tag{4.9}$$

When the potentials are shared and chosen as $\phi_1 = \phi_2 = (x \mapsto x^2/2)$, the resulting regularizer can be compactly written as

$$x \mapsto \frac{\lambda}{2} \sum_{i,j=1}^{m,n} \sum_{k=1}^{2} \big((Dx)_{i,j,k}\big)^2 = \frac{\lambda}{2} \|Dx\|_{2,2}^2. \tag{4.10}$$

Here, the subscript in the norm emphasizes that we take a two-norm over both the pixel- and gradient dimensions. This regularizer encodes the assumption that image gradients have Gaussian marginal distributions.

The corresponding optimization problem

$$\arg\min_{x\in\mathbb{R}^{m\times n}} \frac{1}{2}\|Ax - y\|_2^2 + \frac{\lambda}{2}\|Dx\|_{2,2}^2 \tag{4.11}$$

can again be solved in closed form as

$$x^* = (A^*A + \lambda D^*D)^{-1} A^* y. \tag{4.12}$$

Figure 4.5: Reconstruction using quadratic gradient penalization.

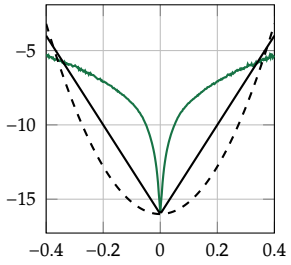

Figure 4.6: Negative log-histogram of horizontal edges in natural images ——. The quadratic - - -, and absolute potentials —— correspond to the choices in eq. (4.10) and eq. (4.13) respectively.

The operator $A^*A + \lambda D^*D$ is invertible when $\ker(A) \cap \ker(D) = \{0\}$. Without going into too much detail, this condition is fulfilled as long as M captures the DC-component of the signal, since $\ker(D) = \{\text{constant signals in } \mathbb{R}^{m\times n}\}$. However, to avoid storing the operator in memory we utilize Nesterov's accelerated gradient method (algorithm 3) to solve the optimization problem. To ensure convergence, we choose the step size sequence $\tau^k = (1 + \lambda\|D\|^2)^{-1}$ using $\sqrt{8}$ as an upper bound for $\|D\|$ from [33].

In the resulting reconstruction in fig. 4.5, some back-folding artifacts are removed, and the image appears less noisy than previous reconstructions. However, edges appear overly smooth and small details such as the blood vessels in the fat tissue are almost completely lost. This is because the quadratic potentials poorly match the empirical marginal distributions of edges in natural images.

It has been known at least since 1999 that the empirical marginal distributions of edges in natural images are highly non-Gaussian [120]. This is illustrated in fig. 4.6, which shows the empirical negative log-histograms of edges in natural images. These empirical marginal distributions are highly leptokurtic (definition 2.2.17).

Huang and Mumford report an excess kurtosis of more than 14 in [120]. Additionally, there is a very sharp peak at zero, indicating that the vast majority of edges in natural images are very close to zero. Simultaneously, edges with large magnitude occur much more often than would be expected under a Gaussian distribution. This *sparsity* of edges in natural images is demonstrated in fig. 4.7.

The analysis of the negative log-marginal distributions of edges in natural images (fig. 4.6) suggests considering other potential functions that match these more closely. Among convex functions, the absolute value is best approximates the negative log-marginal distributions of edges. Choosing $\phi_1 = \phi_2 = |\cdot|$ leads to the well known anisotropic TV regularizer

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sum_{k=1}^{2} \left|(Dx)_{i,j,k}\right| = \lambda \|Dx\|_{1,1}. \tag{4.13}$$

Figure 4.7: Sum of the absolute horizontal and vertical difference of neighboring pixels (bottom right) of an MRI image from the fastMRI [251] dataset (top left).

Again, we adopt this simple interpretation of the discrete anisotropic TV but point out that there exists a large literature on more elaborate, even learned (with respect to approximating some continuous energy), discretizations [23, 38, 60]. Here, *anisotropic* emphasizes that this regularizer favors grid-aligned edges. We discuss different versions of the discrete TV in section 4.2.1.

The one-norm is a highly popular sparsity-inducing function [69, 235]. There are many ways to see the sparsity-inducing property of the one-norm, such as the classical "intersection of level lines" in [109, figure 3.11] or by observing that its proximal map (definition 2.4.18) sets all elements smaller than a threshold to zero (see fig. 2.17). However, from the Bayesian perspective adopted in this thesis, the prior assumptions on the underlying signal are *not* that Dx is sparse: Although the MAP solution favors sparse edges, samples from the Laplace distribution are never zero, e.g. see [3, fig. 1] for samples from a TV prior.

This discrepancy is summarized by the difference between *penalized likelihood estimation*, and *Bayesian estimation* [24, 94]. In the former, the reconstructed signal has some abstract desirable regularity properties (like sparse edges), but the corresponding probabilistic model is in general *not* a "good" model of the underlying distribution. In the latter, the aim is to construct a good probabilistic model of the underlying distribution, whose corresponding MAP estimate might not necessarily exhibit these abstract desirable regularity properties. An illustrative example is fig. 4.7: At almost all pixels, the sum of the absolute horizontal and vertical edges is almost zero. Thus, we derive the abstract property that the underlying signal has sparse gradients. However, a numerical check reveals that actually at exactly *zero* pixels the sum of the absolute horizontal and vertical edges is *exactly* zero.[5] We refer to the papers of Gribonval et al. [94, 95] for other Bayesian interpretations of penalized likelihood estimators and to the thesis of Bohra [24, section 2.2] for a slightly more in-depth discussion.

5: This has been checked on a 32 bit representation of the normalized image (maximum pixel intensity of 1) with a threshold of 1×10^{-7}. Only two pixels have a value lower than 1×10^{-6}.

Optimizing eq. (4.2) under the choice of R in eq. (4.13) is challenging because the absolute value is not differentiable. This can be overcome by using differentiable surrogate functions such as $x \mapsto \sqrt{x^2 + \epsilon^2}$ or $x \mapsto \epsilon^{-1} \log\cosh(\epsilon x)$ where $\epsilon > 0$ [35, 42, 43]. The first choice was used by Charbonnier in 1994 [43]; we use this differentiable surrogate in chapter 5 for the isotropic TV (see section 4.2.1). Another

popular surrogate is the Huber function

$$x \mapsto \begin{cases} x^2/2 & \text{if } |x| \leq \epsilon, \\ \epsilon(|x| - \epsilon/2) & \text{else,} \end{cases} \tag{4.14}$$

which we already introduced in eq. (2.112) as the Moreau envelope (definition 2.4.17) of the absolute value. Figure 4.8 shows these surrogates for $\epsilon = 1$.

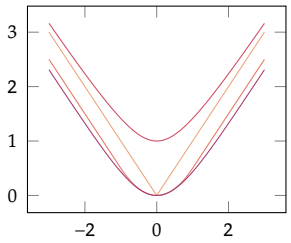

Figure 4.8: The absolute value —— and popular smooth surrogates: The Huber function ——, $\sqrt{(\cdot)^2 + 1}$ ——, and $\log \circ \cosh$ ——.

Using these approximations, we lose the interpretation of gradient *sparsity*, since these functions are not sparsity-inducing (only "small value" inducing) and only approach the absolute value as ϵ approaches zero. As they approach the absolute value, the Lipschitz constant (definition 2.1.10) of the their gradient necessarily has to explode. This affects optimization algorithms since the step size (and hence the convergence speed) is bounded by the reciprocal of the Lipschitz constant of the gradient.

To achieve fast convergence even in the nondifferentiable case, we use primal-dual optimization algorithms. Since the one-norm is a proper closed and convex function, it is equal to its biconjugate, see theorem 2.4.11. Thus, we can write

$$\lambda \|x\|_{1,1} = ((\lambda \|\cdot\|_{1,1})^{**})(x) = \max_{w \in \mathbb{R}^{m \times n \times 2}} \langle w, x \rangle - ((\lambda \|\cdot\|_{1,1})^{*})(w). \tag{4.15}$$

The conjugate of the one-norm is the indicator function of the closed unit infinity-norm ball, see theorem 2.4.10. Additionally, we use theorem 2.4.12 to identify

$$(\lambda \|\cdot\|_{1,1})^{*} = \lambda \delta_{\overline{\mathscr{B}_{\|\cdot\|_{\infty,\infty}}(0,1)}}(\cdot/\lambda) = \delta_{\overline{\mathscr{B}_{\|\cdot\|_{\infty,\infty}}(0,\lambda)}}, \tag{4.16}$$

since indicator functions are invariant with respect to scaling and the scaling of the argument can be absorbed into the radius of the norm ball. Thus,

$$\lambda \|x\|_{1,1} = \max_{w \in \mathbb{R}^{m \times n \times 2}} \langle w, x \rangle - \delta_{\overline{\mathscr{B}_{\|\cdot\|_{\infty,\infty}}(0,\lambda)}}(w). \tag{4.17}$$

The saddle point formulation of the optimization problem eq. (4.2) with the regularizer R being eq. (4.13) is

$$\min_{x \in \mathbb{R}^{m \times n}} \max_{w \in \mathbb{R}^{m \times n \times 2}} \frac{1}{2} \|Ax - y\|_2^2 + \langle w, Dx \rangle + \delta_{\overline{\mathscr{B}_{\|\cdot\|_{\infty,\infty}}(0,\lambda)}}(w). \tag{4.18}$$

This follows the structure eq. (2.118) and we can use the PDHG algorithm (algorithm 6, [36]) to solve it efficiently. In particular, the proximal map w.r.t. the indicator function is just a element-wise projection onto the interval $[-\lambda, \lambda]$,

$$w = \operatorname{prox}_{\delta_{\overline{\mathscr{B}_{\|\cdot\|_{\infty,\infty}}(0,\lambda)}}}(\bar{w}) \iff w_{i,j,k} = \max(\min(\bar{w}_{i,j,k}, \lambda), -\lambda), \tag{4.19}$$

and the proximal map w.r.t. the data fidelity term can be computed by

$$\operatorname{prox}_{\tau \|A \cdot - y\|_2^2/2}(x) = F_{\mathbb{R}}^{*}\Big((\tau M^{*} M + I)^{-1}(F_{\mathbb{R}} x + \tau M^{*} y)\Big). \tag{4.20}$$

There we used that $F_{\mathbb{R}}^{*} F_{\mathbb{R}} = I$ and note that $\tau M^{*} M + I$ is a diagonal operator that can be efficiently inverted. In the algorithm, we used the standard choice $\tau = \sigma = \frac{1}{\sqrt{8}}$.

Figure 4.9: Reconstruction using absolute gradient penalization, also known as the anisotropic total variation.

The reconstruction in fig. 4.9 shows that back-folding artifacts are almost fully removed, the reconstruction has sharp edges, and some anatomical details are recovered. However, it appears overly simplistic and fine anatomical structures such as the blood vessels in the fat tissue are almost entirely missing. This simplification results from the simple model that only considers the distribution of differences between neighboring pixels and assumes these are independent random variables.

Figure 4.6 suggests nonconvex potential functions; many have been proposed in the literature. In 1985 Geman and McClure [89] proposed the potential function

$$x \mapsto -\Big(1 + (x/\gamma)^2\Big)^{-1} \tag{4.21}$$

where $\gamma > 0$ for photon emission tomography reconstruction. Later, Huang and Mumford [120] considered the Student-t and generalized Laplace distributions. The respective potential functions are

$$x \mapsto \alpha \log\Big(1 + (x/\beta)^2\Big) \tag{4.22}$$

and

$$x \mapsto \left|\frac{x}{\alpha}\right|^\beta \tag{4.23}$$

where $\alpha, \beta > 0$ shape the potentials. While these potentials better model the empirical marginal distributions of edges in natural images, the improvements over the absolute value are marginal.[6]

6: ;)

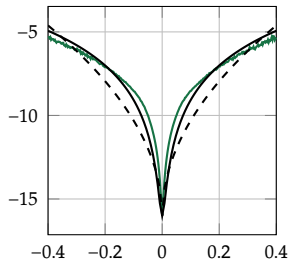

Figure 4.10: The nonconvex Student-t —— and generalized Laplace potentials - - - provide a good fit to the negative log-histograms of horizontal edges in natural images ——.

4.2.1 On the total variation

In the previous section, we discussed "ridge-type" regularizers where a scalar potential function acts on filter responses and identified the discrete *anisotropic* TV

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sum_{k=1}^{2} \big|(Dx)_{i,j,k}\big| = \lambda \|Dx\|_{1,1} \tag{4.24}$$

as an instance of these models. Here, isotropy is with respect to spatial directions, i.e. invariance with respect to rotations of the image. Due to the definition of the forward finite difference operator D, this regularizer favors grid-aligned (horizontal and vertical) structures over oblique structures. This anisotropy is essentially captured in fig. 2.1, where the one-norm ball is not radially symmetric.

To avoid these discretization artifacts, often the *isotropic* TV

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sqrt{\sum_{k=1}^{2} \big((Dx)_{i,j,k}\big)^2} = \lambda \|Dx\|_{2,1} \tag{4.25}$$

is employed. There, the gradient *magnitude* (as in the two-norm) is summed up over all pixels. It turns out that the isotropic TV is also anisotropic, see [60]. The isotropic TV is not of the form eq. (4.6), as the square root acts on the sum of the squares of the responses to two different filters, D_v and D_h. Thus, the isotropic TV is in a more general class of regularizers.

The optimization of inverse problems with isotropic TV regularization is usually performed with primal-dual methods similar to the anisotropic TV discussed previously. The associated saddle point problem problem is

$$\min_{x\in\mathbb{R}^{m\times n}} \max_{w\in\mathbb{R}^{m\times n\times 2}} \frac{1}{2}\|Ax-y\|_2^2 + \langle w, Dx\rangle + \delta_{\overline{\mathscr{B}_{\|\cdot\|_{2,\infty}}(0,\lambda)}}(w), \tag{4.26}$$

which can be solved efficiently using PDHG (algorithm 6). The proximal map with respect to the indicator function is a pixel-wise projection onto two-norm balls with radius λ,

$$w = \operatorname{prox}_{\delta_{\overline{\mathscr{B}_{\|\cdot\|_{2,\infty}}(0,\lambda)}}}(\bar{w}) \iff w_{i,j} = \frac{\bar{w}_{i,j}}{\max(1, \|\bar{w}_{i,j}\|_2/\lambda)}. \tag{4.27}$$

Figure 4.11: Reconstruction using the isotropic TV regularizer (30.35 dB). The difference to the reconstruction obtained with anisotropic TV (fig. 4.9) is almost negligible (30.32 dB).

For the image reconstruction problems considered in this thesis, the differences between the discretizations is largely irrelevant. Figure 4.11 shows the reconstruction using the isotropic TV which quantitatively improves the reconstruction from 30.32 dB to 30.35 dB in PSNR compared to the anisotropic TV. However, for more complex applications where the TV is a building block, the choice of discretization can be crucial [39].

In chapter 5, we require a smooth version of the TV due to the structure of the optimization problems. There, we use the isotropic TV with the popular smoothing proposed by Charbonnier [43]. In this case, the regularizer reads

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sqrt{\sum_{k=1}^{2} \big((Dx)_{i,j,k}\big)^2 + \epsilon^2}, \tag{4.28}$$

where $\epsilon > 0$ controls the smoothness.

4.3 Data-driven regularizers

All regularizes discussed in the previous section were concerned with modeling the histograms of differences of neighboring pixels. However, natural images are highly structured and the distribution of any pixel heavily depends on its neighbors. Therefore, a good statistical model of images should take this structure into account.

Up to now, we have only considered first order forward finite difference filters with a receptive field of two pixels. One way to account for the spatial structure in images is to use larger filters, where classical choices include derivative filters on different scales, Laplacian of Gaussian filters, and Gabor filters [255, 257]. However, it becomes increasingly difficult to choose the filters and model the potential functions manually. A more feasible approach is to derive both the filters and the potential functions from data. These types of models are called *data-driven*.

The degree to which models of natural images are data-driven can vary greatly. The regularizers discussed in the previous section are also in some sense data-driven: We motivated different choices of potential functions via the empirical

("data-driven") negative log-histogram of filter responses. Generalizing this idea, we can derive the filters themselves within the framework described by

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sum_{k=1}^{o} \phi_k\Big((K_k x)_{i,j}\Big). \tag{4.29}$$

To emphasize the image *patches* in this formulation, we write this as

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sum_{k=1}^{o} \phi_k\Big(\langle f_k, P_{i,j} x\rangle\Big). \tag{4.30}$$

Here, the convolution operators $K_1, K_2, \dots, K_o$ implement the convolution with filters $f_1, f_2, \dots, f_o$ of size $b \times b$, and $P_{i,j}$ extracts patches at pixel location (i, j) with appropriate boundary handling. When treating overlapping patches as independent, it becomes evident that under the Bayesian interpretation that we adopt in this thesis, the above is a good model of the negative log-density of images when

$$p \mapsto \lambda \sum_{k=1}^{o} \phi_k\Big(\langle f_k, p\rangle\Big) \tag{4.31}$$

is a good model of the negative log-density of image patches. This independence assumption introduces a modeling error (except in the trivial case when $b = 1$), which we address and resolve rigorously below.

A popular statistical model for image patches, which can be readily computed from a set of reference images, is given by the principal component analysis (PCA) [260]. Following this idea, we plot the principal directions of all overlapping 7×7 patches in the 300 images contained in the Berkeley segmentation data set (BSDS) [161] training and validation data set[7] in fig. 4.12. The principal directions associated with the largest singular value are low-frequency horizontal and vertical gradients. As the singular value decreases, the filters obtain more and more high-frequency components. Over all, the obtained filters closely resemble the basis images of the two-dimensional discrete cosine transform (DCT) and classical Gabor and Fourier filters. For comparison, the basis images of the DCT are shown in fig. 4.13. The observation that PCA of image patches recovers the DCT is related to the observation that the DCT is a good approximation of the Karhunen-Loève transform for stationary processes, as has been pointed out by Unser [234].

7: We use the BSDS images here because they serve as training data in chapter 6. The results would be similar for knee MRIs.

When the filters are aligned with the principal directions, it remains to detail the potential functions. Since PCA provides an orthogonal basis, optimizing the potentials involves fitting each potentials to the corresponding observed negative log-marginal. Using a suitable parametrization, such as piecewise constant functions [257] or radial basis functions [46], this task becomes a least-squares fitting on the negative log-marginals.

The undercomplete model described in chapter 6 shares similarities these ideas, although the motivation is different. This model also learns an orthogonal basis and parametrized potentials from a set of patches. After training, the learned filters and potentials in fig. 4.14 resemble the principal directions and negative log-marginals shown in fig. 4.12. The negative log-marginals of the learned filters, shown in the bottom row of fig. 4.14, closely match the learned potentials. The

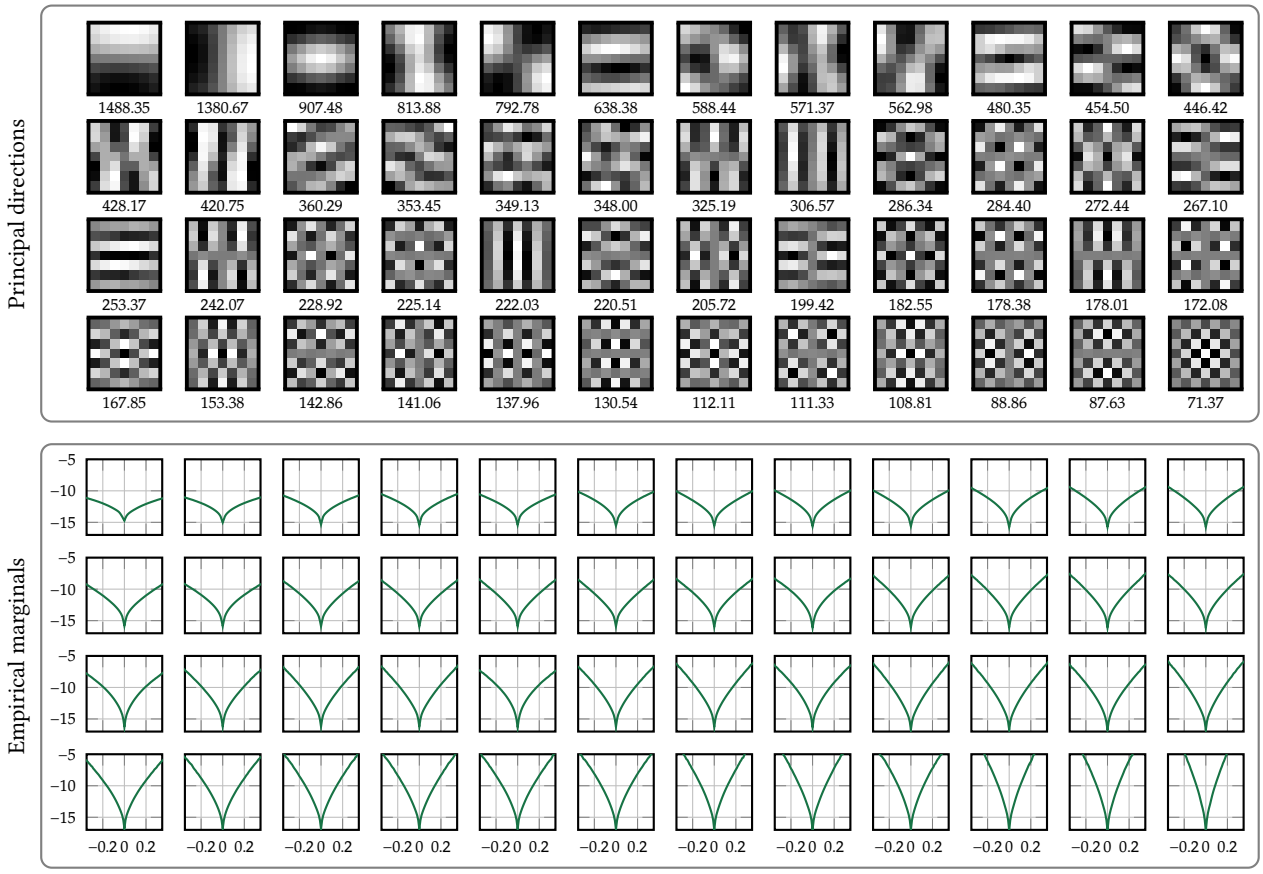

Figure 4.12: The principal directions of overlapping zero-mean patches extracted from natural images shown on top resemble the basis images of the DCT shown in fig. 4.13. The principal directions here are ordered by decreasing singular value shown below the directions; the last principal direction (which is the constant image) is omitted. On the bottom, the negative-log histogram of the filter responses is drawn.

shape of the learned negative log-marginals in fig. 4.14 remains consistent across all filters, while those of the principal directions in fig. 4.12 appear "squashed" as the singular value decreases. This difference is due to the choice of parametrization of the learned model where the filter norms change instead of the potentials being squashed. The norm of the filters is related to the smallest and largest entry in the filter; these are shown below the filters in fig. 4.14.

Due to the parametrization, the potential functions of this model are infinitely often differentiable but not convex, making the optimization problem eq. (4.2) smooth and non-convex. Algorithms for solving these problems include iPiano (see algorithm 7) or non-convex FISTA (see algorithm 5). We use the latter with a proximal step on the data fidelity which we recall from eq. (4.20), and determine the step sizes with backtracking [16, 179].

The reconstruction using the learned regularizer[8] shown in fig. 4.15 appears more natural than the one using anisotropic TV in fig. 4.9. The back-folding artifacts are nearly eliminated and more details are retrieved without producing the staircasing artifacts associated with anisotropic TV. However, the reconstruction appears over-smoothed and important anatomical structures are missing.

8: The regularizer used to produce fig. 4.15 is not exactly the one shown in fig. 4.14. Instead, to ease optimization we used the regularizer at diffusion time $t = 5 \times 10^{-5}$, see section 6.3.1.

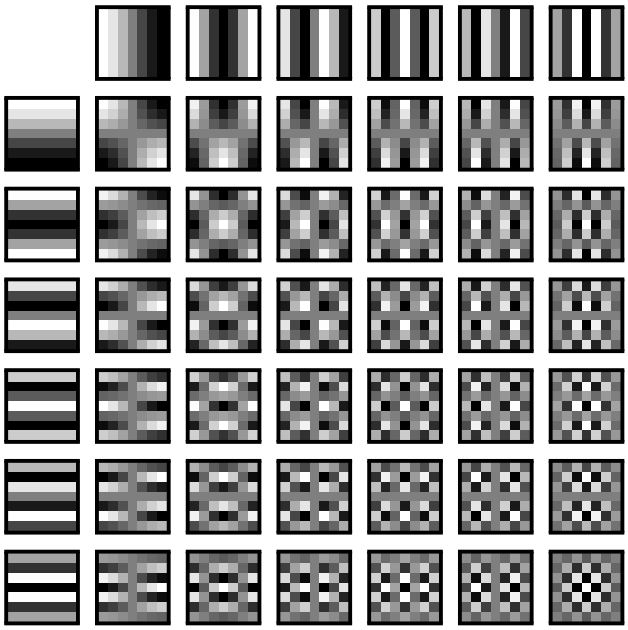

Figure 4.13: The basis images of the two-dimensional DCT, in increasing horizontal frequency (left-to-right) and vertical frequency (top-to-bottom). The basis images are normalized such that they have a minimum and maximum of −1 and 1 respectively; the first basis image (which is the constant image) is omitted.

In the next section, we discuss more expressive models that yield even better reconstructions.

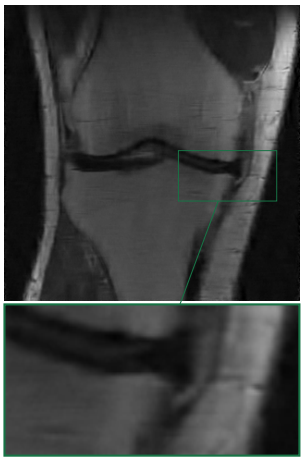

Figure 4.15: Reconstruction using the undercomplete regularizer discussed in chapter 6.

4.4 Overcomplete models and maximum entropy

In terms of the statistical model on the *image*, the subtle but crucial assumption behind the previous models is the *independence* of *patches* at different pixel locations. This introduces a modeling error, which we demonstrate with an example. Consider three-by-three images where the marginal distribution of differences of neighboring pixels follow a normal distribution with unit variance. Formally, let X be a random variable on $\mathbb{R}^{3\times 3}$ where

$$(X_{i,j} - X_{i,j+1}) \sim \mathcal{N}_{0,1} \text{ for all } i = 1,\ 2 \text{ and } 3, \text{ and } j = 1 \text{ and } 2, \tag{4.32}$$

and

$$(X_{i,j} - X_{i+1,j}) \sim \mathcal{N}_{0,1} \text{ for all } i = 1 \text{ and } 2, \text{ and } j = 1,\ 2 \text{ and } 3. \tag{4.33}$$

Using the approach from the previous sections where the density of X is modeled via the product of its marginals, we get

$$\begin{aligned} & p_{\hat{X}}(x) \\ & \propto \exp\left(-\frac{(x_{1,1} - x_{1,2})^2}{2}\right) \cdot \cdots \cdot \exp\left(-\frac{(x_{2,3} - x_{3,3})^2}{2}\right) \exp\left(-\frac{\langle 1, x\rangle^2}{2\sigma^2}\right) \\ & = \exp\left(-\frac{\|Dx\|_{2,2}^2}{2}\right) \exp\left(-\frac{\langle 1, x\rangle^2}{2\sigma^2}\right). \end{aligned} \tag{4.34}$$

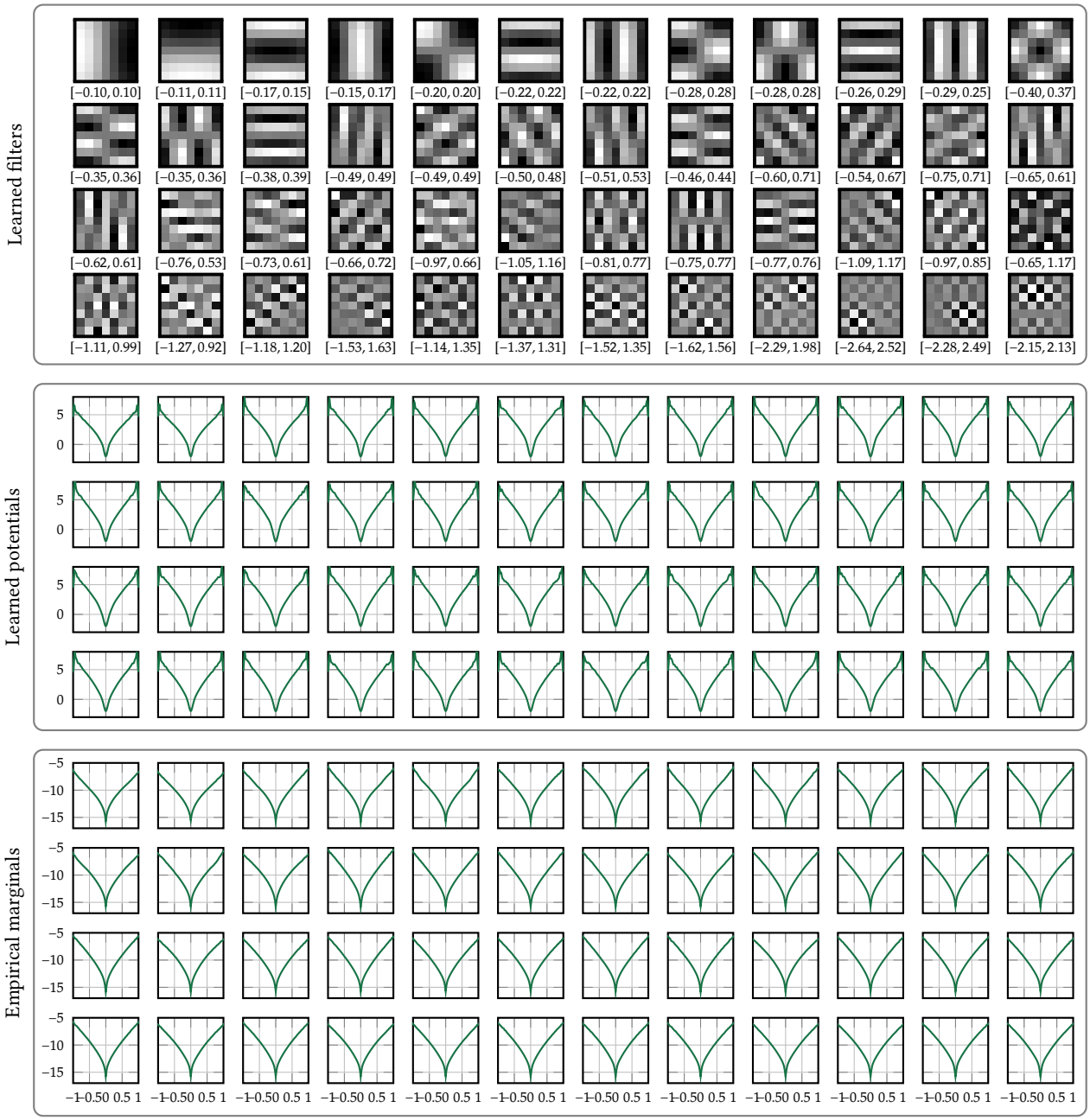

Figure 4.14: A learned undercomplete model based on filter responses. The learned filters (top) share striking similarity to the PCA basis shown in fig. 4.12. The intervals below the filters show the value of black and white respectively. The learned potential functions (middle) match the negative-log empirical marginal distributions (bottom) almost perfectly. For more details, see chapter 6 and [250].

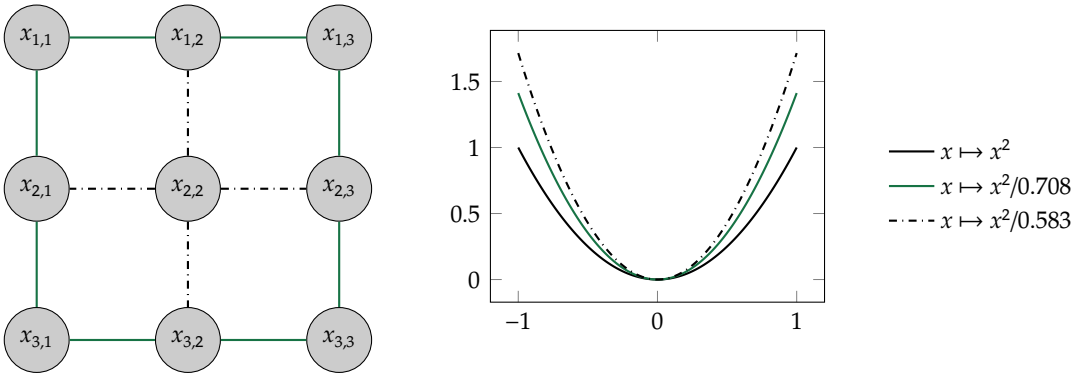

Figure 4.16: In overcomplete models, prescribing the empirical marginals leads to a modeling error: When each edge is modeled with Gaussian potentials with unit variance, the variances in the resulting product model vary spatially, and none of them is one.

Here, the density $p_{\hat{X}}$ should match the density of X. The second tie-breaking factor ensures that $p_{\hat{X}}$ is a density with respect to the Lebesgue measure on $\mathbb{R}^{3\times3}$.[9] $p_{\hat{X}}$ is the density of a normal distribution on $\mathbb{R}^{3\times3}$ with covariance

$$\left(D^*D + \frac{11^*}{\sigma^2}\right)^{-1}. \tag{4.35}$$

9: It is needed such that the covariance matrix has full rank, but does not influence the main point of the discussion.

Through a linear change of variables, the random variable $D\hat{X}$ is normally distributed with mean 0 and covariance

$$D\left(D^*D + \frac{11^*}{\sigma^2}\right)^{-1}D^*, \tag{4.36}$$

see e.g. [102, theorem 3.1] for a proof. The resulting random variables $(D\hat{X})_{i,j,k}$ have different variances, with border edges having a variance of approximately 0.708, and edges involving the central pixel approximately 0.583, as illustrated in fig. 4.16.[10] The variance of the probabilistic tie-breaking factor, σ^2, has no effect on these numbers.

10: We were unable to come up with a general analytic expression for these numbers as a function of image size and pixel index; they were computed by inverting the covariance matrix.

In this example, prescribing Gaussian marginals led to observed Gaussian marginals in the product model due to closure properties of random variables with normal distributions, albeit with different and spatially dependent variances. In the general non-Gaussian case the observed marginals are not necessarily from the same family of distributions. For instance prescribing Laplacian marginals $p_{\hat{X}}(x) \propto \exp\left(-\|Dx\|_{1,1}\right)$ through TV regularization leads to observed marginals that do not have Laplacian distributions.

The mismatch between the prescribed and observed marginals is due to loops in the graph structure of the three-by-three image. Loops in this graph structure make the model *overcomplete*, with more factors than variables. However, any interesting models of the form eq. (4.6) need necessarily be overcomplete, since each convolution operator already contributes as many factors as there are variables. Thus, when we wish to have an accurate model of the underlying density, we cannot simply prescribe the marginals.

In light of this discussion, previous regularizers should be viewed as *penalized likelihood estimators*: While they yield good quantitative results through the MAP estimate, they are generally not accurate models of the negative log-prior.

4.4.1 The correct way to think about marginals

The previous example illustrated that modeling a distribution's density as a product of its marginals leads to modeling errors. How should we correctly approach marginals in general models? We address this by framing the problem in the maximum-entropy framework, inspired by the seminal works of Zhu, Wu, and Mumford [257, 256]. This derivation is similar to those found in [240, section 3.1] and [64, section 12.1].

The setup is as follows: We are given M samples $x_1, x_2, \dots, x_M$ from an unknown distribution with density p_X, which we wish to model. Let

$$\phi = \frac{1}{M} \sum_{i=1}^{M} s(x_i) \tag{4.37}$$

represent the empirical moment of this distribution under the statistic s. Our goal is to find a distribution $\hat{p}_X$ that is consistent with this empirical moment. Among all possible distributions consistent with the empirical moment, the principle of maximum entropy (definition 2.2.18) selects the one with the highest entropy, leading to the optimization problem

$$\begin{array}{ll} \max_{\hat{p}_X} & H(\hat{p}_X) \\ \text{s.t.} & \mathbb{E}_{\hat{p}_X}[1] = 1 \\ & \mathbb{E}_{\hat{p}_X}[s] = \phi. \end{array} \tag{4.38}$$

The Lagrangian of this problem is given by

$$\mathscr{L}(\hat{p}_X, \theta_0, \theta) = H(\hat{p}_X) + \langle \theta_0, \mathbb{E}_{\hat{p}_X}[1] - 1 \rangle + \langle \theta, \mathbb{E}_{\hat{p}_X}[s] - \phi \rangle \tag{4.39}$$

where θ_0 and θ are the dual variables. It follows from the optimality condition

$$\nabla_1 \mathscr{L}(\hat{p}_X, \theta_0, \theta) = -\log \hat{p}_X(x) - 1 + \theta_0 + \langle \theta, s(x) \rangle = 0 \tag{4.40}$$

for all x, that

$$\hat{p}_X(x) = \frac{\exp(\langle \theta, s(x) \rangle)}{\exp(1 - \theta_0)}. \tag{4.41}$$

Combining this with the optimality condition on θ_0

$$\begin{aligned} \nabla_2 \mathscr{L}(\hat{p}_X, \theta_0, \theta) = \mathbb{E}_{\hat{p}_X}[1] - 1 &= \int \hat{p}_X(x)\, \mathrm{d}x - 1 \\ &= \int \frac{\exp(\langle \theta, s(x) \rangle)}{\exp(1 - \theta_0)}\, \mathrm{d}x - 1 \\ &= 0 \end{aligned} \tag{4.42}$$

implies that $\hat{p}_X$ is an exponential family distribution with normalization constant $Z(\theta) = \exp(1 - \theta_0)$.

Thus, the canonical parameters θ of the exponential family $\hat{p}_X$ are the *dual variables* in the maximum entropy problem (4.38) ensuring the moment matching condition

$$\mathbb{E}_{\hat{p}_X}[s] = \phi. \tag{4.43}$$

Hence, the potential functions should not model the negative log-marginal distributions. Instead, potential functions should be chosen such that *samples from the model reproduce the marginal distributions observed in the reference data.*

Fulfilling the moment matching condition is notoriously difficult for models of the form

$$-\log \hat{p}_X(x) = \sum_{i,j=1}^{m,n} \sum_{k=1}^{o} \phi_k((K_k x)_{i,j}) + Z(\phi_1, \phi_2, \dots, \phi_k, K_1, K_2, \dots, K_k). \tag{4.44}$$

Typically, learning these models involves computationally expensive MCMC algorithms to approximate the normalization constant Z [206, 255, 256, 257]. In contrast, chapter 6 uses score matching and ideas from diffusion models to learn a model of this type. Details are provided in chapter 6; here we present the resulting model and key differences from previously discussed models.

The learned filters and potential functions of this overcomplete model are shown in fig. 4.17. In contrast to the potentials discussed previously, they have multiple local minima and sometimes zero is not in the set of local minimizers. Thus, structures in the image can be enhanced under minimization of this regularizer.

With this regularizer, the MAP inference problem remains smooth and nonconvex and we again resort to nonconvex FISTA with proximal steps on the data term. The resulting reconstruction shown in fig. 4.15, appears sharper and recovers details such as the blood vessels in the fat tissue.

The distinction between modeling under- and overcomplete regularizers is well-known [46, 203, 257, 260]. Often, undercomplete models in this context are called "patch priors" or "patch-based priors" [260].[11] However, sometimes this distinction is overlooked, as is the case of Roth and Black's [206] Student-t potential function for their overcomplete model.

Theoretically, any probability density function is determined by *all* its marginal distributions, see [257, theorem 2].[12] Thus, models of the form eq. (4.6) can represent arbitrary distributions. However, this is only true for infinitely many (image-sized) filters. This limitation motivates exploring more general function families. In the next section, we introduce *deep neural regularizers*, where the regularizer is encoded as a deep neural network.

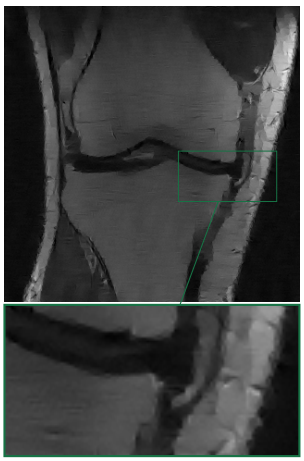

Figure 4.18: Reconstruction using the overcomplete regularizer discussed in chapter 6.

11: Patch distributions can also be modeled with overcomplete models but the challenges remain the same.

12: Their theorem is essentially the Fourier-slice theorem, which is closely related to tomography.

4.5 Deep neural regularizes

In this section, we depart from modeling probability densities trough structured models involving the product of one-dimensional experts and instead explore more general function classes. Specifically, we consider functions N from $\mathbb{R}^{m \times n}$ to $\mathbb{R}$ implemented by a NN. NNs are layered maps of the form

$$N = L_l \circ L_{l-1} \circ \cdots \circ L_2 \circ L_1 \tag{4.45}$$

where L_i is the i-th *layer* (for $i = 1, \dots, l$). Each layer is a linear operator followed by a point-wise non-linearity; detailed definitions can be found in section 3.1. From this perspective, the regularizers in discussed previously can be viewed as one-layer networks where the linear operator is a convolution and the non-linearity is

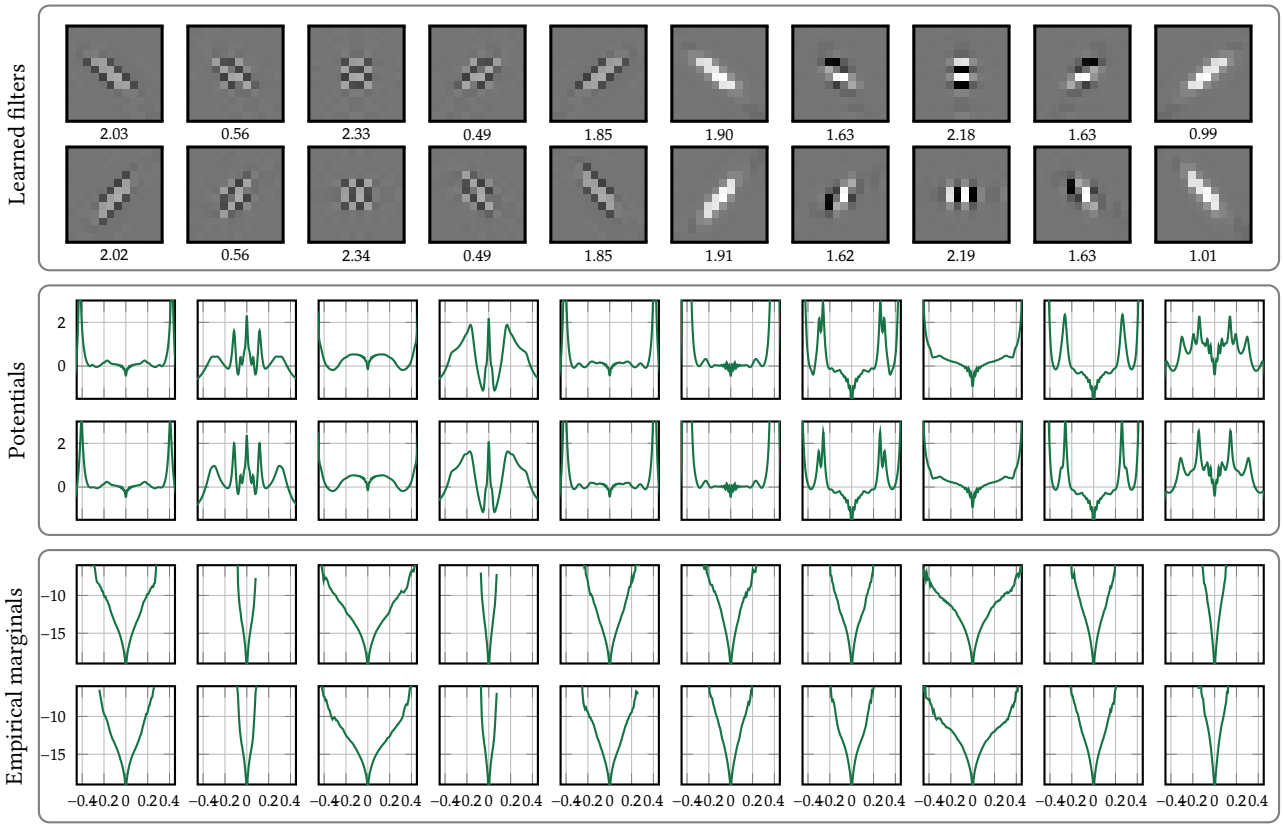

Figure 4.17: The learned overcomplete model based on shearlet responses. The filters (top) are a shearlet system and do not resemble the PCA-like filters in the undercomplete model shown in fig. 4.14. The number below the filters are their associated weights by which their output is multiplied. The learned potential functions (middle) are distinctly different from the negative-log empirical marginal distributions (bottom). In particular, they have multiple local minima such that they can enhance certain structures. Sometimes (e.g. the second, third and fourth potential functions from the left) zero is not even in the set of minima. For more details, see chapter 6 and [250].

the potential function composed with the sum over all pixels. In contrast to this, we now allow a cascade of functions to act on the features extracted in the first layer. Training of the model is discussed in detail in chapter 5. Here, we assume a pre-trained model.

Unlike previous methods, deep neural regularizers cannot be easily inspected through filter and potential function visualization. Although the first layer of the deep neural regularizer has a similar structure, it only becomes interpretable through considering the downstream cascade of functions. Instead, we we visualize *samples* drawn from the Gibbs distribution. A good regularizer accurately models the negative log-prior if samples from its Gibbs distribution are indistinguishable from samples from the reference distribution.[13] The deep neural regularizer we examine is tailored for MRI reconstruction of human knees, with the reference distribution being MRI scans from the fastMRI dataset.

13: In terms of eq. (4.38), the statistic we aim to match is the delta distribution.

To sample from the Gibbs distribution of the regularizer, we resort to MCMC sampling, the details of which are discussed in section 5.3.2. In fig. 4.19, the sampling trajectory describes a path from uniform noise to an image resembling a human knee MRI. This confirms the regularizer prefers high-level structures present in

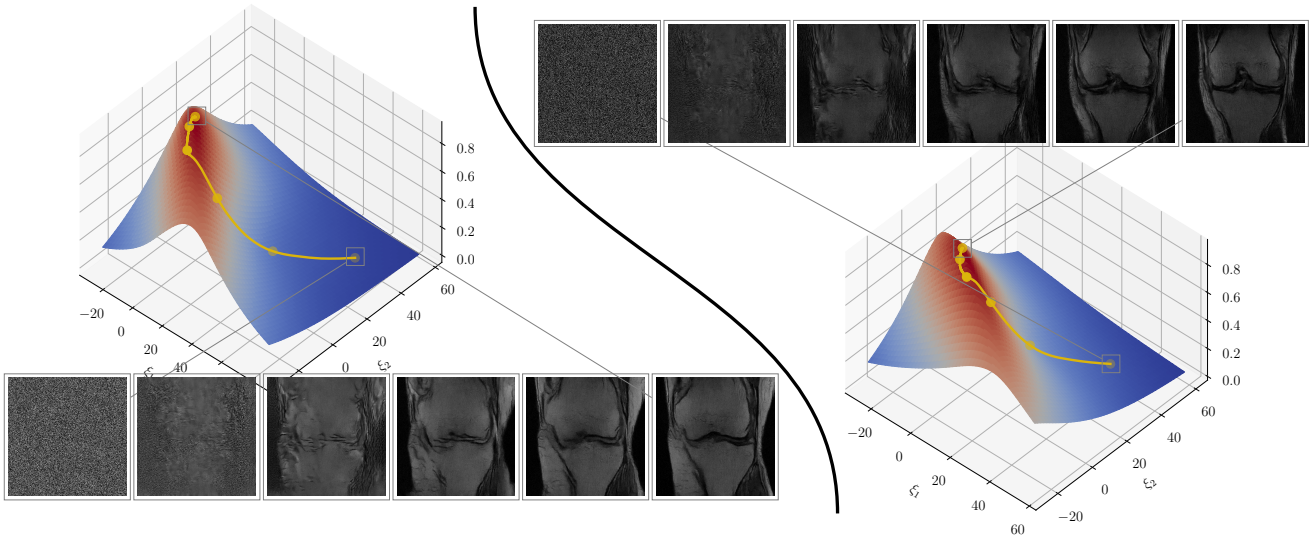

Figure 4.19: We visualize preferred structures of the regularizer via MCMC sampling. Uniform noise is unlikely under the deep neural regularizer whereas natural anatomical structures are very likely. The golden trajectory are the Langevin iterations; see the details in section 5.5.1.

the training data and that the regularizer is an accurate model of the negative log-prior.

The MAP inference problem with the deep neural regularizer is challenging. The deep neural regularizer is not guaranteed to be convex[14] and common neural network activation functions are not differentiable. Traditional methods handle the non-differentiability is via proximal maps, but computing the proximal map with respect to the neural network is as hard as the original optimization problem, making this approach impractical.

14: This is somewhat of an understatement, it is almost surely not convex.

Instead, we rely on nonconvex (nonsmooth) FISTA for optimization, where the gradient step is replaced with a step on the Clarke subgradient (definition 2.4.19) of the deep neural regularizer. Specifically, at the point of nondifferentiability of the leaky rectified linear activation (0), we choose zero out of the subdifferential. While we lack theoretical convergence guarantees, empirical results show fast convergence.

We observed that a good initialization is critical, which we achieve by running the optimization algorithm with a slightly noisy signal, matching the noise level used in Langevin iterations during training.see also section 5.3.2. Since accelerated algorithms can accumulate errors, we inject noise before gradient and function evaluation, not directly on the optimization variable. After achieving suitable initialization, we proceed with standard algorithm. In the resulting algorithm algorithm 13 we chose $N_\eta = 1000$, $\gamma_1 = 2$, $\gamma_2 = 2/3$. After N_η iterations, the algorithm reduces to nonconvex nonsmooth FISTA.

The resulting reconstruction shown in fig. 4.20 is highly satisfactory. Backfolding artifacts are entirely removed, small anatomical details such as the blood vessels in the fat tissue are restored well. There are no obvious artifacts and the

Algorithm 13: Nonconvex nonsmooth noisy FISTA with backtracking.

Input: Set $L = 1$, choose initial point $x_0 \in \mathbb{R}^{m\times n}$, number of iterations with noise $N_\eta \in \mathbb{N}$, noise level $\epsilon > 0$, backtracking multipliers $\gamma_1 \in (1,\infty), \gamma_2 \in (0,1)$.

$x^{-1} = x^0$
while *not converged* **do**
  $\bar{x}^k = x^k + \frac{1}{\sqrt{2}}\left(x^k - x^{k-1}\right)$
  $\eta \sim \mathcal{N}_{0,\epsilon I}$
  $\tilde{x}^k = \bar{x}^k + \chi_{\{1,\dots,N_\eta\}}(k)\eta$
  for *ever* **do**
    $x^{k+1} = \operatorname{prox}_{L^{-1}h}\left(\bar{x}^k - L^{-1}\nabla g(\tilde{x}^k)\right)$
    if $g(x^{k+1} + \chi_{\{1,\dots,N_\eta\}}(k)\eta) < g(\tilde{x}^k) + \langle \nabla g(\tilde{x}^k), x^{k+1} - \bar{x}^k\rangle + \frac{L}{2}\|x^{k+1} - \bar{x}^k\|_2^2$ **then**
      break
    $L = \gamma_1 L$
  $L = \gamma_2 L$
  $k = k + 1$

reconstruction looks natural over all. In summary, the data-driven deep neural regularizer effectively captures the underlying distribution. Samples drawn from its Gibbs distribution are almost indistinguishable from reference samples. This carries over to inverse problem where the regularizer is able to restore anatomical features it has learned from the reference distribution. Although without theoretical guarantees, the optimization of the deep neural regularizer is straightforward and the resulting images are artifact-free.

Combining the discussion on decision theory in chapter 1 with the Bayesian model, we expect that the MMSE will outperform the MAP estimate.[15] For the sake of simplicity, we do not show MMSE estimates in this chapter, but quantitative results in chapter 5 improve significantly from the MAP estimate to the MMSE estimate.

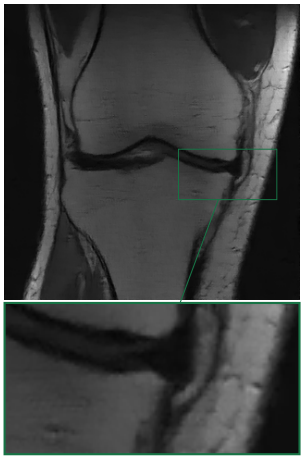

Figure 4.20: Reconstruction using the deep neural regularizer discussed in chapter 5.

15: At least quantitatively with respect to PSNR.

4.6 Conclusion

In this section, we reviewed the historical development of regularizers. Starting with classical methods such as quadratic penalization of gradients, we observed that fitting empirical statistics significantly improves the reconstructions in inverse problems. The classical anisotropic TV regularizer uses the absolute values as the best convex approximation to the empirical marginal distributions of pixel differences. The nondifferentiability of the TV motivated extensive research on nonsmooth optimization, exemplified by the development of the PDHG algorithm [36]. Compared quadratic gradient penalization, these algorithms offer superior recon-

struction quality.

The structure of the TV can be viewed as a composition of *prescribed* discrete gradient filters and absolute value potentials. *Data-driven* regularizers instead learn filters and potentials from a reference distribution. For (under-)complete models on the space of patches, PCA provides a suitable filter basis and the optimal potentials can immediately be fit to the empirical marginal distributions. This richer, data-fitting model outperforms classical TV regularization.

Extending an undercomplete model on image patches to images involves applying the learned filters as convolutions, resulting in an overcomplete model. A common assumption for natural scenes is translation invariance of their statistics, implying that the potentials should be shared amongst all responses of an image on filter. Even with shared potentials, the learning problem is significantly more complex. However, once learned, the model captures richer statistics of the reference distribution. Recently developed deep neural regularizers further elevate the reconstruction performance.

To conclude the chapter, we summarize the historical development of regularizers in fig. 4.21. Our best regularizer improves the naive reconstruction by 5.16 dB in PSNR, and improves over anisotropic TV regularization by 2.78 dB in PSNR.

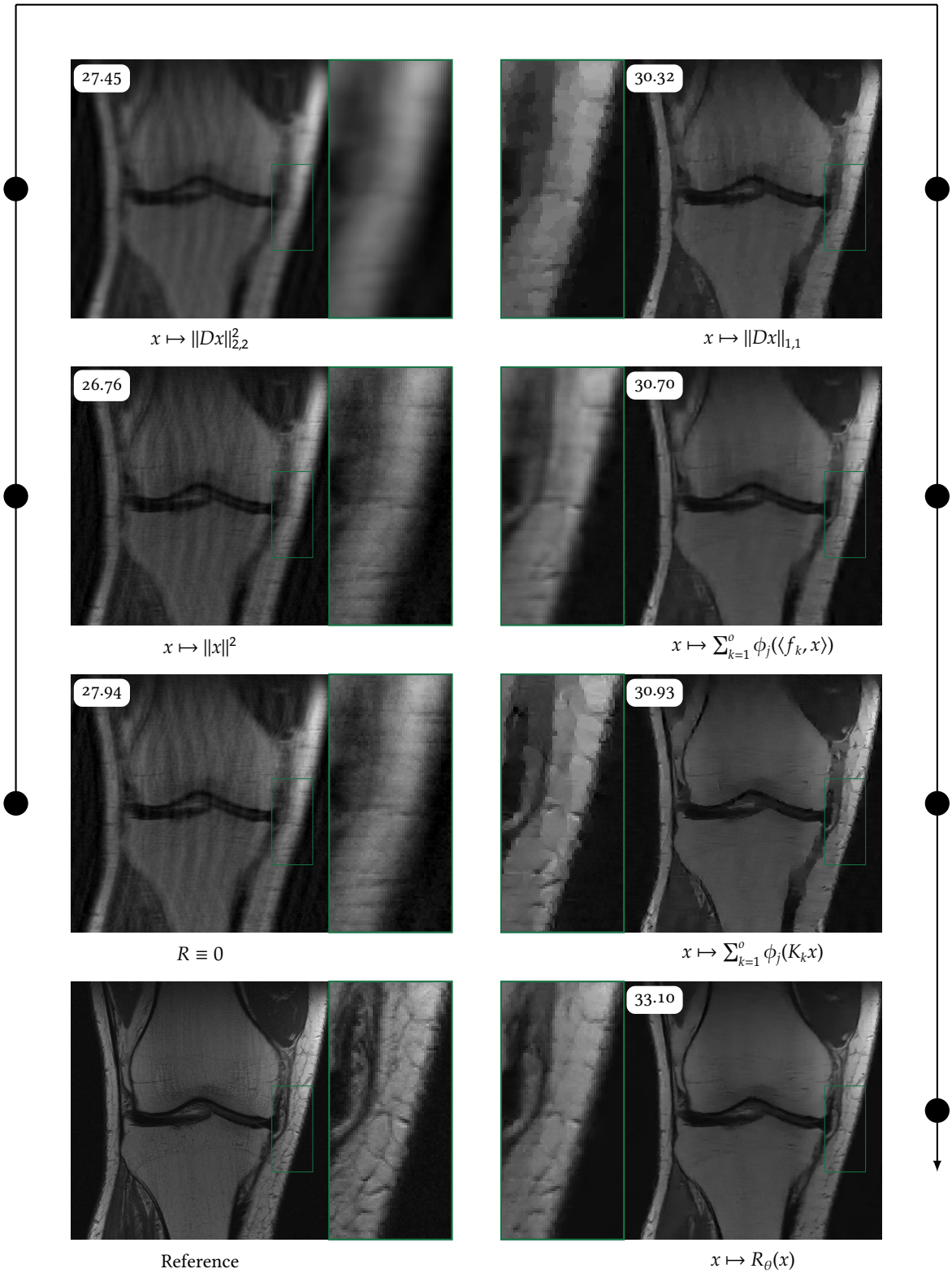

Figure 4.21: The historical development of regularizers: Total variation regularization improves over the naive reconstruction (27.94 dB) by 2.38 dB, our best model improves over the total variation regularization by 2.78 dB.

Chapter 5

Deep neural regularizers

Contents:

In the previous chapter, we outlined the historical development of regularizers in imaging. Starting from classical regularizers based on first principles, such as bounded norm or gradient sparsity of the signal, we highlighted the challenges of manually modeling image statistics and tuning parameters. This led to the idea of *learning* statistics from data.

In this chapter, we follow this idea by learning a generic deep neural regularizer that prefers structures present in the training data instead of prescribing any particular form of regularizer. We design a NN that maps from the input space, images of size $m \times n$, to the real numbers and associate the output with a Gibbs distribution. This Gibbs distribution is matched to the reference distribution by minimizing the Kullback-Leibler divergence (definition 2.2.19). This chapter largely focuses on developing this idea in the context of MRI reconstruction, as some of our contributions are specific to this domain. In particular, we propose a joint nonlinear inversion algorithm for parallel MRI that utilizes the data-driven regularizer. However, we also present some results on CT to demonstrate the approach's generality.

This chapter is structured as follows: In section 5.1 we give an overview of the problems encountered in MRI imaging and review data-driven reconstruction approaches. In particular, we discuss approaches to parallel MRI. Section 5.2 highlights issues with discriminative approaches through a striking example and section 5.3 outlines how we solve these problems through a generative approach. This section encompasses discussion about the motivation behind the architecture of the deep neural regularizer, the probabilistic setup for parameter identification, and the fast joint nonlinear inversion algorithm. The implementation details and practical considerations discussed in section 5.4 are vital for reproducing the results and serve as a reference if anything remains unclear. We show results on single-coil and multi-coil imaging in section 5.5. Additionally, this section focuses on interpretability of the results by visualizing the preferred structures of the regularizer in a data-independent analysis and through uncertainty quantification using the marginal posterior variance. Further, we demonstrate that the regularizer remains stable when confronted with out-of-distribution data and that our joint nonlinear inversion algorithm yields superior coil sensitivities compared to offline estimation algorithms when few data are available. Finally, in section 5.6

we discuss the implications and limitations of our approach and conclude that chapter in section 5.7.

This chapter is based on the following publications:

Martin Zach, Erich Kobler, and Thomas Pock. "Computed Tomography Reconstruction Using Generative Energy-Based Priors". In: *Proc. of the OAGM Workshop 2021*. Graz: Verlag der Technischen Universität Graz, Dec. 2021, pp. 52–58

Martin Zach, Florian Knoll, and Thomas Pock. "Stable Deep MRI Reconstruction Using Generative Priors". In: *IEEE Transactions on Medical Imaging* 42.12 (2023), pp. 3817–3832

Code for training, validation, and visualization, along with pre-trained models, is available at https://github.com/VLOGroup/stable-deep-mri. A related publication, showing preliminary results of joint nonlinear inversion with diffusion priors, is [81].

5.1 Introduction

MRI is a crucial imaging techniques in clinical practice [243]. In MRI, a scanner's receiver coil measures the changes in magnetism of nuclei excited by radiofrequency pulses, and these measurements correspond to Fourier coefficients of the underlying signal via the two-dimensional discrete Fourier transform. However, populating the observation with frequency data is time consuming and long examination times severely limit patient throughput. Therefore, extensive research is dedicated to reducing examination time while retaining diagnostic value.

On the hardware side, parallel MRI [202] exploits spatially varying sensitivities of coil arrays. This has become standard in clinical systems, but noise amplification limits the potential speed up with classical reconstruction techniques [199]. On the algorithmic side, compressed sensing [71] and variational approaches enable greater acceleration. As a prominent example, TV regularization has been successfully applied to parallel MRI [134].

As shown in the previous chapter, variational approaches with classical penalties like TV significantly improve reconstruction. However, modern data-driven methods such as [5, 53, 107, 169, 193, 251, 253] now outperform these traditional methods. Data-driven approaches have been applied successfully both as pre-processing steps in Fourier space [5] and as post-processing steps in image-space [251]. Variational networks (VNs) [47, 49, 107, 136] unroll an optimization algorithm to imitate the iterative reconstruction schemes, while purely data-driven methods like AUTOMAP [254] bypass physical measurement model entirely.

Despite their superior performance, data-driven approaches come with drawbacks that are especially relevant in the context of medical imaging. First, these discriminative approaches act as a point estimators, mapping data directly to reconstructions. In contrast, variational approaches enjoy a probabilistic interpretation through Bayes theorem, as discussed in section 1.2. Data are mapped to a *distribution of reconstructions* which enables uncertainty quantification. Second,

point estimators are typically tied to a particular data likelihood, which is subject to change in MRI due to different frequency selections or anatomical features. It has even been shown that reconstruction quality of some data-driven approaches deteriorates with increased data availability [9]. We discuss similar findings in section 5.2. Third, these approaches typically rely on vast amounts of paired training data. In the context of parallel MRI, this amounts to requiring fully-sampled frequency data which is extremely scarce. [1] In combination, these drawbacks severely hamper the adoption of these methods in clinical practice.

1: The target can be computed from the fully-sampled data, and the input to the reconstruction algorithm can be created retrospectively.

This chapter combines the strengths of modern data-driven methods with the benefits of classical variational approaches by learning a deep neural regularizer that serves as a plug-and-play replacement for classical variational penalties. The accompanying probabilistic interpretation allows experts to explore the posterior distribution of any reconstruction problem. We synthesizing realistic images *without* any data, demonstrating that the regularizer encodes the negative log-prior faithfully. Combining the data-driven regularizer with suitable data-likelihoods for different frequency selections achieves state-of-the-art performance. Unlike most data-driven approaches, training our model only requires access to a database of reference reconstructions.[2] In addition, we propose a joint nonlinear inversion algorithm based on iPALM [188] for parallel MRI. A sketch of our proposed approach is shown in fig. 5.1.

2: Reference reconstructions are much more abundantly available than the corresponding fully sampled data, and constructing the data from the reconstructions is non-trivial.

5.1.1 Parallel MRI

Traditional MRI encodes the signal location by adjusting precession frequencies using gradient fields in spatial directions. By exciting the nuclei with a radiofrequency pulse, the entry in the frequency plane encoded by the strength of the gradient fields could be filled. The introduction of fast gradient-echo and spin-echo sequences, such as echo-planar imaging [160] or turbo spin-echo [113], allowed greater portions of the frequency plane to be filled with one excitation pulse. To achieve even faster imaging, modern MRI systems employ coil arrays with spatially varying coil sensitivities, as pioneered by Roemer et al. [202] in 1990. The spatially varying coil sensitivities can be exploited to reduce the number of required spatial modulations.

The following overview of reconstruction algorithms for parallel MRI draws from the review paper of Blaimer et al. [22]. The conceptually simplest reconstruction algorithm for such data is parallel imaging illustrated with the partially parallel imaging with localized sensitivities (PILS) due to Griswold et al. [97]. It assumes localized, non-overlapping coil sensitivities and that the entries in the frequency plane are such that the aliased coil images from undersampled reconstructions are non-overlapping. The idealized situation for a Cartesian frequency selection with acceleration 2 is illustrated in fig. 5.2. The idealized coil sensitivities of the two receiver coils each cover half of the image and the underlying signals can be recovered exactly. However, the restriction to non-overlapping coil sensitivities and the required adaptation to the frequency selection renders the algorithm impractical.

The sensitiviy encoding (SENSE) algorithm due to Pruessmann et al. [192] extends this idea to arbitrary coil sensitivities and frequency selections. Recon-

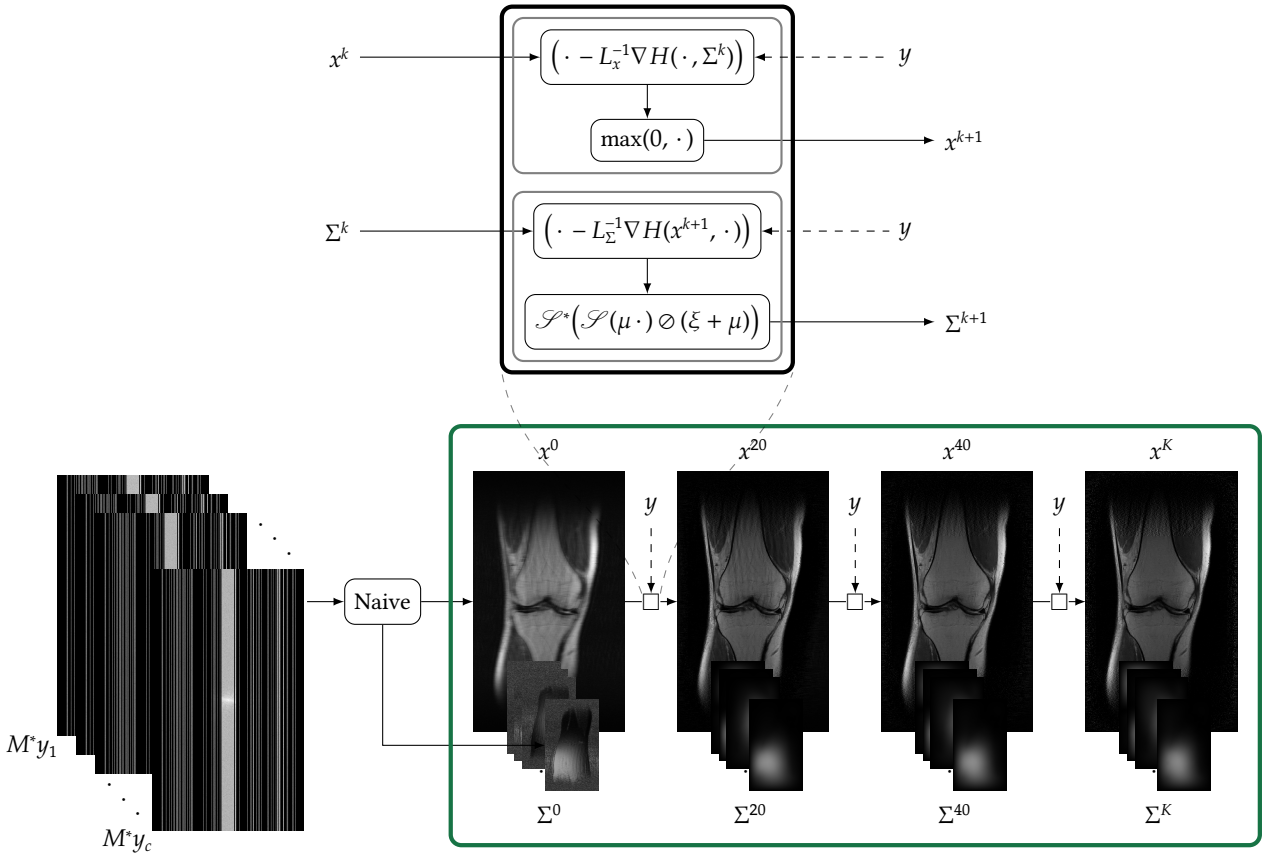

Figure 5.1: Sketch of the reconstruction algorithm: To jointly reconstruct the spin density x and the coil sensitivities $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_c)$, we impose data-fidelity, image-regularity, and coil-regularity in the iterations of iPALM [188]. The function H incorporates our data-driven regularizer acting on x.

structions are calculated by a linear weighting of the aliased coil images, where the weights are derived from the coil sensitivities. Knowledge of the coil sensitivities necessitates a calibration scan in addition to the examination, and the speed-up is limited with classical linear reconstruction techniques due to noise amplification in regions where the sensitivities have significant overlap [22, 192, 199]. The overlap is often characterized by the "geometry factor" or g-factor in the literature[199].

PILS and SENSE reconstruct the image from the aliased coil images. In contrast to this are simultaneous acquisition of spatial harmonics (SMASH) [215] (more specifically, AUTO-SMASH [124]) and generalized autocalibrating partially parallel acquisitions (GRAPPA) [96]. These techniques reconstruct missing frequency data prior to Fourier inversion with the help of autocalibration data, which are typically low-frequency data in the center of the Fourier space.

In the intersection of these methods lies the popular ESPIRiT due to Uecker at al. [232]. They demonstrated that coil sensitivities used in image domain approaches can be derived from autocalibration data used in frequency domain approaches. Today, MRI reconstruction works typically use ESPIRiT to estimate coil sensitivities from autocalibration data [4, 68, 107, 125, 153, 254]. However, this explicit modeling of the coil sensitivities from autocalibration data has the drawbacks of prolonged

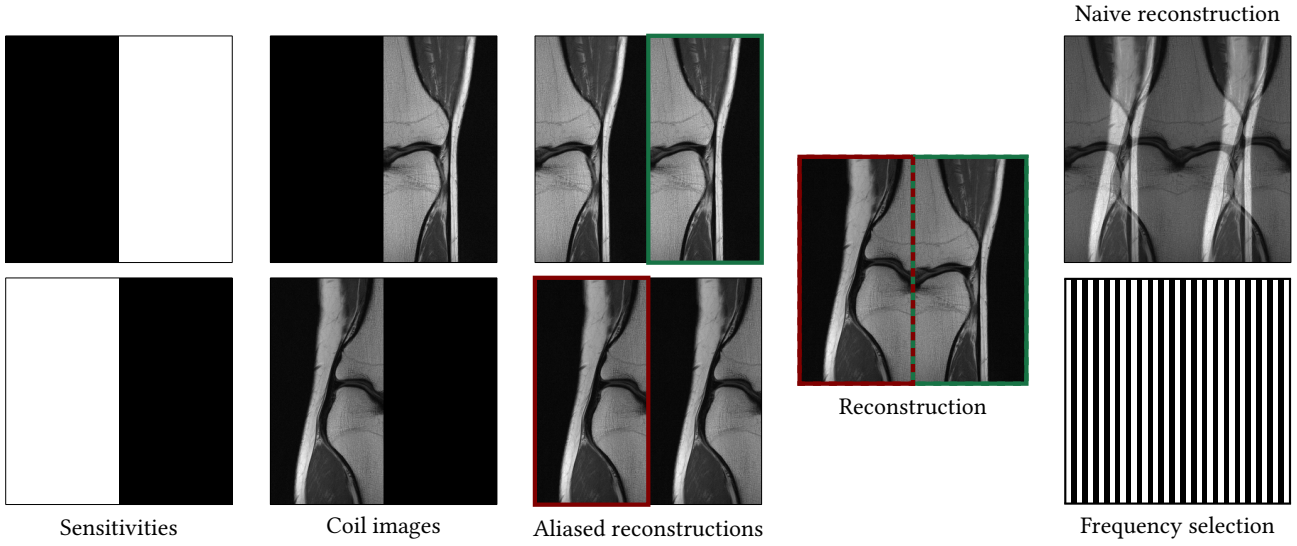

Figure 5.2: The fundamental idea behind PILS algorithm [97] in an idealized case: The first column shows idealized coil sensitivities, each covering half of the measured area. The second column shows the underlying signal as seen by the coils; the third column are the per-coil naive reconstructions where the backfolding does not overlap the image. By taking the appropriate sections of each of the aliased coil images, the signal can be reconstructed exactly. On the right, the naive reconstruction with typical back-folding artifacts due to the frequency selection on the bottom is shown.

scanning time and misalignment artifacts [134, 245]. Additionally, most of the previous works assume a Cartesian frequency selection with measurements on a Cartesian grid,[3] although radial, spiral or random sampling can be advantageous in different situations [154, 155, 233].

3: Acquiring frequencies off of the Cartesian grid is often called *non-uniform* Fourier sampling.

To overcome the limitations of previous approaches, Bauer and Kannengiesser [14] propose to jointly estimate the image with the coil sensitivities. Treating both as unknowns makes the reconstruction problem nonlinear; this is outlined later in section 5.3.3. We refer to the general principle as *joint nonlinear inversion.* They solve the joint nonlinear inversion using the iteratively regularized Gauss Newton method without any regularizers on the image or the coil sensitivities. They treat the image and the coil sensitivities equally and do not resolve the ambiguity discussed in section 5.3.3 and consequently observe that the reconstruction is extremely sensitive to initialization, choice of regularization, and number of iterations of the algorithm.

Ying and Sheng independently proposed joint nonlinear inversion in [245] and resolve the ambiguity between the coil sensitivities and the image by observing that the coil sensitivities are *much smoother than the image.* They explicitly parametrize the coil sensitivities with low-order polynomials and solved the inversion using alternating minimization. Uecker et al. [233] built on this work but revisit the iteratively regularized Gauss Newton algorithm with smoothness of the coil sensitivities enforced through appropriate Sobolev norm penalties. This approach was extended to include classic variational penalties on the image by Knoll et al. [134] who report improved quantitative results using TV and second-order total generalized variation (TGV) regularization.

The work of Knoll et al. [134] highlights the flexibility of formulating reconstruction as a nonlinear inverse problem. Specifically, it accounts for smoothness penalties on coil sensitivities and sophisticated image regularization. It is also not tied to Cartesian frequency selection and can account for nonuniform sampling. However, their algorithm requires choosing multiple regularization parameters at each step and involves nontrivial subproblems to account for the variational penalties.

Our approach builds on the work of Knoll et al. [134] but differs as follows: Instead of handcrafted regularizers, we use modern generative learning techniques to learn an expressive regularizer from data, which leads to state-of-the-art reconstructions. We enforce smoothness of the coil sensitivities quadratic gradient penalization gradient which simultaneously resolves ambiguities. Instead of the iteratively regularized Gauss-Newton algorithm with non-trivial sub-problems, we employ the iPALM algorithm [188] (algorithm 8) for optimization, ensuring convergence and requiring tuning of only two parameters (see section 5.3.3). Our reconstruction algorithm is sketched in fig. 5.1 and takes about 5 s on consumer hardware.

For completeness, we mention that the end-to-end variational network of [222] also estimates the sensitivities jointly with the image. However, their approach learns a mapping from frequency-space to image-space *discriminatively*, using the estimated coil sensitivities to enforce data fidelity. Thus, the network only works well with a particular frequency selection and coil configuration. In contrast, we learn image features *generatively* and impose handcrafted regularity onto the coil sensitivities.

5.1.2 Related work

In this section we review related work on data-driven MRI reconstruction, focusing on methods similar to our proposed approach. For a broader overview of data-driven MRI reconstruction, refer to [252]. Antun et al. [9] provide a comprehensive overview of the potential risks of using modern techniques for medical image reconstruction.

Guan at al. [98] explored learning a deep neural regularizer for MRI simultaneously to our work. Unlike our approach but similar to the diffusion-based approaches discussed below, they apply their regularizer to the individual coil images in the reconstruction algorithm, requiring multiple evaluations of the gradient of the regularizer per iteration. In addition, they estimate the sensitivity maps via ESPIRiT.[4] In contrast, our joint nonlinear inversion algorithm eliminated the need for offline sensitivity estimation and requires only one gradient evaluation per iteration.

4: Their exact reconstruction algorithm is unclear from their exposition.

The authors extended their work in [231] by learning an energy-based model (EBM) in image and frequency domain. This eliminates the need for sensitivity estimation but requires training two independent networks that need to be balanced at inference time. Additionally, the frequency-domain EBM requires fully-sampled reference data, which is scarcely available. Our method only requires reference DICOM (magnitude) images to train one regularizer.

Diffusion models are closely related to deep neural regularizers. As discussed in chapter 6, the score of a diffusion model "at time o" and the gradient of our regularizer model the same object: The gradient of the negative log-prior. Solving inverse problems with diffusion models is computationally demanding as it requires solving a SDE with high accuracy, typically requiring thousand of gradient evaluations [53]. Additionally, it is not clear how to optimally incorporate data-fidelity into the SDE. Proposed solutions include data projection [221], annealed Langevin dynamics [125] and diffusion posterior sampling [54]. These methods require parameter tuning, in some cases at each step of the reverse diffusion [125, 54]. Feng at al. [83] argue that none of these methods generate samples from the true posterior distribution and propose augmenting the score models with normalizing flows. While their inference algorithm is parameter-free, the approach is opaque due to the introduction of the normalizing flow and is still computationally demanding.

Deep neural regularizers enjoy a natural probabilistic interpretation through Bayes theorem as discussed in section 1.2. MAP inference does not require solving an SDE and can be achieved by efficient through efficient optimization algorithms. Access to the function value (not just the gradient) allows practical applications like inspecting the regularization landscape (see fig. 4.19) and using backtracking in optimization algorithms [188].

Works by Luo et al. [153] and Jalal et al. [125] tackle parallel imaging by offline sensitivity estimation, which has drawbacks outlined in section 5.1.1. Chung et al. [53] propose reconstructing individual coil images using a model trained solely on root-sum-of-squares (RSS) reconstructions. While the results are impressive, the computational cost depends on the number of coils and the authors report reconstruction times of up to 10 min. In our joint nonlinear inversion algorithm, the network's gradient is only evaluated once per iteration, regardless of the number of coils. Imposing spatial regularity on the coils is extremely fast, requiring very few fast Fourier transforms at each iteration (see section 5.3.3).

Finally, our generative approach has a practical benefit in the context of MRI reconstruction: although we require *reconstructions* from fully sampled data for training, it does not require fully sampled reference *data*. This distinction is important because reconstructions are more readily available in hospitals picture archiving and communication systems [251]. Constructing synthetic fully-sampled data from reference reconstructions is challenging due to the non-trivial interaction between coil sensitivities and the signal. Most popular data-driven reconstruction algorithms like the end-to-end VN [222], AUTOMAP [254], and dual-domain approaches [231, 253] require image-data pairs for training. Data-driven methods that need reference images include generative adversarial networks (GANs) [169]. GANs suffer from the range-dilemma [26], and authors have proposed to optimize the parameters of the GAN at inference time [169], effectively turning them into deep image priors [144]. Deep image priors are challenging to optimize and don't offer a natural probabilistic interpretation.

5.2 The pitfalls of discriminative signal recovery

The classical way to utilizing deep learning methods in inverse problems involves learning a map from the data to the reconstruction *discriminatively*. However, this approach has significant drawbacks, particularly in medical imaging. In this section we outline these drawbacks with a striking example.

5: We keep the setup general and don't specify any dimensionality or specific forward operator, A, here.

The setup that we consider is as follows:[5] We assume the signal to be reconstructed is an instance of a random variable X with density p_X. The data is summarized by a random variable related to the signals via

$$Y = AX + N, \tag{5.1}$$

where A is a linear operator with appropriate dimensions and N is Gaussian noise with known variance. Thus, the signal and the data form a joint distribution $p_{X\times Y}$ with the relationship between X and Y given by eq. (5.1).

The discriminative approach in data-driven reconstruction methods involves a parametrized map $f(\cdot, \theta)$ that takes an instance of Y and outputs an estimation of X. The optimal parameters of the map are identified via the optimization problem

$$\arg\min_{\theta} \mathbb{E}_{(X,Y)\sim p_{X\times Y}}\Big[l(f(Y,\theta), X)\Big], \tag{5.2}$$

6: Here, the lower case variables are instantiations of the corresponding random variables.

where l is a loss function comparing the reconstruction $f(y, \theta)$ with the reference signal x.[6] Common loss functions include the negative SSIM (definition 2.5.4), and the two-norm or the one-norm of the difference. This paradigm is extremely popular and includes pre-processing approaches [108], post-processing approaches [251], learned iterative networks [107], and AUTOMAP [254].

To emphasize the drawback of this approach, we consider the setup and baseline model from the original fastMRI publication [251]. Here, X is a random variable in $\mathbb{R}^{320\times 320}$ and $A = M_K F \colon 320 \times 320 \to \mathbb{C}^{320\times k}$ is the Fourier transformation F followed by a Cartesian frequency selection, selecting $k \in \mathbb{N}$ lines and 8 % autocalibration lines, encoded in M_k; this frequency selection operator is exemplified in the second column of fig. 5.10. The map $f_k \colon \mathbb{C}^{320\times k} \to \mathbb{R}^{320\times 320}$ is given by

$$f_k(\cdot, \theta) = \mathrm{UNet}(\cdot, \theta) \circ |\cdot| \circ F^* M_k^* \tag{5.3}$$

Figure 5.4: Reconstruction with radial frequency selection; the white arrow indicates obvious artifacts.

where $|\cdot|$ is the complex modulus acting element-wise on its argument and $\mathrm{UNet}(\cdot, \theta) : \mathbb{R}^{320\times 320} \to \mathbb{R}^{320\times 320}$ is a parametrized UNet. This is a post-processing approach refining the zero-filling reconstruction with a learned UNet. In the training, l is the one-norm and $k = 80$ and the acceleration is 4.

Let x be an image from the reference distribution, and let $y_k = A_k x + n$ with n Gaussian noise. $k \in \mathbb{N}$ indicates the number of frequency lines that are available in the data. For instance, when $k = 320$ the frequency space is completely filled. Figure 5.3 shows the PSNR of the reconstruction $f_k(y_k, \theta)$ as k varies. Notably, we observe that the PSNR decreases as k increases after around $k = 140$. The reconstruction of the discriminative reconstruction network *becomes worse as more data become available*. This is *not* an out-of-distribution application of the reconstruction network: The underlying signal *is* drawn from the reference distribution p_X, but we slightly change its relationship with the data y_k.

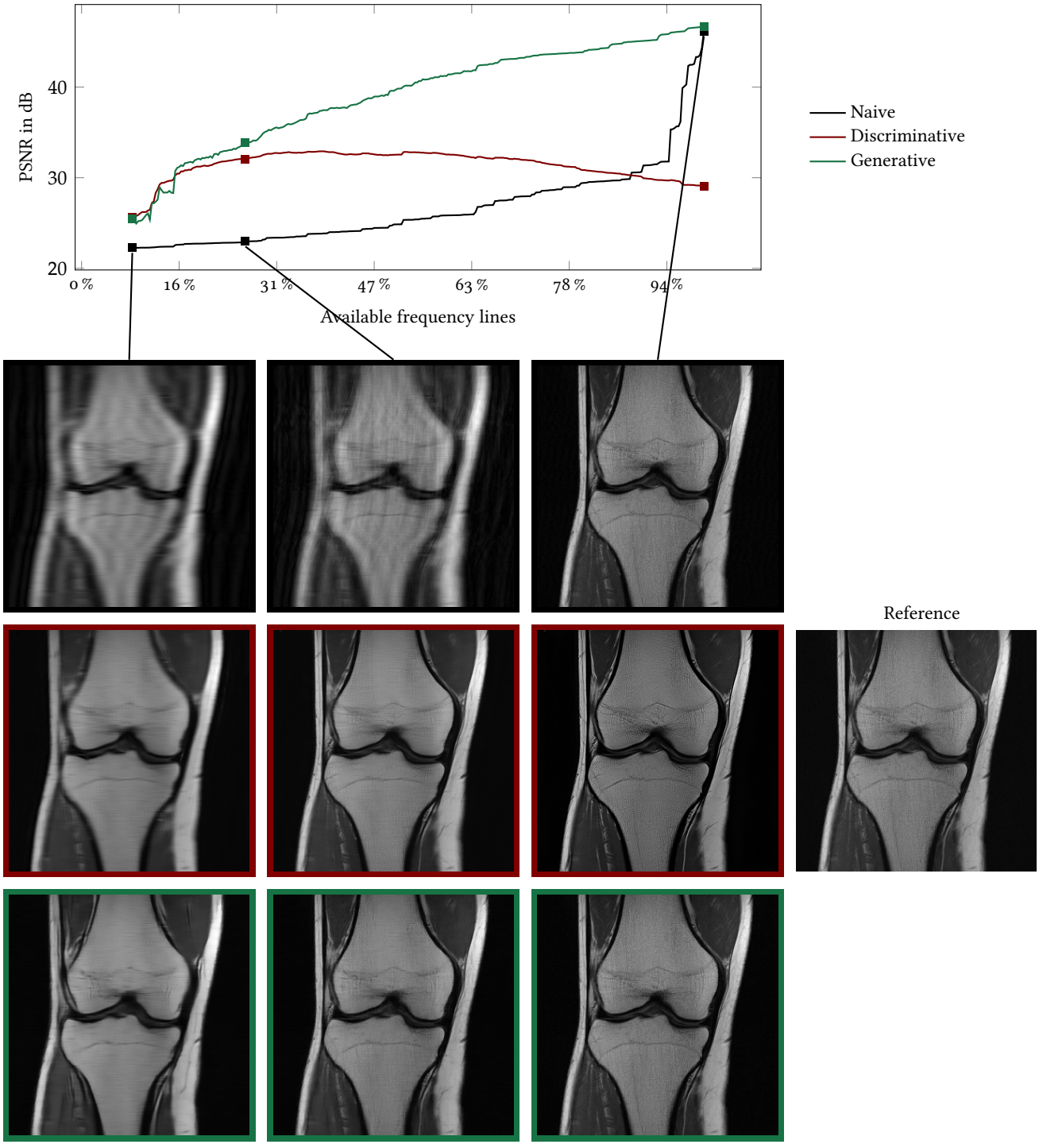

Figure 5.3: The signal recovered by the discriminative method deteriorates as more frequency lines become available: The reconstructions become overly sharp with accentuated details; when all data are available (320 lines), the reconstruction is 3.46 dB worse in PSNR than the reconstruction using 80 available frequency lines, which corresponds to the training setup. In contrast, due to the separation of likelihood and prior, the reconstruction of the generative approach improves with the availability of the data.

While the decrease in PSNR is significant, the resulting reconstruction remains artifact-free when using Cartesian frequency selection. However, switching to a radial frequency selection leads to reconstructions with obvious artifacts. The reconstruction in fig. 5.4 suggests that the network cannot distinguish between anatomical features and artifacts in the zero-filled reconstruction. This issue can even occur with Cartesian frequency selection (see fig. 5.11).

The major drawback of discriminative reconstruction methods is their performance deterioration when the acquisition operator changes, even when the underlying reference distribution remains unchanged. This is due to the acquisition operator being integrated in the training setup, causing the learned components to entangle features of the reference data and characteristics of the acquisition operator.

Despite this drawback, data-driven reconstruction methods offer significant benefits when the data matches the training setup, outperforming hand-crafted regularizers. In the remainder of this chapter, we explore how modern generative learning methods can be used to learn the reference distribution. We adopt a Bayesian approach, separating the likelihood (which uses the acquisition operator) from the prior assumptions on the reference distribution. We learn the latter using generative learning methods and utilize the learned model in a Bayesian variational reconstruction framework.

5.3 Methods

In this section, we first detail the neural network architecture used as the regularizer. Then, we discuss how the parameters of this regularizer can be learned from the reference distribution. To use the regularizer in a joint nonlinear inversion framework, we derive an algorithm based on iPALM that enforces nonnegativity of the spin density and spatial regularity of the coil sensitivities.

5.3.1 Network architecture

Typical regularizers used in imaging applications are of the ridge-type, as defined in eq. (4.6). These regularizers apply potentials to the responses of linear filters, summing over all filters and pixels. This structure makes the regularizers translation invariant, which is desired in natural image reconstruction where the probability of a feature is independent of the spatial location.[7] However, this is not the case for for the highly structures MRI images of the human knee where specific structures like vertically aligned bones surrounded by muscle- and fat tissue are expected. In addition, these types of images usually come with nonlocal dependencies: If a large amount of fat tissue is observed medially from the bone, usually a large amount of fat tissue is present laterally, see the example in fig. 5.5. Classical regularizes are limited in their ability to resolve these large scale dependencies.

7: This class of images is often referred to as texture-like.

Ridge-type regularizers are also often preferred due to their interpretability: Filters and corresponding potentials can be visualized, showing experts which structures are enhanced or diminished. The architecture that we will now describe

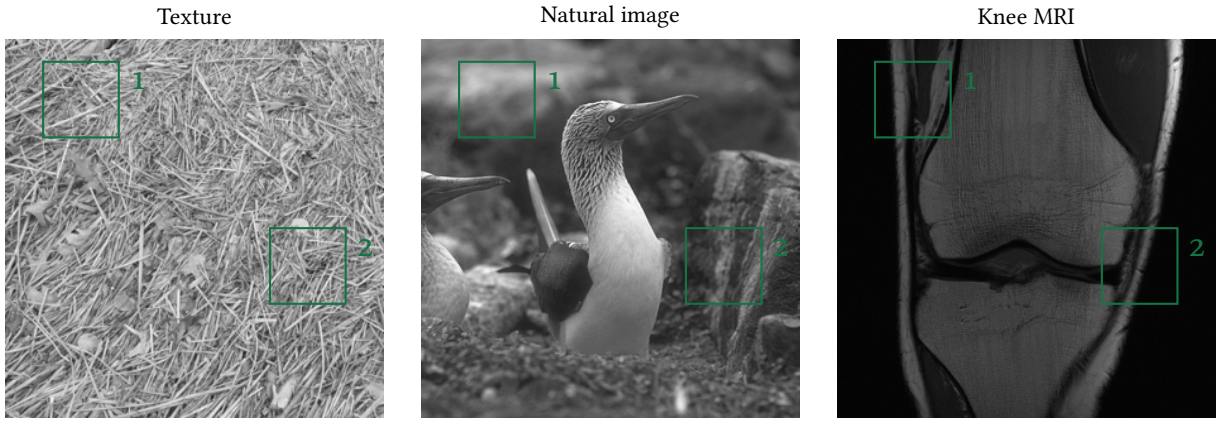

Figure 5.5: Translation (in)variance and (non)local dependencies in images: In the texture-like image on the left (`matted_0166.jpg` from the describable textures dataset [56]), the probability of a feature does not depend on the spatial location. This is also a common assumption for natural images as the one in the middle (`103070.jpg` from BSDS [161]), since the image we see is a (more or less) arbitrary crop of the infinitely extending scene. On the contrary, for the knee image on the right (the central slice of `file1001057.h5` from the fastMRI knee dataset [251]), we always see the femur on top of the tibia, which are both surrounded by muscle- and fat tissue. In addition, knee MRI images carry nonlocal dependencies: Since there is very little fat tissue medially (e.g. □ 1), we also expect very little fat tissue laterally (e.g. □ 2).

can not be interpreted in this way. However, in section 5.5.1 we will provide an alternative way of visualizing preferred structures by means of drawing samples from the Gibbs distribution.

In this chapter, we depart from the ridge-type architecture, opting for a more general architecture that is *not* translation invariant and *can* resolve nonlocal dependencies. We model the regularizer $R(\cdot\,, \theta)\colon \mathbb{R}^{m\times n} \to \mathbb{R}$ as a cascade of convolutional layers with leaky rectified linear unit activation functions. The parameters θ of the regularizer are made explicit in the second argument; they are detailed below.

Formally, the regularizer has the structure

$$R(\cdot\,, \theta) = |\cdot| \circ W_{l+1} \circ L_l \circ L_{l-1} \circ \ldots \circ L_2 \circ L_1. \tag{5.4}$$

All l layers $L_1, L_2, \ldots, L_l$ follow the structure

$$\begin{aligned} L_i\colon \mathbb{R}^{n_i\times n_i\times c_i} &\to \mathbb{R}^{n_{i+1}\times n_{i+1}\times c_{i+1}}, \\ x &\mapsto \nearrow(\widetilde{W}_i \nearrow(W_i x)). \end{aligned} \tag{5.5}$$

Here, leaky rectified linear unit activation function $\nearrow\colon x \mapsto \max\{\gamma x, x\}$ is applied point-wise and has a leakage parameter $\gamma > 0$. The linear operator $W_i\colon \mathbb{R}^{n_i\times n_i\times c_i} \to \mathbb{R}^{n_i\times n_i\times c_i}$ encodes[8] a convolution with a 3×3 kernel with zero-boundary conditions, while the linear operator $\widetilde{W}_i\colon \mathbb{R}^{n_i\times n_i\times c_i} \to \mathbb{R}^{n_{i+1}\times n_{i+1}\times c_{i+1}}$ encodes a *strided* convolution with stride 2, reducing the feature map dimensions progressively. Thus, $n_{i+1} = n_i/2$ and $n_1 = n$.[9] The final linear operator $W_{l+1}\colon \mathbb{R}^{n_{l+1}\times n_{l+1}\times c_{l+1}} \to \mathbb{R}$ is a dense, fully learnable layer that maps to a real number, essentially encoding a learned weighted sum. Denoting with $w_i \in \mathbb{R}^{3\times 3\times c_i\times c_i}$ and $\widetilde{w}_i \in \mathbb{R}^{3\times 3\times c_i\times c_{i+1}}$ the learnable filters of the convolution operators W_i and $\widetilde{W}_i$ respectively, the learnable parameters in our model are $\theta = \{w_l, \widetilde{w}_l\}_{i=1}^{l} \cup \{W_{l+1}\}$.

8: In the special case $i = 1$, $W_1\colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n\times c_1}$.

9: In our setting, this always results in a natural number. We discuss the choice of the number of layers l and the progression of the number of channels c_i in the implementation details.

Unlike ridge-type architectures where feature map responses are summed, our model progressively downsamples the feature maps via learned convolutions and nonlinearities until they are reduced to a scalar via a learned weighted sum. This regularizer is not translation invariant and it can prefer certain structures in certain regions, accommodation nonlocal structure arrangements.

The architecture of our regularizer can also be motived by considering its gradient with respect to the input image.[10] To derive the gradient, we note that $L_i = \nearrow \circ \widetilde{W}_i \circ \nearrow \circ W_i$ and denote the *activations* $\nearrow'$ of the point-wise activation functions after the convolution operators W_i and $\widetilde{W}_i$ as

$$a_i(x) = \Big(\nearrow' \circ W_i \circ L_{i-1} \circ \ldots \circ L_1\Big)(x) \tag{5.6}$$

and

$$\tilde{a}_i(x) = \Big(\nearrow' \circ \widetilde{W}_i \circ \nearrow \circ W_i \circ L_{i-1} \circ \ldots \circ L_1\Big)(x). \tag{5.7}$$

We denote the activation of the final fully connected layer as

$$a_{l+1}(x) = \Big(|\cdot|' \circ W_{l+1} \circ L_l \circ \ldots \circ L_1\Big)(x). \tag{5.8}$$

Then, by backpropagation

$$\begin{aligned}&(\nabla R(\,\cdot\,,\theta))(x) = \\ &W_1^*\Big(a_1(x) \cdot \widetilde{W}_1^*\Big(\ldots\Big(\tilde{a}_{l-1}(x) \cdot W_l^*\Big(a_l(x) \cdot \widetilde{W}_l^*\Big(\tilde{a}_l(x) \cdot W_{l+1}^* a_{l+1}(x)\Big)\Big)\Big)\ldots\Big)\Big)\end{aligned} \tag{5.9}$$

which has the structure of a UNet.

The UNet architecture, introduced by Ronneberger et al. in 2015 [205], has gained immense popularity in many applications requiring maps from the signal space to itself.[11] The key idea behind UNet is to capture context in the contracting path (via consecutive strided convolutions) and subsequently retrieve details in the expanding path (via consecutive adjoint strided convolutions). To ensure detailed recovery in the expanding path and avoid loss of information, UNet employs *skip connections* that retrieve features from the layers in the contracting path. This is typically realized by concatenating the feature maps.

In modeling the gradient of a probability density, UNet-type architectures have become the standard backbone for diffusion models.[12] Works include the seed papers on diffusion models [217, 219] and recent works that utilize diffusion models to solve inverse problems [53, 125, 55, 54]. A common criticism of these models (that we also make in this chapter and in chapter 6) is that these networks are in general not a gradient. This issue was noted in the paper on sliced score matching [220, footnote 1 on page 4], a precursor to diffusion model papers, where the authors argued that the lack of gradient property does not significantly impact practical applications.

In contrast, our approach naturally retrieves a UNet-type model as the gradient of our scalar function. The expanding path in each layer uses the adjoint operators of the corresponding contracting path layer. The skip connections are realized by scalar multiplication of the feature maps in the expanding path by the activation from the contracting path. This structure is schematically illustrated in fig. 5.6.

10: The network is not differentiable; the gradient can be understood either via a differentiable surrogate of the rectified linear unit or in the sense of the Clarke subdifferential definition 2.4.19.

11: The output space can also be "slightly different". This is the case for instance in dense labeling or segmentation.

12: We discuss diffusion models in more detail in chapter 6. The relationship between our approach and diffusion models is covered superficially in section 5.1.2

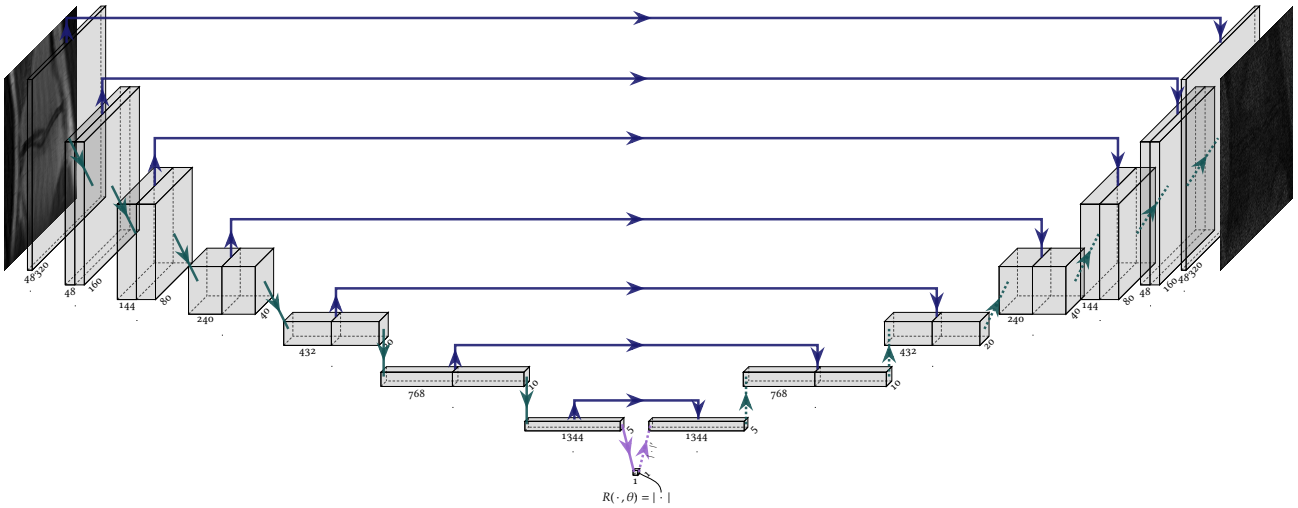

Figure 5.6: The gradient of the regularizer has a UNet structure. At the i-th layer, the symbol → is the map $x \mapsto \nearrow(\widetilde{W}_i x)$, and - → is the map $x \mapsto \widetilde{W}_i^* x$; the convolutions with unit stride are between the gray blocks and (for the sake of simplicity) are not endowed with an arrow. The skip connection → is understood as the map $x \mapsto \nearrow'(x)$, the result of which pointwise multiplies the features in the expanding path. The function value of the network is given as the absolute value of the scalar-valued feature map at the bottom, which is computed via the learned weighed sum (the fully connected layer W_{l+1}) - →. The annotations show the size of the feature maps. The illustration is slightly simplified; we neither explicitly show the nonstrided convolutions (these would be arrows between the adjacent blocks) nor their corresponding skip connections.

While our approach imposes some restrictions compared to a classical UNet, it accurately models the gradient of the log-prior, as empirically demonstrated in fig. 5.9. Moreover, our model allows for the computation of function values, not just gradients, while preserving classical symmetry properties of gradients. In the next section, we discuss how to learn the parameters of the regularizer such that is accurately models the negative log-prior.

5.3.2 Parameter identification

In the strict Bayesian view of inverse problems that we adopt in this thesis, we assume that in any reconstruction problem, the underlying signal is a realization of a random variable X with density p_X. In this chapter, the random variable has values in $\mathbb{R}^{m\times n}$ and represents the magnitude of MRI scans of the human knee, see the details in section 5.4. In this view, the regularizer in the inverse problem models the negative log-prior, up to an additive constant.

Under mild conditions we can associate any regularizer $R(\,\cdot\,,\theta)$ with a density

$$\hat{p}_X(x) = \frac{\exp\bigl(-R(x,\theta)\bigr)}{\int_{\mathbb{R}^{m\times n}} \exp\bigl(-R(\xi,\theta)\bigr)\,\mathrm{d}\xi}. \tag{5.10}$$

When $R(\,\cdot\,,\theta)$ is of ridge-type, then this construction is called a *Gibbs distribution*, see [257, theorem 1 and the two preceding definitions] and [90, page 5]. Modern literature extend this to more general functions, including neural networks, which can also induce a Gibbs distribution via the equation above [174, section 3.1], [175,

section 2.1].

$R(\cdot, \theta)$ admits a Gibbs density if it is bounded from below and grows "fast enough" radially, such that $\int_{\mathbb{R}^{m\times n}} \exp\big(-R(\xi, \theta)\big)\, \mathrm{d}\xi < \infty$. We ensure boundedness from below by using an absolute value activation in the last layer. The growth condition can be achieved by adding a small quadratic term to the regularizer[13] but in practice this is unnecessary.

13: This is also a popular strategy to assign a proper distributions to ridge-type regularizers, see e.g. [209, eq. (1)] and [85, eq. (6)].

Identifying the optimal parameters of the regularizer becomes a problem of density fitting, and we adopt the popular approach of minimizing the Kullback-Leibler divergence from p_X to $\hat{p}_X$. The Kullback-Leibler divergence is popular due to the links to maximum likelihood and maximum entropy estimation. Seminal papers that utilize it include [116, 206, 257]; it has been called the "standard" divergence by Teh, Welling, Osindero and Hinton in [225]. The Kullback-Leibler divergence from p_X to $\hat{p}_X$ (see definition 2.2.19) reads

$$(p_X \,\|\, \hat{p}_X)_{\mathrm{KL}} = \int_{\mathbb{R}^{m\times n}} p_X(x) \log \frac{p_X(x)}{\hat{p}_X(x)}\, \mathrm{d}x. \tag{5.11}$$

As is common in the literature, we do not ensure that p_X is absolutely continuous (see definition 2.1.30) with respect to $\hat{p}_X$. Even without guaranteeing this, we can "notationally" derive a learning objective that is useful in practice.

To derive the objective for parameter identification, we make the parametrization more explicit and write $\hat{p}_X := \hat{p}_X(\cdot\,; \theta)$. The minimization objective to identify the optimal parameters[14] is

14: We discuss the set of admissible parameters Θ in section 5.4.

$$\begin{aligned} &\arg\min_{\theta\in\Theta} (p_X \,\|\, \hat{p}_X(\cdot\,;\theta))_{\mathrm{KL}} = \\ &\arg\min_{\theta\in\Theta} \int_{\mathbb{R}^{m\times n}} p_X(x) \log \frac{p_X(x)}{\hat{p}_X(x;\theta)}\, \mathrm{d}x = \\ &\arg\min_{\theta\in\Theta} \int_{\mathbb{R}^{m\times n}} p_X(x)\big(-\log \hat{p}_X(x;\theta)\big)\, \mathrm{d}x \end{aligned} \tag{5.12}$$

where the last equality holds since the density of the reference distribution does not depend on the parameters. Rewriting in terms of expectations (definition 2.2.14) reveals the celebrated maximum likelihood objective

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{X\sim p_X}\big[-\log \hat{p}_X(X;\theta)\big]. \tag{5.13}$$

Inserting the Gibbs density eq. (5.10) into the above yields

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{X\sim p_X}\big[R(X,\theta)\big] + \log \int_{\mathbb{R}^{m\times n}} \exp\big(-R(\xi,\theta)\big)\, \mathrm{d}\xi. \tag{5.14}$$

We solve the parameter identification via first order algorithms and consequently find the gradient with respect to the parameters by the chain rule as

$$\int_{\mathbb{R}^{m\times n}} \frac{\exp\big(-R(x,\theta)\big)}{\int_{\mathbb{R}^{m\times n}} \exp\big(-R(\xi,\theta)\big)\, \mathrm{d}\xi} \big(-\nabla_2 R(x,\theta)\big)\, \mathrm{d}x = \mathbb{E}_{X\sim \hat{p}_X}\big[-\nabla_2 R(X,\theta)\big]. \tag{5.15}$$

Thus, parameter identification amounts to finding

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{X\sim p_X}\Big[R(X,\theta)\Big] - \mathbb{E}_{X\sim \hat{p}_X}\Big[R(X,\theta)\Big]. \tag{5.16}$$

This derivation has been used in the context of parameter identification of densities at least since the eighties [2]. In some sense, eq. (5.16) is the most general[15] way of fitting densities. In the context of Boltzmann machines, it was termed the *wake-sleep* algorithm [115] (the earliest reference we could find that mentions the word "sleep" in this context was [86]). When the expectation over the Gibbs distribution is approximated via few-step MCMC initialized with reference samples, the resulting algorithm is known as *contrastive divergence* [116].

15: In the sense that it follows directly from maximum likelihood learning of the completely general family of Gibbs distributions with arbitrary potential.

The primary challenge in eq. (5.16) is computing the expectation over the Gibbs distribution $\hat{p}_X$. In the context of Boltzmann machines, *restricted* Boltzmann machines [208] address this by restricting the architecture of the learned functions. For data-driven regularizers in imaging, Schmidt, Gao, and Roth [209] exploit the special structure of their regularizer and derive an efficient auxiliary variable Gibbs sampler to approximate the expectation. Conversely, Weiss and Freeman [85] focus on regularizers where the partition function is parameter independent, allowing them only to learn a rotation of predefined filters in a ridge-regularizer with Gaussian scale mixture potentials. They develop an efficient expectation-maximization algorithm to learn the weights of the Gaussian scale mixture and the rotation.

The method we adopt in this work aligns more closely with the FoE work by Roth and Black [206] and recent literature on generative models [174, 75]. Specifically, we resort to MCMC algorithms to approximate the expectation. Following [174, 75] we use the unadjusted Langevin algorithm shown in algorithm 1 to sample from the Gibbs distribution. The algorithm's motivation, properties and detailed analyses are covered in section 2.3.2; here, we recall is key properties and adapt the notation.

The unadjusted Langevin algorithm

$$x^{k+1} = x^k + \tau\nabla \log \hat{p}_X(x^k) + \sqrt{2\tau}z^k \tag{5.17}$$

with x^0 arbitrary and $z^k \sim \mathcal{N}_{0,I}$, results from an Euler-Maruyama discretization[16] of the continuous Langevin diffusion (eq. (2.94)). The continuous Langevin diffusion has invariant distribution $\hat{p}_X$ as shown in theorem 2.3.2. Although the discretization introduces bias, Durmus, Moulines, and Pereyra [79] found that adding a Metropolis-Hastings correction step often worsens results in practice, so we exclude this step.

16: with equispaced discretization time points, separated by $\tau > 0$

By inserting the Gibbs distribution in eq. (5.17),

$$x^{k+1} = x^k - \tau\nabla R(x^k,\theta) + \sqrt{2\tau}z^k. \tag{5.18}$$

which simplifies to a "noisy" gradient descent[17] on the regularizer since the normalization constant vanishes.

17: Again we remark that our regularizer is not differentiable. The gradient here is understood as a Clarke subdifferential and we discuss works that deal with this setup in section 2.3.2.

During learning, we run the unadjusted Langevin algorithm after every parameters update since parameter updates change the Gibbs distribution. To make the

learning algorithm practical, it is crucial to minimize the required steps until the Markov chain (definition 2.3.6) converges. This can be achieved by an informed initialization: Since parameter updates are usually small,[18] the Gibbs distribution should remain similar between parameter updates. Thus, *samples* from the distribution prior to the update serve as good *initializations* for the Markov chain post-update. This algorithm due to Tieleman [228] is known as *persistent contrastive divergence*. We use this method to facilitate training our regularizer, with implementation details discussed in section 5.4.

18: This depends on the learning rate of the optimizer, but typically this is true.

5.3.3 Reconstruction algorithm

In this work we revisit joint nonlinear inversion for parallel MRI reconstruction. This approach aims to estimate spatially varying coil sensitivities along with the image in a nonlinear inverse problem, thereby utilizing the available data optimally. Detailed motivation for this is discussed in section 5.1.2; here we outline our proposed algorithm and point again to the works of Bauer and Kannengiesser [14], Ying and Sheng [245] and Uecker, Hohage, Block, and Frahm [233] which sparked this line of research.

We assume that the parallel MRI machine has $c \in \mathbb{N}$ receiver coils that sense the spin density of the scanned object. The sensitivities of the receiver coils are complex images $\sigma_1, \sigma_2, \dots, \sigma_c \in \mathbb{C}^{m\times n}$ defined on the same grid as the image. The measurement data are then c vectors $y_1, y_2, \dots, y_c \in \mathbb{C}^f$ where $f \in \mathbb{N}$ is the number of acquired frequencies. We assume that these frequencies are grid-aligned, though nonuniform sampling is possible (see [214]).

The data relate to the underlying image via a discrete Fourier transform $F\colon \mathbb{C}^{m\times n} \to \mathbb{C}^{m\times n}$ and a binary diagonal operator $M\colon \mathbb{C}^{m\times n} \to \mathbb{C}^f$ which picks the acquired frequencies.[19] Thus, the spin density, coil sensitivities and data relate through[20]

$$\begin{aligned} y_1 &= MF(\sigma_1 \odot x) + \eta_1, \\ y_2 &= MF(\sigma_2 \odot x) + \eta_2, \\ &\vdots \\ y_c &= MF(\sigma_c \odot x) + \eta_c, \end{aligned} \tag{5.19}$$

19: The notation is chosen in analogy to chapter 4.

20: Nonuniform sampling is easily realized by substituting MF with the appropriate nonuniform Fourier transform. The remaining derivation carries over to this case.

where $\eta_1, \eta_2, \dots, \eta_c \in \mathbb{C}^f$ are coil-wise additive noise terms.

The data $y_1, y_2, \dots, y_c$ have a *nonlinear* relation to the estimation variables $x, \sigma_1, \sigma_1, \dots, \sigma_c$. The bilinear structure introduces ambiguity: multiplying the spin density with an arbitrary image $w \in \mathbb{C}^{m\times n}$ and dividing the coil sensitivities by w yields the same solution: $(\sigma_i \odot x) = \Big((\sigma_i \oslash w) \odot (x \odot w)\Big)$. To resolve this, we regularize the image x and the coil sensitivities $\sigma_1, \sigma_2, \dots, \sigma_c$. We use the data-driven regularizer described in section 5.3.1 for the image and classical quadratic gradient penalization for the coil sensitivities.

We formalize the nonlinear inversion problem as an optimization problem. Let $\Sigma = (\sigma_1, \sigma_2, \dots, \sigma_c) \in \mathbb{C}^{m\times n\times c}$. Data consistency is ensured using a quadratic data

fidelity

$$\begin{aligned} D\colon \mathbb{R}^{m\times n} \times \mathbb{C}^{m\times n\times c} &\to \mathbb{C}^{m\times n\times c} \\ (x, \Sigma) &\mapsto \frac{1}{2}\sum_{i=1}^{c} \|MF(\sigma_i \odot x) - z_i\|_2^2. \end{aligned} \tag{5.20}$$

We regularize the image with our data-driven regularizer $R(\,\cdot\,, \theta)$ from section 5.3.1 [21] and explicitly enforce nonnegativity of the image during optimization. To resolve the ambiguity in the bilinear form, we ensure that the coil sensitivities are sufficiently smooth by utilizing classical quadratic gradient penalization through

$$\begin{aligned} S\colon \mathbb{C}^{m\times n\times c} &\to \mathbb{R} \\ \Sigma &\mapsto \frac{1}{2}\sum_{c=1}^{C} \left(\|D_{\mathrm{D}}\mathfrak{Re}(\sigma_c)\|_2^2 + \|D_{\mathrm{D}}\mathfrak{Im}(\sigma_c)\|_2^2\right), \end{aligned} \tag{5.21}$$

where $D_{\mathrm{D}}\colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n\times 2}$ is a forward finite difference operator with Dirichlet boundary conditions:

$$\begin{aligned} (D_{\mathrm{D}}x)_{i,j,1} &= \begin{cases} x_{i+1,j} - x_{i,j} & \text{if } 1 \le i < m, \\ -x_{i,j} & \text{else,} \end{cases} \\ (D_{\mathrm{D}}x)_{i,j,2} &= \begin{cases} x_{i,j+1} - x_{i,j} & \text{if } 1 \le j < n, \\ -x_{i,j} & \text{else.} \end{cases} \end{aligned} \tag{5.22}$$

These conditions assume coil sensitivities are zero outside the image domain, reflecting physical reality.

Recovering the image x and coil sensitivities $\sigma_1, \sigma_2, \dots, \sigma_c$ from the data $y_1, y_2 \dots, y_c$ amounts to finding

$$\operatorname*{arg\,min}_{x\in\mathbb{R}^{m\times n}, \Sigma\in\mathbb{C}^{m\times n\times c}} D(x, \Sigma) + \lambda R(x, \theta) + \delta_{\mathbb{R}_+^{m\times n}}(x) + \mu S(\Sigma), \tag{5.23}$$

where $\delta_{\mathbb{R}^{m\times n}}$ is the indicator function (definition 2.4.1) of the nonnegative orthant, and $\lambda, \mu > 0$ tune the influence of the image and the coil sensitivity regularization.

We now detail the algorithm that solves the optimization problem eq. (5.23).[22] Despite being nonconvex and nonsmooth, the problem has a favorable structure: the smooth part has a Lipschitz continuous gradient in both variable blocks, and the proximal map for the nonsmooth functions can be efficiently computed. For a fixed Σ, the map $x \mapsto D(x, \Sigma) + \lambda R(x, \theta)$ has a Lipschitz continuous gradient[23]

$$x \mapsto \sum_{i=1}^{c} \overline{\sigma_i} \odot (F^* M^* (MF(\sigma_i \odot x) - z_i)) + \lambda \nabla_1 R(x, \theta). \tag{5.24}$$

Analogously, for a fixed x, the map[24] $\Sigma \mapsto D(x, \Sigma) + \lambda R(x, \theta)$ has a Lipschitz continuous gradient

$$\Sigma \mapsto \begin{pmatrix} x \odot (F^* M^* (MF(\sigma_1 \odot x) - z_i)) \\ \vdots \\ x \odot (F^* M^* (MF(\sigma_c \odot x) - z_c)) \end{pmatrix}. \tag{5.25}$$

21: The algorithm that we derive does not explicitly require our data-driven regularizer; also classical regularizers such as the TV can be used. The numerical results in section 5.5.4 use the TV regularizer embedded in the algorithm.

22: All gradients are understood in the $\mathbb{CR}$-sense [138].

23: when $\nabla R(\,\cdot\,, \theta)$ is Lipschitz. This is the case when we utilize differentiable surrogates of the rectified linear activation.

24: We include the term $\lambda R(x, \theta)$ here to emphasize the structure.

The proximal maps (definition 2.4.18) for $\delta_{\mathbb{R}^{m\times n}}$ and μS are efficiently computed: The proximal map of the indicator function of the nonnegative orthant, $\operatorname{prox}_{\delta_{\mathbb{R}^{m\times n}}} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$, retrieves the positive part of its argument:

$$\operatorname{prox}_{\delta_{\mathbb{R}^{m\times n}}}(x) = \max(x, 0). \tag{5.26}$$

The proximal map of the quadratic gradient penalization, $\operatorname{prox}_{\delta_{\mathbb{R}^{m\times n}}} : \mathbb{C}^{m\times n\times c} \to \mathbb{C}^{m\times n\times c}$ uses the discrete sine transform: Denoting the imaginary unit with $\hat{\imath} \coloneqq \sqrt{-1}$,

$$\operatorname{prox}_{\mu S}(\Sigma) = \begin{pmatrix} Q_\mu(\mathfrak{Re}(\sigma_1)) + \hat{\imath} Q_\mu(\mathfrak{Im}(\sigma_1)) \\ \vdots \\ Q_\mu(\mathfrak{Re}(\sigma_c)) + \hat{\imath} Q_\mu(\mathfrak{Im}(\sigma_c)) \end{pmatrix}. \tag{5.27}$$

Here,

$$Q_\mu \colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n} \colon x \mapsto \mathscr{S}^*\Big(\mathscr{S}(\mu x) \oslash (\xi + \mu)\Big) \tag{5.28}$$

uses the two-dimensional discrete sine transform $\mathscr{S}$, where ξ are the eigenvalues of the two-dimensional discrete Laplace operator[25]

$$\xi_{i,j} = 4 - 2\Big(\cos\frac{\pi i}{m} + \cos\frac{\pi j}{n}\Big) \tag{5.29}$$

25: This is essentially a fast Poisson solver, see [191, Chapter 19.4] for a more rigorous discussion.

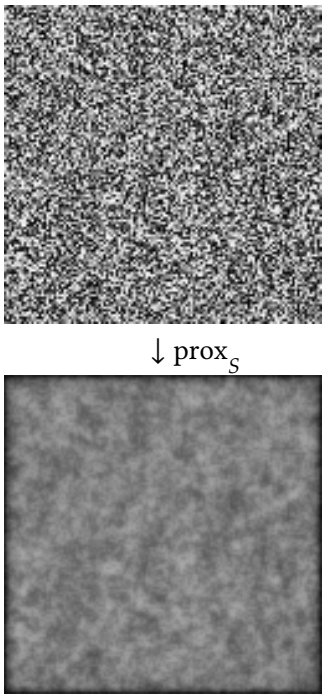

Figure 5.7: The smoothing action of the proximal operator with respect to quadratic gradient penalization with Dirichlet boundary conditions.

We show the smoothing action of this proximal map in fig. 5.7. There, in the image on top the real and imaginary part of each pixel is drawn uniformly in $[0, 1]$, we set $\mu = 3$ and only show the real part. The image is significantly smoother and smoothly approaches zero at the boundaries.

The structure outlined above is exploited in the iPALM algorithm by Pock and Sabach [188]. In the iterations, we use backtracking to estimate the Lipschitz constants; the algorithm is summarized in algorithm 14. As already pointed out, this algorithm can use any image regularizer. For our experiments we tweak it for optimal performance under our particular setup with the details discussed in section 5.4. A sketch of the reconstruction algorithm is shown in fig. 5.1.

Algorithm 14: iPALM [188] instantiation to solve (5.23). bt is algorithm 15.

Input : Initial points $x^0 \in \mathbb{R}^{m\times n}$, $\Sigma^0 \in \mathbb{C}^{m\times n\times c}$, number of iterations $K \in \mathbb{N}$, backtracking multipliers $\gamma_1 \in (0,1)$, $\gamma_2 \in (0,1)$, initial guesses for the Lipschitz constants $L_x, L_\Sigma > 0$

Output : (x^K, Σ^K)

$(x^1, \Sigma^1) = (x^0, \Sigma^0)$

for $k = 1, \dots, K-1$ **do**

  $\bar{x} = x^k + \frac{k}{k+3}(x^k - x^{k-1})$

  $(x^{k+1}, L_x) = \operatorname{bt}(x \mapsto D(x, \Sigma^k) + \lambda R(x), \delta_{\mathbb{R}^{m\times n}_+}, \bar{x}, L_x, \gamma_1, \gamma_2)$

  $\bar{\Sigma} = \Sigma^k + \frac{k}{k+3}(\Sigma^k - \Sigma^{k-1})$

  $(\Sigma^{k+1}, L_\Sigma) = \operatorname{bt}(\Sigma \mapsto H(x^{k+1}, \Sigma), \mu S, \bar{\Sigma}, L_\Sigma, \gamma_1, \gamma_2)$

Algorithm 15: Backtracking procedure to find the local Lipschitz constants in algorithm 8.

Input : $E, P, x_0, L_0, \gamma_1, \gamma_2$
Output: (x, L)

$L \leftarrow L_0$
for *ever* **do**
  $x = \operatorname{prox}_{L^{-1}P}(x_0 - L^{-1}\nabla E(x_0))$
  $d = x - x_0$
  if $E(x) \leq E(x_0) + \langle \nabla E(x_0), d\rangle + \frac{L}{2}\|d\|_2^2$ **then**
    $L \leftarrow \gamma_1 L$
    break
  else $L \leftarrow L/\gamma_2$

We assume a real-valued spin-density and empirically demonstrate good performance on the fastMRI dataset [251] in section 5.5.4. For phase-sensitive imaging, a complex spin-density needs to be assumed. Our approach generalizes to this setting by splitting real and imaginary channels as in [169, 222], and training the regularizer on this data.

5.4 Implementation details

In this section provide details about the training and evaluation setup. In addition, we discuss how we overcome some limitations of our regularizer (such as that it can only act on images of fixed size) in practice in section 5.4.2. This section is important to understand how we utilize the data-driven regularizer for practical problems, but can also be revisited later if some details remain unclear.

5.4.1 Experimental data

We train and evaluate on the fastMRI knee dataset [133]. Specifically, the training data are the RSS reconstructions of size 320 × 320 of the multi-coil coronal proton density weighted (CORPD) training split; we reserve the coronal proton density weighted with fat suppression (CORPDFS) images for an out-of-distribution evaluation. In order to avoid training on empty images, we only consider the central 11 slices.[26] This results in a total of 5324 training images.

Due to vendor-specific implementation differences, the scans vary greatly in magnitude. In order to have consistent intensity ranges during training, we normalize each individual slice to have a maximum of 1 and a minimum of 0: Denoting with $x \in \mathbb{R}^{320\times320}$ an original image from the dataset, we normalize it by $x \mapsto \frac{x - \min_{i,j} x_{i,j}}{\|x\|_\infty - \min_{i,j} x_{i,j}}$. This normalization was not performed for any of the reference methods.

For validation and testing, we used the multi-coil CORPD validation split. For the sake of a simple implementation, we discard all samples that have a

26: Training on a "central" subset of the slices is common with this dataset, see e.g. [53, section 4.1]. The extremal slices sometimes only contain noise.

width different from 368 and 372, leaving 91 scans.[27] The scans were split into 30 validation samples and 61 test samples by lexicographic ordering of the filenames. To be consistent with training, we again restrict our interest to the central 11 slices, resulting in 330 validation slices and 671 test slices. For the out-of-distribution experiments, we used the central 11 slices of the CORPDFS scans (again excluding samples with width different from 368 and 372) in the fastMRI knee validation dataset.

27: Nine out of 100 scans are discarded.

5.4.2 Practical considerations

A particular characteristic of the fastMRI dataset is that the reference RSS reconstructions are only available for a crop of size 320 × 320, whereas the images to be reconstructed are of size 640 × 368 or 640 × 372. In other words, only a subregion of the underlying signal follows the distribution whose negative log our regularizer models. We overcome this by simply cropping the region that our regularizer models: When we utilize our data-driven regularizer in eq. (5.23), the regularizer is given by

$$x \mapsto (R(\,\cdot\,, \theta) \circ \text{⌜}_{\cdot w})(x) \tag{5.30}$$

where $\text{⌜}_{\cdot w} \colon \mathbb{R}^{640 \times w} \to \mathbb{R}^{320 \times 320}$ crops the validation images of size $640 \times w$ (with w either 368 or 372) to 320 × 320, the size of the training images. In particular, the crop extracts only the central regions whose distribution is modeled by our regularizer. Thus, in the reconstruction problems, we only regularize the central 320 × 320 pixels; the effect of this is clearly visible in fig. 5.1 after zooming. The qualitative and quantitative evaluation is only done in this central region[28] and we did not observe any benefit to additionally regularizing the rest of the image with, e.g., TV. The gradient of eq. (5.30) is the zero-padding of $(\nabla R(\,\cdot\,, \theta))(\text{⌜}_{\cdot w}(x))$ to $640 \times w$.

28: This is also standard for the fastMRI dataset, since the original challenge was set up this way, and reference reconstructions are only available for this region.

Utilizing our data-driven regularizer in eq. (5.23) has another small caveat: Due to the ambiguity in the coil sensitivities and the image, the resulting image is not necessarily identical to a standard RSS reconstruction, see [233, section on postprocessing]. This can be fixed with a postprocessing step where the image is "normalized" by the pixel-wise RSS of the estimated coil sensitivities: Denoting with $x, \sigma_1, \sigma_2, \dots, \sigma_c$ a solution to the minimization problem, the normalized image is

$$x \odot \sqrt{\sum_{i=1}^{c} |\sigma_i|^2}, \tag{5.31}$$

where $|\cdot|$ is the complex modulus acting element-wise on the argument.

However, in this case there is a slight mismatch between the optimization variable and the data-driven regularizer during optimization: The optimization variable is the unnormalized image, while the regularizer was trained on the standard RSS reconstruction. We found it slightly beneficial to change the optimization problem to directly optimize over the normalized image, such that this mismatch is corrected. Essentially, this amounts to changing the data term to consider the normalized coil sensitivities:[29]

29: Clearly, this map does not have a Lipschitz continuous gradient in the second argument, which is theoretically required for convergence in iPALM. In practice, we still observe a convergent behavior, and the results are improved.

$$D\colon \mathbb{R}^{m\times n}\times\mathbb{C}^{m\times n\times c}\to\mathbb{C}^{m\times n\times c},\qquad (x,\Sigma)\mapsto \frac{1}{2}\sum_{i=1}^{c}\left\|MF(\sigma_i\odot x\oslash|\Sigma|_{\mathrm{rss}})-y_i\right\|_2^2, \tag{5.32}$$

where $|\Sigma|_{\mathrm{rss}}=\sqrt{\sum_{i=1}^{c}|\sigma_i|^2}$ denotes the pixel-wise RSS of the coil sensitivities.

Additionally, we found it crucial to provide a good initialization for the reconstruction algorithm. In particular, we initialize the algorithm with the zero-filled (ZF) RSS reconstruction

$$x^0=\sqrt{\sum_{i=1}^{c}|F^*M^*y_i|^2} \tag{5.33}$$

and the corresponding coil sensitivities

$$\sigma_i^0=(F^*M^*y_i)\oslash x^0 \text{ for all } i=1,2,\dots,c. \tag{5.34}$$

Then, as already discussed in section 4.5 we run the first iterations by "probing" the regularizer with a slightly noisy signal. The noise level corresponds to that of the iterations in the unadjusted Langevin algorithm we use to sample the regularizer during training. In summary, the algorithm that we use to reconstruct the images in practice is thus noisy iPALM with the data fidelity term given by eq. (5.32). Here, "noisy" is understood as in algorithm 13; we do not explicitly note down the resulting algorithm.

5.4.3 Network and training details

The architecture of the network is discussed in detail in section 5.3.1. Here, we discuss the exact choices for the hyperparameters of the network as well as the training. For the leaky rectified linear activation, we use the leak coefficient $\gamma=0.05$. We use $l=6$ layers and the number of channels in each layer is $c_i=48\lfloor 1.75^i\rfloor$. Since we use 3×3 filters, the number of learnable parameters is

$$3^2\Big(48(1+48)+\sum_{i=1}^{5}c_i^2+c_ic_{i+1}\Big)+c_6 5^2=21\,350\,640, \tag{5.35}$$

where the first layer is handled explicitly (see marginnote 8) and the last term is due to the fully connected layer (note that 5 is the size of the feature map after downsampling 320 by a factor of two six times). We do not impose any constraints onto the parameters, hence the space of admissible parameters Θ in the parameter identification problem eq. (5.16) is isomorphic to $\mathbb{R}^{21\,350\,640}$. Although our network is quite sizable, it has significantly less learnable parameters than the reference methods: The discriminative end-to-end VN of [222] has 3×10^7 learnable parameters and the score-based diffusion models of [53] has 6.7×10^7 learnable parameters.

The training images are the 5324 images described in section 5.4.1. However, to stabilize training, we smooth this empirical measure by convolving it with a Gaussian with variance 1.5×10^{-2}. Thus, denoting with $x_1,x_2,\dots,x_{5324}\in\mathbb{R}^{320\times 320}$

the training images, the reference density p_X in the parameter identification problem eq. (5.16) is the density of

$$\sum_{i=1}^{5324} \delta_{x_i} * \mathcal{N}_{0,1.5\times10^{-2}}. \tag{5.36}$$

Note that this density is supported in all of $\mathbb{R}^{320\times320}$, thus making the Kullback-Leibler divergence in eq. (5.11) slightly "more well defined". We optimize the parameter identification problem eq. (5.16) with AdaBelief (algorithm 12) using the standard choice $\beta_1 = 0.9, \beta_2 = 0.999$. We use a learning rate of 5×10^{-4}, exponentially decreasing with rate 0.5 at update steps 500, 2000, 3000, 5000, and 7000, using a batch size[30] of 50 for 27 000 parameter updates. To sample from the Gibbs distribution, $\hat{p}_X$, we run the unadjusted Langevin algorithm (see algorithm 1 and the discussion in section 5.3.2 for details) for $K_{\max} = 500$ steps. To accelerate training in the early stages, we use an exponential schedule, detailed by $K_h = \lceil K_{\max}(1 - \exp(\frac{-h}{1000}))\rceil$, at the h^{th} parameter update. To realize persistent initialization, we use a buffer holding 8000 images. Samples persist in this buffer with a chance of 99 %; if a sample does not persist, either a sample from the reference distribution or from the uniform distribution over the hypercube $[0,1]^{320\times320}$ is written into the buffer with equal chance. In contrast to most previous works [75, 98, 231] we did not find it necessary to regularize our model by means of, e.g., Lipschitz regularization, gradient clipping, or similar techniques. Training took approximately one month on a machine equipped with one NVIDIA Quadro RTX 8000.

30: In this context, the batch size is the number of samples taken to approximate the expectations over both the reference and the Gibbs distribution. Although not necessary, we use the same number for both.

5.4.4 Synthetic experiments and posterior sampling

In order to assess the data-driven regularizer in a controlled setting without the need for coil sensitivity estimation, we perform experiments on synthetic single-coil data: We take the reference RSS images from the validation and test data detailed in section 5.4.1, and construct the data ourselves as

$$y = MFx + \eta, \tag{5.37}$$

where $x \in \mathbb{R}^{320\times320}$ is a reference RSS image, F is the discrete Fourier transform and M is a frequency selection operator; different choices of M are visualized in fig. 5.10. η is Gaussian noise whose variance is 1 % of the largest pixel intensity in x. Recall that the regularizer was trained on RSS images whose intensity range was normalized to $[0,1]$. To approximately map the reconstructions to the same intensities seen during training, we normalize the data by $y \mapsto \frac{y}{\|x^0\|_\infty}$, and undo this normalization for the final reconstruction by $x^K \mapsto x^K\|x^0\|_\infty$.[31] Since we do not need to estimate coil sensitivities here, the inference algorithm to solve

$$\arg\min_x \frac{1}{2}\|MFx - y\|^2 + \lambda R(x,\theta) \tag{5.38}$$

essentially reduces to noisy nonconvex FISTA with backtracking as discussed in algorithm 13. We found the optimal regularization parameter λ by grid search on the validation dataset.

31: x^0 is the zero-filled RSS reconstruction, see eq. (5.33). K is the number of iterations in the inference algorithm.

In addition to computing MAP estimator eq. (5.38), we approximate the MMSE estimator—the expectation of the posterior—as well as the pixel-wise marginal variance via MCMC. In detail, the unadjusted Langevin algorithm for sampling the posterior reads

$$x^{k+1} = x^k + \tau\Big(F^*M^*(MFx - y) + \lambda\nabla_1 R(x^k, \theta)\Big) + \sqrt{2\tau}z^k \tag{5.39}$$

where $z^k \sim \mathcal{N}_{0,I}$. The algorithm is initialized with the MAP estimate and we discard the first 10 000 samples ("burn-in", see section 2.3.2 and [32, section 1.11]). In order to reduce autocorrelation between the samples, for the computation of the MMSE as well as the pixel-wise marginal variance, we only consider every 15^{th} iteration.[32] We prescribe 10 000 samples for the computation of the statistics and therefore run the unadjusted Langevin algorithm for a total of 160 000 iterations. We found that the regularization parameter λ barely influenced the results of the posterior sampling and consequently set it to $\lambda = 1$ for all frequency selections.

32: This is commonly referred to as *thinning*, see [149]. We were made aware that thinning is typically not advantageous after writing the manuscript, see also [170, remark 4.1 and appendix D]

5.4.5 Parallel imaging

For the parallel imaging experiments, we run the iPALM algorithm algorithm 8 with $K = 100$. As in the experiments on synthetic data, we normalize the data by $z_i \mapsto \frac{z_i}{\|x^0\|_\infty}$ for all $i = 1, 2, \dots, c$, and normalize the final reconstruction by $x^K \mapsto x^K\|x^0\|_\infty$.[33] To find the optimal regularization parameters, we fix $\mu = 10$ and obtain λ by linear least-squares regression of the initial residuum $\sum_{c=1}^{C}\|MF(\sigma_c^0(z^{\text{val},i}) \odot x^0(z^{\text{val},i})) - (z^{\text{val},i})_c\|_2^2$ against $\min_\lambda \|x^K(z^{\text{val},i}, \lambda) - u^{\text{val},i}\|_2^2$ (found by grid search) for all image-data pairs $(x^{\text{val},i}, z^{\text{val},i})$ in the validation set. Here, we view the initial image x^0, the coil sensitivities $\sigma_1, \sigma_2, \dots, \sigma_c$, and the reconstruction x^K as maps to make dependencies explicit. This regression is performed *only* for the Cartesian frequency selection with acceleration 4 and 8 % autocalibration lines. The reconstruction problems with different frequency selections use the same fit. In the generalization experiments, experiments marked with † also use the linear λ-fit calculated on CORPD data. Experiments marked with ∗ use a linear fit computed on the CORPDFS data, again only on a Cartesian frequency selection with acceleration 4 and 8 % autocalibration lines.

33: x^0 is the zero-filled RSS reconstruction, see eq. (5.33).

A particular characteristic of our reconstruction approach is that its intensities are not quantitatively comparable to the reference. In detail, although we normalize the reconstruction by the RSS of the coil sensitivities, we observed that especially in low-intensity regions (e.g. air) the intensities in the reconstruction did not match the reference. To remedy this and allow for fair quantitative evaluation, we utilize the validation data to fit a spline curve (cubic splines, 5 equally spaced knots) against the scatter of reconstructed and reference intensities. For the generalization experiments, we fit the spline curve again on an independent CORPDFS validation dataset. The spline curves for both CORPD and CORPDFS are shown in fig. 5.8. The insets show that our reconstructions prefer zero-intensity in background regions, whereas the reference images have non-zero background intensity.

To compute the MMSE estimate in the parallel imaging case, we fix the coil sensitivities to the MAP estimate. Thus, the iterations of the unadjusted Langevin

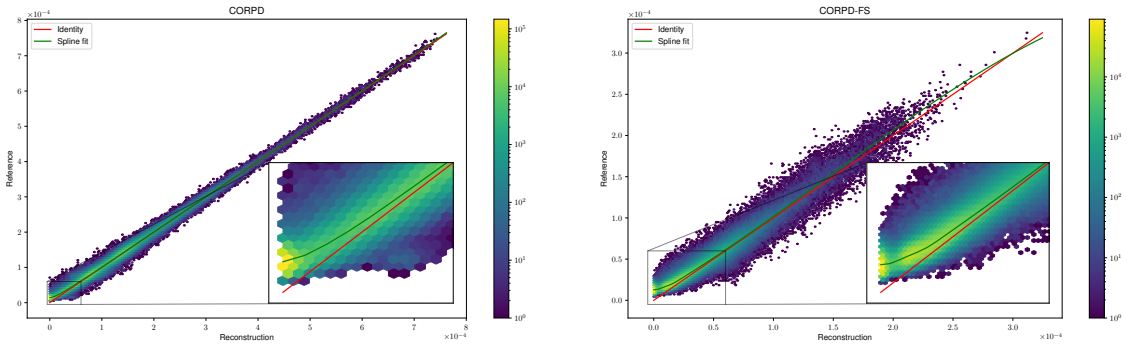

Figure 5.8: We account for for the systematic overestimation of the intensities near zero by utilizing a spline fit on the reconstructed intensities versus the reference intensities. The left plot shows the fit for the CORPD data, the right plot shows the fit for the CORPDFS data.

algorithm take the form

$$x^{k+1} = x^k + \tau\Big(\sum_{i=1}^{c} \overline{\sigma_i} \odot (F^* M^* (MF(\sigma_i \odot x^k) - y_i)) + \lambda \nabla_1 R(x^k, \theta)\Big) + \sqrt{2\tau} z^k, \quad (5.40)$$

where $z^k \sim \mathcal{N}_{0,I}$ and $\sigma_1, \sigma_2, \dots, \sigma_c$ are previously estimated coil sensitivities of the corresponding reconstruction problem. The coil sensitivities may also be included in the Langevin procedure, but we empirically found no noticeable difference to fixing them. We believe that this is due to the strong imposed spatial regularity, which corresponds to a "narrow" prior.

We evaluate the quality of the estimated coil sensitivities by computing the null-space residual [232]: Let $x_1, x_2, \dots, x_c$ the individual coil images from fully sampled data. The null-space residual ρ_i of the i-th coil

$$\rho_i = \frac{\sigma_i}{|\Sigma|^2_{\text{rss}}} \sum_{j=1}^{c} \overline{\sigma}_j x_j - x_i, \quad (5.41)$$

should only contain noise since $x_i = \sigma_i x$ when σ_i is exact:

$$\frac{\sigma_i}{|\Sigma|^2_{\text{rss}}} \sum_{j=1}^{c} \overline{\sigma}_j x_j - x_i = \frac{\sigma_i}{|\Sigma|^2_{\text{rss}}} \sum_{j=1}^{c} \overline{\sigma}_j \sigma_j x - x_i = \sigma_i x - x_i = 0. \quad (5.42)$$

Thus, any residual signal in ρ_i points to sub-optimal sensitivity estimates. In the results in fig. 5.19, for the sake of conciseness we do not inspect the null-space residual coil-wise, but instead show the RSS of the residual, $\sqrt{\sum_{i=1}^{c} |\rho_i|^2}$.

5.4.6 Comparison and evaluation

In the experiments on synthetic data, we compare against the fastMRI baseline method [251] as well as the diffusion-based approach of [53]. To ensure a fair comparison, we trained the fastMRI baseline method on the subset of the fastMRI dataset detailed in section 5.4.1. To train the model, the data was constructed

using a Cartesian frequency selection with acceleration 4 and 8 % autocalibration lines; other training hyperparameters are taken from the github repository of the authors.[34] In contrast, training the diffusion model is extremely time- and resource intensive. Thus, to avoid training a model we took the implementation as well as the weights of a trained model of the diffusion-based approach the github repository of the authors.[35] We note that their model was trained on a larger corpus of data, as their training data includes the CORPDFS scans in addition to the CORPD scans. As in their paper, we use 2000 steps in the reverse diffusion. However, due to time and computational constraints, we limit our comparison to a subset of the validation data and hence provide results separately.

34: Can be found at `https://github.com/facebookresearch/fastMRI`, accessed 2024–06–07.

35: Can be found at `https://github.com/HJ-harry/score-MRI`, accessed 2024–06–07.

In the experiments on synthetic data as well as the parallel MRI experiments, we use the Charbonnier-smoothed isotropic total variation

$$x \mapsto \lambda \sum_{i,j=1}^{m,n} \sqrt{\sum_{k=1}^{2} \big((Dx)_{i,j,k}\big)^2 + \epsilon^2}, \tag{5.43}$$

with $\epsilon = 10^{-3}$ as a hand-crafted prior. In the experiments on synthetic data, $m = n = 320$; in the parallel MRI experiments, $m = 640$ and $n = 368$ or $n = 372$, depending on the width of the scan. Thus, in contrast to our regularizer, we apply the Charbonnier-smoothed isotropic total variation regularizer to the entire image, not only to the central region. We emphasize again that, for parallel MRI, we use the regularizer in conjunction with our proposed algorithm for join estimation of the image and the coil sensitivities.

As a state-of-the-art discriminative approach for parallel MRI, we compare against the end-to-end VN approach from [222]. The implementation was taken from the fastMRI github repository with default parameters. As before, to ensure a fair comparison, the end-to-end VN was trained on the subset of the fastMRI dataset detailed section 5.4.1. To train the model, the data was constructed using a Cartesian frequency selection with acceleration 4 and 8 % autocalibration lines.

We compare the reconstructions quantitatively using the PSNR, NMSE and SSIM, see definition 2.5.3, definition 2.5.2, and definition 2.5.4 respectively. We compute the SSIM with the standard setup; a 7×7 uniform filter and parameters $K_1 = 0.01$, $K_2 = 0.03$. We define the acceleration as the ratio of the image size and the acquired frequencies. This is commonly done but only reflects the ill-posedness of the problem and not necessarily the achieved speed-up of a scan in practice. The practical speed-up is determined by the alignment of the acquired frequencies with respect to each other due to the finite speed of the gradient coils in the MRI scanner. The discussion of MRI physics and sequence design is out of the scope of this work, we refer to the thesis of Lazarus [142] for a discussion of sequence design in the context of accelerated MRI.

5.5 Results

In this section, we first visualize preferred structures of our regularizer in section 5.5.1 using the unadjusted Langevin algorithm on the trained model. Then,

in section 5.5.2 we consider synthetic single-coil data, similar to chapter 4, leveraging the natural Bayesian interpretation to access posterior distributions for reconstructions. This allows computation of different Bayesian estimators, such as MAP and MMSE estimates, and visualization of pixel-wise marginal variance of the posterior distribution. In section 5.5.4, we apply our reconstruction algorithm and data-driven regularizer to parallel MRI, achieving state-of-the-art results with different frequency selections. Here, our algorithm jointly estimates the image and coil sensitivities, showing superiority over offline estimation, especially with limited autocalibration data. Finally, in section 5.5.5, we discuss generalization properties of the data-driven regularizer.

5.5.1 Data-independent analysis

In chapter 4, we reasoned about preferred structures of different ridge-type regularizers by visualizing filters and their corresponding potentials. This is hardly possible with the deep neural regularizer, which includes 2 368 560 filters with nontrivial interaction[36] and a final weighted sum containing 33 600 weights. However, we can visualize preferred structures by drawing samples from the Gibbs distribution (eq. (5.10)), making the model more interpretable compared to learned discriminative reconstruction approaches, thus facilitating clinical adoption.

36: via cascaded nonlinearities and downsampling

Additionally, we can *visualize the density* of the Gibbs distribution around the samples. The local regularization landscape provides insights into properties such as (local) convexity, generalization capabilities, and adversarial robustness [224]. To visualize the (320×320)-dimensional landscape, we follow [145]: Let $x^0, x^1, \dots, x^K \in \mathbb{R}^{320\times320}$ be iterates in the unadjusted Langevin algorithm[37] eq. (5.18), with x^0 drawn from uniformly from $[0,1]^{320\times320}$. Let v_1, v_2 be the first two principal directions of the point cloud $\{x^0 - x^K; x^1 - x^K; \dots; x^{K-1} - x^K\}$ and denote $\bar{x} = \frac{1}{K-1}\sum_{k=1}^{K-1} x^k$.

37: we set $K = 10\,000$

In fig. 5.9 we show[38] $\mathbb{R}^2 \ni (\xi_1, \xi_2) \mapsto \exp\big(-R(\bar{x} + \xi_1 v_1 + \xi_2 v_2)/T\big)$ along with the Langevin trajectory. The samples from the Gibbs distribution closely resemble those from the reference distribution[39], suggesting that the model accurately represents high-level statistics of the reference data. For example, the knee is centered and the femur and tibia are visible and anatomically plausibly separated, surrounded by muscle- and fat tissue with blood vessels, reflecting plausible anatomy. This empirical evidence suggests that the data-driven regularizer is a good model of the negative log prior, which is highly desirable in reconstruction problems with signals drawn from the reference distribution. For out-of-distribution signals, the regularizer might impart features of the reference distribution, discussed further in section 5.5.5.

38: $T = 7$ yields visually pleasing results. This scaling is arbitrary; it can make the density more or less peaky. In the context of a reconstruction problem, it is related to the regularization parameter.

39: Samples from the reference distribution are shown in e.g. fig. 5.16.

The two-dimensional landscape appears smooth on the considered domain and almost log-concave around modes of the Gibbs distribution, indicating that the high-dimensional landscape is also reasonably well-behaved. This is corroborated by the ease of optimization: For all reconstruction tasks, we only need 50 to 100 iterations of iPALM.

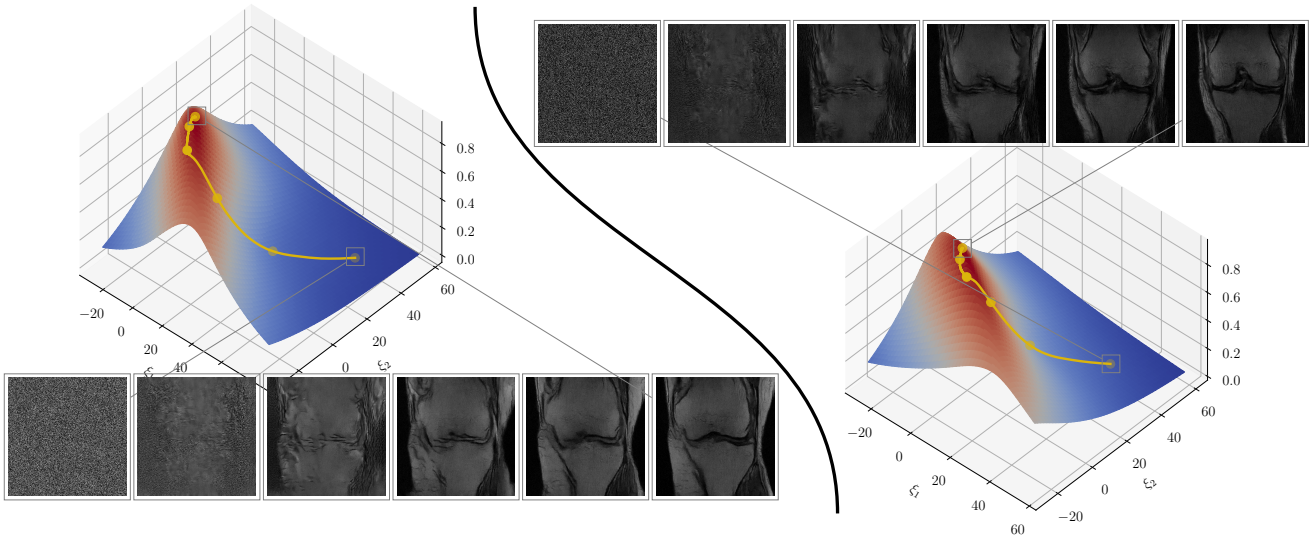

Figure 5.9: We visualize preferred structures of the regularizer via MCMC sampling. Uniform noise is unlikely under the deep neural regularizer—natural anatomical structures are very likely. The golden line is the trajectory of the unadjusted Langevin algorithm.

5.5.2 Simulation study

This chapter' contributions include designing and training the regularizer, and developing the algorithm for joint nonlinear inversion. In this section, we focus on the data-driven regularizer by constructing a reconstruction problem that does not require coil sensitivity estimation. Details of this construction and reference methods are provided in section 5.5.2; here, we state that we have access to data y given by

$$y = MFx + \eta, \tag{5.44}$$

where the aim is to recover the underlying signal $x \in \mathbb{R}^{320\times320}$. Crucially, the random variable of the underlying signal x is distributed as p_X, the reference distribution on which the regularizer was trained. We will demonstrate that for different frequency selection operators, encoded in M, the optimization problem

$$\arg\min_{x\in\mathbb{R}^{320\times320}} \frac{1}{2}\|MFx - y\|^2 + \lambda R(x, \theta) \tag{5.45}$$

yields satisfactory reconstructions.[40]

We consider four different frequency selection operators:

1. A random selection with acceleration 3,
2. a Cartesian selection with a densely samples frequencies in the phase encoding direction, 8 % autocalibration lines, and acceleration 4,
3. a spiral selection with acceleration 5, and
4. a radial selection with acceleration 6.

These are visualized in in fig. 5.10.

40: In this respect, there is also nothing special about the Fourier transform F. Indeed, the regularizer will work well for any forward operator (Radon imaging, superresolution, etc.), so long as the underlying signal is drawn from the reference distribution.

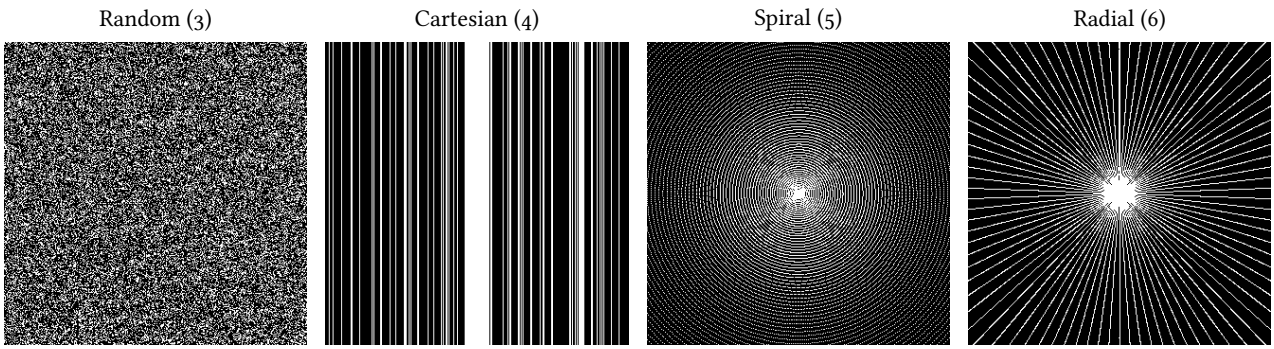

Figure 5.10: A visualization of the frequency selection operators in the synthetic experiments: Low frequencies are understood to be in the center. Frequencies overlaid with a white pixel are understood to be present in the data, frequencies overlaid with a black pixel are not present in the data. With the random frequency selection, in contrast to all others, low frequencies are not more densely sampled. The number in parenthesis shows the acceleration, i.e. the number of all possible frequencies divided by the number of selected frequencies.

Qualitative reconstruction results for the different reconstruction problems are shown in fig. 5.11, corresponding quantitative results in terms of PSNR, NMSE, and SSIM in table 5.1. The table also shows the number of learnable parameters and the reconstruction time.

We start with the prototypical reconstruction problem, Cartesian frequency selection with 8 % autocalibration lines and acceleration 4, shown in the second row. This setup matches the training conditions of the discriminative UNet, resulting in satisfactory reconstructions. However, even in this case the network hallucinates structures into the reconstruction that are not reflected in the data; an example is highlighted in fig. 5.11.[41] The isotropic TV regularizer reconstruction shows typical back-folding artifacts, and increasing the regularization would lead to significant loss of detail. Our data-driven regularizer produces reconstructions qualitatively (at least) on par with the discriminative U-Net. Quantitatively, our reconstructions consistently beat the reference methods. In particular, the our MMSE reconstruction beats the discriminative UNet by over 2 dB in PSNR, even though the UNet was specifically trained for this task. Generally, the MMSE estimate is superior to the MAP estimate quantitatively and qualitatively, which is expected by the discussion on Bayes estimates in chapter 1.

41: Recall that data consistency is not explicitly enforced here. See also the previous discussion on the pitfalls of discriminative signal recovery in section 5.2.

The third row shows the reconstructions spiral frequency selection, with acceleration 5. The spirals sample low frequencies more densely compared to high frequencies. Despite less data availability compared to the Cartesian case, the reconstruction obtained by the TV regularizer shows less back-folding artifacts, likely due to the availability of more high-frequency data, especially horizontal frequencies (see the visualization of the frequency selection operators in fig. 5.10). However, some back-folding artifacts remain, especially in the background. The U-Net struggles to discriminate between details in the anatomy and back-folding artifacts in the intermediary zero-filling reconstruction, leading to unnatural reconstructions and loss of detail. As an example, a large dark spot in the femur in the zero-filling reconstruction is interpreted as an anatomical detail and persists in the final reconstruction. In contrast, our approach is able to faithfully reconstruct the knee with no visible artifacts. The reconstructions obtained with the

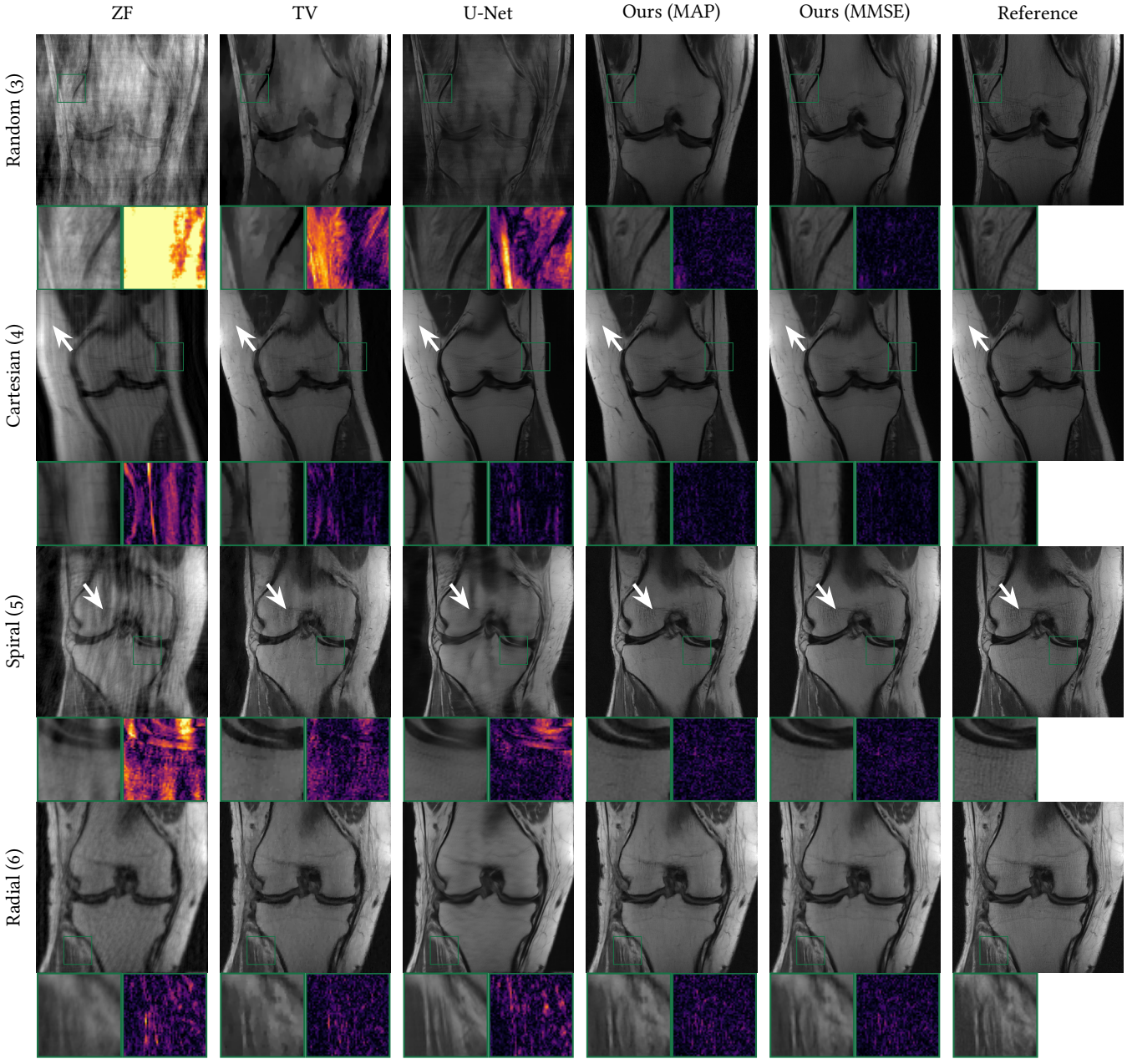

Figure 5.11: Results on synthetic single-coil data. First row: Random frequency selection with acceleration 3. Second row: Cartesian frequency selection acceleration 4 and 8 % autocalibration lines. Third row: Spiral frequency selection with acceleration 5. Fourth row: Radial frequency selection spokes with acceleration 6. A visualization of the frequency selection operators can be seen in fig. 5.10. The inlays show a detailed zoom and the magnitude of the difference to the reference (0 0.2). The white arrow highlights features hallucinated by the discriminative UNet. This is especially notable in the second row (Cartesian frequency selection), where the reconstruction task corresponds to the training setup. Best viewed electronically.

	Acc.		ZF	TV	U-Net	Ours		ScoreMRI	Ours	
						MAP	MMSE		MAP	MMSE
Random	3	P	12.93	20.81	19.52	30.87	**32.44**	33.31	31.78	**34.76**
		N	64.24	10.22	12.83	1.34	**1.02**	—	—	—
		S	0.48	0.73	0.57	0.86	**0.90**	—	—	—
Cartesian	4		24.16	31.47	34.16	35.53	**36.17**	34.65	34.97	**35.55**
			5.96	0.88	0.44	0.32	**0.28**	—	—	—
			0.70	0.85	0.89	0.89	**0.91**	—	—	—
Spiral	≈ 5		21.21	31.25	27.76	35.35	**36.21**	35.66	36.07	**37.00**
			10.24	0.92	1.90	0.34	**0.28**	—	—	—
			0.62	0.85	0.78	0.88	**0.90**	—	—	—
Radial	≈ 6		27.02	32.86	31.76	35.04	**35.47**	34.98	36.02	**36.56**
			3.29	0.64	0.76	0.36	**0.33**	—	—	—
			0.48	0.71	0.65	0.86	**0.89**	—	—	—
Learnable parameters			0	0	5.0×10^8	2.1×10^7		6.8×10^7	2.1×10^7	
Time in seconds			3×10^{-4}	0.59	0.10	3.13	333.30	251.60	3.13	333.30

Table 5.1: Quantitative results for the synthetic experiments with different frequency selections. The rows alternate between PSNR (P, ↑), NMSE (N, ↓), and SSIM (S, ↑). Acc. is the acceleration, bold typeface indicates the best method. The comparison against the ScoreMRI method of [53] is shown separately because the evaluation was done on only on a subset of the data.

radial frequency selection shown in the fourth row are similar; the radial frequency selection also samples low frequencies more densely compared to high frequencies.

In contrast, random frequency selection does not sample low frequencies more densely, which manifests in the zero-filling reconstruction by large-scale intensity shifts. None of the reference methods can correct this: The TV regularizer can not resolve nonlocal dependencies, and the UNet fails to distinguish between anatomical features and back-folding artifacts in the intermediary reconstruction. Our approach can restore knee's general shape and retains details present in the data, alleviating the need for frequency selection operators to densely sample low frequencies.

During the writing of the paper and the thesis, the use of diffusion models in inverse problems gained popularity. We compare our approach to Chung and Ye's diffusion-based method [53]. However, since this approach is very time consuming, we limit the evaluation to a subset of the data used for the other models. Quantitative results in table 5.1 are shown separately; qualitative results in fig. 5.12. Our MMSE estimate consistently outperforms the diffusion-based approach, and our MAP estimate is only inferior for the random frequency selection. Computing the MMSE estimate takes only about 30 % more time than computing one sample with the diffusion-based approach,[42] while computing our MAP estimate is about 80 times faster.

42: This is in our particular setup, see the details in section 5.4.4.

Our approach allows us to trade off reconstruction quality with computational time: The MMSE estimate gracefully approaches the MAP estimate as we take

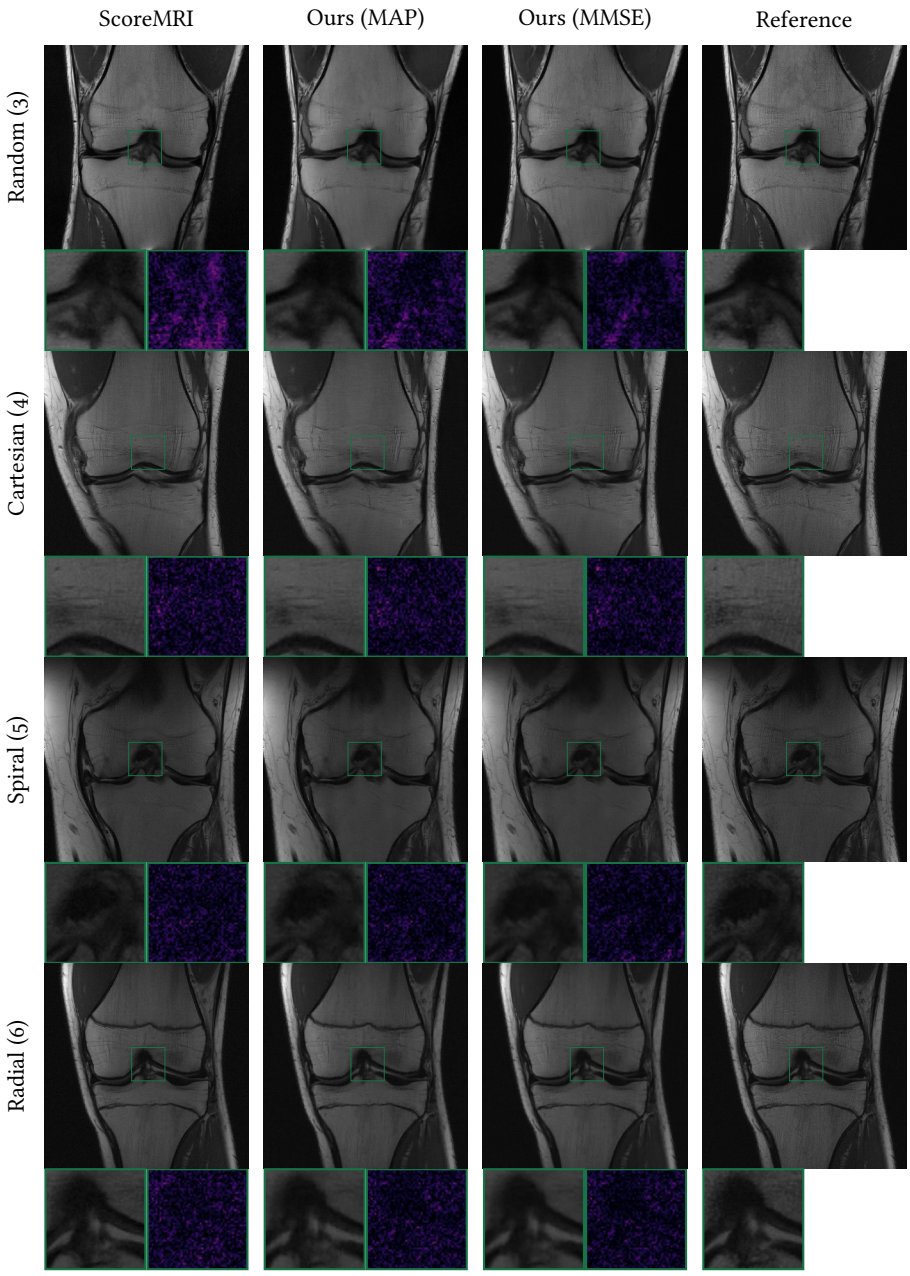

Figure 5.12: Qualitative comparison against [53] on in-distribution data: First row: Random frequency selection with acceleration 3, Second row: Cartesian frequency selection with 8 % autocalibration lines and acceleration 4, Third row: Spiral frequency selection with acceleration 5, Fourth row: Radial frequency selection with acceleration 6. The inlays show a detail zoom, and the magnitude of the difference to the reference (0 0.2).

fewer and fewer samples, since we initialize the unadjusted Langevin algorithm with the MAP estimate. Although significant advances have been made to speed up diffusion models (see e.g. [128, 52]), they are in general still slower.

5.5.3 Uncertainty quantification

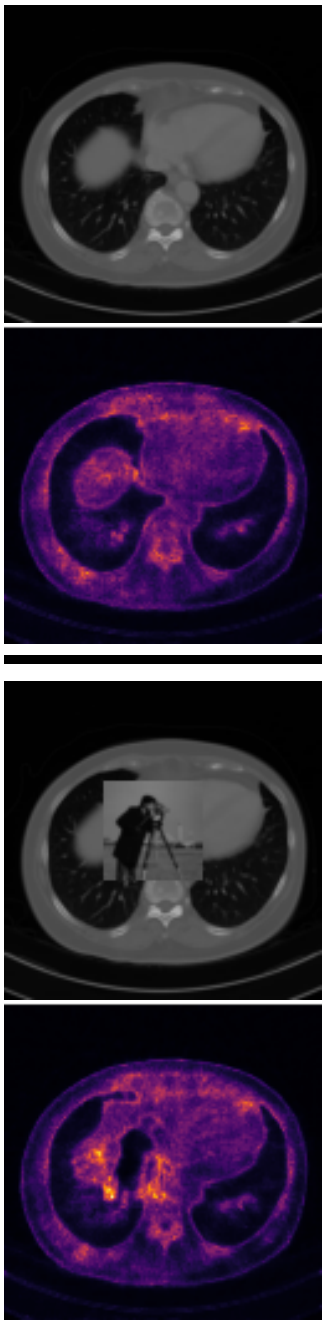

Figure 5.14: The pixel-wise marginal variance in the region around the unnatural cameraman is significantly higher than in the reference scan.

The strict Bayesian approach taken in this thesis allows access to a *posterior distribution* of reconstructions for a datum. In the previous section, we computed the MAP as well as the MMSE estimate of this distribution. The MMSE estimate is the expectation (definition 2.2.14, the first moment) of the posterior. In this section, we discuss higher order moments, such as the variance definition 2.2.15—the second moment.

Computing the second moment of the posterior distribution for our reconstruction problems is infeasible: This moment is described by $(320\times320)\times(320\times320) = 10\,485\,760\,000$ real numbers. Therefore, we restrict our focus to the pixel-wise marginal variance, as is commonly done in this context [53, 170, 185]. Notably, computing the pixel-wise marginal variance does not incur additional computational cost when computing the MMSE estimate: The MMSE estimate is obtained by taking the pixel-wise average of all samples from the posterior, while the pixel-wise marginal variance is computed by taking the pixel-wise variance.

In fig. 5.13, we present the MMSE estimate along with the pixel-wise marginal variance and the magnitude of the error for the same reconstruction problems discussed in the previous section. As expected, the pixel-wise marginal variance increases with the acceleration: random frequency selection with acceleration 3 shows the smallest variance, while radial frequency selection with acceleration 6 shows the highest. The additional variance information can aid clinicians in decision making and enhance interpretability of the results. In addition, it can be combined with conformal prediction techniques to provide coverage guarantees for error bounds, as proposed by Narnhofer, Habring, Holler, and Pock [170].

In our proof-of-concept paper dealing with Radon imaging, we investigated the possibility of using the pixel-wise marginal variance for pathology detection. We briefly discuss the results here, referring to the publication [248] for more details on the experimental setup and discussion of the results. We introduce an artificial structure (a "pathology") in the image by overlaying the "cameraman" onto the anatomy. Comparing the pixel-wise marginal variance in fig. 5.14, the variance around the artificial structures is significantly higher than in the reference scan. This suggests the potential for developing a pathology detection system based on this variance. Pathologies, seen as anomalies, align with the broader field of anomaly detection methods discussed in [182].

5.5.4 Parallel imaging

The previous sections outlined the benefits of our data-driven regularizer. In this section, we demonstrate that our proposed algorithm, combined with the data-driven regularizer, achieves state-of-the-art results in parallel MRI reconstruction problems encountered in clinical practice. We first focus on the reconstructions,

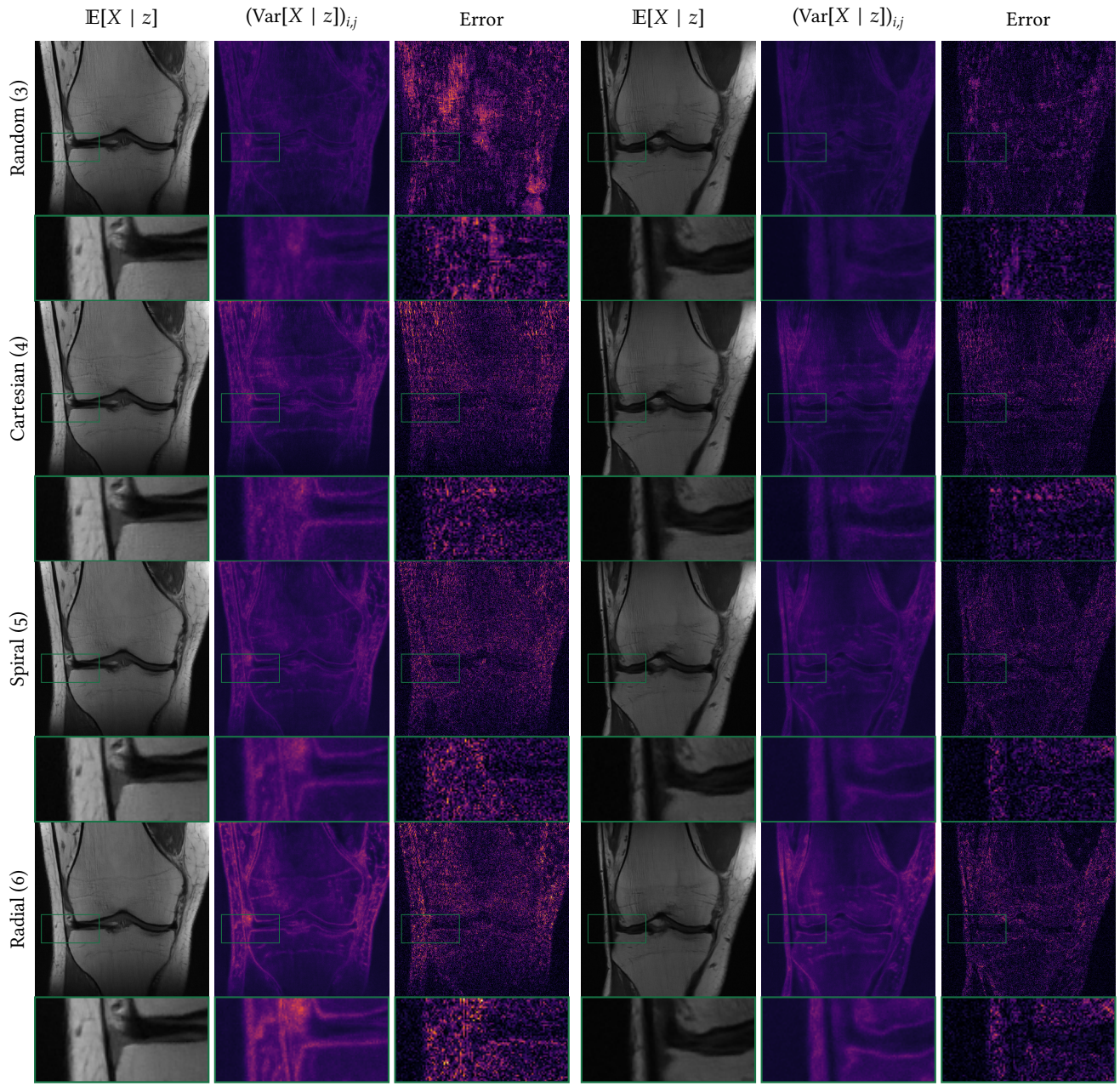

Figure 5.13: The MMSE estimate along with the pixel-wise marginal variance (0 0.0025) and the magnitude of the error to the reference signal. First row: Random frequency selection with acceleration 3. Second row: Cartesian frequency selection with 8 % autocalibration lines and acceleration 4. Third row: Spiral frequency selection with acceleration 5. Fourth row: Radial frequency selection with acceleration 6. The variance increases as less data is available and could be used to obtain coverage guarantees for error bounds [170]. A visualization of the frequency selection operators can be seen in fig. 5.10.

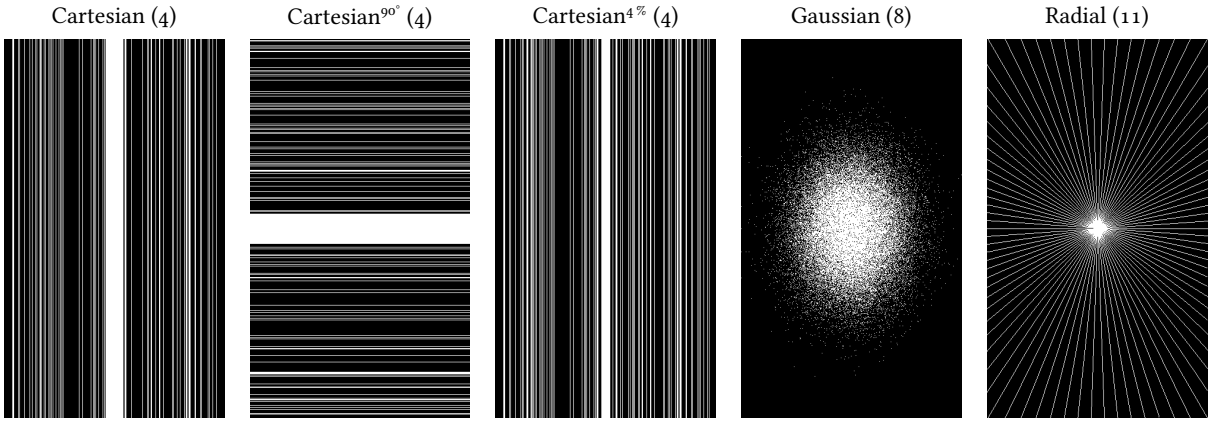

Figure 5.15: A visualization of the frequency selection operators we consider in the parallel imaging experiments: Low frequencies are understood to be in the center. Frequencies overlaid with a white pixel are understood to be present in the data, frequencies overlaid with a black pixel are not present in the data. The number in parenthesis shows the acceleration, i.e. the number of all possible frequencies divided by the number of selected frequencies.

showing excellent results for numerous frequency selection operators. Then, we examine the coil sensitivity estimates, highlighting the advantages of our approach over offline coil sensitivity estimation, especially when little data are available.

We assume that the data $z_1, z_2, \dots, z_c$ of $c \in \mathbb{N}$ coils are given by eq. (5.19), with both the coil sensitivities and the underlying signal are unknown. As in previous sections, we assume that the underlying signal comes from the same distribution that the regularizer was trained on, p_X. We consider five different frequency selection operators:

1. A Cartesian selection with densely sampled frequencies in the phase-encoding direction, 8 % autocalibration lines, and acceleration 4,
2. the same setup as in item 1 but with swapped phase-encoding direction,
3. the same setup as in item 1 but with 4 % autocalibration lines,
4. a two-dimensional Gaussian selection with acceleration 8, and
5. a radial selection with acceleration 11.

The two-dimensional Gaussian selection is understood as assigning each frequency a probability of being included that falls off as a Gaussian from the center. These are visualized in fig. 5.15.

We present qualitative reconstruction results for the different reconstruction problems in fig. 5.16, and the corresponding quantitative results in terms of PSNR, NMSE, and SSIM in table 5.2.

The first row of fig. 5.16 shows the prototypical Cartesian frequency selection with 8 % autocalibration lines and acceleration 4, corresponding to the training setup of the discriminative end-to-end VN of [222]. Consequently, the reconstructions are satisfactory, with the VN achieving the best quantitative results with a PSNR of 36.92 dB on the test set. Our method also achieves competitive results with a PSNR of 35.23 dB. This is expected, as generative approaches typically do not outperform discriminative counterparts.

	A	ACL	In-distribution (CORPD)					Out-of-distribution (CORPDFS)				
			ZF	TV	VN	Ours		ZF	TV	VN	Ours	
						MAP	MMSE				†	∗
C	4	8 %	27.19	31.87	**36.92**	35.23	35.28	26.09	31.30	30.00	30.60	**31.71**
			2.24	0.79	**0.24**	0.36	0.36	5.35	1.48	2.38	1.95	**1.35**
			0.74	0.81	**0.92**	0.89	0.89	0.68	0.73	**0.77**	0.73	0.73
		8 %	31.13	33.03	24.72	**36.23**	36.01	26.58	31.56	28.57	30.89	**31.65**
			0.93	0.59	4.01	**0.28**	0.30	5.15	1.40	2.90	2.24	**1.37**
			0.81	0.83	0.67	**0.90**	**0.90**	0.71	0.73	0.71	**0.75**	0.73
		4 %	24.14	25.81	32.16	**35.33**	35.22	24.98	29.91	29.12	29.93	**31.26**
			4.51	3.48	0.70	**0.35**	0.36	6.65	2.09	2.79	2.92	**1.53**
			0.69	0.70	**0.89**	**0.89**	**0.89**	0.65	0.71	**0.76**	0.73	0.72
R	11	—	28.76	32.47	20.56	**34.46**	34.16	25.06	31.15	26.26	**31.43**	31.36
			1.57	0.67	10.13	**0.42**	0.45	7.37	1.53	4.97	**1.45**	1.57
			0.75	0.81	0.69	**0.86**	**0.86**	0.62	0.71	0.69	**0.73**	0.70
G	8	—	32.10	34.14	20.74	35.35	**35.41**	26.75	31.52	23.46	31.87	**32.09**
			0.74	0.45	9.95	**0.34**	**0.34**	5.46	1.42	9.93	1.43	**1.25**
			0.84	0.85	0.68	0.88	**0.89**	0.71	0.75	0.68	**0.76**	0.75

Table 5.2: Quantitative results for parallel imaging with different frequency selections on in- and out-of-distribution data. The rows alternate between PSNR, NMSE, and SSIM. The † column shows results using the CORPD λ-fit, while the ∗ column has CORPDFS-adapted parameters (see section 5.4.5).

The benefits of our approach become apparent when we slightly change the frequency selection: the performance of the end-to-end VN deteriorates significantly when the phase-encoding direction is swapped, or when less autocalibration lines are acquired. Notably, the acceleration remains *fixed*. These minimal changes in the frequency selection cause the end-to-end VN to struggle with back-folding artifacts or introduces severe hallucinations. In contrast, our approach maintains stable performance across all tasks, reflected in the quantitative evaluation in table 5.2. For example, the PSNR of the reconstructions of the end-to-end VN drops by 12.20 dB when phase-encoding direction is swapped, whereas our approach's PSNR increases by 1 dB.[43] When reducing autocalibration lines from 8 % to 4 %, the PSNR of the end-to-end VN drops by 4.76 dB, whereas the PSNR of our approach remains almost constant.[44]

The situation is similar for the two-dimensional Gaussian and radial frequency selection, but the artifacts introduced by the end-to-end VN become even more pronounced. This is due the sensitivity estimation sub-network in the end-to-end VN failing, which assumes a densely samples low frequencies, similarly to offline coil sensitivity estimation algorithms. In addition, the image estimation sub-network is confronted with unseen features since it is coupled to the sensitivity estimation network. Quantitatively, the end-to-end VN performs much worse than the TV reconstruction for these frequency selections, although the TV reconstructions

43: Due to the knee's longitudinal arrangement in the scanner, swapped phase encoding is advantageous because more horizontal high-frequency information is available, see the visualization in fig. 5.15.

44: It increases by 0.1 dB.

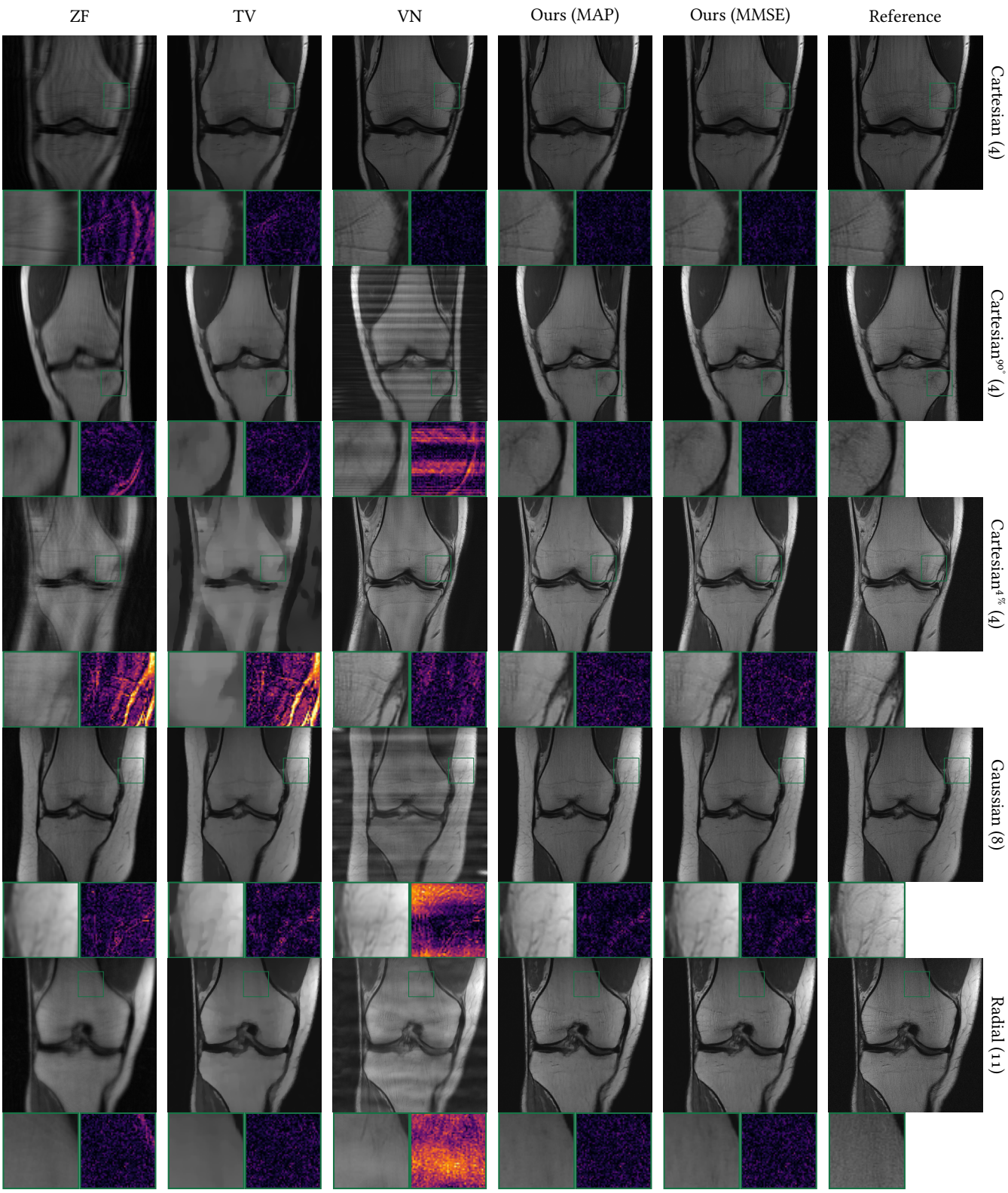

Figure 5.16: Parallel imaging on in-distribution data: First row: 4-fold Cartesian frequency selection with 8 % autocalibration lines and acceleration 4. Second row: As in the first row but with swapped phase encoding direction. Third row: As in the first row but with 4 % autocalibration lines. Fourth row: 2D Gaussian frequency selection with acceleration 8. Fifth row: Radial frequency selection with acceleration 11. The inlays show a detail zoom and the magnitude of the difference to the reference (0 0.2).

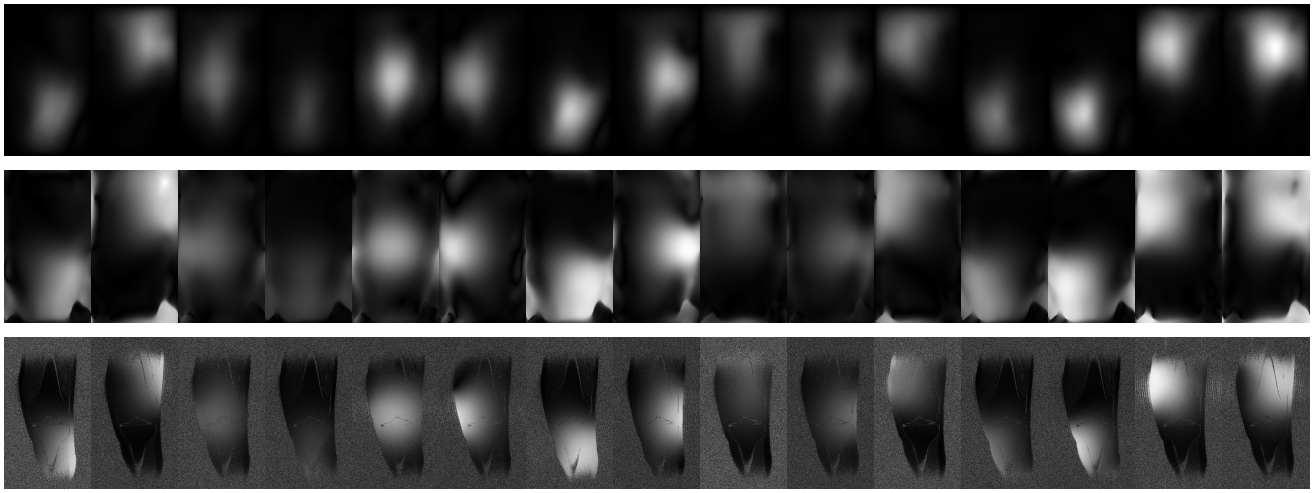

Figure 5.18: Magnitude of the estimated sensitivities using our joint nonlinear inversion algorithm (top) versus the ESPIRiT [232] estimation (middle) for the reconstruction problem shown in the first row of fig. 5.16. The bottom row shows the reference coil sensitivities computed with the fully-sampled data.

appear overly smooth. Our approach satisfactorily reconstructs the image, yielding the best performance both quantitatively and qualitatively.

In the third row of fig. 5.16, we highlight a failure case of our algorithm: In the second columns, with the Charbonnier-smoothed isotropic TV regularizer, parts of the details of the anatomy have "slipped" into the coil sensitivities, some of which we show in fig. 5.17. As a result, the reconstruction appears extremely smooth and anatomically implausible. However, this can be corrected by appropriately choosing the respective regularization parameters, μ and λ in eq. (5.23).[45] We never observed these types of failures when using our data-driven regularizer in the reconstruction algorithm and believe that they are extremely rare: As indicated in fig. 4.19, it prefers anatomically plausible structures over smooth images, facilitating the separation between the anatomy in the image and the coil sensitivities.

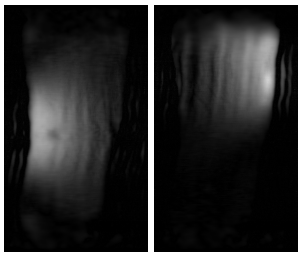

Figure 5.17: Under-smoothed coil sensitivities for the failure case of our algorithm in the third row and the second column of fig. 5.16.

45: The details of our choice are in section 5.4.

Finally, we analyze our proposed reconstruction algorithm with respect to the estimated coil sensitivities. We consider the reconstruction problem shown in the first and third row of fig. 5.16: Cartesian frequency selection with acceleration 4, and 8 % and 4 % autocalibration lines respectively. As a reference offline estimation method, we chose the ESPIRiT algorithm [232]. The estimated coil sensitivities for the 8 % autocalibration lines reconstruction problem are shown in fig. 5.18. Due to the data fidelity term that we use in practice (see the discussion in section 5.4.2, in particular eq. (5.32)), the coil sensitivities from our estimation algorithm need not be pixel-wise normalized. Hence, they look very physically plausible and match the reference well.

To better understand the quality of the estimation, we follow [232] and visualize the null-space residual in fig. 5.19. As discussed in section 5.4.5, any residual signal components indicate a suboptimal estimation of the coil sensitivities. While the ESPIRiT estimation leads to slightly better results with 8 % autocalibration lines, its performance deteriorates with only 4 % autocalibration lines. In contrast, our estimation remains stable, as the joint nonlinear inversion accounts for all available

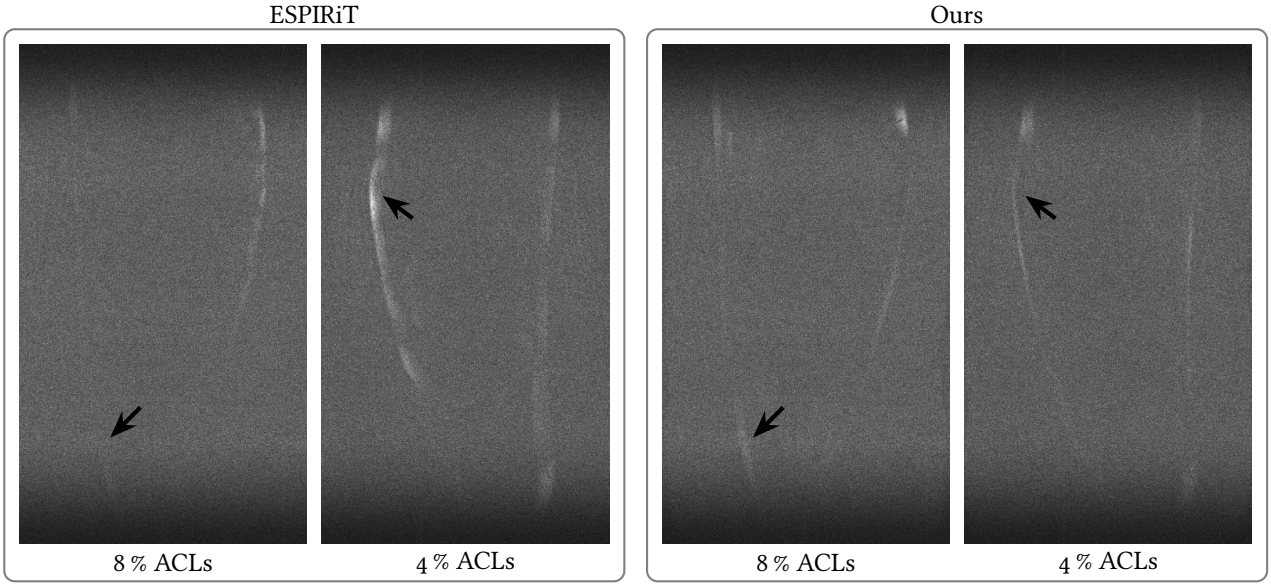

Figure 5.19: Qualitative comparison of the coil sensitivity estimation using the null-space residual: Any residual signal components point to a suboptimal estimation. We consider the Cartesian frequency selections with acceleration 4 in the first and third row of fig. 5.16: 8 % autocalibration lines and 4 % autocalibration lines.The proposed joint nonlinear inversion algorithm (right) provides superior estimates compared to offline estimation using ESPIRIT [232] (left) when little autocalibration data are available.

data, nut just autocalibration lines.

5.5.5 Generalization

A large part of this chapter was dedicated to designing a regularizer that can encode high-level features of the reference distribution. We have demonstrated in fig. 5.9 that the regularizer (also due to the generative training) has indeed learned high-level features of the reference distribution. In the previous sections we demonstrated thoroughly that utilizing the regularizer in a variational reconstruction framework leads to high-quality reconstructions for a variety problems where the underlying signal is a sample from the reference distribution. In this section, we investigate how well our regularizer works for reconstruction problems where the underlying signal is *not* a sample from the reference distribution. Therefore, we consider three scenarios where the underlying signal is a

1. CORPDFS MRI scan of a human knee, an
2. MRI scan of a human brain, or an
3. MRI examination of a prostate.

In some sense, the items in the enumeration are increasingly out-of-distribution: The fat-suppression—although it drastically changes the over-all appearance of the image—leaves edges at the same positions as the non-fat-suppressed scans and the image is still easily identified as a knee. The brain scans are similar to the knee scans in the local structures appear similar and in that they have a region of almost all air around the region containing the signal, which is not the case for the prostate scans. This is shown in fig. 5.20.

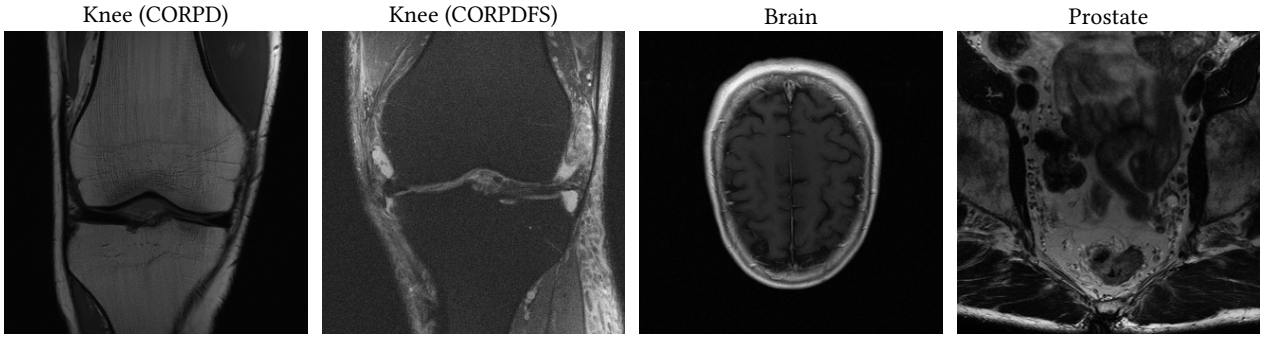

Figure 5.20: From left to right, the signals become increasingly different: The fat-suppressed knees retain the shape and edges. The brains local structures appear similar and it is surrounded by a region of almost zero-signal. The prostate scans appear completely different overall.

First, we evaluate the data-driven regularizer on the fat-suppressed knee scans, following exactly the same setup as in the previous section. In particular, we also evaluate the Charbonnier-smoothed isotropic TV regularizer as well as the end-to-end VN on this dataset. We show qualitative qualitative results in fig. 5.21. The corresponding quantitative results were already shown in table 5.2. In summary, the quantitative results indicate our method generalizes slightly better to unseen data than the end-to-end VN approach of [222], although the performance degrades significantly in both cases.

In contrast to the end-to-end VN, we can adapt the regularization strength to the new data. To highlight the advantage of the tunable regularization parameters in our approach, we show results using the λ-fit (see section 5.4.5) calculated on non-fat-suppressed data, as well as using parameters adapted to the task: Obviously, the ability to tune the influence of the regularizer leads to improved performance when confronted with previously unseen data.

For the brain and prostate scan, we restrict ourselves to a setup as in chapter 4; the reference images and the data are shown in fig. 5.22 and fig. 5.23 for the brain and the prostate scan respectively. The scans are taken from the fastMRI brain dataset [168] and the fastMRI prostate dataset [227].[46]

We show the reconstructions depending on the regularization parameter in fig. 5.24. The reconstructions of the brain appear reasonable, with the reconstruction becoming smoother as the regularization becomes stronger. The reconstructions of the prostate have knee-like structures, especially at the boundary. These boundary effects deserve more investigation and highlight that the regularizer is only expected to work well when the underlying signal is from the distribution it models.

46: The images are the files `AXT1_202_2020190_s8` (brain, eighth slice) and `AXT2_013` (prostate).

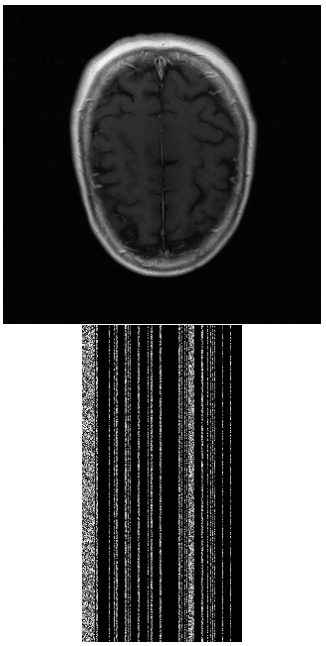

Figure 5.22: The reference signal and the zero-filled data for the brain scan.

5.6 Discussion

This chapter's first part focused on designing an architecture suitable for modeling the negative log-prior. Unlike classical regularizers used in imaging applications, the regularizer is *not* translation invariant and can model non-local dependencies.

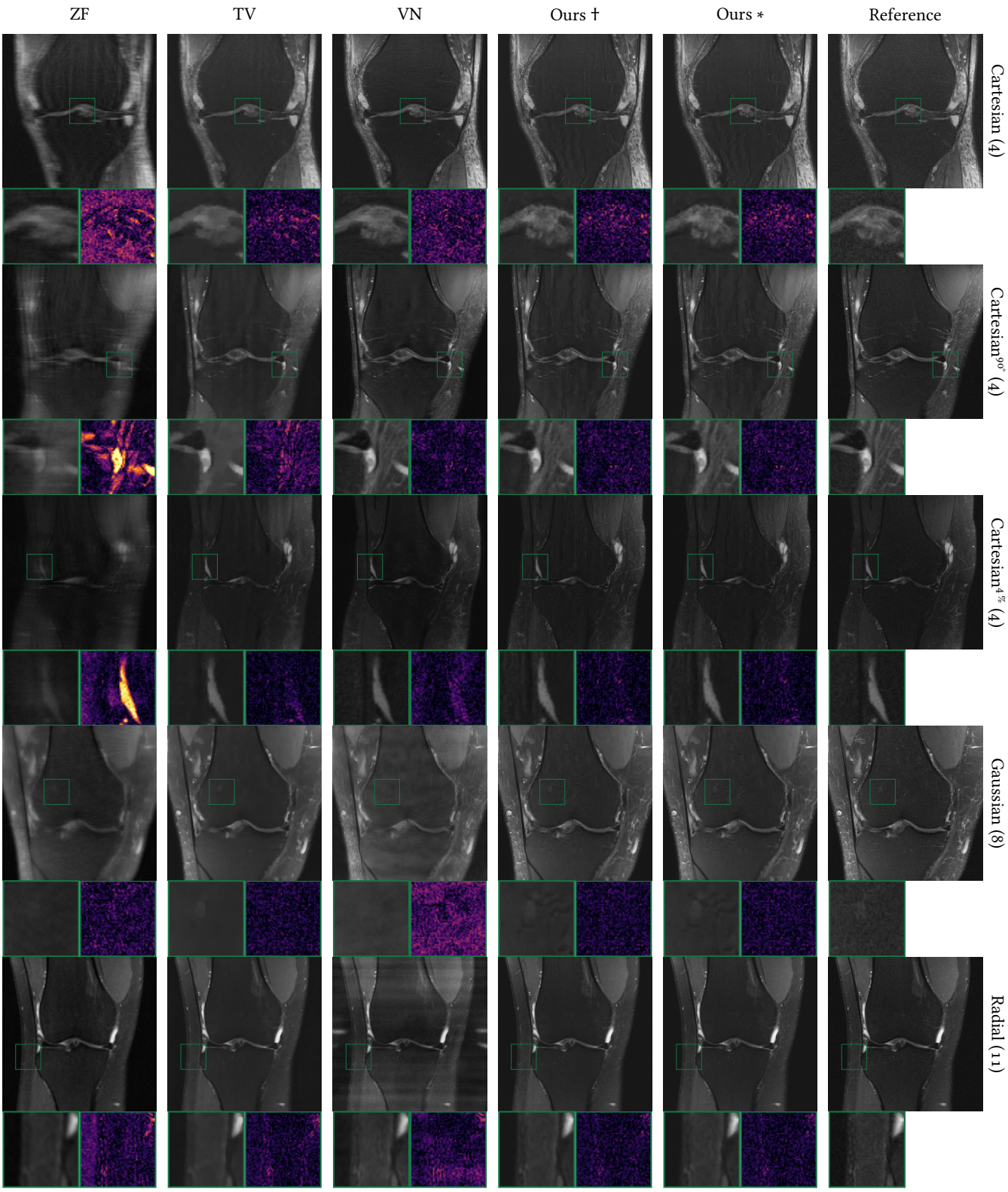

Figure 5.21: Parallel imaging on out-of-distribution data. First row: Cartesian frequency selection with 8 % autocalibration lines and acceleration 4. Second row: As in the first row but with swapped phase encoding direction. Third row: As in the first row but with 4 % autocalibration lines. Fourth row: 2D Gaussian frequency selection with acceleration 8. Fourth row: Radial frequency selection with acceleration 11. The inlays show a detail zoom and the magnitude of the difference to the reference (0 0.2). The † column shows the results using the regularization parameters from non-fat-suppressed data, the ∗ column shows the results using adapted parameters.

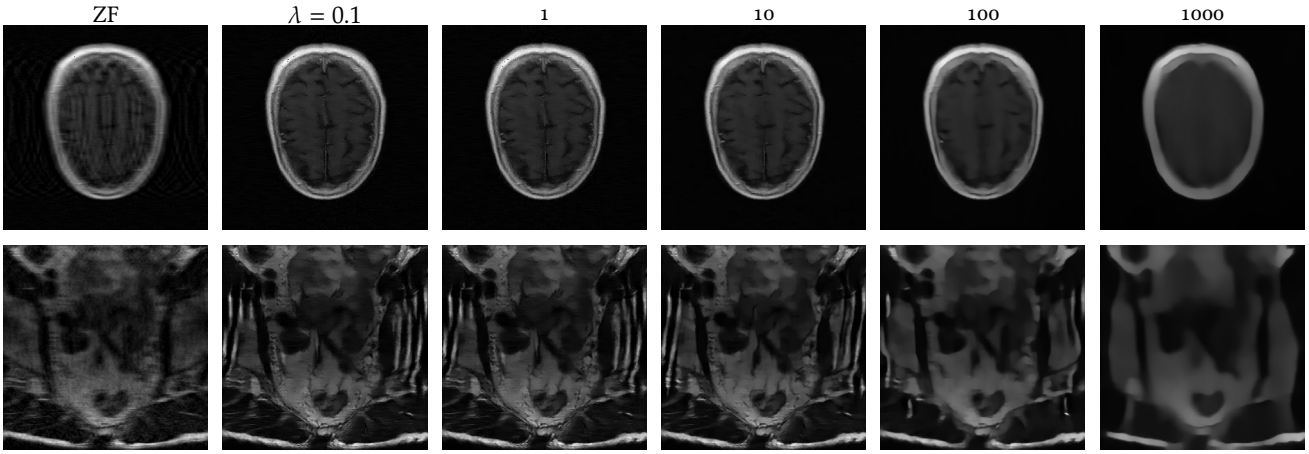

Figure 5.24: Simulation study on out-of-distribution signals: When the underlying signal is an MRI scan of the brain, the regularizer yields reasonable solutions. When the underlying signal is an MRI scan of the prostate, it tries to impart knee-like structure, especially at the boundaries.

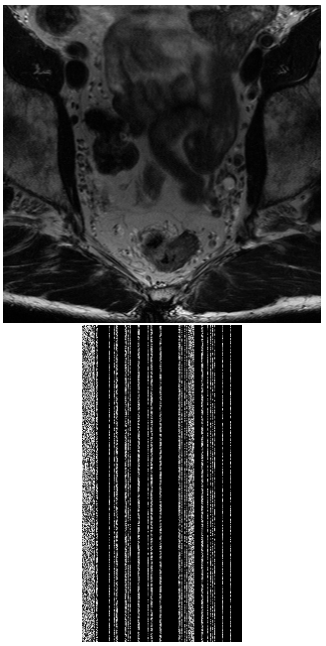

Figure 5.23: The reference signal and the zero-filled data for the prostate scan.

Coupled with a generative learning approach, the data-driven regularizer encodes high-level domain statistics as demonstrated by synthesizing realistic knee MRIs without any data (see fig. 4.19). Consequently, reconstructions from severely ill-posed problems, such as MRI with a random frequency selection and acceleration 3, appear natural.

Using a joint nonlinear inversion algorithm, we achieved state-of-the-art reconstructions in clinically relevant parallel MRI settings. The strict separation of likelihood and prior, combined with our generative learning approach, ensures stable reconstruction quality across various frequency selection, as shown in fig. 5.16. This is a significant advantage over traditional methods, where reconstruction quality often depends on specific frequency selections, such as phase encoding direction, which may need to be adjusted when encountering problematic blood vessel positions. Our regularizer does not require retraining for new advantageous frequency selections.

Our regularizer also performs well in out-of-distribution experiments, underscoring the importance of controlling its influence. table 5.2 shows that adjusting regularization strength to the underlying data strongly improves performance. Regularization strength was tuned for Cartesian frequency selection with 8 % autocalibration lines (see section 5.4.5) and acceleration 4. Results for the radial frequency selection indicate a sub-optimal fit for the out-of-distribution data. The fit on the CORPD data († column) performs better than the fit calculated on CORPDFS data. fig. 5.21 suggests that the adapted parameters lead to an over-smoothed reconstruction, which could be improved by adapting the regularization strength to the CORPDFS data *and* the radial frequency selection. Generally, adapting the regularization strength to the frequency selection and the data is beneficial, but typically not possible with other data-driven approaches.

The probabilistic interpretation of our approach is significant in two ways: First, the regularizer can be inspected through data-independent analysis (see fig. 4.19) allowing experts to visualize preferred structures and improving clini-

cian confidence in reconstructions. Second, since any datum provides a posterior distribution of reconstructions, we can compute compute pixel-wise marginal variances, offering clinicians additional information. For example, fig. 5.13 shows high variance around small anatomic structures, suggesting clinicians might need more information if decisions are based on these areas. Most other data-driven approaches only provide a point estimate.

Experiments on synthetic single-coil data show the MMSE's quantitative and qualitative superiority of over the MAP estimate. In light of the introductory chapter this is not surprising, but in addition to the rational outlined there we believe that this is related to the training procedure: During training, the regularizer encounters slightly noisy images in the Langevin process (see section 5.3.2), and injecting the same noise in the reconstruction algorithm improves performance. However, this performance improvement does not translate to parallel imaging experiments, possibly because our joint nonlinear inversion biases the reconstruction such that this effect is no longer observable.

Our parallel MRI reconstruction algorithm is fast: algorithm 8 converges in around 100 iterations, taking around 5 s on an NVIDIA Titan RTX using approximately 2 GB of memory. This contrasts with diffusion models where reconstruction times of up to 10 min are reported [53]. Speed is advantageous in practice, allowing images to be viewed while the patient is still in the scanner and enabling immediate adjustments to the frequency selection if needed.

5.7 Conclusion

We leverage modern generative learning techniques to train a regularizer that faithfully encodes the underlying distribution of the training data. Embedding this regularizer in a variational reconstruction framework enables satisfactory reconstruction of both single-coil and parallel MRI by adapting the data fidelity term. Quantitative and qualitative analyses show that our approach achieves competitive reconstruction performance, matching or surpassing discriminative approaches. Furthermore, our approach is robust to changes in the frequency selection, while the discriminative reference methods introduce severe hallucinations when confronted with previously unseen data. The anatomical knowledge encoded in our regularizer allows us to reconstruct high quality images even from random sampling masks. On out-of-distribution data, our method performs as well as or better than hand-crafted regularizers such as TV or other discriminative methods, demonstrating superior generalization to distribution shifts.

Our method also provides a natural accompanying probabilistic interpretation through statistical modeling and Bayesian inference. This allows experts to explore a distribution of reconstructions, whereas other data-driven approaches typically provide only point estimates. Experiments suggest that this distribution encodes important diagnostic information, such as high uncertainty around small anatomical structures, potentially aiding clinical decision making. Additionally, we can perform a data-independent analysis of the model by visualizing the Gibbs distribution, providing insight into the information encoded in the regularizer

unlike the black-box nature of other data-driven methods.

For parallel MRI reconstruction, we propose a fast algorithm that jointly estimate the image and the coil sensitivities. Unlike offline sensitivity estimation, our approach does not require autocalibration lines, can be applied to non-Cartesian sampling trajectories and nonuniform sampling, and utilizes all available data. The resulting optimization problem can be solved efficiently using non-convex optimization algorithms, delivering high quality reconstructions in approximately 5 s on consumer hardware.

We believe that reconstruction approaches based on generative priors hold significant potential. A natural extension of our work would be to combine the image-prior with a learned sensitivity-prior. Future research could also explore enhancing the simple architecture used in this paper with more modern building blocks, such as attention layers. Additionally, investigating whether local convolutional models can replicate the performance of our generative prior is a promising direction for further study.

Chapter 6

Product of Gaussian mixture diffusion models

In the previous chapter, we moved beyond classical ridge-type regularizers, as discussed in chapter 4, and introduced an architecture for a deep neural regularizer capable of resolving nonlocal dependencies and without translation invariance. This was desirable in order to model the distribution of MRI scans of the human knee. In this chapter, we revisit the structure of the classical ridge-type regularizers but with a novel twist.

Contents:

Typically, the potentials of the ridge-type regularizers discussed in chapter 4 have rigid parametric forms inspired by classical regularization theory. For instance, the potentials for Gaussian, generalized Laplacian, or Student-t distributions exhibit a global minimum at zero and lack other local minima. As noted in [255], this choice is motivated by classical regularization theory.

However, under the strict Bayesian view that we adopt in this thesis—where the regularizer should model the negative log-prior—this class of potentials is too restrictive. Amongst the works that have pointed this out is the seminal paper [255] by Zhu and Mumford, who use piecewise constant potentials[1] with arbitrary shapes to model the statistics of natural images. Their generative maximum entropy approach recovers potentials that sometimes feature a *maximum* at zero and decrease monotonically away from zero [255, fig. 9]. However, the use of piecewise constant potential functions hinders the application in first-order optimization-based image reconstruction approaches.

1: They work with quantized images and the piecewise constant potentials are essentially negative-log histograms, where the bins arise naturally through the quantization.

Similar observations were made in the context of data-driven discriminative approaches: For example, Chen and Pock's trainable nonlinear reaction diffusion model [46], which falls under the class of learned optimization schemes (such as also the VN in the previous chapter) employs a general parametrization of the potential functions[2] using radial basis functions. They recover more complex potentials, such as negative Mexican-hat- or double-well-type potentials [46, fig. 5] with multiple local minima that do not contain zero.

2: In their work, the filters and potential functions change during the iterations of the learned iterative scheme, thus their role is slightly different.

The choice of the Student-t potentials in the FoE model due to Roth and Black [206] was a limitation in this respect. In [209] Schmidt, Gao, and Roth revisited this model and identified that it could not reproduce the edge statistics of natural images, despite generative training. They attribute this largely to the

inefficient MCMC sampler used in the training of the original model. To remedy this, they propose to use Gaussian scale mixture experts than enable efficient sampling via an auxiliary variable Gibbs sampler. Although this improved results, they still fell short of reproducing marginal statistics of random filters, likely due to the restrictive choice of Gaussian scale mixture experts.[3] While more general that Student-t experts, the potentials of Gaussian scale mixtures (Student-t experts can be represented by Gaussian scale mixtures, see section 6.5.2) are still monotonically increasing away from zero.

3: We believe that their work is easily extended to more general Gaussian mixtures, such as mixtures of Gaussian scale mixtures.

In unrelated work, Heess, Williams, and Hinton [111] identified the limitation of the choice of experts in the original FoE model. The authors propose to replace the unimodal Student-t experts with slightly more general bimodal experts, thereby improving the performance on texture synthesis tasks.

This chapter addresses both deficiencies of the original FoE model: First, we use more general Gaussian mixture experts, translating to constrained log-sum-exp potentials. This versatile parametrization allows multi-well potentials with arbitrarily steep valleys, depending on the choice of the smallest variance in the Gaussian mixture experts. Second, we replace inefficient maximum-likelihood-type training (which requires sampling the model) with score matching-based training. In addition to an efficient training, we exploit connections between score matching and empirical Bayes theory to derive a one-step MMSE optimal denoiser for Gaussian noise with arbitrary variance.

This chapter is structured as follows: In section 6.1 we motivate our work by emphasizing connections to diffusion models and classical ridge-type regularizers in imaging. In section 6.2, we give background information on diffusion and its use in parameter estimation of learned densities, including an overview of related work. In section 5.3, we introduce the backbone of our models and derive conditions under which they obey the diffusion PDE. We demonstrate the practical applicability of our models in section 6.4 with numerical experiments. We explore alternative parametrizations and possible extensions of our models in section 5.6 and finally conclude the paper, providing future research directions, in section 6.6.

This chapter is based on the following publications:

Martin Zach et al. “Explicit Diffusion of Gaussian Mixture Model Based Image Priors”. In: *Proc. of the International Conference on Scale Space and Variational Methods in Computer Vision*. Cham: Springer International Publishing, 2023, pp. 3–15. ISBN: 978-3-031-31975-4

Martin Zach et al. “Product of Gaussian Mixture Diffusion Models”. In: *Journal of Mathematical Imaging and Vision* (Mar. 2024). ISSN: 1573-7683. DOI: 10.1007/s10851-024-01180-3

Code for training, validation, and visualization, along with pre-trained models is available at https://github.com/VLOGroup/PoGMDM.

6.1 Introduction

In the previous chapter, we approached generative modeling as the problem of minimizing the Kullback Leibler divergence (definition 2.2.19) of the reference density from the model density. The fundamental challenge in this approach lies in the computation of the partition function: For general models, such as deep neural networks without specific structure, computing the partition function is intractable, necessitating the use of time-consuming MCMC methods for estimation.

In this chapter, we circumvent the computation of the partition function. Inspired by the literature on diffusion models [217, 219], we use the *denoising score matching* objective to train our models. Introduced by Vincent [238] as an extension to the classical score matching introduced by Hyvärinen [123], denoising score matching minimizes the Fisher divergence (definition 2.2.20) of the *smoothed* reference distribution from the model distribution. Here, smoothing refers to convolving the reference density with a Gaussian.

In practice, the reference distribution is usually an empirical measure, and the variance of Gaussian determines the regularity of the smoothed distribution. However, the desired degree of smoothing is often unclear: A small variance respects the reference distribution locally around the modes but results in bad estimates in low-density regions. Conversely, a high variance removes details from the reference distribution but facilitates the estimation of low-density regions.

This problem has been identified by Song and Ermon in [217] in the context of generative models. They propose to consider an increasing sequence variances for the Gaussian distribution and to learn a network conditioned on the variances. By conditioning the network on the smoothing variance, it encodes the data distribution at various smoothing stages. To generate new samples, they utilize the unadjusted Langevin algorithm while *annealing* the variance conditioning toward zero, calling the resulting algorithm *annealed Langevin dynamics*. By generalizing the increasing sequence of variances to a monotonically increasing function, they recover *diffusion models* in [218]. In these models, a diffusion process (definition 2.3.11) is constructed, where the boundary condition is the random variable representing the reference distribution. Intuitively, the diffusion equilibrates high- and low-density regions over time, thus easing the estimation problem.

The ideas behind diffusion models have been independently discovered Sohl-Dickstein et al. in [216] in 2015. Ho, Jain, and Abbeel revisited this work in 2020 and demonstrated that such models can achieve state-of-the-art results in generation [118]. The approaches of Song and Ermon, and Sohl-Dickstein et al., differ mainly in the construction of the diffusion process and were later unified by Song, Sohl-Dickstein et al. in [218].

To formalize the setup, let X be the random variable of the reference distribution with density p_X. The general form of a diffusion process is[4]

$$\begin{aligned} Y_0 &= X, \\ \mathrm{d}Y_t &= f(Y_t, t)\,\mathrm{d}B_t + g(Y_t, t)\,\mathrm{d}t, \end{aligned} \tag{6.1}$$

where f is the matrix of diffusion coefficients, B is the vector of Brownian motion, and g is the vector of drift coefficients; these concepts are treated in more detail

4: Compared to the introduction of diffusion processes in definition 2.3.11, we have exchanged the roles of X and Y here to be in accordance with our original publication.

in section 2.3.2.

In this chapter, we restrict ourselves to the choice $f \equiv \sqrt{2}$ and $g \equiv 0$, yielding the process

$$\begin{aligned} Y_0 &= X, \\ \mathrm{d}Y_t &= \sqrt{2}\,\mathrm{d}B_t. \end{aligned} \tag{6.2}$$

We choose this due to the nice interpretation in terms of the heat diffusion on densities and its connection to empirical Bayes theory: In terms of densities, let $p_Y(\cdot, t)$ denote the density of Y_t. By the construction of the diffusion process, it fulfills the heat diffusion PDE $(\partial_t - \Delta_1) p_Y(\cdot, t) = 0$ with initial condition $p_Y(\cdot, 0) = p_X$. Empirical Bayes theory [197] provides a machinery for reversing the diffusion PDE: Given an instance of Y_t, the Bayesian least-squares estimate of X can be expressed solely using $p_Y(\cdot, t)$. Importantly, this holds for all positive t, as long as p_Y is properly constructed.

In practice, we aim to have a parametrized model of p_Y, say p_θ where θ is a parameter vector, such that $p_Y(x, t) \approx p_\theta(x, t)$ for all x and all $t \in [0, \infty)$. Recent work [217, 218] has focused on functions $p_\theta(\cdot, t)$ tailored towards good generative performance: Instead of an analytic expression for $p_\theta(\cdot, t)$ at any time $t > 0$, a time-conditioned network is used to learn behaviour akin to having undergone the diffusion PDE. Further, instead of ensuring $\int p_Y(\cdot, t) = 1$ for all $t \in [0, \infty)$, the *score* $-\nabla_1 \log p_Y(\cdot, t)$ is often estimated directly with some UNet-type NN. However, the architecture of the UNet is typically not constrained to by the gradient of a scalar function and lacks symmetry properties.

In contrast, we pursue a more principles approach. With a focus on inverse problems in imaging we revisit classical translation invariant MRF modeling and combine them with ideas from diffusion models. Our fundamental aim is to express the action of the diffusion PDE on the high-dimensional density function by adapting the one-dimensional experts. To this end, we utilize Gaussian mixture experts due to the closure properties of Gaussians under multiplication and convolution, see section 6.2.2. We derive conditions under which $p_Y(\cdot, t)$ can be expressed analytically from $p_Y(\cdot, 0)$. We call our model PoGMDM to reflect the building blocks: products of Gaussian mixture experts and diffusion.

6.1.1 Contributions

We introduce a new parametrization of classical ridge-type regularizers where the action of the diffusion PDE on the high-dimensional density is expressed by adapting the one-dimensional experts. Specifically, we analyze three models: a complete model on a filter basis, a complete model on a wavelet basis, and an overcomplete model on a convolutional basis. For each model, we derive conditions on the basis under which it suffices to adapt the one-dimensional experts to respect the diffusion equation. Additionally, we provide algorithms to learn a suitable basis and present numerical denoising results, demonstrating our models can be used for robust noise level estimation and blind heteroscedastic denoising.

6.2 Background

In this section, we highlight the importance of diffusion in density estimation and sampling in high dimensional spaces. Then, we explore the relationship between the action of the diffusion PDE on density function, empirical Bayes, and denoising score matching.

6.2.1 Generative modeling and diffusion

A major challenges in learning high dimensional densities lies in the curse of dimensionality: the number of required samples to maintain a constant (average) number of samples per unit-hypervolume in a d-dimensional space grows exponentially in d. Real-world data often concentrates in lower dimensional manifolds within high dimensional spaces. Consequently, empirical datasets tend to concentrate on this manifold and leave "most" of the high-dimensional space empty. This is a challenge for generative models, since modeling these low-density regions becomes increasingly difficult.

To address this, Gaussian noise can be added to the empirical dataset. In terms of density, the target density becomes the empirical distribution convolved with a Gaussian.[5] However, selecting the variance of the Gaussian noise is not straightforward: A small variance keeps the target density close to the empirical distribution has little effect on low-density regions. Conversely, a large variance facilitates the estimation of low-density regions but details of the empirical distribution are lost.

5: We did this in chapter 5 to stabilize the training.

In the seminal work on diffusion models [217], Song and Ermon propose perturbing the empirical distribution with a sequence of Gaussian noise with increasing variance. They used a single network conditioned on the noise variance to learn the target densities via denoising score matching [238]. To generate samples, they employ the unadjusted Langevin algorithm, progressively annealing the noise variance conditioning towards zero.

Generalizing the sequence of increasing variances to a monotonically increasing function in [218] connects this approach to modeling a diffusion process (definition 2.3.11). In this chapter, we consider the diffusion process given by eq. (6.2). In terms of densities, this diffusion process is equivalent to the head diffusion

$$(\partial_t - \Delta_1) p_Y(\cdot, t) = 0 \text{ with initial condition } p_Y(\cdot, 0) = p_X. \tag{6.3}$$

Here, ∂_t denotes the standard partial derivative with respect to time and Δ_1 is the Laplace operator applied to the first argument. The next section details the evolution of p_X under this PDE and relations to empirical Bayes.

6.2.2 Diffusion, empirical Bayes, and denoising score matching

We adopt the interpretation that the evolution in (6.3) defines the density of a random variable Y_t. It is well known that Green's function of (6.3) is a Gaussian (see e.g. [58]) with zero mean and covariance $2tI$. For $t > 0$, this can be expressed

as

$$p_Y(\cdot, t) = G_{0,2tI} * p_X, \tag{6.4}$$

where

$$G_{\mu,\Sigma}(x) = \det(2\pi\Sigma)^{-1/2} \exp\Big(-\|x-\mu\|^2_{\Sigma^{-1}}/2\Big). \tag{6.5}$$

Thus, the diffusion PDE constructs a (linear) *scale space in the space of probability densities* and we refer to Y_t (or p_{Y_t}) as the *smoothed* random variable (or density). In terms of the random variables, we can write $Y_t = X + \sqrt{2t}N$ where N is a random variable with normal distribution $\mathcal{N}_{0,I}$.

Next, we use empirical Bayes methods to estimate X from an observed Y_t. The goal of empirical Bayes is to derive Bayes estimators of a random variable from corrupted observations [197]. For our setup, the corruption model is

$$y_t = x + \sqrt{2t}\eta, \tag{6.6}$$

where $x \sim p_X$ is a sample from the reference distribution, and $\eta \sim \mathcal{N}_{0,I}$ is Gaussian noise. Given the corrupted observation y_t we aim to estimate x through the Bayesian MMSE estimate

$$y_t \mapsto \operatorname*{arg\,min}_{x\in\mathscr{X}} \int_{\mathscr{X}} \|x - x'\|^2 p_{X|Y_t}(x', y_t)\,\mathrm{d}x' \tag{6.7}$$

where the right hand side is the posterior mean

$$\int_{\mathscr{X}} x p_{X|Y_t}(x, y_t)\,\mathrm{d}x, \tag{6.8}$$

see e.g. [127, page 172]. Classical Bayes estimators use Bayes theorem to write $p_{X|Y_t} = \frac{p_{Y_t|X}p_X}{p_{Y_t}}$ and choose an appropriate prior p_X. However, a result from empirical Bayes estimation reveals that a map $y_t \mapsto \int x p_{X|Y_t}(x \mid y_t)\,\mathrm{d}x$ can be constructed *solely* from p_{Y_t}. This result is known as the Miyasawa estimate [167] or Tweedie's formula [80, 195], which we derive here for completeness.

From the corruption model (6.6), we can write

$$p_{Y_t|X}(y \mid x) = \det\left(2\pi\sigma^2 I\right)^{-\frac{1}{2}} \exp\left(-\frac{\|y-x\|^2}{2\sigma^2}\right), \tag{6.9}$$

where we use the relation $\sigma^2 = 2t$, and we can write

$$\begin{aligned} p_{Y_t}(y) &= \int p_{X,Y_t}(x, y)\,\mathrm{d}x \\ &= \int p_{Y_t|X}(y \mid x) p_X(x)\,\mathrm{d}x \\ &= \int \det\left(2\pi\sigma^2 I\right)^{-\frac{1}{2}} \exp\left(-\frac{\|y-x\|^2}{2\sigma^2}\right) p_X(x)\,\mathrm{d}x. \end{aligned} \tag{6.10}$$

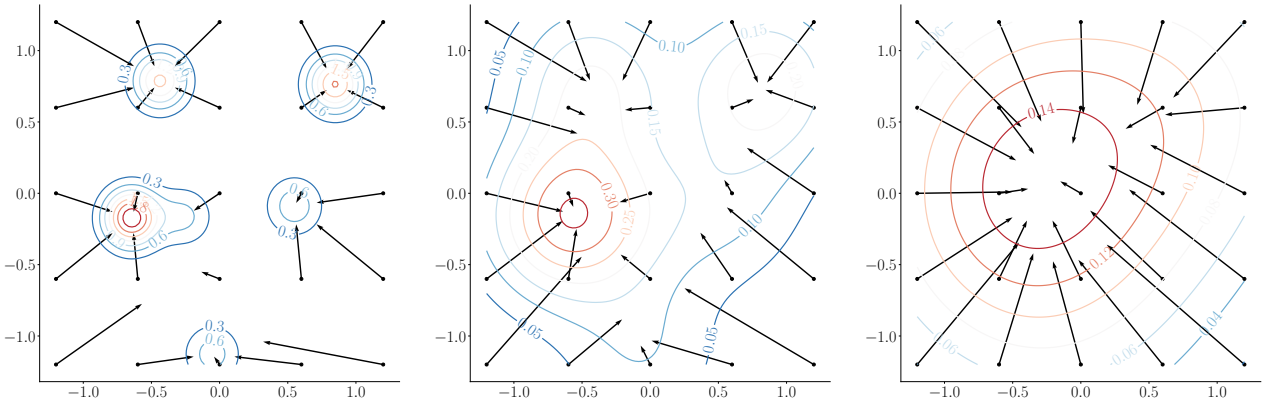

Figure 6.1: Diffusion of an empirical distribution consisting of weighted Dirac measures at diffusion times $t = 0.008, 0.078, 0.3$. The arrows show the empirical Bayes estimate $y \mapsto y + 2t\nabla \log p_{Y_t}(y)$. As t approaches infinity, p_{Y_t} becomes log-concave and $-\log p_{Y_t}$ approaches a quadratic function.

Taking the gradient and multiplying by σ^2 yields

$$\begin{aligned}
&\sigma^2 \nabla p_{Y_t}(y) \\
&= \int (x - y) \det(2\pi\sigma^2 I)^{-\frac{1}{2}} \exp\left(-\frac{\|y - x\|^2}{2\sigma^2}\right) p_X(x)\, \mathrm{d}x \\
&= \int (x - y) p_{X,Y_t}(x, y)\, \mathrm{d}x \\
&= \int x p_{X,Y_t}(x, y)\, \mathrm{d}x - y p_{Y_t}(y)
\end{aligned} \tag{6.11}$$

and after dividing by p_{Y_t} it follows that

$$y + \sigma^2 \frac{\nabla p_{Y_t}(y)}{p_{Y_t}(y)} = \int x p_{X|Y_t}(x \mid y)\, \mathrm{d}x, \tag{6.12}$$

where we used that $p_{X|Y_t} = \frac{p_{X,Y_t}}{p_{Y_t}}$. Finally, since $\frac{\nabla p_{Y_t}}{p_{Y_t}} = \nabla \log p_{Y_t}$,

$$y + \sigma^2 \nabla \log p_{Y_t}(y) = \int x p_{X|Y_t}(x \mid y)\, \mathrm{d}x. \tag{6.13}$$

For more genera corruptions, see the work of Raphan and Simoncelli [195]. They refer to this type of estimator as nonparametric empirical Bayesian least squares (NEBLS).

We illustrate the idea of empirical MMSE estimation on a toy example in fig. 6.1, where the data distribution consists of Dirac measures: $p_X = \sum_{i=1}^{6} w_i \delta_{x_i}$.[6] The figure illustrates that that p_{Y_t} approaches a simple form as t approaches infinity. Indeed, it has been shown [135] that p_{Y_t} is log-concave for large enough t, and $-\log p_{Y_t}$ approaches a quadratic function.

Recently, (6.13) has been used for parameter estimation [218, 238]: Let $p_\theta : \mathscr{X} \times [0, \infty) \to \mathbb{R}_+$ denote a parametrized model for which we wish that $p_\theta(\cdot, t) \approx p_{Y_t}$,

6: The Dirac measures are centered at

$$\begin{aligned}
x_1 &= (0.588, 0.966), \\
x_2 &= (0.289, 0.112) \\
x_3 &= (-0.313, -0.924), \\
x_4 &= (-0.696, 0.990), \\
x_5 &= (-0.906, 0.030), \\
x_6 &= (-0.516, 0.039),
\end{aligned}$$

and have weights $w = (0.23, 0.1, 0.09, 0.19, 0.29, 0.08)$.

for all $t > 0$. Then, both the left- and right-hand side of (6.13) are known in expectation, which leads to the loss function

$$\min_{\theta\in\Theta} \int_0^\infty \mathbb{E}_{(x,y_t)\sim p_{X,Y_t}}\Big[\|x - y_t - \sigma^2(t)\nabla_1 \log p_\theta(y_t, t)\|^2\Big]\,\mathrm{d}t \tag{6.14}$$

for estimating θ such that $p_\theta(\,\cdot\,, t) \approx p_{Y_t}$ for all $t > 0$. Here, p_{X,Y_t} denotes the joint distribution of the clean and smoothed random variables. Efficient sampling from this distribution is possible via ancestral sampling: A pair $(x, y_t) \sim p_{X,Y_t}$ can be constructed by sampling $x \sim p_X$ and then computing $y_t = x + \sqrt{2t}\eta$, where $\eta \sim \mathcal{N}_{0,I}$. Θ encodes constraints on the learnable parameters; In our case, Θ encodes constraints on the learnable parameters θ that are required to be able to solve the diffusion PDE.

This learning problem is known as denoising score matching in the literature [218, 238]. It can be justified more formally as minimizing the Fisher divergence (definition 2.2.20) of p_{Y_t} from $p_\theta(\,\cdot\,, t)$ for all $t > 0$:

$$\begin{aligned}&\min_{\theta\in\Theta} \int_0^\infty (p_{Y_t} \,\|\, p_\theta(\,\cdot\,, t))_{\mathrm{F}}\,\mathrm{d}t = \\ &\min_{\theta\in\Theta} \int_0^\infty \mathbb{E}_{y_t\sim p_{Y_t}}\Big[\|\nabla \log p_{Y_t}(y_t) - \nabla_1 \log p_\theta(y_t, t)\|^2\Big]\,\mathrm{d}t\end{aligned} \tag{6.15}$$

where $p_{Y_t} = G_{0,2tI} * p_X$. This is intractable as it is written since computing $\nabla \log p_{Y_t} = \nabla \log(G_{0,2tI} * p_X)$ is complex.[7] The key insight of the equivalence between the intractable formulation in eq. (6.15) and the tractable "ancestral" formulation in eq. (6.14) is due to Vincent [238]; the proof is provided in their paper.

7: It has the infamous "log-integral-exp" structure. When p_X is an empirical distribution, this is the problem of computing the gradient of the kernel density estimate with a Gaussian kernel.

6.3 Methods

As mentioned in the introduction, we revisit the classical ridge-type structure of regularizers. We endow them with a rigorous statistical interpretation and fit the potential functions by minimizing the Fisher divergence (definition 2.2.20) from the reference density to the Gibbs density. A suitable parametrization of the experts gives access to an MMSE optimal denoiser for Gaussian noise with arbitrary variance. In the next section, we demonstrate our approach by learning a complete model on the space of image patches.

6.3.1 A complete model on the space of image patches

We introduce the general concept of PoGMDMs through a complete model on the space of image patches. We approximate the distribution of image patches of size $b \times b$ by a product of $o = b \times b \in \mathbb{N}$ one-dimensional experts. In detail, the density on $\mathbb{R}^{b\times b}$ is of the form

$$p_\theta^{\mathrm{filt}}(x, t) = Z\Big(\{f_k\}_{k=1}^o, \sigma_0, t\Big)^{-1} \prod_{k=1}^{o} \psi_k\Big(\langle f_k, x\rangle, w_j, t\Big) \tag{6.16}$$

where $\psi_1, \psi_2, \dots, \psi_o$ are one-dimensional experts on the responses of linear filters $f_1, f_2, \dots, f_o \in \mathbb{R}^{b\times b}$. $Z(\{f_k\}_{k=1}^o, \sigma_0, t)$ the partition function ensuring that p_θ^{filt} is properly normalized.

The empirical Bayes construction discussed in section 6.2.2 requires finding $p_\theta^{\text{filt}}(\cdot, t) = G_{0,2tI} * p_\theta^{\text{filt}}(\cdot, 0)$. The closure properties of Gaussians under convolution motivates the choice of *Gaussian mixture experts*. In detail, the backbone of our models are one-dimensional Gaussian mixture experts $\psi_1, \psi_2, \dots, \psi_o \colon \mathbb{R} \times \Delta^p \times [0, \infty) \to \mathbb{R}_+$ with p components of the form

$$\psi_k(x, w_k, t) = \sum_{l=1}^{p} w_{k,l} G_{\mu_l, \sigma_k^2(t)}(x). \tag{6.17}$$

For the experts to be normalized, the weights of each expert $w_k = (w_{k,1}, w_{k,2}, \dots, w_{k,p})$ must be lie within the unit simplex Δ^p (definition 2.4.6). For simplicity, we assume that all experts have the same number of components and share the same means $\mu_1, \mu_2, \dots, \mu_p \in \mathbb{R}$, which are fixed a priori, see the details in section 6.4.1. In addition, within each expert, all components share the same variance. This parametrization is versatile, as illustrated in Figure 6.2, and can approximate classical sparsity-inducing potentials like the absolute, and value and the potentials of popular leptokurtic densities, such as the density of the Student-t distribution. It also allows for multi-well potentials which are advantageous in Markov random field modeling [111, 256] as discussed in the introduction.

Our main contribution is demonstrating that under certain assumptions, adapting the variances of the one-dimensional experts suffices to implement the convolution of p_θ^{filt} with a Gaussian. Specifically, we show that the variance of the k-th expert, σ_k^2, can be modeled as

$$\sigma_k^2(t) = \sigma_0^2 + c_k 2t, \tag{6.18}$$

where $\sigma_0 > 0$ is chosen a-priori to support the discretization of the means and $c_k > 0$ is derived from the filters f_k.

We leverage two well-known properties of Gaussians to determine how to adapt the variances of the one-dimensional experts with diffusion time t. First, the product of GMMs is a GMM up to normalization, see e.g. [211], allowing us to work with expressive models that enable efficient *evaluations* due to factorization. Second, there exists an analytical solution to the diffusion PDE if p_X is a GMM. Green's function associated with the linear isotropic diffusion PDE (6.3) is a Gaussian with isotropic covariance $2tI$. Using previous notation, if X is a random variable with normal distribution $\mathcal{N}_{\mu_X, \Sigma_X}$, then Y_t follows the distribution $\mathcal{N}_{\mu_X, \Sigma_X + 2tI}$.[8] In particular, the mean remains unchanged and it suffices to adapt the covariance matrix with the diffusion time.

8: Due to the linearity of the convolution, it suffices to consider a single Gaussian; the GMM follows by a linear combination.

Due to closure properties of Gaussians under multiplication, p_θ^{filt} is a Gaussian mixture model on $\mathbb{R}^{b\times b}$. In order to find an expression for the variances of the one-dimensional Gaussian mixture experts under diffusion, we first establish the exact form of $p_\theta^{\text{filt}}(\cdot, t)$ as a GMM on $\mathbb{R}^{b\times b}$. We denote with $\hat{l} : \{1, \dots, o\} \to \{1, \dots, p\}$ a fixed but arbitrary selection from the index set $\{1, \dots, p\}$.

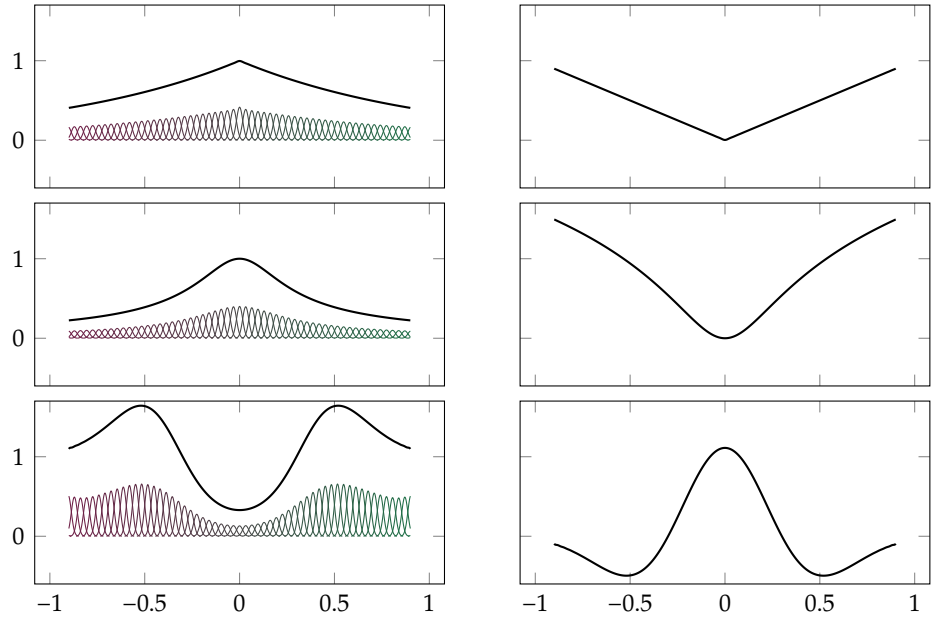

Figure 6.2: The GMM experts can model popular experts used in imaging: The Laplace expert (top), the Student-t expert (middle), and more general multimodal experts (bottom) are shown in the first column, and the corresponding absolute value potential (top), the leptokurtic Student-t potential (middle), and a Mexican-hat-like potential in the second column. Individual components of the GMM are shown on the left, with the discretization as described in section 6.4.1 (125 components, equidistant means over $[-1, 1]$, $\sigma_0^2 = 2/(125-1)$). To avoid clutter we only plot every second component.

Theorem 6.3.1. *$p_\theta^{\text{filt}}(\cdot, t)$ is a homoscedastic GMM on $\mathbb{R}^{b\times b}$ with precision*

$$(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} (f_k \otimes f_k). \tag{6.19}$$

It has p^o components whose means is identified by the choice of the index map $\hat{l}$

$$\mu_{\hat{l}} = \Sigma(t) \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} f_k \mu_{\hat{l}(k)}. \tag{6.20}$$

Proof. By inserting the definition of the one-dimensional GMM experts,

$$\prod_{k=1}^{o} \psi_k\Big(\langle f_k, x\rangle, w_j, t\Big) = \prod_{k=1}^{o} \sum_{l=1}^{p} \frac{w_{kl}}{\sqrt{2\pi\sigma_k^2(t)}} \exp\left(-\frac{1}{2\sigma_k^2(t)}\Big(\langle f_k, x\rangle - \mu_l\Big)^2\right). \tag{6.21}$$

We develop the product over the components and inspect a general term of the resulting sum, which corresponds to one component of the resulting GMM. The

general term of the resulting sum can be written as

$$\left(\prod_{k=1}^{o} \frac{w_{k\hat{l}(k)}}{\sqrt{2\pi\sigma_k^2(t)}}\right) \exp\left(-\sum_{k=1}^{o} \frac{1}{2\sigma_k^2(t)}\left(\langle f_k, x\rangle - \mu_{\hat{l}(k)}\right)^2\right). \tag{6.22}$$

To find the covariance, we match the gradient of the familiar quadratic form: $\nabla_x\left(\sum_{k=1}^{o} \frac{1}{2\sigma_k^2(t)}\left(\langle f_k, x\rangle - \mu_{\hat{l}(k)}\right)^2\right) = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)}\left((f_k \otimes f_k)x - f_k\mu_{\hat{l}(k)}\right)$. From the first term, we immediately identify $(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)}(f_k \otimes f_k)$, and we find $\mu_{\hat{l}}$ by left-multiplying $(\Sigma(t))$ onto $\sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} f_k\mu_{\hat{l}(k)}$. □

As discussed in section 6.2.2, the convolution of p_θ^{filt} with a Gaussian with covariance $2tI$ can be implemented by letting $\Sigma(t) = \Sigma(0) + 2tI$. Now, the challenge lies in finding a way to express this map by adapting the variances of the one-dimensional GMM experts given the structure in eq. (6.16). We show that this is not possible in the general case with an example on $\mathbb{R}^2$: Let $f_1 = (1,0)$, $f_2 = (1,1)/\sqrt{2}$, and $\sigma_0 = 1$.[9] Then, we have that

9: For the sake of this example, the means are irrelevant.

$$f_1 \otimes f_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \text{ and } f_2 \otimes f_2 = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}, \tag{6.23}$$

and at time $t = 0$,

$$\Sigma^{-1} = \begin{pmatrix} 1.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix} \text{ and } \Sigma = \begin{pmatrix} 1 & -1 \\ -1 & 3 \end{pmatrix}. \tag{6.24}$$

At time $t = 0.5$,

$$\Sigma + I = \begin{pmatrix} 2 & -1 \\ -1 & 4 \end{pmatrix} \text{ and } (\Sigma + I)^{-1} = \frac{1}{7}\begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix}. \tag{6.25}$$

It is easy to see that $(\Sigma + I)^{-1}$ is not in the span of $f_1 \otimes f_1$ and $f_2 \otimes f_2$. I.e., it can not be represented by $\frac{1}{\sigma_1^2(t)}(f_1 \otimes f_1) + \frac{1}{\sigma_2^2(t)}(f_2 \otimes f_2)$, irrespective of the adaptation of the variances σ_1^2 and σ_2^2 of the one-dimensional GMMs.

However, the next theorem establishes a tractable analytical expression for the diffusion process under the assumption of pair-wise orthogonal filters, i.e. that

$$\langle f_j, f_i\rangle = \begin{cases} 0 & \text{if } i \neq j, \\ \|f_j\|^2 & \text{else,} \end{cases} \text{ for all } i, j \in \{1, \dots, o\}. \tag{6.26}$$

Theorem 6.3.2 (Diffusion of a complete model)**.** *Under assumption* (6.26)*, p_θ^{filt} satisfies the diffusion PDE $(\partial_t - \Delta_1)p_\theta^{\text{filt}}(\cdot, t) = 0$ if the variances of the one-dimensional GMM experts are adapted as $\sigma_k^2(t) = \sigma_0^2 + \|f_k\|^2 2t$.*

Proof. We exploit that the orthogonality assumption on the filters, (6.26), immediately gives the Eigendecomposition of the precision: At $t = 0$, the Eigendecomposition of the precision is

$$(\Sigma(0))^{-1} = \sum_{k=1}^{o} \frac{\|f_k\|^2}{\sigma_0^2}\Big(\frac{f_k}{\|f_k\|} \otimes \frac{f_k}{\|f_k\|}\Big). \tag{6.27}$$

The covariance can be computed by taking the reciprocal of the Eigenvalues:

$$\Sigma(0) = \sum_{k=1}^{o} \frac{\sigma_0^2}{\|f_k\|^2}\Big(\frac{f_k}{\|f_k\|} \otimes \frac{f_k}{\|f_k\|}\Big). \tag{6.28}$$

Adding a multiple of the identity to the covariance can be expressed on the level of Eigenvalues as

$$\Sigma(0) + 2tI = \sum_{k=1}^{o} \frac{\sigma_0^2 + 2t\|f_k\|^2}{\|f_k\|^2}\Big(\frac{f_k}{\|f_k\|} \otimes \frac{f_k}{\|f_k\|}\Big), \tag{6.29}$$

and inverting is again just taking the reciprocal of the Eigenvalues:

$$\begin{aligned}(\Sigma(0) + 2tI)^{-1} &= \sum_{k=1}^{o} \frac{\|f_k\|^2}{\sigma_0^2 + 2t\|f_k\|^2}\Big(\frac{f_k}{\|f_k\|} \otimes \frac{f_k}{\|f_k\|}\Big) \\ &= \sum_{k=1}^{o} \frac{1}{\sigma_0^2 + 2t\|f_k\|^2}(f_k \otimes f_k).\end{aligned} \tag{6.30}$$

This is exactly of the form of eq. (6.19), with $\sigma_k^2(t) = \sigma_0^2 + \|f_k\|^2 2t$. Thus, p_θ^{filt} satisfies the diffusion PDE if $\sigma_k^2(t) = \sigma_0^2 + \|f_k\|^2 2t$. □

Denoting with X the random variable whose density we aim to model at time 0, we argued in section 4.4.1 that in general, the one-dimensional expert acting on $\langle f_k, \cdot\rangle$ does not model the distribution of the random variable $\langle f_k, X\rangle$. However, in the undercomplete case with orthogonal filters, this is the case:

Corollary 1. *Let X be the random variable whose density we model with eq.* (6.16)*, and let Y_t be the random variable given by eq.* (6.2)*. Then, with assumption* (6.26) *the experts $\psi_k(\cdot, w_k, t)$ in* (6.16) *model the marginal distribution of the random variable $U_{k,t} = \langle f_k, Y_t\rangle$ for any $t \geq 0$.*

Proof. Consider the component of the resulting homoscedastic GMM identified by the choice of the index map $\hat{l}$: $Y_{\hat{l},t} \sim \mathcal{N}_{\mu_{\hat{l}}, \Sigma+2tI}$. The distribution of $\hat{U}_{k,t} = \langle f_k, Y_{\hat{l},t}\rangle$ is $\hat{U}_{k,t} \sim \mathcal{N}_{\langle f_k, \mu_{\hat{l}}\rangle, \langle f_k, (\Sigma+2tI) f_k\rangle}$ (see e.g. [102, theorem 3.1]). Under our orthogonality assumptions, this simplifies to $\mathcal{N}_{\mu_{\hat{l}(k)}, \sigma_0^2 + 2t\|f_k\|^2}$. The claim follows from the linear combination of the different components. □

In addition, we can compute the normalization constant of $p_\theta^{\text{filt}}(\cdot, t)$ for any $t \geq 0$ as

$$Z\big(\{f_k\}_{k=1}^o, \sigma_0, t\big) = \sqrt{\prod_{k=1}^{o} 2\pi \frac{\sigma_0^2 + 2t\|f_k\|^2}{\|f_k\|^2}}, \tag{6.31}$$

which is just the square root of the product of the eigenvalues in eq. (6.30) multiplied by 2π.

We presented the analysis above assuming that the number of one-dimensional experts is exactly equal to the dimensionality of the space; a complete model. Coupled with the orthogonality constraint, this effectively ensures that the distribution is *proper*. However, in practice fewer experts can be utilized, for instance to enforce invariance with respect to radiometric shifts.[10] In this case, $p_\theta^{\text{filt}}(\cdot, t)$ does not admit a density with respect to the Lebesgue measure for any $t \geq 0$. Nevertheless, the theoretical analysis can be carried out by restricting the Lebesgue measure to the span of the filters (invoking the disintegration theorem) and replacing inverses by pseudo-inverses and determinants by pseudo-determinants [194].

10: In this case, there are $b^2 - 1$ experts acting on the responses of zero-mean filters; we do this in section 6.4.2.

6.3.2 Wavelet model

The key ingredient in the previous section was the orthogonality of the filters. In other words, the filters $f_1, f_2, \dots, f_o$ form an orthogonal (not necessarily orthonormal) basis for (possibly a subspace of) $\mathbb{R}^{b\times b}$. In this section, we discuss the application of explicit diffusion models in another well-known orthogonal basis: Wavelets.

A brief discussion about the wavelet transform can be found in section 2.1.9; for the purposes of this section it suffices to note that any image $x \in \mathbb{R}^{m\times n}$ can be decomposed as

$$x = \text{proj}_{V_m} x + \sum_{j\leq m} \text{proj}_{W_j} x \tag{6.32}$$

where $\text{proj}_{W_1}, \text{proj}_{W_2}, \dots, \text{proj}_{W_m} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ are the projections onto the *orthogonal detail spaces* $W_1, W_2, \dots, W_m$. We utilize the shorthand notation

$$\mathscr{W}_j = \text{proj}_{W_j}, \tag{6.33}$$

and recall the properties of an orthogonal projection,

$$\begin{aligned} &\text{(self-adjoint)} && (\mathscr{W}_j)^* &&= \mathscr{W}_j, \\ &\text{(idempotency)} && \mathscr{W}_j \circ \mathscr{W}_j &&= \mathscr{W}_j, \text{ and} \\ &\text{(identity on subspace)} && \mathscr{W}_j|_{W_j} &&= I_{W_j} \end{aligned} \tag{6.34}$$

where $\mathscr{W}_j|_{W_j}$ denotes the restriction of $\mathscr{W}_j$ to W_j.

As in the previous section, we model the wavelet-responses with Gaussian mixture experts. For simplicity, we discard the partition function and build an undercomplete model on $\oplus_{j=1}^{o} W_j$:[11]

$$p_\theta^{\text{wave}}(x, t) \propto \prod_{i,j=1}^{m,n} \prod_{k=1}^{o} \psi_k\Big((\mathscr{W}_k x)_{i,j}, w_k, t\Big). \tag{6.35}$$

11: Again, this could be more formally dealt with via the disintegration theorem.

Following the approach utilized in theorem 6.3.2, we first describe the exact form of (6.35) as a GMM on $\mathbb{R}^n$.

Theorem 6.3.3. *$p_\theta^{\text{wave}}(\cdot, t)$ is a homoscedastic GMM on $\mathbb{R}^{m\times n}$ with precision*

$$(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} \mathscr{W}_k. \tag{6.36}$$

Proof. As in theorem 6.3.2, this follows immediately from the expansion of eq. (6.35):

$$p_\theta^{\text{wave}}(x,t) \propto \prod_{i,j=1}^{m,n} \prod_{k=1}^{o} \sum_{l=1}^{p} \frac{w_{kl}}{\sqrt{2\pi\sigma_k^2(t)}} \exp\left(-\frac{((\mathscr{W}_k x)_{i,j} - \mu_l)^2}{2\sigma_k^2(t)}\right). \tag{6.37}$$

By expanding the product over the features as well as the experts we find that the general component has the form[12]

12: Expressions for the weight and the mean $\mu_{\hat{l}(i,j,k)} \in \mathbb{R}^{m\times n}$ of this component can be derived easily but are omitted for simplicity.

$$\exp\left(-\sum_{k=1}^{o} \frac{1}{2\sigma_k^2(t)} \|\mathscr{W}_k x - \mu_{\hat{l}(i,j,k)}\|^2\right). \tag{6.38}$$

The precision of this component is

$$(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} \left(\mathscr{W}_k\right)^* \mathscr{W}_k, \tag{6.39}$$

which simplifies to

$$(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} \mathscr{W}_k \tag{6.40}$$

since projections are self-adjoint and idempotent, see (6.34). □

It remains to show how the $\Sigma(t) = \Sigma(0) + 2tI$ can be implemented by adapting the variances of the one-dimensional Gaussian mixture experts. However, this becomes trivial due to the orthogonality of the detail spaces.

Theorem 6.3.4 (Wavelet diffusion). *$p_\theta^{\text{wave}}(\cdot, t)$ satisfies the diffusion PDE $(\partial_t - \Delta_1)p_\theta^{\text{wave}}(\cdot, t) = 0$ if the variances of the one-dimensional Gaussian mixture experts are adapted as $\sigma_k^2(t) = \sigma_0^2 + 2t$.*

Proof. Due to the orthogonality of the detail spaces, the precision in eq. (6.40) on $\oplus_{k=1}^{o} W_k$ is the identity (see also the decomposition eq. (6.32)). Thus, it suffices to adapt the variance of the one-dimensional GMMs ψ_k with $\sigma_k^2(t) = \sigma_0^2 + 2t$. □

We can endow the different sub-bands of the wavelet transformation with scalars to weight their influence as follows: Replacing $\mathscr{W}_k$ with $\lambda_k \mathscr{W}_k$ in (6.39) (the derivation does not change up to this point), we find that the diffusion PDE is satisfied when $\sigma_k^2(t) = \sigma_0^2 + 2t\lambda_k^2$. Thus, the scaling parameters λ_k are analogous to the filter-norms in theorem 6.3.2.

In the above, we played a slight of hand by using the one-dimensional theory from the preliminaries in a two-dimensional context. However, the derivation

is essentially the same for the two-dimensional wavelet transform by replacing the detail index (k in the above) with a two-index to account for the direction (vertical, horizontal, diagonal). The details are discussed in slightly more detail in the original publication [250] and rigorously in [30, chapter 4.4].

6.3.3 Overcomplete model

The undercomplete model on filter-responses discussed in section 6.3.1 can not account for the correlation of overlapping patches when used for whole image restoration [206, 260]. Similarly, the model based on wavelet-responses is limited in expressiveness since it only models the distribution of a scalar random variable per sub-band. In what follows, we describe a convolutional PoGMDM that avoids the extraction and combination of patches in patch-based image priors and can account for the local nature of low-level image features.

In analogy to the product-of-experts-type model acting on filter-responses, here we extend the FoE model [206] to our considered diffusion setting by accounting for the diffusion time t and obtain

$$p_\theta^{\text{conv}}(x, t) \propto \prod_{i,j=1}^{m,n} \prod_{k=1}^{o} \psi_k\Big((K_k x)_{i,j}, w_k, t\Big). \tag{6.41}$$

Here, the experts $\psi_1, \psi_2, \dots, \psi_o$ act the responses to convolution operators $K_1, K_2, \dots, K_o \colon \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$. In detail, the convolution operators implement circular boundary conditions and hence are circulant Toeplitz operators (see [177, section 5.5.2]) described by

$$(K_k x)_{i,j} = \sum_{p,q=1}^{b,b} (f_k)_{p,q} \cdot x_{\imath(p),\jmath(q)}, \tag{6.42}$$

where[13]

$$\imath(p) = 1 + (i - p + \lfloor b/2 \rfloor) \bmod m \tag{6.43}$$

and

$$\jmath(q) = 1 + (j - q + \lfloor b/2 \rfloor) \bmod n. \tag{6.44}$$

Further, $w_k \in \Delta^p$ are the weights of the components of the k-th expert ψ_k (see (6.17)). In the above, $f_1, f_2, \dots, f_o \in \mathbb{R}^{b\times b}$ are the $b \times b$ filters of the respective convolution operators. As with the models based on filter- and wavelet-responses, under some assumptions on the convolution operators it suffices to adapt the variances σ_k^2 by the diffusion time, which we show by the following analysis.

We start by outlining the structure of (6.41) as a GMM on $\mathbb{R}^{m\times n}$.

Theorem 6.3.5. *$p_\theta^{\text{conv}}(\,\cdot\,, t)$ is a homoscedastic GMM on $\mathbb{R}^{m\times n}$ with precision*

$$(\Sigma(t))^{-1} = \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} K_k^* K_k. \tag{6.45}$$

Proof. The proof is essentially given in the proof of theorem 6.3.3: The form of the covariance matrix above is eq. (6.39), which holds for general linear operators. □

13: Here, mod is understood as the least positive residue, e.g. $-1 \bmod 5 = 4$.

As previously, the challenge now lies in finding a way to express the map $\Sigma(t) = \Sigma(0) + 2tI$ by adapting the variances of the one-dimensional Gaussian mixture experts. The next theorem establishes an analytic expression for the diffusion process under the following assumption: The frequency spectra of the filters $f_1, f_2, \dots, f_o$ are non-overlapping and constant on the support. More formally, let the size of the filters be $b \times b$ and let the image size be $m \times n$. Let $P\colon \mathbb{R}^{b\times b} \to \mathbb{R}^{m\times n}$ be a padding and shifting operator[14] $P = \tilde{P}S$ where $\tilde{P}\colon \mathbb{R}^{b\times b} \to \mathbb{R}^{m\times n}$ is zero padding (putting the filter in the upper left hand corner) and $(Sx)_{i,j} = x_{i(\lfloor b/2 \rfloor), j(\lfloor b/2 \rfloor)}$. Then, denoting by Γ_k the support of FPf_k, that is $\Gamma_k = \{(i,j) \mid (FPf_k)_{i,j} \neq 0\}$, we require that

14: This padding operator is formally needed for the convolution theorem to hold in the next theorem. In practice, we will use filters that are directly defined in the frequency domain, see section 6.4.4.

$$\Gamma_k \cap \Gamma_l = \varnothing \text{ for all } k, l = 1, 2, \dots, o \text{ where } k \neq l. \tag{6.46}$$

In addition, let

$$\Big(|FPf_k|\Big)_{i,j} = \xi_k \chi_{\Gamma_k}\Big((i,j)\Big), \tag{6.47}$$

where $\xi_k > 0$ is the magnitude and χ_{Γ_k} is the characteristic function (definition 2.1.25) of Γ_k. In the language of classical signal processing, the filters $k_1, k_2, \dots, k_o$ should be *ideal*[15], see [6, fig. 2.17, fig. 2.18]; we discuss the implications of this in after the theorem.

15: up to scaling; typically ideal filters have no gain in their passband

Theorem 6.3.6 (Diffusion of an overcomplete model)**.** *Under assumptions* (6.46) *and* (6.47)*, p_θ^{conv} satisfies the diffusion PDE $(\partial_t - \Delta_1)p_\theta^{\text{conv}}(\cdot, t) = 0$ if the variances of the one-dimensional Gaussian mixture experts are adapted as $\sigma_k^2(t) = \sigma_0^2 + \xi_k^2 2t$.*

Proof. To efficiently invert the precision, we exploit a diagonalization of the circulant Toeplitz convolution operators $K_1, K_2, \dots, K_o$ via the discrete Fourier transform. The convolution operators are diagonalized as (see [177, section 5.5.4])

$$K_k = F^* \operatorname{diag}(FPf_k) F. \tag{6.48}$$

Thus, the precision in eq. (6.45) can be expressed as

$$\begin{aligned}(\Sigma(t))^{-1} &= \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} (F^* \operatorname{diag}(FPf_k)F)^* F^* \operatorname{diag}(FPf_k)F \\ &= \sum_{k=1}^{o} \frac{1}{\sigma_k^2(t)} F^* \operatorname{diag}(\overline{FPf_k}) \operatorname{diag}(FPf_k)F \\ &= F^* \operatorname{diag}\Big(\sum_{k=1}^{o} \frac{|FPf_k|^2}{\sigma_k^2(t)}\Big)F,\end{aligned} \tag{6.49}$$

where we used that $FF^* = I$ and $\bar{z}z = |z|^2$. To compute the covariance at time zero, it suffices to invert the diagonal operator:

$$\Sigma(0) = F^* \operatorname{diag}\Big(\sum_{k=1}^{o} \frac{|FPf_k|^2}{\sigma_0^2}\Big)^{-1} F. \tag{6.50}$$

Critically, at each entry of the diagonal operator only one filter is active:

$$\Big(\sum_{k=1}^{o} \frac{|FPf_k|^2}{\sigma_0^2}\Big)_{i,j} = \frac{|FPf_a|^2}{\sigma_0^2} \text{ where } a \text{ is such that } (i,j) \in \Gamma_a. \tag{6.51}$$

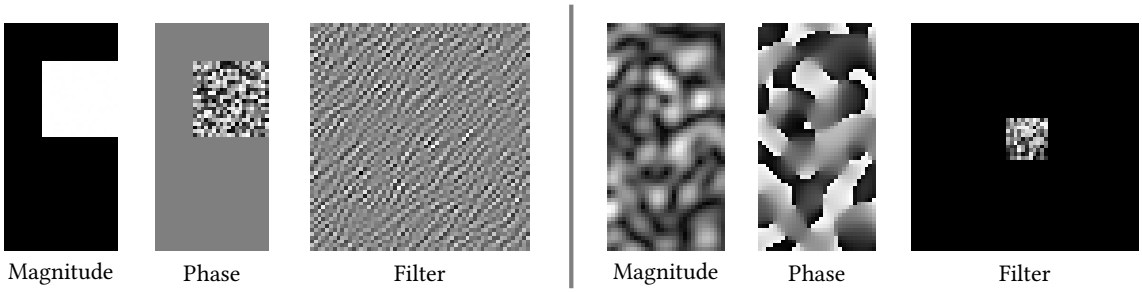

Figure 6.3: Windowing leads to infinite support in the transformed domain: On the left, a partitioning of the spectrum is prescribed, which leads to a filter with image-sized support. On the right, the finite-support filter is prescribed, which leads to a spectrum with image-sized support. Since we assume real filters, we only show half of the spectrum as it enjoys conjugate symmetry.

Thus, the covariance is

$$\Sigma(0) = F^* \operatorname{diag}\Big(\sum_{k=1}^{o} \frac{\sigma_0^2}{|FPf_k|^2}\Big)F \tag{6.52}$$

and adding a multiple of the identity amounts to

$$\Sigma(0) + 2tI = F^* \operatorname{diag}\Big(\sum_{k=1}^{o} \frac{\sigma_0^2 + 2t|FPf_k|^2}{|FPf_k|^2}\Big)F. \tag{6.53}$$

Finally, for the same reason as previously this can be inverted as

$$(\Sigma(0) + 2tI)^{-1} = F^* \operatorname{diag}\Big(\sum_{k=1}^{o} \frac{|FPf_k|^2}{\sigma_0^2 + 2t|FPf_k|^2}\Big)F. \tag{6.54}$$

This is exactly of the form eq. (6.49) with $\sigma_k(t) = \sigma_0^2 + 2t\xi_k^2$. □

The overcomplete model (6.41) is fundamentally different from the model based on filter-responses discussed in section 6.3.1. Specifically, the one-dimensional GMM experts $\psi_k(\cdot, w_k, t)$ do *not* model the distribution of the filter-responses of their corresponding filter kernels f_k. Instead, the "overcompleteness through convolution" allows the model to capture the non-trivial correlation of overlapping patches.

The next step is to select filters $k_1, k_2, \dots, k_o$ that meet our assumptions. One approach might be to construct ideal filters by partitioning the Fourier spectrum, aligning with our assumption. However, this does leads to filters with infinite spatial support, making the construction impractical: The MRF-type models (6.41) aim to share potential for responses extracted at different locations, requiring filters much smaller than the image. Conversely, any finite-length filter's spectrum has infinite support and special care is needed to ensure spectra of different filters do not overlap. This issue is related to the classical windowing problem, see [6, section 7.5] and fig. 6.3.

We balance the trade-off between satisfying theoretical assumptions on the spectra and having compactly supported time-domain filters, by using compactly supported shearlets [139, 147], specifically the non-separable version from [146]. As

an extension to the wavelet transform, the shearlet transform represents directional information in multidimensional signals via shearing and provides an optimally sparse representation of signals within a certain smoothness class [100, 140]. We use the non-separable digital shearlet transform [146], an improved discretization of the compactly supported shearlets introduced by Lim in [147]. The induced frequency tiling is schematically shown in fig. 6.4, where the frequency plane is partitioned into non-overlapping cones indexed by the scaling and shearing parameters.

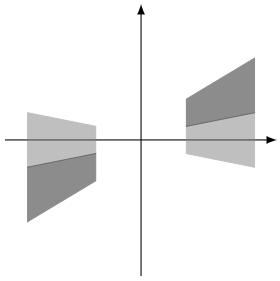

Figure 6.4: Frequency tiling of the non-separable shearlet transform [146].

6.4 Numerical results

In this section, we detail the setup for numerical optimization. Specifically, we discuss the joint learning of the one-dimensional GMM experts and their corresponding transformation (filters, wavelets, and shearlets). We then present denoising results using a simple one-step empirical Bayes scheme and algorithms derived from diffusion models. Additionally, we demonstrate our models' capability for noise level estimation and blind heteroscedastic denoising, and derive a direct sampling scheme using corollary 1.

6.4.1 Numerical optimization

For our numerical experiments, the reference random variable X reflects the 400 gray-scale images in the BSDS 500 [161] training and test set, including rotates and flipped versions, with pixel values ranging from 0 to 1. We optimize the score matching objective (6.14) using projected AdaBelief [258] for 100 000 steps. The infinite-time diffusion PDE is approximated by uniformly drawing $\sqrt{2t}$ from the interval $[0, 0.4]$.

For the denoising experiments, we use the validation images from [206] (also known as "Set68"). Due to computational constraints, we utilize only the first 15 images of the dataset according to a lexicographic ordering of the filenames. In addition, our wavelet- and shearlet-toolboxes only allow the processing of square images, so we only utilize the central region of size 320×320.

In all experiments, the one-dimensional Gaussian mixture experts have $p = 125$ components, with means equispaced over the interval $[-\eta, \eta]$:[16]

$$\mu_l = \eta\left(2\frac{l-1}{p-1} - 1\right) \text{ for all } l = 1, 2, \dots, p. \tag{6.55}$$

16: Due to the choice of the discretization of the means, it is important to have p odd to ensure that the GMMs are sufficiently peaky around zero.

We discuss the choice of $\eta > 0$ for the different models in their respective sections; for the wavelet-model it varies between the experts. To support the uniform discretization of the means, the base-variance is set to $\sigma_0^2 = \frac{2\eta}{p-1}$.

To normalize the GMM experts, after each parameter update their weight vectors are projected onto the unit simplex (definition 2.4.6). We implement the projection $\operatorname{proj}_{\Delta^p} : \mathbb{R}^p \to \mathbb{R}^p$ using the sorting-based method proposed by [112], summarized in algorithm 17. Additionally, we enforce symmetry of the one-dimensional

GMM experts around 0, making the models invariant to image inversion (i.e., x and $1 - x$ are equally likely). This is often implicit (e.g., Gaussian scale mixtures (GSMs) are symmetric around 0) or learned if not explicitly enforced (see e.g. [47, fig. 5]). We achieve this by storing only $\lceil p/2 \rceil$ weights and mirroring the tail of $\lceil p/2 \rceil - 1$ elements prior to the projection algorithm and function evaluations.

In the following sections, we detail the constraints the building blocks of the learned transformations have to fulfill and how to satisfy them in practice.

6.4.2 Learning orthogonal filters

Let

$$\begin{aligned} K \colon \mathbb{R}^{b \times b} &\to \mathbb{R}^o \\ x &\mapsto \Big(\langle f_1, x \rangle, \langle f_2, x \rangle, \dots, \langle f_o, x \rangle \Big) \end{aligned} \tag{6.56}$$

be the linear operator that computes all filter responses. Finding orthogonal filters can be formalized by finding

$$\operatorname{proj}_{\mathscr{O}}(K) = \operatorname*{arg\,min}_{M \in \mathscr{O}} \frac{1}{2} \|M - K\|_F^2 \tag{6.57}$$

where

$$\mathscr{O} = \{A \colon \mathbb{R}^{b \times b} \to \mathbb{R}^o \mid AA^* = D^2\} \tag{6.58}$$

and $D \colon \mathbb{R}^o \to \mathbb{R}^o$ is a diagonal operator. Here, with slight abuse of notation, $\|\cdot\|_F$ is the Frobenius norm of the matrix representation of its argument. Since $\operatorname{proj}_{\mathscr{O}}(K) \operatorname{proj}_{\mathscr{O}}(K)^* = D^2$ we can represent it as $\operatorname{proj}_{\mathscr{O}}(K) = DO$ with $O \colon \mathbb{R}^{b \times b} \to \mathbb{R}^o$ semi-unitary: $OO^* = I$ (on $\mathbb{R}^o$). Thus, we rewrite the objective

$$\operatorname{proj}_{\mathscr{O}}(K) = \operatorname*{arg\,min}_{\substack{O \text{ semi-unitary} \\ D \text{ diagonal}}} \mathscr{E}(O, D) \tag{6.59}$$

where

$$\mathscr{E}(O, D) \coloneqq \|DO - K\|_F^2 = \|K\|_F^2 - 2\langle K, DO \rangle_F + \|D\|_F^2, \tag{6.60}$$

with $\langle \cdot, \cdot \rangle_F$ denoting the Frobenius inner product.

We propose the following alternating minimization scheme for finding O and D. The solution for the sub-problem in O can be computed via the polar decomposition: Let UP, with $U \colon \mathbb{R}^o \to \mathbb{R}^{b \times b}$ semi-unitary and $P \colon \mathbb{R}^o \to \mathbb{R}^o$ self-adjoint and positive semi-definite, be the polar decomposition of K^*D. The solution in O is setting $O = U$. The sub-problem in D is solved by setting $D_{i,i} = \max\Big((O^*K^*)_{i,i}, 0\Big)$. The algorithm is summarized in algorithm 16.[17] We empirically observe fast convergence: three steps already yielded satisfactory results.

17: There, we have removed the linear operator O and directly used U in the computation of D.

This algorithm constitutes a generalization of the projection onto the Stiefel manifold; the set of all orthonormal linear operators [1, section 3.3.2]. The projection onto the Stiefel manifold is given by the polar decomposition, see [114, eq. (4)]. The problem is also related to the unitary Procrustes problem, see [119, section 7.4.5]. A preliminary theoretical analysis of the algorithm is presented in the supplementary material of the conference paper [249].

Algorithm 16: Algorithm for orthogonalizing a set of filters K.

Input : Linear operator $K\colon \mathbb{R}^{b\times b} \to \mathbb{R}^o$
Output: $UD = \operatorname{proj}_{\mathscr{O}}(K)$

$D^1 = I$
while *not converged* **do**
  $U^k P^k = K^* D^k$ `// Polar decomposition`
  $D^{k+1}_{i,i} = \max\Big(((U^{k+1})^* K^*)_{i,i}, 0\Big)$
  $k = k + 1$

We continue by detailing the setup of the numerical experiments. We explicitly enforce invariance with respect to radiometric shifts by utilizing only zero-mean filters. In detail, assuming $b \times b$ image patches, we use $o = b^2 - 1$ filters spanning the space $\mathscr{Z} = \{x \in \mathbb{R}^{b\times b} \mid \sum_{i,j=1}^{b} x_i = 0\}$. Thus, we have two constraints on the filters $f_1, f_2, \dots, f_o$: The filters $f_1, f_2, \dots, f_o$ must be in the set $\mathscr{Z}$, and the operator K constructed from the filters by eq. (6.56) must be in the set $\mathscr{O}$ defined by eq. (6.58), We enforce this by first projecting $f_1, f_2, \dots, f_o$ onto $\mathscr{Z}$ via

$$f \mapsto f - \sum_{i,j=1}^{b,b} f_{i,j}, \tag{6.61}$$

and subsequently projecting the corresponding K onto $\mathscr{O}$ via algorithm 16. In practice, by this procedure both constraints were always almost exactly fulfilled. To ensure the correct projection, an alternative would be to utilize Dykstra's projection algorithm [28].[18]

18: Our procedure can be interpreted as one iteration of Dykstra's projection algorithm.

We draw the initial weights of the filters independently from $\mathscr{N}_{0,b^{-2}}$. The interval over which the means of the one-dimensional GMM experts are gridded is not so critical since we do not constrain the norm of the filters. Thus, we simply choose $\eta_k = 1$ for all $k = 1, 2, \dots, o$.

Due to corollary 1, the potentials of the undercomplete model should approximate the negative-log empirical marginal response histograms

$$z \mapsto -\log \mathbb{E}_{x\sim p_{Y_t}}\Big[\chi_{\{0\}}(z - \langle f_k, x\rangle)\Big] \tag{6.62}$$

for all $t > 0$. To evaluate this we plot the learned 7×7 orthogonal filters, the learned potential functions $-\log \psi_k(\,\cdot\,, w_k, t)$ and activation functions $-\nabla \log \psi_k(\,\cdot\,, w_k, t)$ along with the negative-log empirical marginal response histograms in fig. 6.5. Indeed, the learned potential functions match the negative empirical marginal response log-histograms almost perfectly even at low-density tails.

The filters bare striking similarity to the Eigenimages of the covariance of [260, Fig. 6], who learn a GMM directly on the space of image patches (i.e. without any factorizing structure). This comes as no surprise, since the construction of the patch-model (6.16) can be interpreted as "learning the Eigendecomposition", see theorem 6.3.1 and the proof of theorem 6.3.2.

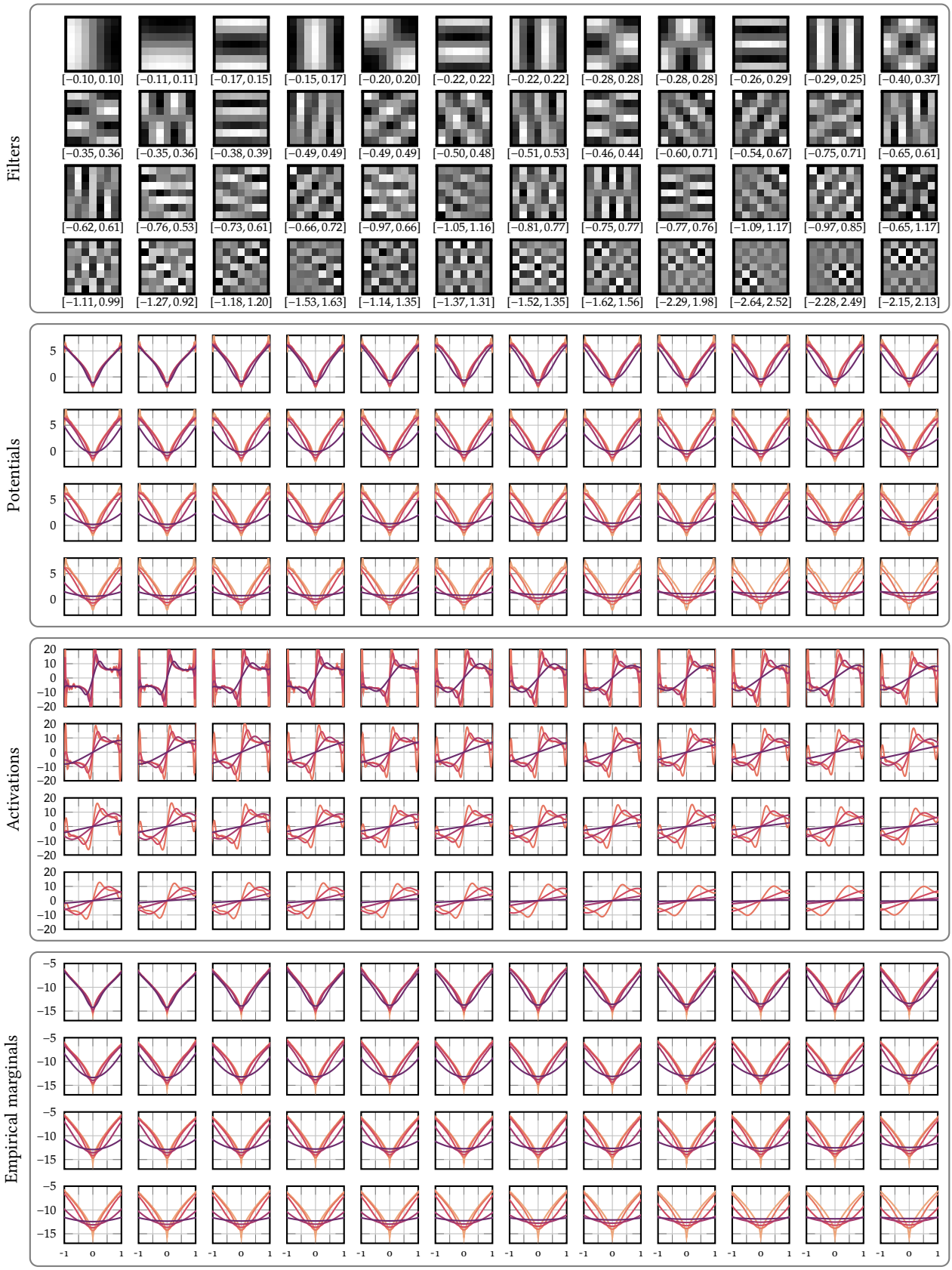

Figure 6.5: The learned undercomplete model utilizing Gaussian mixture experts. The learned potentials match the empirical marginals almost perfectly. The colors indicate the diffusion time $\sqrt{2t} = 0$, 0.025, 0.05, 0.1, 0.2.

6.4.3 Learning wavelets

As discussed in section 2.1.9, the discrete wavelet transformation is characterized by the sequence $h \in \mathbb{R}^K$. In addition to learning the parameters of the one-dimensional GMM, we follow [93] and also learn h. From the sequence h, the scaling-function ϕ and wavelet-function ω are defined by

$$\phi(x) = \sum_{k=1}^{K} h_k \sqrt{2} \phi(2x - k) \tag{6.63}$$

and

$$\omega(x) = \sum_{k=1}^{K} \Big(g(h)\Big)_k \sqrt{2} \phi(2x - k) \tag{6.64}$$

where $\Big(g(h)\Big)_k = (-1)^k h_{K-k-1}$. For ω to be a wavelet, it must follow the admissibility criterion

$$\int_0^\infty \frac{|(F\omega)(\zeta)|^2}{\zeta} \, \mathrm{d}\zeta < \infty, \tag{6.65}$$

cf [158], from which it immediately follows that $(F\omega)(0) = \int_{\mathbb{R}} \omega = 0$. For practical reasons, the transform should be normalized such that $\int_{\mathbb{R}} \phi = 1$. In addition, it has to be orthonormal to integer translates, i.e.

$$\int_{\mathbb{R}} \phi(x)\phi(x - n) \, \mathrm{d}x = \chi_{\{0\}}(n) \text{ for all } n \in \mathbb{Z}. \tag{6.66}$$

From these constraints, the feasible set of wavelet-generating sequences is described by

$$\Omega = \left\{ h \in \mathbb{R}^K \,\middle|\, \sum_{k=1}^{K} (g(h))_k = 0, \sum_{k=1}^{K} h_k = \sqrt{2}, \langle h, \circlearrowleft_{2n} h \rangle = \chi_{\{0\}}(n) \right\}. \tag{6.67}$$

Here, the last orthonormality constraint goes over all natural numbers n less than $K/2$ and

$$\begin{aligned} \circlearrowleft_n \colon \mathbb{R}^K &\to \mathbb{R}^K \\ (x_1, x_2, \dots, x_K) &\mapsto (x_{K-n+1}, x_{K-n+2}, \dots, x_K, x_1, x_2, \dots, x_{K-n}) \end{aligned} \tag{6.68}$$

rolls its argument by n entries. Observe that the orthogonality condition encodes $K/2$ constraints (we assume that K is even), since $\circlearrowleft_0 = \circlearrowleft_K = I$. To project onto Ω, we write the projection problem

$$\mathrm{proj}_\Omega(\bar{x}) = \arg\min_{x \in \Omega} \frac{1}{2} \|x - \bar{x}\|_2^2 \tag{6.69}$$

in its Lagrangian form using

$$\begin{aligned}\mathscr{L}\colon \mathbb{R}^K \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{K/2} &\to \mathbb{R} \\ (x, \Lambda_{\text{scal}}, \Lambda_{\text{adm}}, \Lambda) &\mapsto \frac{1}{2}\|x - \bar{x}\|_2^2 \\ &\quad + \Lambda_{\text{scal}}\Big(\sum_{k=1}^{K} h_k - \sqrt{2}\Big) + \Lambda_{\text{adm}} \sum_{k=1}^{K} \Big(g(h)\Big)_k \\ &\quad + \sum_{n=0}^{\frac{K}{2}-1} \Lambda_{n+1}\Big(\langle h, \circlearrowleft_{2n} h\rangle - \chi_{\{0\}}(n)\Big).\end{aligned} \tag{6.70}$$

and find stationary points by solving the associated nonlinear least-squares problem

$$\min_{x, \Lambda_{\text{scal}}, \Lambda_{\text{adm}}, \Lambda} \frac{1}{2}\|\nabla \mathscr{L}(x, \Lambda_{\text{scal}}, \Lambda_{\text{adm}}, \Lambda)\|_2^2 \tag{6.71}$$

using 10 iterations of the Levenberg-Marquardt algorithm (algorithm 9) with step size 1 and regularization parameter 10^{-10}. To facilitate convergence, we warm start the Lagrange multipliers $\Lambda_{\text{scal}}, \Lambda_{\text{adm}}, \Lambda$ with the solution from the previous outer iteration. We initialize the sequence h with the generating sequences of the `db2`- ($K = 4$) and `db4`-wavelet ($K = 8$). For both, we utilize 2 detail levels, but to account for the directionality of the two-dimensional wavelet transform, i.e., horizontal, vertical, and diagonal details, we have a total of $o = 6$ experts in our learned model. We use the `pytorch_wavelets` [63] implementation of the discrete wavelet transformation.

In contrast to the model based on filter-responses, the model based on wavelet-responses does not have the freedom to adapt the scaling of filters. To overcome this, we discretize the means over the real line individually for each sub-band. In detail, for the j-th detail level and d-th direction, $d \in \{\text{horizontal}, \text{vertical}, \text{diagonal}\}$, we choose $\eta_{j,d} = 1.1 q_{j,d}$, where $q_{j,d}$ is the 0.999-quantile (definition 2.2.6) of corresponding responses calculated on the training set.

The initial and learned generating sequences, their corresponding scaling- and wavelet-functions, along with the learned potential functions and MMSE-shrinkage are shown in fig. 6.6. For the potentials and the MMSE shrinkage functions, the first row is the finest detail level, $j = 1$, and the second row is the coarser detail level, $j = 2$. Within each detail level, the plots show vertical, horizontal, and diagonal details from left to right. In these figures, it is apparent that our chosen parametrization is sub-optimal. In particular, in order to represent the heavy tails (especially for the finest detail level $j = 1$), many intermediate weights are set to 0. This leads to the MMSE shrinkage functions becoming step-like. We emphasize that this is a practical problem of choosing the appropriate parametrization; we discuss alternatives to our equispaced GMM in section 6.5.

6.4.4 Learning shearlets

The construction of the shearlet system is discussed briefly in section 2.1.10 and we refer to [146] for mode details. For the purposes of this chapter we recall that the system is constructed from a one-dimensional low-pass filter h_1 and a

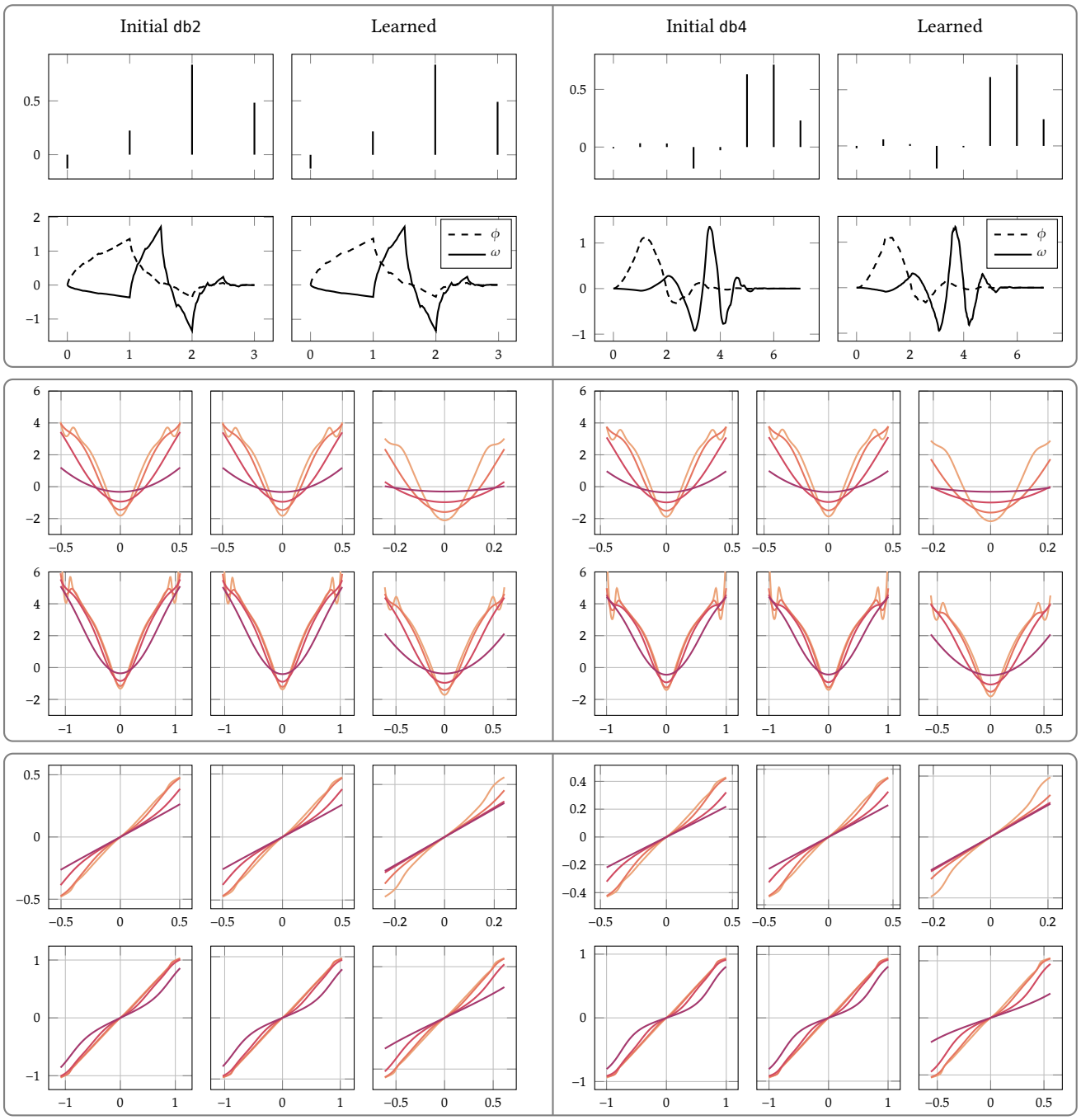

Figure 6.6: The learned wavelet model: On the left, we used the initial db2 generating sequence with $K = 4$, on the right we used the db4 generating sequence with $K = 8$. From top to bottom, the initial generating sequence along with the scaling- and wavelet functions ϕ and ω, the learned generating sequence along with the scaling- and wavelet functions, the learned potential functions $-\log \psi(\,\cdot\,, w, t)$ and the MMSE shrinkage functions $y_t \mapsto y_t + 2t\nabla \log \psi(y_t, w, t)$. For the potentials and the MMSE shrinkage functions, the first row is the finest detail level and the second row is the coarser detail level. Within each detail level, the plots show vertical, horizontal, and diagonal details from left to right. The colors indicate the diffusion time $\sqrt{2t} = 0.025$——, 0.05——, 0.1——, 0.2——.

Algorithm 17: Simplex projection from [112].

Input : $x \in \mathbb{R}^m$
Output: $y = \text{proj}_{\Delta^m}(x)$

$u = \text{sort}(x)$ // $u_1 \geq \ldots \geq u_m$
$K = \max_{1 \leq k \leq m}\{k \mid \left(\sum_{r=1}^{k} u_r - 1\right)/k < u_k\}$
$\tau = (\sum_{k=1}^{K} u_k - 1)/K$
$y = \max(x - \tau, 0)$ // element-wise

two-dimensional directional filter P. For the numerical experiments, we chose 2 detail levels, 5 shearings and learned experts for both the vertical and horizontal cones. Thus, the overcomplete models has a total of $o = 5 \times 2 \times 2 = 20$ experts, indexed by the detail levels, $j = 1, 2$, the shearings, $k = -2, -1, \ldots, 2$, and the cones $l =$ horizontal, vertical. We can summarize the learnable parameters for the model based on shearlet-responses as $\theta = \{h_1, P, \{\lambda_{j,k,l}\}_{j,k,l}, \{w_{j,k,l}\}_{j,k,l}\}$. We initialize the one-dimensional low-pass filter h_1 and the two-dimensional directional filter P with the standard choices from [141]: h_1 is initialized as maximally flat 9-tap symmetric low-pass filter, P is initialized as the maximally flat fan filter of size 17×17 described in [67]. Furthermore, $\lambda_{j,k,l}$ is initialized as 1 for all detail levels j, shearings k, and cones l, and we set $\eta_{j,k,l} = 0.5$.

We enforce the following constraints on the parameter blocks: For all detail levels, shearings, and cones, the weight parameter $\lambda_{j,k,l}$ must be non-negative. The one-dimensional low-pass filter h_1 must be gain-free, i.e. $h_1 \in \mathscr{H} := \{x \in \mathbb{R}^9 \mid \sum_{i=1}^{9} x_i = 1\}$. The two-dimensional directional filter P must have unit-one-norm, i.e. $P \in \mathscr{P} := \{x \in \mathbb{R}^{17 \times 17} \mid \sum_{i,j=1}^{17} |x_{i,j}| = 1\}$.

The projection operators can be realized as follows: For all detail levels j shearings k, and cones l, the weight parameter can be projected onto the non-negative real line via

$$\lambda_{j,k,l} \mapsto \max(\lambda_{j,k,l}, 0). \tag{6.72}$$

The map

$$x \mapsto x - \frac{1}{9}\left(\sum_{i=1}^{9} x_i - 1\right) \tag{6.73}$$

realizes the projection onto the linear constrain encoded in $\mathscr{H}$. The projection onto the unit-one-norm-sphere $\mathscr{P}$ is

$$x \mapsto \text{sgn}(x) \odot \text{proj}_{\Delta^{17 \times 17}}(|x|), \tag{6.74}$$

see e.g. [61, proposition 2.1]. We ignore the degenerate case of projecting the origin where $\text{proj}_{\mathscr{P}}$ is not well defined. Our implementation of the shearlet transformation is based on the ShearLab 3D [141] toolbox.[19]

19: The toolbox is available at http://shearlab.math.lmu.de/.

We show the initial and learned the one-dimensional low-pass filter h_1, and the two-dimensional directional filter P in fig. 6.7 and the resulting shearlet system in frequency- and time-domain along with the learned experts in fig. 6.8. There, the rows alternate between horizontal and vertical cones, and in reach row the first five

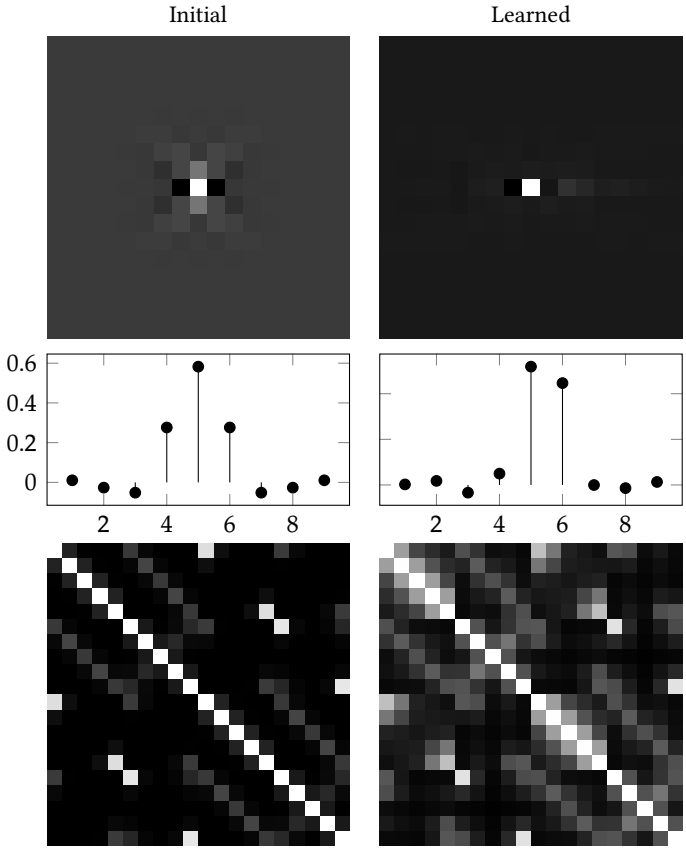

Figure 6.7: The first two rows show the two-dimensional directional filter P and the one-dimensional low-pass filter h. The last row shows the cosine similarity of the magnitude of the spectra of the resulting shearlet system. A system exactly fulfilling the assumption (6.46) would be unpopulated off the main diagonal.

plots show the five different shearings for the coarse detail level, $j = 1$ and the last five plots show the five different shearings for the finer detail level, $j = 2$. We again emphasize that the learned one-dimensional experts $\psi_{j,k,l}(\cdot, w_{j,k,l}, t)$ are distinctly different from the other models. In particular, the potentials $-\log \psi_{j,k,l}(\cdot, w_{j,k,l}, t)$ exhibit multiple local minima, sometimes different from 0, such that certain image structures can be enhanced under this prior. This is in stark contrast to the learned filter- and wavelet-responses, which show a single minimum at 0 and the classical heavy-tailed shape.

Visual inspection of the spectra shown in fig. 6.8 immediately reveals that the shearlet system only approximately fulfills the assumption (6.46) and (6.47). We analyze the shearlet system with respect to the assumption of disjoint support (6.46) by visualizing the pair-wise cosine similarity of the magnitude of the spectra in fig. 6.7. In detail, the figure shows $\left\langle \frac{|\gamma_{\tilde{j},\tilde{k},\tilde{l}}|}{\||\gamma_{\tilde{j},\tilde{k},\tilde{l}}|\|}, \frac{|\gamma_{j,k,l}|}{\||\gamma_{j,k,l}|\|} \right\rangle$, for $\tilde{j}, j = 1, 2$, $\tilde{k}, k = -2, \dots, 2$ and $\tilde{l}, l =$ horizontal, vertical.[20] Although less for the learned shearlet system, the plot is dominated by the main diagonal, indicating that the corresponding spectra are almost non-overlapping. To meet the theoretical assumptions, it would be

20: the exact arrangement irrelevant

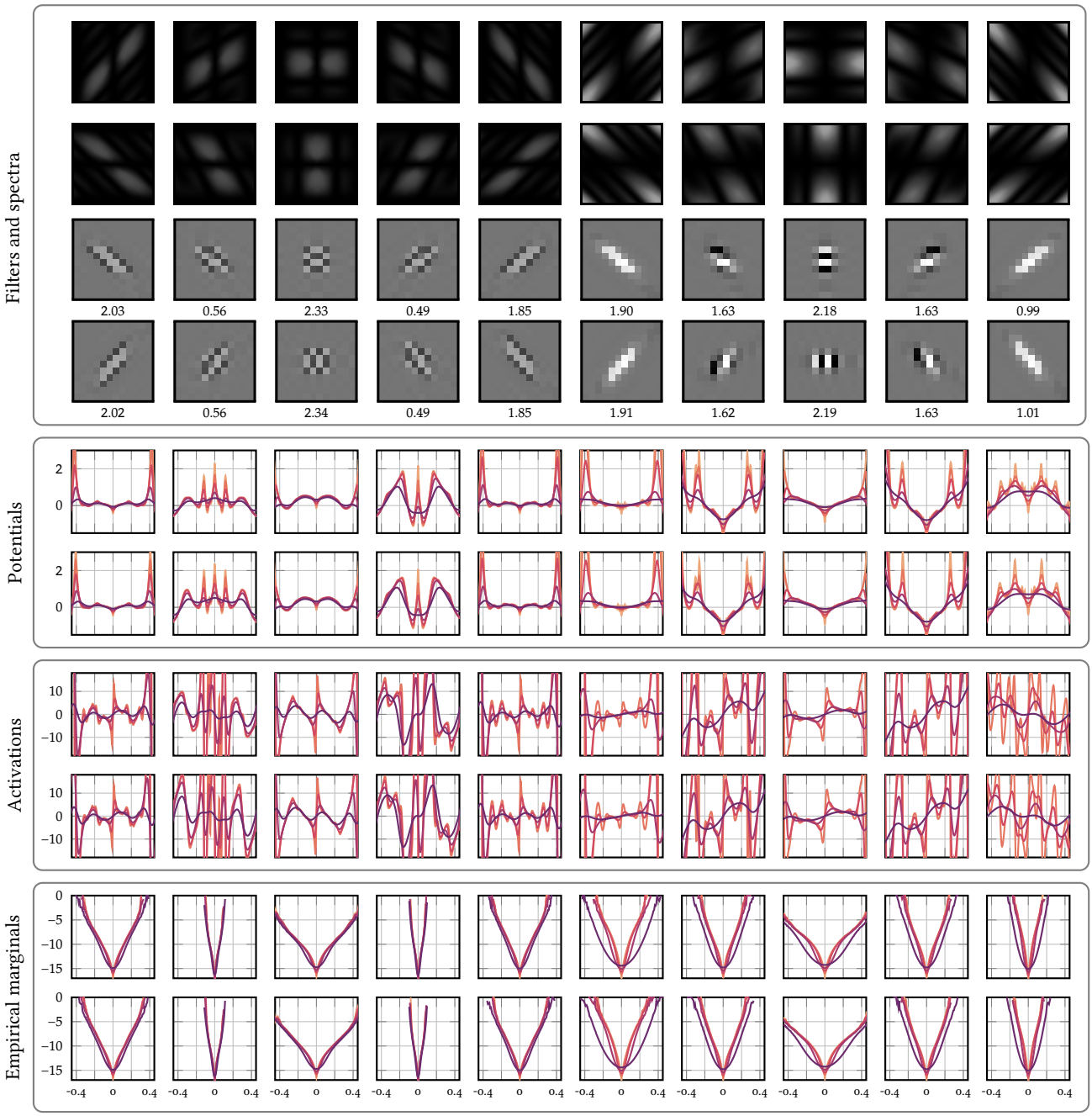

Figure 6.8: The learned overcomplete model utilizing Gaussian mixture expters. Below each filter its associated weight is shown. The learned potential functions (middle) are distinctly different from the negative-log empirical marginals (bottom). In particular, they have multiple local minima such that they can enhance certain structures. Sometimes (e.g. the second potential function from the left) zero is not even in the set of minima. The rows alternate between horizontal and vertical cones, and in reach row the first five plots show the five different shearings for the coarse detail level, $j = 1$ and the last five plots show the five different shearings for the finer detail level, $j = 2$. The colors indicate the diffusion time $\sqrt{2t} = 0$——, 0.025——, 0.05——, 0.1——, 0.2——.

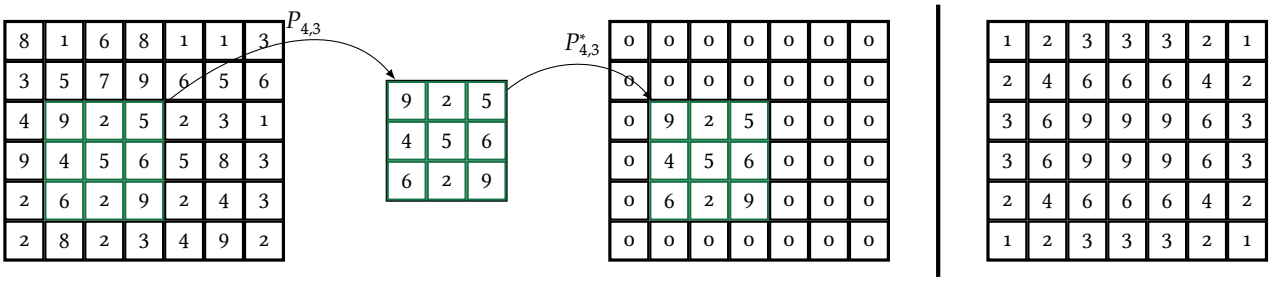

Figure 6.9: $P_{i,j}$ extracts a patch at pixel location (i, j), the adjoint $P^*_{i,j}$ operator puts the patch into the corresponding location in a zero-image. On the right, the number of overlapping patches at each pixel location is shown. In the example, we consider a 6×7 image and 3×3 patches.

possible to penalize $\langle \frac{|\gamma_{\tilde{j},\tilde{k},\tilde{l}}|}{\| |\gamma_{\tilde{j},\tilde{k},\tilde{l}}| \|}, \frac{|\gamma_{j,k,l}|}{\| |\gamma_{j,k,l}| \|} \rangle$ for $\tilde{j} \neq j$, $\tilde{k} \neq k$, and $\tilde{l} \neq l$ during training.

The fact that the spectra are not constant over their support raises the question of how to choose $\xi_{j,k,l}$ that best approximates (6.47). During training and evaluation, we simply chose $\xi_{j,k,l} = \| |\gamma_{j,k,l}| \|_\infty$. It remains an open question how the violation of the constraints (6.46) and (6.47) influences the diffusion, and if there exists a better choice for $\xi_{j,k,l}$.

6.4.5 Image denoising

This section addresses the prototypical image restoration problem: image denoising. To utilize the undercomplete model for image denoising, we employ the expected patch log-likelihood [260]. Assuming that $p_\theta^{\text{filt}}(\cdot, t)$ models the distribution of $b \times b$ image patches, we define the expected patch log-likelihood of a noisy image $y \in \mathbb{R}^{m \times n}$ with variance $\sigma^2(t) = 2t$ as

$$\text{epll}_\theta^{\text{filt}}(y, t) = \sum_{i,j=\tilde{b}}^{m-\tilde{b},n-\tilde{b}} p_{i,j}^{-1} \log p_\theta^{\text{filt}}(P_{i,j} y, t), \tag{6.75}$$

where $\tilde{b} = \lfloor b/2 \rfloor$. The sum ranges over all *overlapping* patches, with $P_{i,j} \colon \mathbb{R}^{m \times n} \to \mathbb{R}^{b \times b}$ extracting the $b \times b$ image patch centered at (i, j) and $p_{i,j} = \left(\sum_{k,l=\tilde{b}}^{m-\tilde{b},n-\tilde{b}} P^*_{k,l} P_{k,l} \mathbf{1} \right)_{i,j}$ counts the number of overlapping patches at (i, j) to compensate for boundary effects.[21] An example is illustrated in fig. 6.9. Wavelet- and shearlet-based priors can act on images of arbitrary size.

21: Here, $\mathbf{1}$ is the one-image in $\mathbb{R}^{m \times n}$.

Let $\log p_\theta$ represent either $\text{epll}_\theta^{\text{filt}}$, $\log p_\theta^{\text{wave}}$, or $\log p_\theta^{\text{conv}}$. We consider three inference methods: In the classical variational formulation

$$\min_{x \in \mathbb{R}^{m \times n}} \frac{1}{2} \|x - y\|^2 - \lambda \log p_\theta(x, t), \tag{6.76}$$

different choices of $t > 0$ are possible. $\log p_\theta(\cdot, t)$ approaching p_X as t approaches o motivates choosing t small. However, smaller t make optimizing eq. (6.76) more challenging as. We chose $t = 0.01$ balancing optimization feasibility with model performance.

Algorithm 18: Stochastic image denoising algorithm from [129].

Input : Variance schedule $\{\sigma_i\}_{i=1}^{C}$, noisy image $y \in \mathbb{R}^{m\times n}$, inner iterations $B \in \mathbb{N}$, noise level σ_0 in y

Output : Stochastically denoised image x_B

for $i \in 1, 2, \dots, C$ **do**
  $\alpha_i \leftarrow \epsilon\sigma_i^2/\sigma_C^2$
  for $b \in 1, \dots, B-1$ **do**
    $z_b \sim \mathcal{N}_{0,I}$
    $g_b \leftarrow \nabla \log p_\theta(x_{b-1}, \sigma_i) + (y - x_{b-1})/(\sigma_0^2 - \sigma_i^2)$
    $x_b \leftarrow x_{b-1} + \alpha_i g_b + \sqrt{2\alpha_i} z_b$
  $x_0 \leftarrow x_B$

The empirical Bayes estimate

$$\hat{x}_{\mathrm{EB}}(y,t) = y + \sigma^2(t)\nabla_1 \log p_\theta(y,t), \tag{6.77}$$

is the most natural inference strategy in the context of this chapter. This estimator provides the Bayesian MMSE through one gradient evaluation. For the undercomplete model, $\log p_\theta = \mathrm{epll}_\theta^{\mathrm{filt}}$, the estimator

$$\begin{aligned}\hat{x}_{\mathrm{EB}}(y,t) &= y + \sigma^2(t)\nabla_1 \mathrm{epll}_\theta^{\mathrm{filt}}(y,t) \\ &= y + 2t \sum_{i,j=\tilde{b}}^{m-\tilde{b},n-\tilde{b}} p_{i,j}^{-1} P_{i,j}^* \nabla_1 \log p_\theta^{\mathrm{filt}}(P_{i,j} y, t)\end{aligned} \tag{6.78}$$

computes patch-wise MMSE estimates and combines them by averaging. This is known to be a sub-optimal inference strategy, since the averaged patches are not necessarily likely under the model [260]. We refer the interested reader to the works of [203, 260] for a detailed discussion on utilizing patch-based priors for whole-image restoration.

Lastly, we use recently proposed algorithms for inverse problems that utilize diffusion models. We consider the stochastic denoising algorithm by [129] (see algorithm 18) with standard parameters $\epsilon = 5\times10^{-6}$, $\sigma_C = 0.01$ and the exponential schedule $\sigma_i = \sqrt{2t}\left(\frac{\sigma_C}{\sqrt{2t}}\right)^{i/C}$, using $B = 3$ inner loops and $C = 100$ diffusion steps. The algorithm samples from the posterior of a denoising problem when utilizing diffusion priors, effectively balancing the prior $\nabla \log p_\theta(\,\cdot\,, t)$ and the data term as t approaches 0. Sampling from the posterior often produce sharper results with modern diffusion models [128, 129].

Table 6.1 shows quantitative results using PSNR (definition 2.5.3) and SSIM (definition 2.5.4), computed with a uniform 7×7 filter and parameters $K_1 = 0.01$ and $K_2 = 0.03$. The column titled "Patch-GSM" utilizes the Gaussian scale mixture parametrization discussed in section 6.5.2. The results are for one run of the algorithms without computing the expectation over the noise (neither in the construction of y_t nor during the iterations of the stochastic denoising algorithm), thus confidence intervals reflect only image variability in the test dataset. We do

not discuss posterior variance in this chapter but techniques for analyzing the posterior induced by diffusion models are also readily applicable to our models. We refer to [129] or related papers such as [54] for an in-depth discussion of these techniques.

For the variational formulation, the overcomplete model performs best when the noise level is small. However, the undercomplete model with GSM experts surpasses other models at high noise levels dur to its smooth, non-oscillatory tails. While the models are trained to approximate the reference density p_X at time 0, the potentials become oscillatory in practice. Oscillatory potentials, as seen in the wavelet model (see fig. 6.6), hinder optimization, but GSM experts' smooth tails avoid this issue.[22]

22: However, GSMs cannot represent multi-modal densities like GMMs.

The oscillation problem can be mitigated by considering a decreasing time sequence and optimizing the variational objective, where subsequent problems are initialized with previous solutions. This has strong relations to classical continuation methods used in computer vision, see e.g. [244, 246]. A proximal-gradient continuation method is presented in our preliminary work [249]. Kobler and Pock's joint minimization approach using preconditioned proximal-gradient [135, figure 4] also shows promising results. In addition, they show that *learning* a decreasing time sequence and step sizes of an optimization algorithm improves the results, which is is strongly related to variational networks. However, in their approach only the "parameters" of the optimization algorithm are learned whereas the model remains fixed.

The empirical Bayes estimator demonstrates the overcomplete model's parameter efficiency, performing close to GMM-expected patch log-likelihood (EPLL) [260] with significantly fewer parameters. *One* full covariance matrix on $\mathbb{R}^{7\times7}$ has 1226 learnable parameters, slightly less than the 1642 parameters in our overcomplete model. With one component, GMM-EPLL effectively reduces to quadratic potentials acting on filter responses, which is a bad model for natural images (see chapter 4). Thus, we use the setup of the original paper [260] with 200 components, totaling 200(1 + 49 + 49 choose 2) = 245 200 learnable parameters. Leveraging symmetries in the shearlet system could increase parameter efficiency further: Sharing the potentials between the cones would half the number of parameters (the symmetry is apparent in fig. 6.8).

The overcomplete model's performance deteriorates with higher noise levels, likely due to a mismatch between theoretical filter assumptions and the practical shearlet characteristics. Specifically, the magnitude of the spectra is only approximately constant on the support and we adapting the variances with the maximum may not be the optimal choice.

The empirical Bayes estimator outperforms stochastic denoising in all quality metrics, likely because only one sample is drawn in the stochastic algorithm; multiple samples would likely improve quantitative results. The empirical Bayes estimator consistently beats the variational approach only for the overcomplete model. The undercomplete models appear to be better suited for MAP estimation than MMSE estimation, highlighting the difference between penalized likelihood estimation and Bayesian estimation: good results in MAP estimation can occur even with a poor prior model[23] Conversely, Nikolova [176] and Schmidt et al. [209,

23: The most prominent example is image restoration with total variation penalization.

section 5] point out that good prior models are not well suited for MAP estimation.

Qualitative results using the variational approach, empirical Bayes estimation, and stochastic denoising are shown in fig. 6.10, fig. 6.11, and fig. 6.12, respectively. The overcomplete model produces more natural reconstructions particularly with Empirical Bayes estimation. Prominent structures, like the tiger stripes, are sometimes emphasized due to complex learned potentials. In contrast, the reconstructions of the undercomplete models appear overly smooth, removing rather than enhancing structures. Surprisingly, sharper images were not observed with the stochastic denoising algorithm.

		σ	y_t	Patch-GMM		-GSM	Wavelet		Shearlet	TV
				$b = 7$	$b = 15$	$b = 7$	$K = 4$	$K = 8$		
Optimization	PSNR	0.025	32.04 ± 0.00	34.70 ± 0.29	35.17 ± 0.29	35.41 ± 0.35	33.67 ± 0.28	33.74 ± 0.30	**34.98 ± 0.31**	34.63 ± 0.34
		0.050	26.02 ± 0.00	29.90 ± 0.55	30.91 ± 0.35	31.26 ± 0.42	29.12 ± 0.42	29.03 ± 0.31	**30.88 ± 0.46**	30.42 ± 0.38
		0.100	20.00 ± 0.00	27.29 ± 0.44	27.70 ± 0.55	**27.94 ± 0.53**	24.09 ± 0.27	23.96 ± 0.25	27.03 ± 0.57	27.12 ± 0.54
		0.200	13.98 ± 0.00	24.46 ± 0.46	24.91 ± 0.56	**24.99 ± 0.61**	18.43 ± 0.13	18.42 ± 0.12	18.03 ± 0.12	24.25 ± 0.66
	SSIM	0.025	0.84 ± 0.02	0.93 ± 0.01	**0.94 ± 0.00**	**0.94 ± 0.00**	0.91 ± 0.01	0.91 ± 0.01	**0.94 ± 0.00**	0.93 ± 0.00
		0.050	0.65 ± 0.03	0.85 ± 0.01	0.86 ± 0.01	**0.88 ± 0.01**	0.81 ± 0.01	0.79 ± 0.01	0.87 ± 0.01	0.85 ± 0.01
		0.100	0.41 ± 0.03	0.74 ± 0.01	0.78 ± 0.01	**0.79 ± 0.01**	0.56 ± 0.02	0.56 ± 0.02	0.76 ± 0.02	0.75 ± 0.01
		0.200	0.21 ± 0.02	0.61 ± 0.01	**0.66 ± 0.02**	**0.66 ± 0.02**	0.30 ± 0.02	0.30 ± 0.02	0.34 ± 0.02	0.62 ± 0.03
										GMM-EPLL
Empirical Bayes	PSNR	0.025	32.04 ± 0.00	34.54 ± 0.21	35.00 ± 0.27	35.08 ± 0.29	33.51 ± 0.23	33.61 ± 0.25	35.30 ± 0.38	**35.59 ± 0.35**
		0.050	26.02 ± 0.00	30.44 ± 0.32	30.79 ± 0.37	30.80 ± 0.37	29.32 ± 0.29	29.48 ± 0.31	31.17 ± 0.44	**31.68 ± 0.46**
		0.100	20.00 ± 0.00	27.03 ± 0.42	27.27 ± 0.46	27.20 ± 0.44	25.72 ± 0.36	25.90 ± 0.36	27.50 ± 0.46	**28.12 ± 0.50**
		0.200	13.98 ± 0.00	24.24 ± 0.48	24.45 ± 0.52	24.29 ± 0.48	22.41 ± 0.33	22.54 ± 0.32	23.91 ± 0.40	**24.92 ± 0.51**
	SSIM	0.025	0.84 ± 0.02	0.92 ± 0.01	0.93 ± 0.01	0.93 ± 0.00	0.90 ± 0.01	0.90 ± 0.01	0.94 ± 0.00	**0.95 ± 0.00**
		0.050	0.65 ± 0.03	0.83 ± 0.01	0.85 ± 0.01	0.85 ± 0.01	0.80 ± 0.01	0.80 ± 0.01	0.87 ± 0.01	**0.89 ± 0.01**
		0.100	0.41 ± 0.03	0.72 ± 0.01	0.73 ± 0.01	0.73 ± 0.01	0.65 ± 0.02	0.66 ± 0.02	0.75 ± 0.01	**0.79 ± 0.01**
		0.200	0.21 ± 0.02	0.58 ± 0.01	0.60 ± 0.01	0.59 ± 0.01	0.48 ± 0.02	0.49 ± 0.02	0.55 ± 0.01	**0.63 ± 0.01**
Stochastic Denoising	PSNR	0.025	32.04 ± 0.00	31.34 ± 0.11	31.79 ± 0.16	31.88 ± 0.18	30.68 ± 0.10	30.78 ± 0.11	**32.40 ± 0.27**	—
		0.050	26.02 ± 0.00	27.07 ± 0.18	27.55 ± 0.23	27.62 ± 0.25	26.00 ± 0.14	26.17 ± 0.15	**28.46 ± 0.40**	—
		0.100	20.00 ± 0.00	23.66 ± 0.24	24.06 ± 0.27	24.06 ± 0.28	22.03 ± 0.15	22.24 ± 0.16	**24.94 ± 0.44**	—
		0.200	13.98 ± 0.00	20.90 ± 0.25	**21.24 ± 0.29**	21.12 ± 0.28	18.60 ± 0.13	18.71 ± 0.13	21.10 ± 0.34	—
	SSIM	0.025	0.84 ± 0.02	0.84 ± 0.02	0.85 ± 0.01	0.86 ± 0.01	0.82 ± 0.02	0.82 ± 0.02	**0.88 ± 0.01**	—
		0.050	0.65 ± 0.03	0.69 ± 0.02	0.71 ± 0.02	0.72 ± 0.02	0.65 ± 0.02	0.65 ± 0.02	**0.78 ± 0.01**	—
		0.100	0.41 ± 0.03	0.52 ± 0.02	0.54 ± 0.02	0.54 ± 0.02	0.46 ± 0.03	0.47 ± 0.03	**0.63 ± 0.01**	—
		0.200	0.21 ± 0.02	0.36 ± 0.02	0.37 ± 0.02	0.37 ± 0.02	0.29 ± 0.02	0.30 ± 0.02	**0.41 ± 0.01**	—
Learnable parameters				5376	78 400	3312	389	393	1642	245 200

Table 6.1: Quantitative results in terms of PSNR and SSIM using one-step empirical Bayes denoising the stochastic denoising algorithm from [129]. The intervals indicate the 0.95 confidence region, bold typeface indicates the best method.

6.4.6 Noise estimation and blind image denoising

Within this and the following subsection, we describe two applications that arise as a byproduct of our principled approach: Noise estimation (and, consequently, blind denoising) and analytic sampling. For both, we utilize the undercomplete model as a stand-in but we believe that generalizations to the overcomplete model are possible.

The construction of our model allows us to interpret $p_\theta^{\text{filt}}(\cdot, t)$ as a time-conditional density. Thus, it can naturally be used for noise level estimation: Assume that a noisy image patch $y \in \mathbb{R}^{b\times b}$ is constructed by $y = x + \sigma\eta$, where $x \sim p_X$, $\eta \sim \mathcal{N}_{0,I}$, and the noise level σ is unknown. We can estimate the noise level by maximizing the likelihood of the given patch y w.r.t. to the diffusion time—$\hat{t} = \arg\max_t p_\theta^{\text{filt}}(y, t)$—and recover the noise level via $\hat{\sigma} = \sqrt{2\hat{t}}$.

To demonstrate the feasibility of this approach, we show the expected negative-log density $\mathbb{E}_{x\sim p_X,\eta\sim\mathcal{N}_{0,I}}\left[l_\theta(x+\sigma\eta, t)\right]$ over a range of σ and t in fig. 6.13. Here, for visualization purposes we normalized the negative-log density to have a minimum of zero over t:

$$l_\theta(x,t) = -\log p_\theta^{\text{filt}}(x,t) - \left(\max_t \log p_\theta^{\text{filt}}\theta(x,t)\right). \tag{6.79}$$

The noise level estimate $\sigma \mapsto \arg\min_t \mathbb{E}_{x\sim p_X,\eta\sim\mathcal{N}_{0,I}}\left[l_\theta(x+\sigma\eta, t)\right]$ perfectly matches the identity map $\sigma \mapsto \sqrt{2t}$.

The previous result showed that the noise estimation works in expectation. Now, we provide empirical evidence that the noise estimation is reasonably robust with respect to instances of the noise and the content of the underlying image. To this end, we perform blind heteroscedastic denoising as follows: First, for all overlapping patches $P_{i,j}y$, $i = \tilde{b}, \tilde{b}+1, \dots, m-\tilde{b}$, $j = \tilde{b}, \tilde{b}+1, \dots, n-\tilde{b}$ in the noisy image, we optimize the diffusion time: $\hat{t}_{i,j} = \arg\max_t p_\theta^{\text{filt}}(P_{i,j}y, t)$. Then, given the diffusion times $\hat{t}_{i,j}$, we can estimate the clean image via an empirical Bayes step of the form

$$\hat{x}_{\text{blind}}(y) = y + 2\sum_{i,j=\tilde{b}}^{m-\tilde{b},n-\tilde{b}} \hat{t}_{i,j} p_{i,j}^{-1} P_{i,j}^* \nabla_1 \log p_\theta^{\text{filt}}(P_{i,j}y, \hat{t}_{i,j}), \tag{6.80}$$

where for each patch $P_{i,j}y$ we utilize the estimated diffusion time $\hat{t}_{i,j}$.

In the first column of fig. 6.14, we show six images corrupted by heteroscedastic Gaussian noise, where the standard deviation alternates between 0.1 and 0.2 in a checkerboard pattern. This checkerboard pattern is clearly visible in the visualization of the optimal diffusion time in the second column. The restored image and the absolute difference to the reference image are shown in the third and fourth column. There, the checkerboard pattern is hardly visible, indicating that the noise level estimation is robust also when confronted with little data.

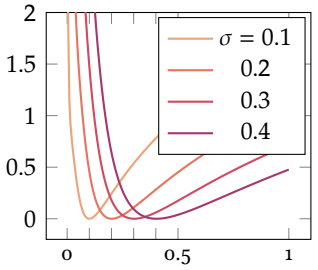

Figure 6.13: Expected normalized negative log-density $\sqrt{2t} \mapsto \mathbb{E}_{x\sim p_X,\eta\sim\mathcal{N}_{0,I}}\left[l_\theta(x+\sigma\eta, t)\right]$ for different noise levels.

6.4.7 Sampling

A direct consequence of corollary 1 is that the undercomplete model admits a simple sampling procedure: The statistical independence of the components allows

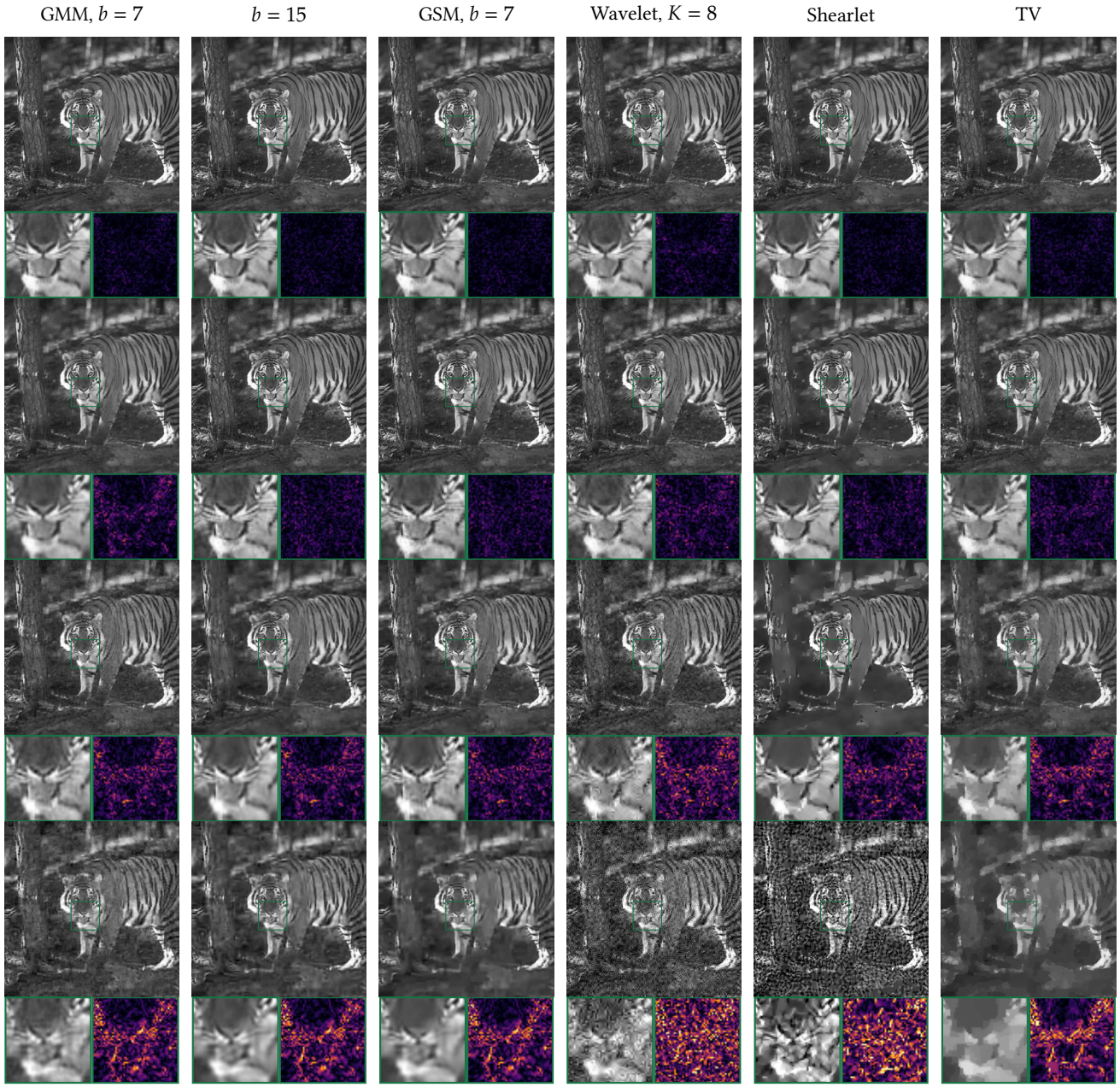

Figure 6.10: Qualitative results for optimization-based denoising. In the rows, the noise standard deviation ranges in $\sigma \in \{0.025, 0.05, 0.1, 0.2\}$. The inlays show a zoomed region (magnifying factor 3), and the absolute difference of the reconstruction to the ground truth image (0 ▬ 1/3). The accompanying quantitative results are shown in table 6.1.

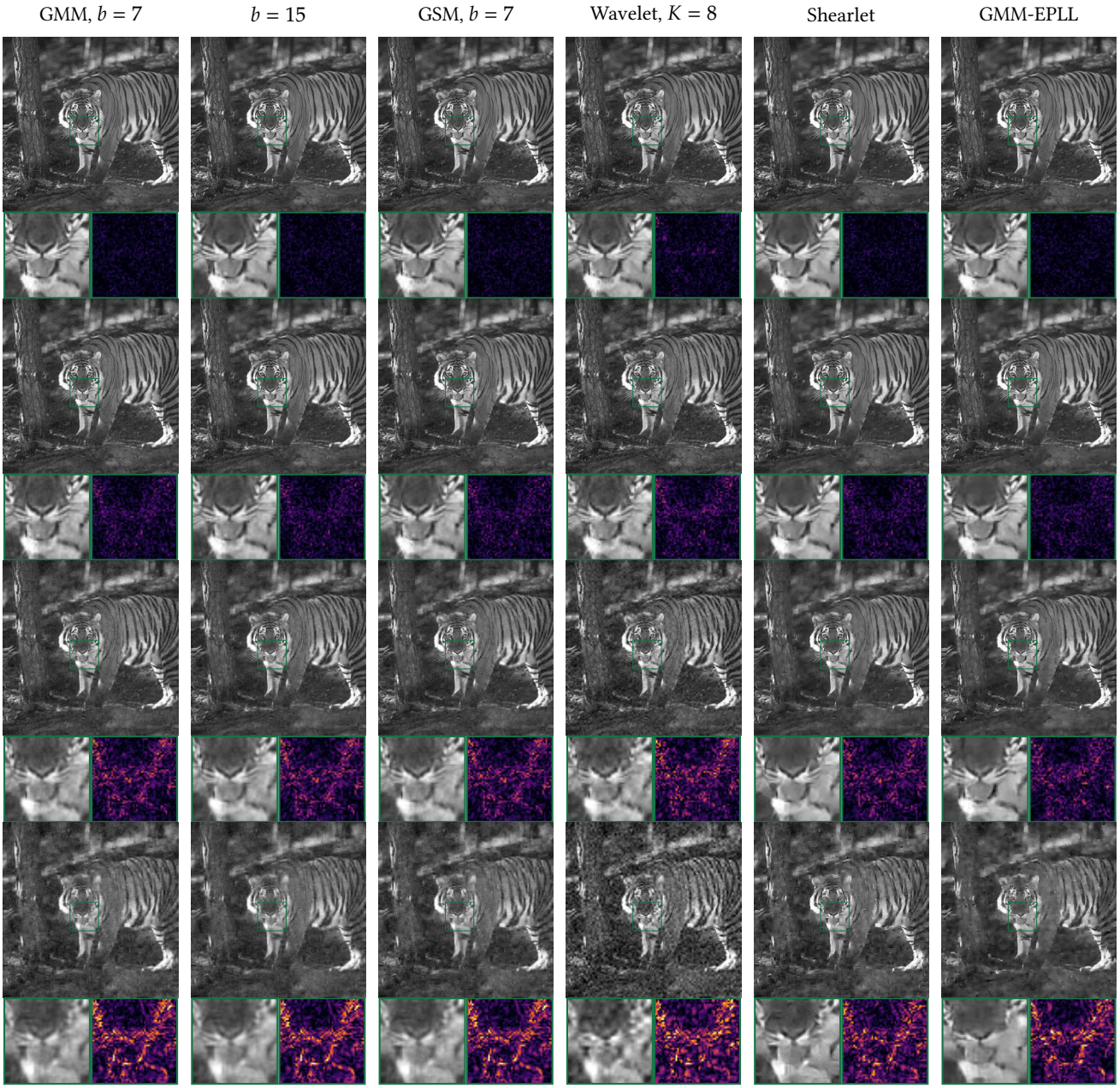

Figure 6.11: Qualitative results for one-step empirical Bayes denoising. In the rows, the noise standard deviation ranges in $\sigma \in \{0.025, 0.05, 0.1, 0.2\}$. The inlays show a zoomed region (magnifying factor 3), and the absolute difference of the reconstruction to the ground truth image (0 1/3). The accompanying quantitative results are shown in table 6.1.

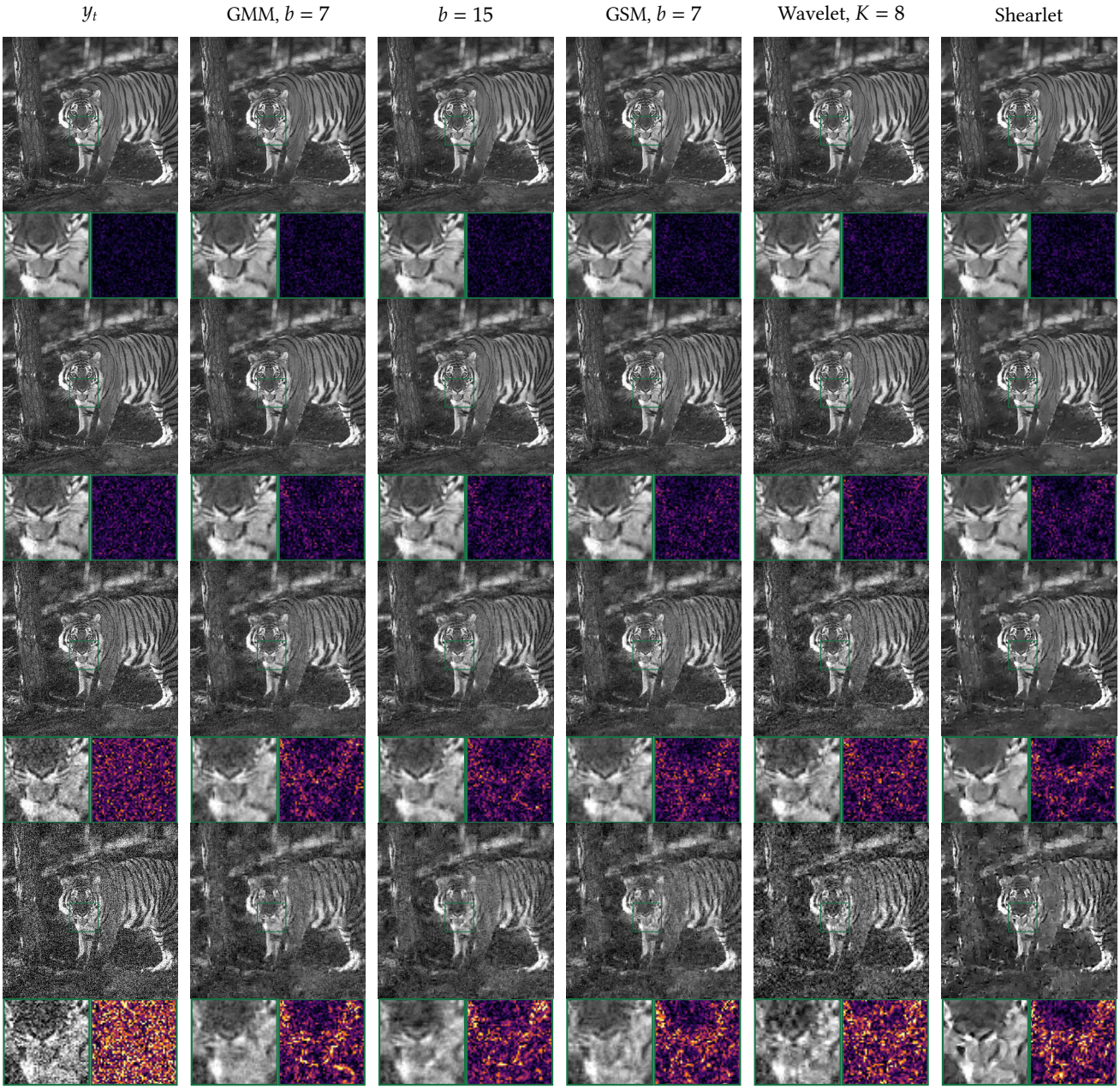

Figure 6.12: Qualitative results using the stochastic denoising algorithm from [129]. In the rows, the noise standard deviation ranges in $\sigma \in \{0.025, 0.05, 0.1, 0.2\}$. The inlays show a zoomed region (magnifying factor 3), and the absolute difference of the reconstruction to the reference image (0 1/3). The accompanying quantitative results are shown in table 6.1.

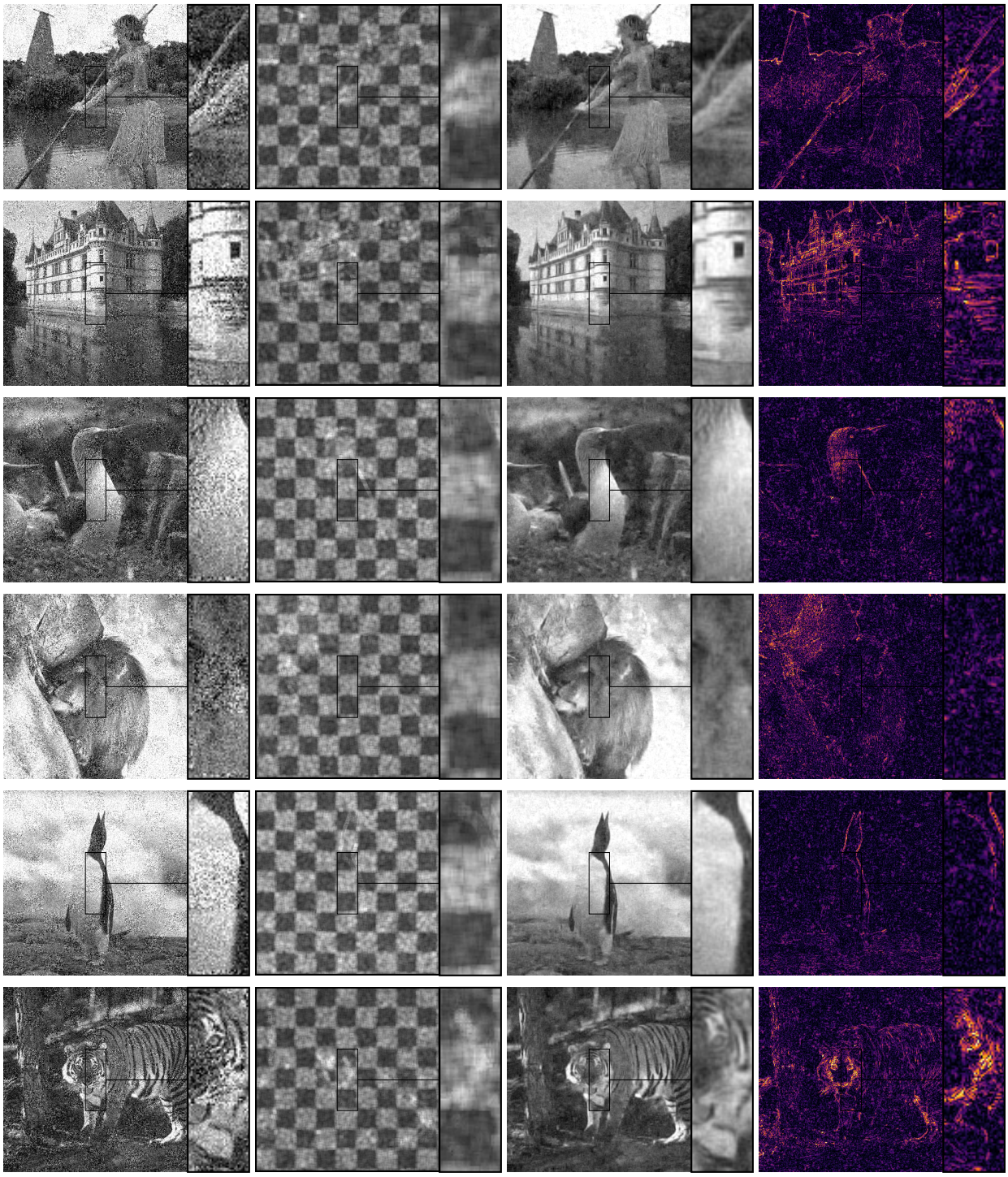

Figure 6.14: Noise estimation and blind denoising: The columns show the input image corrupted with heteroscedastic Gaussian noise in a checkerboard pattern with standard deviation 0.1 and 0.2, the patch-wise noise level estimate (0 0.5), the one-step empirical Bayes denoising result using (6.80), and its absolute difference to the reference image (0 1/3).

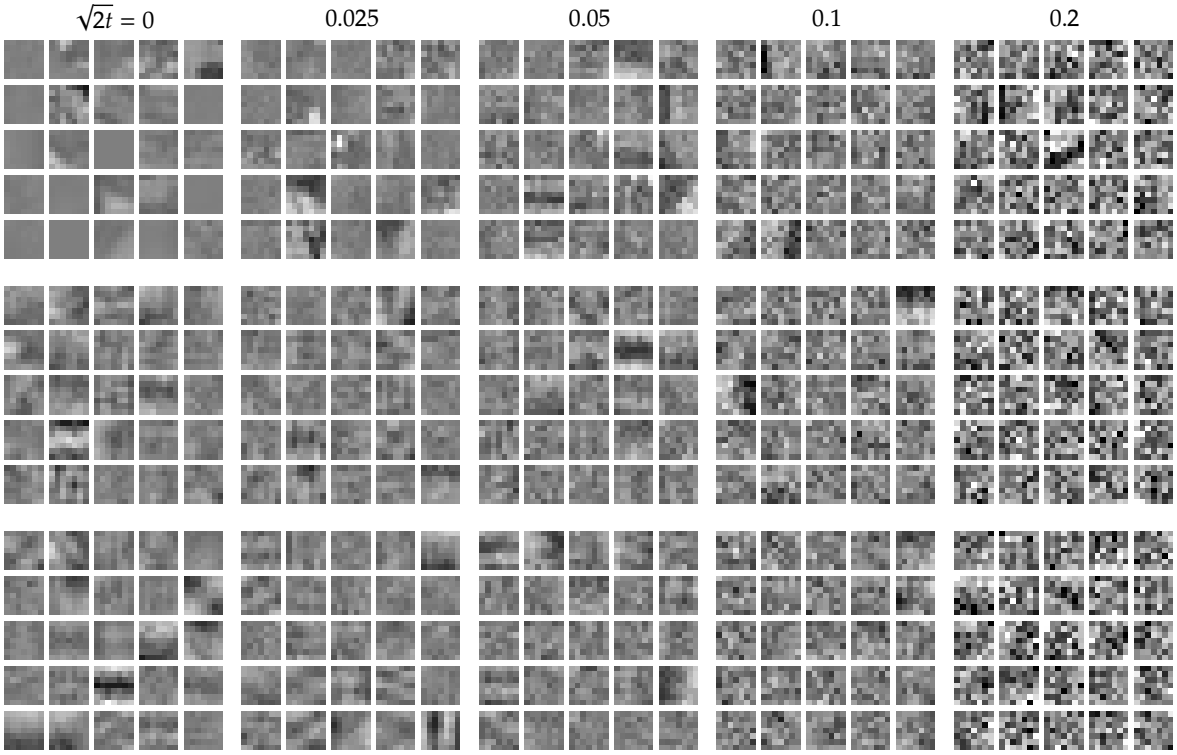

Figure 6.15: Reference samples from the random variable Y_t (top) and samples generated by the analytic sampling procedure (6.81) using GMM experts (middle) and GSM experts (bottom).

drawing random patches by

$$Y_t = \sum_{k=1}^{o} \frac{f_k}{\|f_k\|^2} U_{k,t}, \tag{6.81}$$

where $U_{k,t}$ is a random variable on $\mathbb{R}$ sampled from the one-dimensional expert $\psi_k(\cdot, w_k, t)$.

In fig. 6.15, we show samples from the reference distribution along with samples from the learned models where the experts are GMMs or GSMs. In both cases, the generated patches appear slightly noisy for small t. This indicates that the parametrization is not "peaky enough" around 0. However, the discretization is easily adapted: For the GMM, the discretization of the means as equidistant over the real line could be changed to, e.g., a logarithmic scaling resulting in a denser grid around 0. This would require the base-variance σ_0^2 to change with the mean, which is also possible in our model. For the GSM, the peakiness of the expert is determined by the smallest variance, which is entirely up to choice; see the discussion in section 5.6.

We believe that the structure of the linear operators in the overcomplete models also allows for an efficient sampling procedure; this is subject to future work.

6.5 Discussion

In this section, we first give an interpretation of our learned complete model in the wavelet basis as *diffusion wavelet shrinkage*. Then, we discuss possible extensions to our model in more detail: Alternative parametrizations and the possibilities of building more expressive models. Finally, we again highlight the difference of the undercomplete and the overcomplete model and the implication of overcompleteness.

6.5.1 Interpretation as diffusion wavelet shrinkage

Wavelet shrinkage is a popular class of denoising algorithms. Starting from the seminal works of Donoho [72, 73, 74], a vast literature is dedicated to finding optimal shrinkage parameters for wavelet-based denoising (see, e.g. [40, 51, 57, 65, 126, 213] and the references therein). In what follows, we briefly describe historical approaches to estimating shrinkage parameters.

The key motivation behind wavelet shrinkage denoising algorithms is the observation that wavelet coefficients of natural images are sparse, wheres the wavelet coefficients of noisy images are densely filled with "small" values. Thus, a straight forward denoising algorithm might be to calculate the wavelet coefficients, "shrink" small coefficients towards zero, and calculate the inverse wavelet transform of the shrank coefficients. In the terminology we use in this thesis, wavelet shrinkage algorithms *penalized likelihood estimation* algorithms. Popular shrinkage operators include the soft-shrinkage

$$x \mapsto \operatorname{sgn}(x) \max(|x| - \tau, 0) \tag{6.82}$$

and the hard-shrinkage

$$x \mapsto x \chi_{\{y \in \mathbb{R} | -\tau < y \leq \tau\}}(x). \tag{6.83}$$

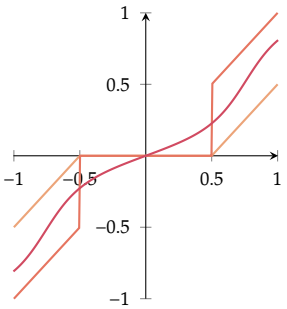

Figure 6.16: Popular wavelet shrinkage functions: The soft-shrinkage —— and the hard-shrinkage —— with threshold parameter $\tau = 0.5$. In addition, the learned MMSE optimal shrinkage function ——.

It is easy to see that these operators promote sparsity in the wavelet coefficients, as they correspond to the proximal maps (definition 2.4.18) with respect to the sparsity inducing norms $\tau\|\cdot\|_1$ and $\tau\|\cdot\|_0$ respectively; they are visualized in fig. 6.16. Here, $\tau > 0$ is a thresholding parameter that has to be chosen depending on the noise level.

Historically, research for wavelet shrinkage models has focused on finding the optimal shrinkage parameter τ (w.r.t. some risk, e.g. the squared error), assuming a particular choice of the shrinkage operator (e.g. the soft-shrinkage). Popular selection methods include *VisuShrink* [74] and *SureShrink* [73]. The former is signal independent and the threshold is essentially determined by the dimensionality of the signal as well as the (assumed known) noise level. In contrast, the latter chooses the thresholding parameter depending on the energy in a particular sub-band and does not depend on the dimensionality of the signal explicitly. The *BayesShrink* [41] method is also sub-band adaptive, and the authors provide expressions (or at least good approximations) for the optimal thresholding parameter under a generalized Gaussian prior on the wavelet coefficients. In particular, they rely on classical noise level estimation techniques to fit the generalized Gaussian to the wavelet

coefficients (of the noisy image) and arrive at a simple expression for a sub-band dependent threshold.

The general methodology outlined in the previous section allows us to take a different approach: Instead of fixing the thresholding function and estimating the threshold solely on the corrupted image, we instead propose to learn the distribution of wavelet coefficients in different sub-bands for all noise levels $\sigma > 0$. Notice that an empirical Bayes step on the wavelet coefficients under our model corresponds to applying a point-wise non-linearity.

In contrast to the traditional wavelet shrinkage, our model does not prescribe a shrinkage function for which an optimal parameter has to be estimated for different noise levels. Rather, by learning the distribution of the wavelet coefficients at "all" noise levels, we have access to an MMSE optimal "shrinkage" function view of the empirical Bayes step on the experts. In addition, our wavelet prior can be used in more general inverse problems whereas classical shrinkage methods are only applicable to denoising (although the denoising engine could be used in regularization by denoising [204] or plug-and-play [236] approaches).

6.5.2 Alternative parametrizations

We start with discussing alternative parametrizations of the undercomplete model. Under the orthogonality assumption eq. (6.26), corollary 1 shows that the one-dimensional experts model the marginal distributions of the underlying random variable along the directions of the filters. In chapter 4, we discussed in-depth that these marginal distributions are always highly leptokurtic, irrespective of the filters. Although the GMM is a natural choice to model these distributions in our setup[24], in some sense it is parameter inefficient: The heavy tails arise only through the large number of components that are gridded along an interval of the real line. At the same time, the discretization of the means over the real line has to be fine enough to model sharp peaks. Hence, the majority of the learnable parameters are actually the weights of the one dimensional Gaussian mixtures. This motivates the consideration of other experts that are more "inherently" leptokurtic.

24: due to the closure properties under the diffusion process

A popular choice is the Student-t expert [117, 206]

$$x \mapsto \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}, \tag{6.84}$$

where $\Gamma(z) = \int_0^\infty t^{z-1}\exp(-t)\,\mathrm{d}t$ is the Gamma function. The convolution of this function with a Gaussian can not be expressed in closed form. However, there exist approximations, such as the one in [84] or [18, theorem 1]: Let X be a random variable on $\mathbb{R}$ with density eq. (6.84), and let Y_t be a random variable defined by the diffusion process eq. (6.2). Then, the density of Y_t is $\lim_{N\to\infty} p^{(N)}_{Y_t}$ where

$$p^{(N)}_{Y_t}(y) = \frac{\exp\left(-\frac{y^2}{4t}\right)\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{4t\pi}\Gamma\left(\frac{\nu}{2}\right)\left(\frac{4t}{\nu}\right)^{\frac{\nu}{2}}}\sum_{n=0}^{N}\left(\frac{1}{n!}\left(\frac{y^2}{4t}\right)^n\Psi\left(\frac{\nu+1}{2}, \frac{\nu}{2}+1-n, \frac{\nu}{4t}\right)\right). \tag{6.85}$$

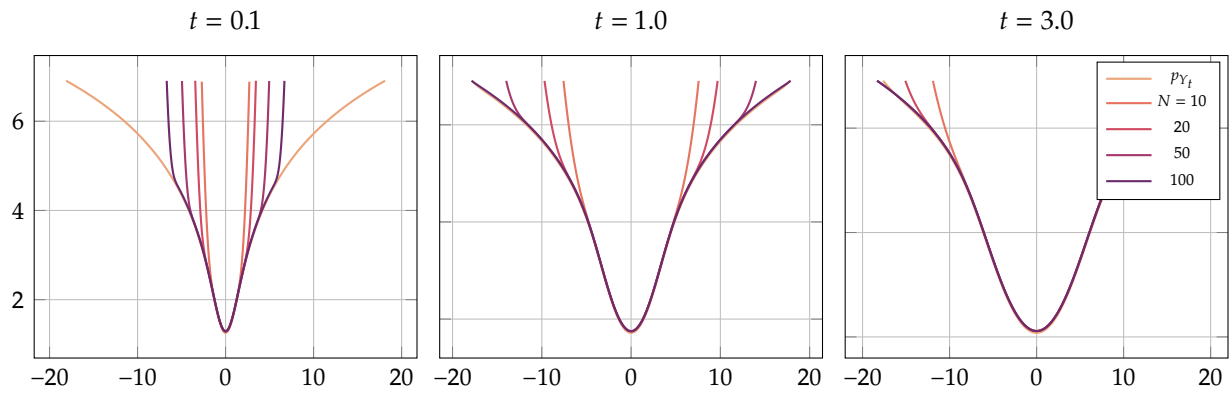

Figure 6.17: Forchini's [84] approximation $-\log p^{(N)}_{Y_t}$ (see (6.85)) of the density of the sum of a t- and a normally distributed random variable with standard deviation $\sqrt{2t}$.

Here, Ψ is the confluent hypergeometric function of the second kind (also known as "Tricomi's function" due to [230], or as "the hypergeometric U function").

We show the potential $-\log p^{(N)}_{Y_t}$ for different N and $t > 0$ in fig. 6.17. We also show $-\log p_{Y_t}$ (i.e. the wanted potential), which we computed numerically by convolving p_X with a Gaussian with appropriate variance. Notice that (6.85) is composed of two terms: A Gaussian with variance $2t$ and an infinite polynomial in the even powers to fill up the tails of the distribution. Thus, it is not surprising that the approximation fails to model the tails of the distribution when t is small, and becomes better as t increases and the density approaches a Gaussian.

Another popular expert function is the GSM

$$x \mapsto \int_{-\infty}^{\infty} (2\pi z^2\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{x^2}{2z^2\sigma^2}\right) p_Z(z)\,\mathrm{d}z \tag{6.86}$$

which has been used in the context of modeling both the distributions of filter- [87, 209] as well as wavelet-responses [190, 239]. Here, p_Z is the *mixing density* of the *multiplier* Z. Thus, GSMs can represent densities of random variables that follow

$$X = ZU \tag{6.87}$$

where Z is a scalar random variable and U is a random variable with normal distribution with zero mean; see [8] for a more rigorous discussion. In practice, the mixing density is usually chosen as a Dirac mixture $p_Z = \sum_{l=1}^{p} w_l \delta_{z_l}$ with $(w_1, w_2, \dots, w_p) \in \Delta^p$ and z_l a-priori fixed. Then, adopting the notation from eq. (6.17), the GSM expert reads

$$\psi_k^{\mathrm{GSM}}(x, w_k, t) = \sum_{l=1}^{p} w_{kl} (2\pi z_l^2(t))^{-\frac{1}{2}} \exp\left(-\frac{x^2}{2z_l^2(t)}\right), \tag{6.88}$$

where without loss of generality we fixed $\sigma = 1$.[25] The notation also reflects that, in the context of the undercomplete model eq. (6.16), it suffices to adapt the variances with diffusion time as $z_l^2 \mapsto z_l^2 + 2t\|f_k\|^2$.[26]

25: There is no loss of generality because σ can be absorbed into the z_l's.

26: The proof is a straight forward adaption of the proof of theorem 6.3.2.

27: Here, the smallest standard deviation, $z_1 = 0.01$, is approximately equal to $\sigma_0 = 0.016$ in the GMM, and we used $p = 20$.

To show the practical merit of this parametrization in our context, we train an undercomplete model on 7×7 patches choosing $z_i = 0.01 \times 1.4^{i-1}$ where $i = 1, 2, \dots, p$.[27] The learned filters, their corresponding potential- and activation functions are shown in fig. 6.18. For 7×7 patches and under our choice of p, the number of learnable parameters is $(7^2 - 1)(7^2 + 20) = 3312$. This is considerably less than the 5376 parameters for the GMM, which, as discussed in section 6.4.7, seems to still be discretized too coarsely. This might indicate that a GSM parametrization is more fit for this purpose. Indeed, the quantitative analysis presented in table 6.1 shows superiority of the patch-based GSM model over the patch-based GMM. However, note that the GMM parametrization is strictly more versatile as it does not assume a maximum at 0. For instance, GSMs can not model the multi-well potential functions of the overcomplete model shown in fig. 6.8.

6.5.3 Going deeper

All models discussed in this chapter are "shallow" in the sense that they are one-layer neural networks. A possible extension of our work would be to consider deeper networks with more than one layer. Popular deep image restoration frameworks, such as trainable non-linear reaction diffusion [46] or the cascade of shrinkage fields [210] use trainable potentials parametrized by a mixture of Gaussian. However, there the Gaussian mixture directly models the potential and usually has no probabilistic interpretation. In addition, they are typically trained as point estimators in a classic discriminative learning setup, and have not been studied in the context of diffusion priors. We note also that trainable non-linear reaction diffusion considers a diffusion in *image space*, whereas our framework considers diffusion in *probability space*. Extending the idea of diffusion in probability space to deep networks is non-trivial and we believe that such models can only be tackled by approximating the diffusion PDE.

In the complete model on wavelet-responses described in section 6.3.2, each expert models the distribution of wavelet coefficients in its sub-band. However—similar to the situation when the complete filter model is applied to whole images—it does not account for the non-trivial correlation of neighboring wavelet coefficient, neither in its own, nor in sibling- or parent-sub-bands. There exist many works that aim to account for this correlation: Guerrero-Colon et al. [99] introduce mixtures of GSMs to model the spatial distribution of wavelet coefficients in and across sub-bands. Gupta at al. [101] extend this idea to mixtures of generalized Gaussian scale model mixtures. We believe that these extensions can be used also in our work: In particular, accounting for the correlation within disjoint neighborhoods leads to a block diagonal structure of the precision which can be efficiently inverted. However, modeling disjoint neighborhoods is known to introduce artifacts [190]. Still, such models can be globalized, e.g. by utilizing ideas similar to the expected patch log-likelihood [260], which amounts to applying a local model to overlapping local neighborhoods individually and averaging the results.

Another interesting research direction with applications to generative modeling would be to condition the distribution of the wavelet coefficients on their parent sub-bands. The wavelet score-based generative model of Guth et al. [103]

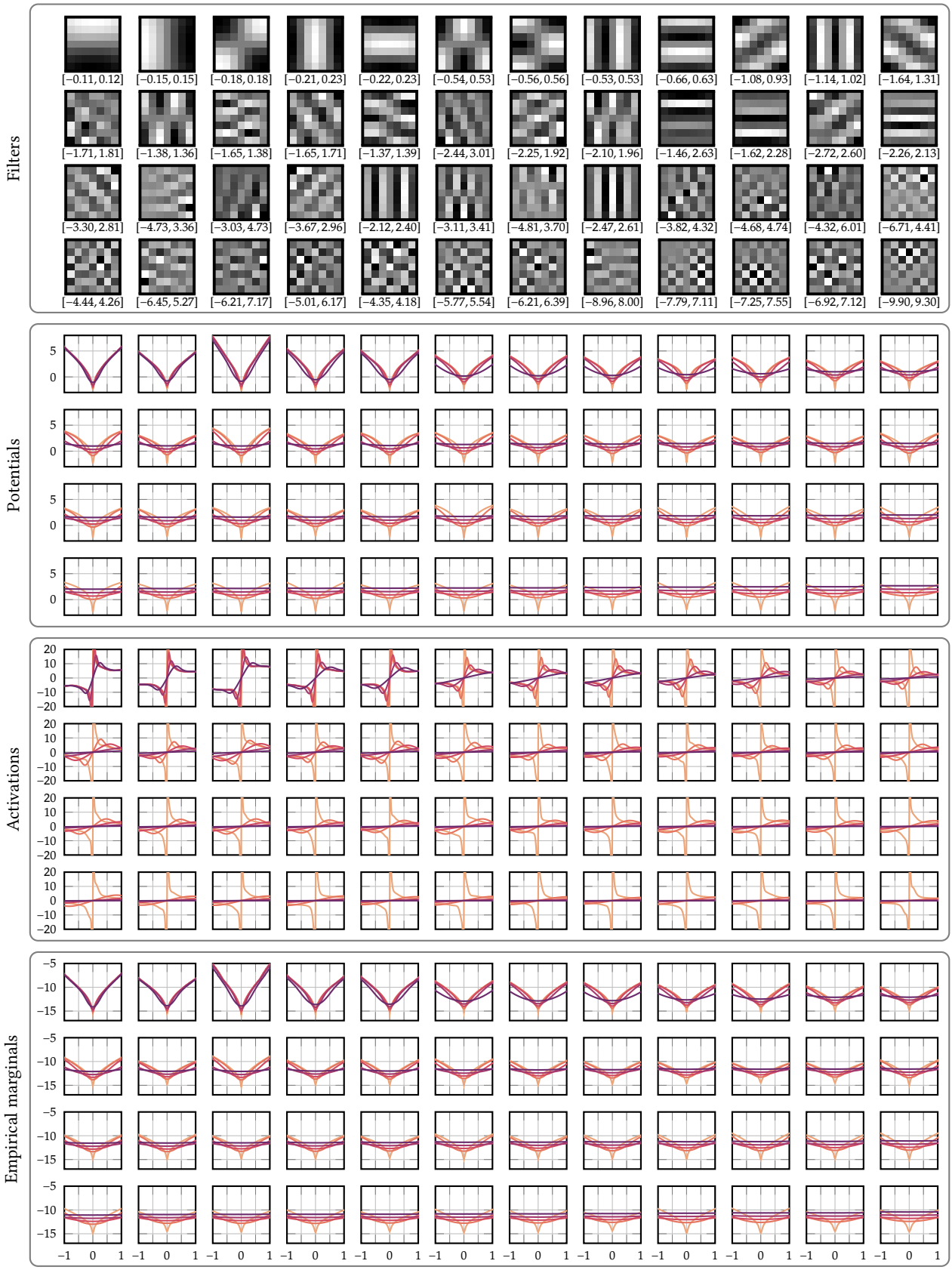

Figure 6.18: The learned undercomplete model utilizing Gaussian scale mixture experts. The colors indicate the diffusion time $\sqrt{2t} = 0$——, 0.025——, 0.05——, 0.1——, 0.2——.

utilizes a conditional diffusion model to modeling local neighborhoods of wavelet coefficients. Their model uses the score network architecture proposed in [173], but we believe that modelling local neighborhoods could yield results that are close to theirs.

6.5.4 Complete versus overcomplete models

We emphasize the difference between the complete model and the overcomplete model. In the complete model the experts "only" model the marginal distribution of the responses of their respective filter due to the orthogonality assumptions. It has been pointed out in the nineties that the distribution of filter responses of natural images is leptokurtic irrespective of the filter [120] and sharply peaked at 0. Our learned potential functions for the complete model in fig. 6.5 reflect these observations.

In contrast, the experts in the overcomplete model do *not* model the marginal distribution of filter responses. Instead, due to the overcomplete structure, it accounts for the non-trivial correlation of overlapping patches. This results in significantly more complex expert functions with multiple minima, sometimes different from zero. This was observed in the context of generative modeling of image priors in the nineties by Zhu and Mumford [255] and in the context of discriminative learning by Chen and Pock [46]. Our learned potential functions of the overcomplete model in fig. 6.8 also reflect these observations.

Although this distinction is well known in the literature [46, 203, 257, 260], it is sometimes overlooked. For example, the choice of the Student-t experts in the FoE model by Roth and Black [206], as well as the Gaussian scale mixture experts in the follow-up paper by Schmidt, Gao, and Roth [209], can be considered deficiencies in this respect.

Unfortunately, the assumption on the filters in the overcomplete model is quite strong; the filters need to be *ideal*. However, ideal filters are necessarily are as large as the image, rendering the construction impractical. Therefore, it is paramount to find a balance between satisfying the assumptions and having useful, compactly supported filters. The optimal filters for this application remain an open question; our choice of the compactly supported non-separable shearlets of Lim [146] is natural but not necessarily optimal. Additionally, determining the scaling parameter of the variance of the experts (ξ_k in theorem 6.3.6) given the non-ideal frequency responses is unclear. Our choice of of the maximal magnitude is likely suboptimal may explain the performance drop of the overcomplete model observed in table 6.1 for high noise levels.

In general, we believe that it is not possible to relax the constraints and still represent the diffusion only by adapting the variances of the one-dimensional experts. However, we think that it is feasible to relax the constraints and derive error bounds within which the diffusion equation is fulfilled. The relaxed constraints might clarify the filters choice, and the error bounds could be tighter than what we achieve with our current choices.

6.6 Conclusion

In this chapter, we introduced PoGMDMs as products of Gaussian mixture experts that allow for an explicit solution of the diffusion PDE by adapting the variance of the one-dimensional experts. For complete and convolutionally-overcomplete models, we derive conditions on the associated filters and experts such that the diffusion PDE is exactly fulfilled. Our explicit formulation enables learning of image priors simultaneously for all diffusion times using denoising score matching. Numerical results demonstrated that PoGMDMs capture the statistics of the underlying distribution well for any diffusion time. As a byproduct, our models can naturally be used for noise estimation and blind heteroscedastic denoising.

Future work will include the design of multi-layer architectures for which the diffusion can be expressed analytically, or approximated within some error bounds. In addition, the learned models will be evaluated on more involved inverse problems such as deblurring or medical imaging. Further, the extensive evaluation of the model based on filter-responses in terms of sampling the distribution and performing heteroscedastic blind denoising can also be applied to the models based on wavelet- and shearlet-responses.

Chapter 7

Conclusion and outlook

In the introduction, we highlighted that many tasks in computer vision and medical imaging can be framed as inverse problems. In such scenarios, the goal is to recover an underlying signal from corrupted observations. Achieving this recovery necessitates a model that describes how the corrupted observation was generated. However, relying solely on this model, several signals could explain an observation equally well, making the recovery problem is ill-posed. Therefore, any signal recovery algorithm must incorporate prior knowledge about the underlying signal. Variational recovery approaches provide a flexible framework for addressing these challenges. In this framework, the observation model is decoupled from the prior knowledge. Prior knowledge is integrated as a regularization term in a minimization problem, whose solution aims to recover the underlying signal.

In chapter 4, we provided a historical overview of regularizers. One of the most classical choice of regularizers is magnitude penalization, which essentially encodes that the underlying signals should have bounded energy. However, this has limited utility in context of inverse problems in imaging. For instance, in Fourier imaging scenarios discussed in our running example, this approach leads to a dimmed version of the naive reconstruction. More sophisticated regularity assumptions include the sparsity of gradients in the signal or wavelet coefficients. However, these regularity assumptions are oversimplified and do not accurately reflect the statistics of the underlying signal.

Modeling these statistics by hand becomes increasingly difficult. To address these challenges, this thesis explores two methods for learning statistical models directly from reference data in the context of inverse problems in imaging. For magnetic resonance imaging, we design a very general deep neural regularizer that encodes nonlocal and translation variant statistics of MRI scans of the human knee. Coupled with a fast nonlinear inversion algorithm, this approach achieves state-of-the-art results for parallel MRI without requiring calibration scans. In contrast, we combined classical modeling techniques from the Markov random field literature with modern ideas from diffusion models. We recover a translation invariant PoGMDM for natural images, that admits a closed form one-step MMSE optimal denoising procedure for Gaussian noise with arbitrary variance.

The two approaches represent opposite ends of the complexity-structure spectrum: The deep neural regularizer is a very general map whereas the PoGMDM

is highly structured and explicitly encodes assumptions about the underlying distribution. Naturally, there is interest in hybrid models that combine aspects of both approaches. For example, the deep neural regularizer could be stripped of some layers and compensated with learnable activation functions as used in the PoGMDM. Conversely, the PoGMDM could be benefit from more complex functions that combine the filter-wise and pixel-wise energies in more sophisticated ways, as opposed to current scalar summation. Effectively, this would place the regularizer in the same class of functions as the isotropic TV.

A different research direction is developing efficient Gibbs-type samplers for PoGMDMs. We believe that an algorithm similar to the auxiliary variable Gibbs sampler in [209] can be used to efficiently sample our model. This would facilitate efficient maximum-likelihood learning and eliminate the need for ideal filters, thereby improving representation abilities. Additionally, refinements of the parametrization of PoGMDMs promise to yield better numerical results. As demonstrated in chapter 4, the applications of PoGMDM extend beyond denoising and we believe that PoGMDMs can be readily plugged into the proposed joint nonlinear inversion algorithm for parallel MRI. Conversely, the joint nonlinear inversion algorithm can integrate ideas from posterior sampling algorithms from diffusion models; we explore this direction with preliminary results in [81].

In summary, in this thesis we discuss principled approaches to utilizing modern generative machine learning approaches in the context of inverse problems in imaging. By adopting a rigorous Bayesian interpretation of inverse problems, finding a good regularizer amounts to fitting a parametric density to reference data. The resulting learned regularizers enjoy great performance in inverse problems due to the data-driven approach while maintaining interpretability through a strict separation of likelihood and prior.

Glossary

ACL auto-calibration lines 127
Adam adaptive moments 58, 59

BSDS Berkeley segmentation data set 81, 103

CLT central limit theorem 27
CORPD coronal proton density weighted 111, 115–117, 127, 133
CORPDFS coronal proton density weighted with fat suppression 111, 112, 115–117, 127, 130, 133
CT computed tomography 1–3, 93

DCT discrete cosine transform 81–83

e.r.v. extended real-valued 42
EBM energy-based model 98
EPLL expected patch log-likelihood 166

FISTA fast iterative shrinkage and thresholding algorithm 53, 54, 82, 87, 89, 90, 114
FoE fields-of-experts 7, 9, 107, 137, 138, 151, 180
FRAME filters, random fields, and maximum entropy 7

GAN generative adversarial network 99
GMM Gaussian mixure model 65, 69, 145–151, 153–156, 158, 159, 166, 174, 176–178
GRAPPA generalized autocalibrating partially parallel acquisitions 96
GSM Gaussian scale mixture 155, 165, 166, 174, 177, 178

i.i.d. independent and identically distributed 28–30
iPALM inertial proximal alternating linearized minimization 56, 95, 96, 98, 102, 110, 112, 113, 115, 118
iPiano inertial proximal algorithm for nonconvex optimization 55, 82
ISTA iterative shrinkage and thresholding algorithm 53

MALA Metropolis adjusted Langevin algorithm 41
MAP maximum a-posteriori 5, 6, 71, 73, 77, 85, 87, 89, 90, 99, 115, 118, 120–124, 127, 128, 134, 166, 167
MCMC Markov chain Monte Carlo 7, 35, 87–89, 107, 115, 119, 138, 139
MMSE minimum mean-squared-error 7, 90, 115, 118, 120–125, 127, 128, 134, 138, 142–144, 159, 160, 165, 166, 175, 176, 183
MRF Markov random field 65, 73, 75, 140, 153
MRI magnetic resonance imaging iii, 1–5, 7–9, 61, 71–74, 77, 81, 88, 93–96, 98, 99, 102, 103, 105, 108, 117, 118, 124, 130, 133–135, 137, 183, 184
MSE mean squared error iii, v, 6, 8, 9, 60–63

NEBLS nonparametric empirical Bayesian least squares 143

NMSE normalized mean-squared error 60–62, 117, 120, 122, 126, 127
NN neural network 65–68, 87, 93, 140

PCA principal component analysis 81, 84, 88, 91
PDE partial differential equation 39, 40, 138, 140–142, 144, 145, 147, 148, 150, 152, 154, 178, 181
PDHG primal-dual hybrid gradient 54, 78, 80, 90
PILS parallel imaging illustrated with the partially parallel imaging with localized sensitivities 95–97
PoGMDM product of Gaussian mixture diffusion model 73, 140, 144, 151, 181, 183, 184
PSNR peak signal-to-noise ratio 6, 61, 62, 80, 90, 91, 100–102, 117, 120, 122, 126, 127, 165, 168

ReLU rectified linear unit 58
RSS root-sum-of-squares 99, 111–116

SDE stochastic differential equation 34, 35, 39, 40, 99
SENSE sensitiviy encoding 95, 96
SMASH simultaneous acquisition of spatial harmonics 96
SSIM structural similarity 62–64, 100, 117, 120, 122, 126, 127, 165, 168

TGV total generalized variation 97
TV total variation 41, 73, 77, 79, 80, 82, 85, 90–92, 94, 97, 109, 112, 120–122, 127–129, 131, 132, 134, 168, 184

ULA unadjusted Langevin algorithm 40

VN variational network 94, 99, 113, 117, 126–128, 131, 132, 137

ZF zero-filled 113, 121, 122, 127, 128, 132

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Dec. 2008. ISBN: 9781400830244. DOI: 10.1515/9781400830244 (cit. on p. 155).

[2] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. "A learning algorithm for Boltzmann machines". In: *Cognitive science* 9.1 (1985), pp. 147–169 (cit. on p. 107).

[3] Jonas Adler and Ozan Öktem. "Deep Bayesian Inversion". In: *arXiv preprint arXiv:1811.05910* (2018) (cit. on p. 77).

[4] Hemant K. Aggarwal, Merry P. Mani, and Mathews Jacob. "MoDL: Model-Based Deep Learning Architecture for Inverse Problems". In: *IEEE Transactions on Medical Imaging* 38.2 (2019), pp. 394–405. DOI: 10.1109/TMI.2018.2865356 (cit. on p. 96).

[5] Mehmet Akçakaya et al. "Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: Database-free deep learning for fast imaging". In: *Magnetic Resonance in Medicine*. 2019 (cit. on p. 94).

[6] Ronald W. Schafer Alan V. Oppenheim. *Discrete-Time Signal Processing (3rd Edition)*. 3rd. Prentice Hall, 2009. ISBN: 0131988425; 9780131988422 (cit. on pp. 152, 153).

[7] Hans Wilhelm Alt. *Linear Functional Analysis*. Springer London, 2016. ISBN: 9781447172802. DOI: 10.1007/978-1-4471-7280-2 (cit. on pp. 11, 15).

[8] D. F. Andrews and C. L. Mallows. "Scale Mixtures of Normal Distributions". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 36.1 (Sept. 1974), pp. 99–102. ISSN: 1467-9868. DOI: 10.1111/j.2517-6161.1974.tb00989.x (cit. on p. 177).

[9] Vegard Antun et al. "On instabilities of deep learning in image reconstruction and the potential costs of AI". In: *Proc. of the National Academy of Sciences* 117.48 (May 2020), pp. 30088–30095. DOI: 10.1073/pnas.1907377117 (cit. on pp. 95, 98).

[10] Simon Arridge et al. "Solving inverse problems using data-driven models". In: *Acta Numerica* 28 (2019), pp. 1–174. DOI: 10.1017/S0962492919000059 (cit. on p. 8).

[11] Hédy Attouch et al. "Proximal Alternating Minimization and Projection Methods for Nonconvex Problems: An Approach Based on the Kurdyka-Łojasiewicz Inequality". In: *Mathematics of Operations Research* 35.2 (2010), pp. 438–457. ISSN: 0364765X, 15265471 (cit. on p. 55).

[12] Sheldon Axler. *Linear algebra done right*. en. 4th ed. Undergraduate texts in mathematics. Cham, Switzerland: Springer International Publishing, Nov. 2023 (cit. on pp. 42, 43, 48).

[13] George Bachman. *Functional Analysis*. 2nd ed. Dover Books on Mathematics. Mineola, NY: Dover Publications, Jan. 1998 (cit. on p. 46).

[14] Frank Bauer and Stephan Kannengiesser. "An alternative approach to the image reconstruction for parallel data acquisition in MRI". In: *Mathematical Methods in the Applied Sciences* 30.12 (2007), pp. 1437–1451. DOI: https://doi.org/10.1002/mma.848 (cit. on pp. 97, 108).

[15] Amir Beck. *First-Order Methods in Optimization*. Philadelphia, PA: Society for Industrial & Applied Mathematics, 2017. DOI: 10.1137/1.9781611974997 (cit. on pp. 11, 42, 44, 49–51, 53).

[16] Amir Beck and Marc Teboulle. "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". In: *SIAM Journal on Imaging Sciences* 2.1 (2009), pp. 183–202. DOI: 10.1137/080716542 (cit. on pp. 53, 54, 56, 82).

[17] Martin Benning and Martin Burger. "Modern regularization methods for inverse problems". In: *Acta Numerica* 27 (2018), pp. 1–111. DOI: 10.1017/S0962492918000016 (cit. on p. 5).

[18] Christian Berg and Christophe Vignat. "On the density of the sum of two independent Student t-random vectors". In: *Statistics & Probability Letters* 80.13 (2010), pp. 1043–1055. ISSN: 0167-7152 (cit. on p. 176).

[19] Dimitri P. Bertsekas. "Incremental Gradient, Subgradient, and Proximal Methods for Convex Optimization: A Survey". In: *Optimization for Machine Learning*. The MIT Press, Sept. 2011. ISBN: 9780262298773. DOI: 10.7551/mitpress/8996.003.0006 (cit. on pp. 58, 59).

[20] Patrick Billingsley. *Probability and Measure*. en. 3rd ed. Wiley Series in Probability & Mathematical Statistics: Probability & Mathematical Statistics. Nashville, TN: John Wiley & Sons, May 1995 (cit. on p. 27).

[21] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738 (cit. on pp. 65, 70, 72).

[22] Martin Blaimer et al. "SMASH, SENSE, PILS, GRAPPA: How to Choose the Optimal Method". In: *Topics in Magnetic Resonance Imaging* 15.4 (Aug. 2004), pp. 223–236. ISSN: 0899-3459. DOI: 10.1097/01.rmr.0000136558.09801.dd (cit. on pp. 95, 96).

[23] Lea Bogensperger et al. "Learned Discretization Schemes for the Second-Order Total Generalized Variation". In: *Proc. of the International Conference on Scale Space and Variational Methods in Computer Vision*. Ed. by Luca Calatroni et al. Cham: Springer International Publishing, 2023, pp. 484–497. ISBN: 978-3-031-31975-4 (cit. on p. 77).

[24] Pakshal Narendra Bohra. "Statistical Inference for Inverse Problems: From Sparsity-Based Methods to Neural Networks". en. PhD thesis. École Polytechnique Fédérale de Lausanne, 2024. DOI: 10.5075/EPFL-THESIS-9824 (cit. on pp. 7, 77).

[25] Jérôme Bolte et al. "Nonsmooth Implicit Differentiation for Machine-Learning and Optimization". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 13537–13549 (cit. on p. 69).

[26] Ashish Bora et al. "Compressed Sensing using Generative Models". In: *Proc. of the International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, Aug. 2017, pp. 537–546 (cit. on p. 99).

[27] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004 (cit. on p. 46).

[28] James P. Boyle and Richard L. Dykstra. "A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Spaces". In: *Advances in Order Restricted Statistical Inference*. New York: Springer New York, 1986, pp. 28–47 (cit. on p. 156).

[29] Kristian Bredies, Karl Kunisch, and Thomas Pock. "Total Generalized Variation". In: *SIAM Journal on Imaging Sciences* 3.3 (Jan. 2010), pp. 492–526. ISSN: 1936-4954. DOI: 10.1137/090769521 (cit. on p. 5).

[30] Kristian Bredies and Dirk Lorenz. *Mathematical Image Processing*. Springer International Publishing, 2018. ISBN: 9783030014582. DOI: 10.1007/978-3-030-01458-2 (cit. on pp. 16, 23, 24, 151).

[31] Steve Brooks et al. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, May 2011. ISBN: 9780429138508. DOI: 10.1201/b10905 (cit. on p. 40).

[32] Steve Brooks et al. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, 2011. ISBN: 9781420079418 (cit. on p. 115).

[33] Antonin Chambolle. "An Algorithm for Total Variation Minimization and Applications". In: *Journal of Mathematical Imaging and Vision* 20.1 (Jan. 1, 2004), pp. 89–97. ISSN: 1573-7683 (cit. on pp. 75, 76).

[34] Antonin Chambolle, Stacey E. Levine, and Bradley J. Lucier. "An Upwind Finite-Difference Method for Total Variation–Based Image Smoothing". In: *SIAM Journal on Imaging Sciences* 4.1 (2011), pp. 277–299. DOI: 10.1137/090752754 (cit. on p. 76).

[35] Antonin Chambolle and Pierre-Louis Lions. "Image recovery via total variation minimization and related problems". In: *Numerische Mathematik* 76.2 (Apr. 1997), pp. 167–188. ISSN: 0945-3245. DOI: 10.1007/s002110050258 (cit. on p. 77).

[36] Antonin Chambolle and Thomas Pock. "A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging". In: *Journal of Mathematical Imaging and Vision* 40.1 (Dec. 2010), pp. 120–145. ISSN: 0924-9907, 1573-7683. DOI: 10.1007/s10851-010-0251-1 (cit. on pp. 5, 54, 55, 75, 78, 90).

[37] Antonin Chambolle and Thomas Pock. "An introduction to continuous optimization for imaging". In: *Acta Numerica* 25 (2016), pp. 161–319. DOI: 10.1017/S096249291600009X (cit. on p. 75).

[38] Antonin Chambolle and Thomas Pock. "Learning Consistent Discretizations of the Total Variation". In: *SIAM Journal on Imaging Sciences* 14.2 (2021), pp. 778–813. DOI: 10.1137/20M1377199 (cit. on p. 77).

[39] Antonin Chambolle and Thomas Pock. "Total roto-translational variation". In: *Numerische Mathematik* 142.3 (Mar. 2019), pp. 611–666. ISSN: 0945-3245. DOI: 10.1007/s00211-019-01026-w (cit. on p. 80).

[40] Antonin Chambolle et al. "Nonlinear Wavelet Image Processing: Variational Problems, Compression, and Noise Removal Through Wavelet Shrinkage". In: *IEEE Transactions on Image Processing* 7.3 (Mar. 1998), pp. 319–335. ISSN: 1941-0042 (cit. on p. 175).

[41] S.Grace Chang, Bin Yu, and Martin Vetterli. "Adaptive Wavelet thresholding for image denoising and compression". In: *IEEE Transactions on Image Processing* 9.9 (2000), pp. 1532–1546 (cit. on p. 175).

[42] Pierre Charbonnier et al. "Deterministic edge-preserving regularization in computed imaging". In: *IEEE Transactions on Image Processing* 6.2 (1997), pp. 298–311. DOI: 10.1109/83.551699 (cit. on p. 77).

[43] Pierre Charbonnier et al. "Two deterministic half-quadratic regularization algorithms for computed imaging". In: *Proc. of the International Conference on Image Processing*. Vol. 2. 1994, 168–172 vol.2. DOI: 10.1109/ICIP.1994.413553 (cit. on pp. 77, 80).

[44] Baiyu Chen et al. "SparseCT: System Concept and Design of Multislit Collimators". In: *Medical Physics* 46.6 (May 2019), pp. 2589–2599. DOI: 10.1002/mp.13544 (cit. on p. 3).

[45] Guang-Hong Chen, Jie Tang, and Shuai Leng. "Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets: Prior image constrained compressed sensing (PICCS)". In: *Medical Physics* 35.2 (Jan. 2008), pp. 660–663. ISSN: 0094-2405. DOI: 10.1118/1.2836423 (cit. on p. 3).

[46] Yunjin Chen and Thomas Pock. "Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (June 1, 2017), pp. 1256–1272. ISSN: 0162-8828, 2160-9292 (cit. on pp. 81, 87, 137, 178, 180).

[47] Yunjin Chen and Thomas Pock. "Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017), pp. 1256–1272. DOI: 10.1109/TPAMI.2016.2596743 (cit. on pp. 94, 155).

[48] H.D. Cheng et al. "Color image segmentation: advances and prospects". In: *Pattern Recognition* 34.12 (2001), pp. 2259–2281. ISSN: 0031-3203. DOI: https://doi.org/10.1016/S0031-3203(00)00149-7 (cit. on p. 1).

[49] Jing Cheng et al. "Model Learning: Primal Dual Networks for Fast MR Imaging". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*. Ed. by Dinggang Shen et al. Cham: Springer International Publishing, 2019, pp. 21–29. ISBN: 978-3-030-32248-9 (cit. on p. 94).

[50] Xiang Cheng and Peter Bartlett. "Convergence of Langevin MCMC in KL-divergence". In: *Proceedings of Algorithmic Learning Theory*. Ed. by Firdaus Janoos, Mehryar Mohri, and Karthik Sridharan. Vol. 83. Proceedings of Machine Learning Research. PMLR, Apr. 2018, pp. 186–211 (cit. on p. 41).

[51] Hugh A. Chipman, Eric D. Kolaczyk, and Robert E. McCulloch. "Adaptive Bayesian Wavelet Shrinkage". In: *Journal of the American Statistical Association* 92.440 (1997), pp. 1413–1421. ISSN: 01621459 (cit. on p. 175).

[52] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. "Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 12413–12422 (cit. on p. 124).

[53] Hyungjin Chung and Jong Chul Ye. "Score-based diffusion models for accelerated MRI". In: *Medical Image Analysis* 80 (2022), p. 102479. ISSN: 1361-8415. DOI: 10.1016/j.media.2022.102479 (cit. on pp. 66, 94, 99, 104, 111, 113, 116, 122–124, 134).

[54] Hyungjin Chung et al. "Diffusion Posterior Sampling for General Noisy Inverse Problems". In: *Proc. of the International Conference on Learning Representations*. 2023 (cit. on pp. 99, 104, 166).

[55] Hyungjin Chung et al. "Solving 3D Inverse Problems using Pre-trained 2D Diffusion Models". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023 (cit. on p. 104).

[56] M. Cimpoi et al. "Describing Textures in the Wild". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2014 (cit. on p. 103).

[57] Merlise Clyde, Giovanni Parmigiani, and Brani Vidakovic. "Multiple Shrinkage and Subset Selection in Wavelets". In: *Biometrika* 85.2 (1998), pp. 391–401. ISSN: 00063444 (cit. on p. 175).

[58] Kevin Cole et al. *Heat Conduction Using Greens Functions*. Boca Raton, FL: CRC Press, 2010 (cit. on p. 141).

[59] Patrick L. Combettes and Valérie R. Wajs. "Signal Recovery by Proximal Forward-Backward Splitting". In: *Multiscale Modeling & Simulation* 4.4 (2005), pp. 1168–1200. DOI: 10.1137/050626090 (cit. on p. 53).

[60] Laurent Condat. "Discrete Total Variation: New Definition and Minimization". In: *SIAM Journal on Imaging Sciences* 10.3 (2017), pp. 1258–1290. DOI: 10.1137/16M1075247 (cit. on pp. 77, 79).

[61] Laurent Condat. "Fast projection onto the simplex and the ℓ_1 ball". In: *Mathematical Programming* 158.1-2 (Sept. 2015), pp. 575–585 (cit. on p. 161).

[62] Shuang Cong and Yang Zhou. "A review of convolutional neural network architectures and their optimizations". In: *Artificial Intelligence Review* 56.3 (June 2022), pp. 1905–1969. ISSN: 1573-7462. DOI: 10.1007/s10462-022-10213-5 (cit. on p. 67).

[63] Fergal Cotter. "Uses of Complex Wavelets in Deep Convolutional Neural Networks". PhD thesis. University of Cambridge, 2020 (cit. on p. 159).

[64] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, Apr. 2005. ISBN: 9780471748823. DOI: 10.1002/047174882x (cit. on pp. 31, 32, 86).

[65] Matthew S. Crouse, Robert D. Nowak, and Richard G. Baraniuk. "Wavelet-based statistical signal processing using hidden Markov models". In: *IEEE Transactions on Signal Processing* 46.4 (Apr. 1998), pp. 886–902. ISSN: 1053587X (cit. on p. 175).

[66] Imre Csiszar. "Why Least Squares and Maximum Entropy? An Axiomatic Approach to Inference for Linear Inverse Problems". In: *The Annals of Statistics* 19.4 (Dec. 1991). ISSN: 0090-5364. DOI: 10.1214/aos/1176348385 (cit. on p. 31).

[67] Arthur L. da Cunha, Jianping Zhou, and Minh N. Do. "The Nonsubsampled Contourlet Transform: Theory, Design, and Applications". In: *IEEE Transactions on Image Processing* 15.10 (2006), pp. 3089–3101 (cit. on p. 161).

[68] Mohammad Zalbagi Darestani, Akshay S Chaudhari, and Reinhard Heckel. "Measuring robustness in deep learning based compressive sensing". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 2433–2444 (cit. on p. 96).

[69] Ingrid Daubechies, Michel Defrise, and Christine De Mol. "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint". In: *Communications on Pure and Applied Mathematics* 57.11 (Aug. 2004), pp. 1413–1457. ISSN: 1097-0312. DOI: 10.1002/cpa.20042 (cit. on pp. 53, 77).

[70] Dimitri P. Bertsekas Dimitri P. Bertsekas. *Nonlinear programming*. 2nd. Athena Scientific, 1999. ISBN: 1886529000; 9781886529007 (cit. on p. 56).

[71] David L. Donoho. "Compressed Sensing". In: *IEEE Transactions on Information Theory* 52.4 (2006), pp. 1289–1306. DOI: 10.1109/TIT.2006.871582 (cit. on p. 94).

[72] David L. Donoho. "De-noising by soft-thresholding". In: *IEEE Transactions on Information Theory* 41.3 (1995), pp. 613–627 (cit. on p. 175).

[73] David L. Donoho and Iain M. Johnstone. "Adapting to Unknown Smoothness via Wavelet Shrinkage". In: *Journal of the American Statistical Association* 90.432 (1995), pp. 1200–1224. ISSN: 01621459 (cit. on p. 175).

[74] David L. Donoho and Iain M. Johnstone. "Ideal Spatial Adaptation by Wavelet Shrinkage". In: *Biometrika* 81.3 (1994), pp. 425–455 (cit. on p. 175).

[75] Yilun Du and Igor Mordatch. "Implicit Generation and Modeling with Energy Based Models". In: *Proc. of the Conferene on Neural Information Processing Systems*. Vol. 32. Red Hook, NY, USA: Curran Associates, Inc., Nov. 6, 2019 (cit. on pp. 107, 114).

[76] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. "Activation functions in deep learning: A comprehensive survey and benchmark". In: *Neurocomputing* 503 (2022), pp. 92–108. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2022.06.111 (cit. on p. 68).

[77] Alain Durmus and Éric Moulines. "High-dimensional Bayesian inference via the unadjusted Langevin algorithm". In: *Bernoulli* 25.4A (Nov. 2019). ISSN: 1350-7265. DOI: 10.3150/18-bej1073 (cit. on p. 41).

[78] Alain Durmus and Éric Moulines. "Nonasymptotic convergence analysis for the unadjusted Langevin algorithm". In: *The Annals of Applied Probability* 27.3 (June 2017). ISSN: 1050-5164. DOI: 10.1214/16-aap1238 (cit. on p. 41).

[79] Alain Durmus, Éric Moulines, and Marcelo Pereyra. "Efficient Bayesian Computation by Proximal Markov Chain Monte Carlo: When Langevin Meets Moreau". In: *SIAM Journal on Imaging Sciences* 11.1 (2018), pp. 473–506. DOI: 10.1137/16M1108340 (cit. on pp. 41, 107).

[80] Bradley Efron. "Tweedie's Formula and Selection Bias". In: *Journal of the American Statistical Association* 106.496 (2011), pp. 1602–1614 (cit. on p. 142).

[81] Moritz Erlacher and Martin Zach. "Joint non-linear MRI inversion with diffusion priors". In: *Proceedings of the AAPR Workshop 2023: Patterns in One Health*. Graz: Verlag der Technischen Universität Graz, 2023, to appear in (cit. on pp. 94, 184).

[82] Tim van Erven and Peter Harremos. "Rényi Divergence and Kullback-Leibler Divergence". In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 3797–3820. DOI: 10.1109/TIT.2014.2320500 (cit. on p. 31).

[83] Berthy T. Feng et al. *Score-Based Diffusion Models as Principled Priors for Inverse Imaging*. 2023 (cit. on p. 99).

[84] G. Forchini. "The distribution of the sum of a normal and a t random variable with arbitrary degrees of freedom". In: *Metron - International Journal of Statistics* 0.2 (2008), pp. 205–208 (cit. on pp. 176, 177).

[85] W. T. Freeman and Y. Weiss. "What makes a good model of natural images?" In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Los Alamitos, CA, USA: IEEE Computer Society, June 2007, pp. 1–8 (cit. on pp. 106, 107).

[86] Yoav Freund and David Haussler. "Unsupervised learning of distributions on binary vectors using two layer networks". In: *Advances in neural information processing systems* 4 (1991) (cit. on p. 107).

[87] Qi Gao and Stefan Roth. "How Well Do Filter-Based MRFs Model Natural Images?" In: *Pattern Recognition*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 62–72. ISBN: 978-3-642-32717-9 (cit. on p. 177).

[88] S. Geman and D. Geman. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6.6 (1984), pp. 721–741 (cit. on p. 75).

[89] S. Geman and D.E. McClure. "Bayesian image analysis: An application to single photon emission tomography". In: *Proceedings of the American Statistical Association*. 1985 (cit. on p. 79).

[90] Stuart Geman and Donald Geman. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6.6 (1984), pp. 721–741. DOI: 11.1109/TPAMI.1984.4767596 (cit. on p. 105).

[91] Paul Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer New York, 2003. ISBN: 9780387216171. DOI: 10.1007/978-0-387-21617-1 (cit. on p. 39).

[92] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016 (cit. on p. 74).

[93] Thomas Grandits and Thomas Pock. "Optimizing Wavelet Bases for Sparser Representations". In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Cham: Springer International Publishing, 2018, pp. 249–262. ISBN: 978-3-319-78199-0 (cit. on p. 158).

[94] Rémi Gribonval. "Should Penalized Least Squares Regression be Interpreted as Maximum A Posteriori Estimation?" In: *IEEE Transactions on Signal Processing* 59.5 (2011), pp. 2405–2410. DOI: 10.1109/TSP.2011.2107908 (cit. on pp. 7, 77).

[95] Rémi Gribonval, Volkan Cevher, and Mike E. Davies. "Compressible Distributions for High-Dimensional Statistics". In: *IEEE Transactions on Information Theory* 58.8 (2012), pp. 5016–5034. DOI: 10.1109/TIT.2012.2197174 (cit. on p. 77).

[96] Mark A. Griswold et al. "Generalized autocalibrating partially parallel acquisitions (GRAPPA)". In: *Magnetic Resonance in Medicine* 47.6 (2002), pp. 1202–1210. DOI: https://doi.org/10.1002/mrm.10171 (cit. on p. 96).

[97] Mark A. Griswold et al. "Partially parallel imaging with localized sensitivities (PILS)". In: *Magnetic Resonance in Medicine* 44.4 (2000), pp. 602–609. DOI: https://doi.org/10.1002/1522-2594(200010)44:4<602::AID-MRM14>3.0.CO;2-5 (cit. on pp. 95, 97).

[98] Yu Guan et al. "Magnetic resonance imaging reconstruction using a deep energy-based model". In: *NMR in Biomedicine* 36.3 (Nov. 2022). DOI: 10.1002/nbm.4848 (cit. on pp. 98, 114).

[99] Jose A. Guerrero-Colon, Eero P. Simoncelli, and Javier Portilla. "Image denoising using mixtures of Gaussian scale mixtures". In: *2008 15th IEEE International Conference on Image Processing*. 2008, pp. 565–568 (cit. on p. 178).

[100] Kanghui Guo and Demetrio Labate. "Optimally Sparse Multidimensional Representation Using Shearlets". In: *SIAM Journal on Mathematical Analysis* 39.1 (Jan. 2007), pp. 298–318. ISSN: 1095-7154. DOI: 10.1137/060649781 (cit. on p. 154).

[101] Praful Gupta et al. "Generalized Gaussian scale mixtures: A model for Wavelet coefficients of natural images". In: *Signal Processing: Image Communication* 66 (2018), pp. 87–94. ISSN: 0923-5965 (cit. on p. 178).

[102] Allan Gut. *An Intermediate Course in Probability*. New York: Springer, 2009 (cit. on pp. 85, 148).

[103] Florentin Guth et al. "Wavelet Score-Based Generative Modeling". In: *Proc. of the Conferene on Neural Information Processing Systems*. 2022 (cit. on p. 178).

[104] Andreas Habring, Martin Holler, and Thomas Pock. *Subgradient Langevin Methods for Sampling from Non-smooth Potentials*. 2024 (cit. on p. 41).

[105] Jacques Hadamard and Philip M. Morse. "Lectures on Cauchy's Problem in Linear Partial Differential Equations". In: *Physics Today* 6.8 (Aug. 1953), pp. 18–18. ISSN: 0031-9228. DOI: 10.1063/1.3061337 (cit. on p. 2).

[106] Ernst Hairer, Wanner Gerhard, and Syvert P. Norsett. *Solving Ordinary Differential Equations I. Nonstiff problems*. Springer Berlin Heidelberg, 1993. ISBN: 9783540788621. DOI: 10.1007/978-3-540-78862-1 (cit. on p. 39).

[107] Kerstin Hammernik et al. "Learning a variational network for reconstruction of accelerated MRI data". In: *Magnetic Resonance in Medicine* 79.6 (Nov. 2017), pp. 3055–3071. DOI: 10.1002/mrm.26977 (cit. on pp. 94, 96, 100).

[108] Yoseo Han, Leonard Sunwoo, and Jong Chul Ye. "k -Space Deep Learning for Accelerated MRI". In: *IEEE Transactions on Medical Imaging* 39.2 (2020), pp. 377–386. DOI: 10.1109/TMI.2019.2927101 (cit. on p. 100).

[109] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York, 2009. ISBN: 9780387848587. DOI: 10.1007/978-0-387-84858-7 (cit. on p. 77).

[110] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57.1 (Apr. 1970), pp. 97–109. ISSN: 0006-3444. DOI: 10.1093/biomet/57.1.97 (cit. on p. 41).

[111] Nicolas Heess, Christopher Williams, and Geoffrey Hinton. "Learning Generative Texture Models with extended Fields-of-Experts." In: Jan. 2009. DOI: 10.5244/C.23.115 (cit. on pp. 138, 145).

[112] Michael Held, Philip Wolfe, and Harlan P. Crowder. "Validation of subgradient optimization". In: *Mathematical Programming* 6.1 (Dec. 1974), pp. 62–88 (cit. on pp. 154, 161).

[113] J. Hennig, A. Nauerth, and H. Friedburg. "RARE imaging: A fast imaging method for clinical MR". In: *Magnetic Resonance in Medicine* 3.6 (Dec. 1986), pp. 823–833. ISSN: 1522-2594. DOI: 10.1002/mrm.1910030602 (cit. on p. 95).

[114] Johannes Hertrich, Sebastian Neumayer, and Gabriele Steidl. "Convolutional proximal neural networks and Plug-and-Play algorithms". In: *Linear Algebra and its Applications* 631 (2021), pp. 203–234. ISSN: 0024-3795. DOI: https://doi.org/10.1016/j.laa.2021.09.004 (cit. on p. 155).

[115] Geoffrey E Hinton et al. "The "wake-sleep" algorithm for unsupervised neural networks". In: *Science* 268.5214 (1995), pp. 1158–1161 (cit. on p. 107).

[116] Geoffrey E. Hinton. "Training Products of Experts by Minimizing Contrastive Divergence". In: *Neural Computation* 14.8 (2002), pp. 1771–1800 (cit. on pp. 75, 106, 107).

[117] Geoffrey E. Hinton and Yee-Whye Teh. "Discovering Multiple Constraints That Are Frequently Approximately Satisfied". In: *Proc. of the Seventeenth Conference on Uncertainty in Artificial Intelligence*. Seattle, Washington: Morgan Kaufmann Publishers Inc., 2001, pp. 227–234. ISBN: 1558608001 (cit. on p. 176).

[118] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising Diffusion Probabilistic Models". In: *Proc. of the Conferene on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, Inc., 2020 (cit. on p. 139).

[119] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985 (cit. on p. 155).

[120] J. Huang and D. Mumford. "Statistics of natural images and models". In: *Proc. of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*. Vol. 1. 1999, 541–547 Vol. 1 (cit. on pp. 76, 77, 79, 180).

[121] Junhwa Hur and Stefan Roth. "Optical Flow Estimation in the Deep Learning Age". In: *Modelling Human Motion: From Human Perception to Robot Design*. Ed. by Nicoletta Noceti, Alessandra Sciutti, and Francesco Rea. Cham: Springer International Publishing, 2020, pp. 119–140. ISBN: 978-3-030-46732-6. DOI: 10.1007/978-3-030-46732-6_7 (cit. on p. 1).

[122] Samuel Hurault et al. "Convergent Plug-and-Play with Proximal Denoiser and Unconstrained Regularization Parameter". In: *Journal of Mathematical Imaging and Vision* (June 2024). ISSN: 1573-7683. DOI: 10.1007/s10851-024-01195-w (cit. on p. 3).

[123] Aapo Hyvärinen. "Estimation of Non-Normalized Statistical Models by Score Matching". In: *Journal of Machine Learning Research* 6.24 (2005), pp. 695–709 (cit. on pp. 7, 33, 139).

[124] Peter M. Jakob et al. "AUTO-SMASH: A self-calibrating technique for SMASH imaging". In: *Magma: Magnetic Resonance Materials in Physics, Biology, and Medicine* 7.1 (Nov. 1998), pp. 42–54. ISSN: 1352-8661. DOI: 10.1007/bf02592256 (cit. on p. 96).

[125] Ajil Jalal et al. "Robust Compressed Sensing MRI with Deep Generative Priors". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 14938–14954 (cit. on pp. 96, 99, 104).

[126] Maarten Jansen, Maurits Malfait, and Adhemar Bultheel. "Generalized cross validation for Wavelet thresholding". In: *Signal Processing* 56.1 (1997), pp. 33–44. ISSN: 0165-1684 (cit. on p. 175).

[127] E. T. Jaynes. *Probability Theory: The Logic of Science*. Ed. by G. Larry Bretthorst. Cambridge University Press, 2003 (cit. on pp. 6, 142).

[128] Tero Karras et al. "Elucidating the Design Space of Diffusion-Based Generative Models". In: *Proc. of the Conferene on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, Inc., 2022 (cit. on pp. 3, 124, 165).

[129] B. Kawar, G. Vaksman, and M. Elad. "Stochastic Image Denoising by Sampling from the Posterior Distribution". In: *Proc. of the IEEE International Conference on Computer Vision Workshops*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2021, pp. 1866–1875 (cit. on pp. 165, 166, 168, 172).

[130] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *Proc. of the International Conference on Learning Representations*. Ed. by Yoshua Bengio and Yann LeCun. 2015 (cit. on p. 58).

[131] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *Proc. of the International Conference on Learning Representations*. Ed. by Yoshua Bengio and Yann LeCun. 2015 (cit. on p. 59).

[132] Achim Klenke. *Probability Theory: A Comprehensive Course*. Springer London, 2014. ISBN: 9781447153610. DOI: 10.1007/978-1-4471-5361-0 (cit. on pp. 11, 17–19, 21, 25, 29, 35–38).

[133] Florian Knoll et al. "fastMRI: A Publicly Available Raw k-Space and DICOM Dataset of Knee Images for Accelerated MR Image Reconstruction Using Machine Learning". In: *Radiology: Artificial Intelligence* 2.1 (2020) (cit. on p. 111).

[134] Florian Knoll et al. "Parallel imaging with nonlinear reconstruction using variational penalties". In: *Magnetic Resonance in Medicine* 67.1 (June 2011), pp. 34–41. DOI: 10.1002/mrm.22964 (cit. on pp. 4, 94, 97, 98).

[135] Erich Kobler and Thomas Pock. *Learning Gradually Non-convex Image Priors Using Score Matching*. 2023 (cit. on pp. 143, 166).

[136] Erich Kobler et al. "Variational Networks: Connecting Variational Methods and Deep Learning". In: *Lect. Notes Comput. Sci.* Springer International Publishing, 2017, pp. 281–293. DOI: 10.1007/978-3-319-66709-6_23 (cit. on p. 94).

[137] Thomas Koesters et al. "SparseCT: interrupted-beam acquisition and sparse reconstruction for radiation dose reduction". In: *SPIE Proceedings*. Ed. by Thomas G. Flohr, Joseph Y. Lo, and Taly Gilat Schmidt. SPIE, Mar. 2017. DOI: 10.1117/12.2255522 (cit. on p. 3).

[138] Ken Kreutz-Delgado. *The Complex Gradient Operator and the CR-Calculus*. 2009. DOI: 10.48550/ARXIV.0906.4835 (cit. on p. 109).

[139] Gitta Kutyniok and Demetrio Labate, eds. *Shearlets*. Boston: Birkhäuser, 2012 (cit. on p. 153).

[140] Gitta Kutyniok and Wang-Q Lim. "Compactly supported shearlets are optimally sparse". In: *Journal of Approximation Theory* 163.11 (2011), pp. 1564–1589. ISSN: 0021-9045. DOI: https://doi.org/10.1016/j.jat.2011.06.005 (cit. on p. 154).

[141] Gitta Kutyniok, Wang-Q Lim, and Rafael Reisenhofer. "ShearLab 3D: Faithful Digital Shearlet Transforms Based on Compactly Supported Shearlets". In: *ACM Transactions on Mathematical Software* 42.1 (Jan. 2016). ISSN: 0098-3500 (cit. on pp. 24, 161).

[142] Carole Lazarus. "Compressed Sensing in MRI: optimization-based design of k-space filling curves for accelerated MRI." PhD thesis. University of Paris-Saclay, France, 2018 (cit. on p. 117).

[143] E L Lehmann and George Casella. *Theory of Point Estimation*. en. 2nd ed. Springer Texts in Statistics. New York, NY: Springer, Dec. 1998 (cit. on p. 33).

[144] Victor Lempitsky, Andrea Vedaldi, and Dmitry Ulyanov. "Deep Image Prior". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9446–9454. DOI: 10.1109/CVPR.2018.00984 (cit. on p. 99).

[145] Hao Li et al. "Visualizing the Loss Landscape of Neural Nets". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018 (cit. on p. 118).

[146] Wang-Q Lim. "Nonseparable Shearlet Transform". In: *IEEE Transactions on Image Processing* 22.5 (2013), pp. 2056–2065 (cit. on pp. 24, 153, 154, 159, 180).

[147] Wang-Q Lim. "The Discrete Shearlet Transform: A New Directional Transform and Compactly Supported Shearlet Frames". In: *IEEE Transactions on Image Processing* 19.5 (2010), pp. 1166–1180. DOI: 10.1109/TIP.2010.2041410 (cit. on pp. 153, 154).

[148] J. Lin. "Divergence measures based on the Shannon entropy". In: *IEEE Transactions on Information Theory* 37.1 (1991), pp. 145–151. DOI: 10.1109/18.61115 (cit. on p. 31).

[149] William A. Link and Mitchell J. Eaton. "On thinning of chains in MCMC". In: *Methods in Ecology and Evolution* 3.1 (June 2011), pp. 112–115. ISSN: 2041-210X. DOI: 10.1111/j.2041-210x.2011.00131.x (cit. on p. 115).

[150] P. L. Lions and B. Mercier. "Splitting Algorithms for the Sum of Two Nonlinear Operators". In: *SIAM Journal on Numerical Analysis* 16.6 (1979), pp. 964–979. DOI: 10.1137/0716071 (cit. on p. 53).

[151] Ziming Liu et al. "Kan: Kolmogorov-arnold networks". In: *arXiv preprint arXiv:2404.19756* (2024) (cit. on p. 65).

[152] S. Lloyd. "Least squares quantization in PCM". In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137. DOI: 10.1109/TIT.1982.1056489 (cit. on p. 70).

[153] Guanxiong Luo et al. "Bayesian MRI reconstruction with joint uncertainty estimation using diffusion models". In: *Magnetic Resonance in Medicine* 90.1 (Mar. 2023), pp. 295–311. DOI: 10.1002/mrm.29624 (cit. on pp. 96, 99).

[154] Michael Lustig, David Donoho, and John M. Pauly. "Sparse MRI: The application of compressed sensing for rapid MR imaging". In: *Magnetic Resonance in Medicine* 58.6 (2007), pp. 1182–1195. DOI: https://doi.org/10.1002/mrm.21391 (cit. on p. 97).

[155] Michael Lustig et al. "Faster imaging with randomly perturbed, under-sampled spirals and l1 reconstruction". In: *Proceedings of the 13th annual meeting of ISMRM*. Citeseer. 2005, p. 685 (cit. on p. 97).

[156] Tung Duy Luu, Jalal Fadili, and Christophe Chesneau. "Sampling from Non-smooth Distributions Through Langevin Diffusion". In: *Methodology and Computing in Applied Probability* 23.4 (Dec. 2021), pp. 1173–1201. DOI: 10.1007/s11009-020-09809- (cit. on p. 41).

[157] Siwei Lyu. "Interpretation and generalization of score matching". In: *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. UAI '09. Montreal, Quebec, Canada: AUAI Press, 2009, pp. 359–366. ISBN: 9780974903958 (cit. on p. 33).

[158] S.G. Mallat. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11.7 (1989), pp. 674–693 (cit. on pp. 23, 158).

[159] Benoit Mandelbrot. "The Pareto-Levy Law and the Distribution of Income". In: *International Economic Review* 1.2 (May 1960), p. 79. ISSN: 0020-6598. DOI: 10.2307/2525289 (cit. on p. 27).

[160] P Mansfield. "Multi-planar image formation using NMR spin echoes". In: *Journal of Physics C: Solid State Physics* 10.3 (Feb. 1977), pp. L55–L58. ISSN: 0022-3719. DOI: 10.1088/0022-3719/10/3/004 (cit. on p. 95).

[161] D. Martin et al. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics". In: *Proc. of the International Conference on Computer Vision*. Vol. 2. 2001, pp. 416–423 (cit. on pp. 81, 103, 154).

[162] Gisiro Maruyama. "Continuous Markov processes and stochastic equations". In: *Rendiconti del Circolo Matematico di Palermo* 4.1 (Jan. 1955), pp. 48–90. ISSN: 1973-4409. DOI: 10.1007/bf02846028 (cit. on p. 40).

[163] Allister Mason et al. "Comparison of Objective Image Quality Metrics to Expert Radiologists' Scoring of Diagnostic Quality of MR Images". In: *IEEE Transactions on Medical Imaging* 39.4 (2020), pp. 1064–1072. DOI: 10.1109/TMI.2019.2930338 (cit. on p. 62).

[164] Nicholas Metropolis et al. "Equation of State Calculations by Fast Computing Machines". In: *The Journal of Chemical Physics* 21.6 (June 1953), pp. 1087–1092. ISSN: 1089-7690. DOI: 10.1063/1.1699114 (cit. on p. 41).

[165] Sean P. Meyn and Richard L. Tweedie. *Markov Chains and Stochastic Stability*. Springer London, 1993. ISBN: 9781447132677. DOI: 10.1007/978-1-4471-3267-7 (cit. on p. 35).

[166] Sean P. Meyn and Robert L. Tweedie. "Stability of Markovian Processes II: Continuous-Time Processes and Sampled Chains". In: *Advances in Applied Probability* 25.3 (1993), pp. 487–517. ISSN: 00018678 (cit. on pp. 35, 40).

[167] Koichi Miyasawa. "An empirical Bayes estimator of the mean of a normal population". In: *Bulletin of the International Statistical Institute*. 1961, pp. 161–188 (cit. on p. 142).

[168] Matthew J. Muckley et al. "Results of the 2020 fastMRI Challenge for Machine Learning MR Image Reconstruction". In: *IEEE Transactions on Medical Imaging* 40.9 (2021), pp. 2306–2317. DOI: 10.1109/TMI.2021.3075856 (cit. on p. 131).

[169] Dominik Narnhofer et al. "Inverse GANs for Accelerated MRI Reconstruction". In: *Wavelets and Sparsity XVIII*. 2019 (cit. on pp. 94, 99, 111).

[170] Dominik Narnhofer et al. "Posterior-Variance–Based Error Quantification for Inverse Problems in Imaging". In: *SIAM Journal on Imaging Sciences* 17.1 (Feb. 2024), pp. 301–333. ISSN: 1936-4954. DOI: 10.1137/23m1546129 (cit. on pp. 115, 124, 125).

[171] Yu. E. Nesterov. "A method for solving the convex programming problem with convergence rate $O(1/k^2)$". In: *Doklady Akademii Nauk SSSR* 269.3 (1983), pp. 543–547. ISSN: 0002-3264 (cit. on p. 52).

[172] Yurii Nesterov. *Introductory Lectures on Convex Optimization*. Springer US, 2004. ISBN: 9781441988539. DOI: 10.1007/978-1-4419-8853-9 (cit. on p. 52).

[173] Alexander Quinn Nichol and Prafulla Dhariwal. "Improved denoising diffusion probabilistic models". In: *Proc. of the International Conference on Machine Learning*. PMLR. 2021, pp. 8162–8171 (cit. on p. 180).

[174] Erik Nijkamp et al. "Learning Non-Convergent Non-Persistent Short-Run MCMC Toward Energy-Based Model". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019 (cit. on pp. 105, 107).

[175] Erik Nijkamp et al. "On the Anatomy of MCMC-Based Maximum Likelihood Learning of Energy-Based Models". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.04 (Apr. 2020), pp. 5272–5280. DOI: 10.1609/aaai.v34i04.5973 (cit. on p. 105).

[176] Mila Nikolova. "Model distortions in Bayesian MAP reconstruction". In: *Inverse Problems and Imaging* 1.2 (2007), p. 399 (cit. on p. 166).

[177] Nikolaï Nikolski. *Toeplitz matrices and operators*. en. Cambridge Studies in Advanced Mathematics. Cambridge, England: Cambridge University Press, Jan. 2020 (cit. on pp. 151, 152).

[178] John P. Nolan. *Univariate Stable Distributions: Models for Heavy Tailed Data*. Springer International Publishing, 2020. ISBN: 9783030529154. DOI: 10.1007/978-3-030-52915-4 (cit. on p. 27).

[179] Peter Ochs et al. "iPiano: Inertial Proximal Algorithm for Nonconvex Optimization". In: *SIAM Journal on Imaging Sciences* 7.2 (2014), pp. 1388–1419. DOI: 10.1137/130942954 (cit. on pp. 55, 56, 82).

[180] F. Otto and C. Villani. "Generalization of an Inequality by Talagrand and Links with the Logarithmic Sobolev Inequality". In: *Journal of Functional Analysis* 173.2 (2000), pp. 361–400. ISSN: 0022-1236. DOI: https://doi.org/10.1006/jfan.1999.3557 (cit. on p. 32).

[181] Guobin Ou and Yi Lu Murphey. "Multi-class pattern classification using neural networks". In: *Pattern Recognition* 40.1 (2007), pp. 4–18. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2006.04.041 (cit. on p. 66).

[182] Guansong Pang et al. "Deep Learning for Anomaly Detection: A Review". In: *ACM Comput. Surv.* 54.2 (Mar. 2021). ISSN: 0360-0300. DOI: 10.1145/3439950 (cit. on p. 124).

[183] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035 (cit. on pp. 8, 68).

[184] Marcelo Pereyra. "Proximal Markov chain Monte Carlo algorithms". In: *Statistics and Computing* 26.4 (May 2015), pp. 745–760. ISSN: 1573-1375. DOI: 10.1007/s11222-015-9567-4 (cit. on p. 41).

[185] Marcelo Pereyra, Luis A. Vargas-Mieles, and Konstantinos C. Zygalakis. "The Split Gibbs Sampler Revisited: Improvements to Its Algorithmic Structure and Augmented Target Distribution". In: *SIAM Journal on Imaging Sciences* 16.4 (Nov. 2023), pp. 2040–2071. ISSN: 1936-4954. DOI: 10.1137/22m1506122 (cit. on p. 124).

[186] David L. Phillips. "A Technique for the Numerical Solution of Certain Integral Equations of the First Kind". In: *J. ACM* 9.1 (Jan. 1962), pp. 84–97. ISSN: 0004-5411. DOI: 10.1145/321105.321114 (cit. on p. 5).

[187] Zygmunt Pizlo. "Perception viewed as an inverse problem". In: *Vision Research* 41.24 (2001), pp. 3145–3161. ISSN: 0042-6989. DOI: https://doi.org/10.1016/S0042-6989(01)00173-0 (cit. on p. 1).

[188] Thomas Pock and Shoham Sabach. "Inertial Proximal Alternating Linearized Minimization (iPALM) for Nonconvex and Nonsmooth Problems". In: *SIAM Journal on Imaging Sciences* 9.4 (2016), pp. 1756–1787 (cit. on pp. 55, 56, 95, 96, 98, 99, 110).

[189] Boris T. Polyak. "Some methods of speeding up the convergence of iteration methods". In: *USSR Computational Mathematics and Mathematical Physics* 4.5 (1964), pp. 1–17. ISSN: 0041-5553. DOI: 10.1016/0041-5553(64)90137-5 (cit. on p. 55).

[190] Javier Portilla et al. "Image denoising using scale mixtures of Gaussians in the Wavelet domain". In: *IEEE Transactions on Image Processing* 12.11 (2003), pp. 1338–1351 (cit. on pp. 177, 178).

[191] William H. Press et al. *Numerical Recipes in C (2nd Ed.): The Art of Scientific Computing*. USA: Cambridge University Press, 1992. ISBN: 0521431085 (cit. on p. 110).

[192] Klaas P. Pruessmann et al. "SENSE: Sensitivity Encoding for Fast MRI". In: *Magnetic Resonance in Medicine* (1999) (cit. on pp. 95, 96).

[193] Patrick Putzky and Max Welling. "Invert to Learn to Invert". In: *Proc. of the Conferene on Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019 (cit. on p. 94).

[194] C. Radhakrishna Rao. *Linear Statistical Inference and its Applications*. Hoboken, NJ, USA: Wiley, Apr. 1973 (cit. on p. 149).

[195] Martin Raphan and Eero P. Simoncelli. "Least Squares Estimation Without Priors or Supervision". In: *Neural Computation* 23.2 (2011), pp. 374–420. ISSN: 0899-7667 (cit. on pp. 34, 142, 143).

[196] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. "On the Convergence of Adam and Beyond". In: *Proc. of the International Conference on Learning Representations*. OpenReview.net, 2018 (cit. on p. 59).

[197] Herbert Robbins. "An Empirical Bayes Approach to Statistics". In: *Proc. of the Berkeley Symposium on Mathematical Statistics and Probability*. 1956, pp. 157–163 (cit. on pp. 34, 140, 142).

[198] Gareth O. Roberts and Richard L. Tweedie. "Exponential Convergence of Langevin Distributions and their Discrete Approximations". In: *Bernoulli* 2.4 (1996), pp. 341–363 (cit. on pp. 39–41).

[199] Philip M. Robson et al. "Comprehensive quantification of signal-to-noise ratio and g-factor for image-based and k-space-based parallel imaging reconstructions". In: *Magnetic Resonance in Medicine* 60.4 (Oct. 2008), pp. 895–907. DOI: 10.1002/mrm.21728 (cit. on pp. 94, 96).

[200] R. Tyrrell Rockafellar and Roger J. B. Wets. *Variational Analysis*. Springer Berlin Heidelberg, 1998. ISBN: 9783642024313. DOI: 10.1007/978-3-642-02431-3 (cit. on p. 58).

[201] Ralph Tyrell Rockafellar. *Convex Analysis*. Princeton: Princeton University Press, 1970. ISBN: 9781400873173. DOI: doi:10.1515/9781400873173 (cit. on pp. 50, 54).

[202] P. B. Roemer et al. "The NMR phased array". In: *Magnetic Resonance in Medicine* 16.2 (Nov. 1990), pp. 192–225. ISSN: 1522-2594. DOI: 10.1002/mrm.1910160203 (cit. on pp. 4, 94, 95).

[203] Yaniv Romano and Michael Elad. "Boosting of Image Denoising Algorithms". In: *SIAM Journal on Imaging Sciences* 8.2 (2015), pp. 1187–1219 (cit. on pp. 87, 165, 180).

[204] Yaniv Romano, Michael Elad, and Peyman Milanfar. "The Little Engine that Could: Regularization by Denoising (RED)". In: *SIAM Journal on Imaging Sciences* 10.4 (2017), pp. 1804–1844 (cit. on p. 176).

[205] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by Nassir Navab et al. Cham: Springer International Publishing, 2015, pp. 234–241. ISBN: 978-3-319-24574-4 (cit. on p. 104).

[206] Stefan Roth and Michael J. Black. "Fields of Experts". In: *International Journal of Computer Vision* 82.2 (2009), pp. 205–229 (cit. on pp. 5, 7, 75, 87, 106, 107, 137, 151, 154, 176, 180).

[207] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60.1 (Nov. 1992), pp. 259–268. ISSN: 0167-2789. DOI: 10.1016/0167-2789(92)90242-F (cit. on pp. 1, 5).

[208] David E. Rumelhart and James L. McClelland. "Parallel distributed processing". In: Cambridge, MA: MIT Press, 1986. Chap. Chapter 6: Information Processing in Dynamical Systems: Foundations of Harmony Theory, pp. 194–281 (cit. on p. 107).

[209] Uwe Schmidt, Qi Gao, and Stefan Roth. "A Generative Perspective on MRFs in Low-Level Vision". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2010, pp. 1751–1758 (cit. on pp. 106, 107, 137, 166, 177, 180, 184).

[210] Uwe Schmidt and Stefan Roth. "Shrinkage Fields for Effective Image Restoration". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Columbus, OH, USA: IEEE, June 2014, pp. 2774–2781. ISBN: 978-1-4799-5118-5 (cit. on p. 178).

[211] O.C. Schrempf, O. Feiermann, and U.D. Hanebeck. "Optimal mixture approximation of the product of mixtures". In: *Proc. of the International Conference on Information Fusion*. Vol. 1. 2005, pp. 85–92 (cit. on p. 145).

[212] Christoph Schuhmann et al. "LAION-5B: An open large-scale dataset for training next generation image-text models". In: *Proc. of the Conferene on Neural Information Processing Systems*. 2022 (cit. on p. 58).

[213] Eero P. Simoncelli and Edawrd H. Adelson. "Noise removal via Bayesian wavelet coring". In: *Proc. of 3rd IEEE International Conference on Image Processing*. Vol. 1. 1996, 379–382 vol.1 (cit. on p. 175).

[214] David S. Smith et al. “Trajectory optimized NUFFT: Faster non-Cartesian MRI reconstruction through prior knowledge and parallel architectures”. In: *Magnetic Resonance in Medicine* 81.3 (Oct. 2018), pp. 2064–2071. ISSN: 1522-2594. DOI: 10.1002/mrm.27497 (cit. on p. 108).

[215] Daniel K. Sodickson and Warren J. Manning. “Simultaneous acquisition of spatial harmonics (SMASH): Fast imaging with radiofrequency coil arrays”. In: *Magnetic Resonance in Medicine* 38.4 (1997), pp. 591–603. DOI: https://doi.org/10.1002/mrm.1910380414 (cit. on p. 96).

[216] Jascha Sohl-Dickstein et al. “Deep Unsupervised Learning using Nonequilibrium Thermodynamics”. In: *Proc. of the International Conference on Machine Learning*. Vol. 37. Lille, France: PMLR, July 2015, pp. 2256–2265 (cit. on p. 139).

[217] Yang Song and Stefano Ermon. “Generative Modeling by Estimating Gradients of the Data Distribution”. In: *Proc. of the Conferene on Neural Information Processing Systems*. Vol. 32. Red Hook, NY, USA: Curran Associates Inc., 2019 (cit. on pp. 104, 139–141).

[218] Yang Song et al. “Score-Based Generative Modeling through Stochastic Differential Equations”. In: *Proc. of the International Conference on Learning Representations*. 2021 (cit. on pp. 7, 39, 139–141, 143, 144).

[219] Yang Song et al. “Score-Based Generative Modeling through Stochastic Differential Equations”. In: *Proc. of the International Conference on Learning Representations*. 2021 (cit. on pp. 66, 104, 139).

[220] Yang Song et al. “Sliced score matching: A scalable approach to density and score estimation”. In: *Uncertainty in Artificial Intelligence*. PMLR. 2020, pp. 574–584 (cit. on p. 104).

[221] Yang Song et al. “Solving Inverse Problems in Medical Imaging with Score-Based Generative Models”. In: *Proc. of the International Conference on Learning Representations*. 2022 (cit. on p. 99).

[222] Anuroop Sriram et al. “End-to-End Variational Networks for Accelerated MRI Reconstruction”. In: *MICCAI’20*. 2020 (cit. on pp. 98, 99, 111, 113, 117, 126, 131).

[223] A. M. Stuart. “Inverse problems: A Bayesian perspective”. In: *Acta Numerica* 19 (2010), pp. 451–559. DOI: 10.1017/S0962492910000061 (cit. on p. 5).

[224] David Stutz, Matthias Hein, and Bernt Schiele. “Relating Adversarially Robust Generalization to Flat Minima”. In: *Proc. of the International Conference on Computer Vision*. Oct. 2021, pp. 7807–7817 (cit. on p. 118).

[225] Yee Whye Teh et al. “Energy-Based Models for Sparse Overcomplete Representations”. In: *Journal of Machine Learning Research* 4 (Dec. 2003), pp. 1235–1260. ISSN: 1532-4435 (cit. on p. 106).

[226] Jean-Baptiste Thibault et al. “A Three-Dimensional Statistical Approach to Improved Image Quality for Multislice Helical CT”. In: *Medical Physics* 34.11 (Oct. 2007), pp. 4526–4544. DOI: 10.1118/1.2789499 (cit. on p. 3).

[227] Radhika Tibrewala et al. “FastMRI Prostate: A public, biparametric MRI dataset to advance machine learning for prostate cancer imaging”. In: *Scientific Data* 11.1 (2024), p. 404 (cit. on p. 131).

[228] Tijmen Tieleman. “Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient”. In: *Proc. of the International Conference on Machine Learning*. ICML ’08. Helsinki, Finland: Association for Computing Machinery, 2008, pp. 1064–1071. ISBN: 9781605582054 (cit. on p. 108).

[229] A. N. Tikhonov. “On the solution of ill-posed problems and the method of regularization”. In: *Doklady Akademii Nauk SSSR* 151 (1963), pp. 501–504. ISSN: 0002-3264 (cit. on p. 5).

[230] Francesco Tricomi. “Sulle funzioni ipergeometriche confluenti”. In: *Annali di Matematica Pura ed Applicata* 26.1 (Dec. 1947), pp. 141–175 (cit. on p. 177).

[231] Zongjiang Tu et al. "K-space and image domain collaborative energy-based model for parallel MRI reconstruction". In: *Magnetic Resonance Imaging* 99 (2023), pp. 110–122. ISSN: 0730-725X. DOI: 10.1016/j.mri.2023.02.004 (cit. on pp. 98, 99, 114).

[232] Martin Uecker et al. "ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA". In: *Magnetic Resonance in Medicine* 71.3 (May 2013), pp. 990–1001. DOI: 10.1002/mrm.24751 (cit. on pp. 96, 116, 129, 130).

[233] Martin Uecker et al. "Image reconstruction by regularized nonlinear inversion—Joint estimation of coil sensitivities and image content". In: *Magnetic Resonance in Medicine* 60.3 (2008), pp. 674–682. DOI: 10.1002/mrm.21691 (cit. on pp. 97, 108, 112).

[234] M. Unser. "On the Approximation of the Discrete Karhunen-Loève Transform for Stationary Processes". In: *Signal Processing* 7.3 (Dec. 1984), pp. 231–249 (cit. on p. 81).

[235] Michael Unser, Julien Fageot, and Harshit Gupta. "Representer Theorems for Sparsity-Promoting ℓ_1 Regularization". In: *IEEE Transactions on Information Theory* 62.9 (2016), pp. 5167–5180. DOI: 10.1109/TIT.2016.2590421 (cit. on p. 77).

[236] Singanallur V. Venkatakrishnan, Charles A. Bouman, and Brendt Wohlberg. "Plug-and-Play priors for model based reconstruction". In: *IEEE Global Conference on Signal and Information Processing*. 2013, pp. 945–948 (cit. on p. 176).

[237] Martin Vetterli and Jelena Kovačevic. *Wavelets and Subband Coding*. USA: Prentice-Hall, Inc., 1995. ISBN: 0130970808 (cit. on p. 23).

[238] Pascal Vincent. "A connection between score matching and denoising autoencoders". In: *Neural Computation* 23.7 (2011), pp. 1661–1674 (cit. on pp. 33, 139, 141, 143, 144).

[239] Martin J Wainwright and Eero Simoncelli. "Scale Mixtures of Gaussians and the Statistics of Natural Images". In: *Proc. of the Conferene on Neural Information Processing Systems*. Vol. 12. Denver, CO: MIT Press, 1999, pp. 855–861 (cit. on p. 177).

[240] Martin J. Wainwright and Michael I. Jordan. "Graphical Models, Exponential Families, and Variational Inference". In: *Foundations and Trends® in Machine Learning* 1.1–2 (2008), pp. 1–305. ISSN: 1935-8237. DOI: 10.1561/2200000001 (cit. on p. 86).

[241] Zhou Wang et al. "Image quality assessment: from error visibility to structural similarity". In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612. DOI: 10.1109/TIP.2003.819861 (cit. on p. 62).

[242] Max Welling, Simon Osindero, and Geoffrey E Hinton. "Learning Sparse Topographic Representations with Products of Student-t Distributions". In: *Proc. of the Conferene on Neural Information Processing Systems*. Ed. by S. Becker, S. Thrun, and K. Obermayer. Vol. 15. MIT Press, 2002 (cit. on p. 7).

[243] Catherine Westbrook and John Talbot. *MRI in Practice*. 5th ed. Standards Information Network, Aug. 2018 (cit. on p. 94).

[244] Andrew Witkin, Demetri Terzopoulos, and Michael Kass. "Signal matching through scale space". In: *International Journal of Computer Vision* 1.2 (1987), pp. 133–144. ISSN: 1573-1405. DOI: 10.1007/bf00123162 (cit. on p. 166).

[245] Leslie Ying and Jinhua Sheng. "Joint image reconstruction and sensitivity estimation in SENSE (JSENSE)". In: *Magnetic Resonance in Medicine* 57.6 (2007), pp. 1196–1202. DOI: 10.1002/mrm.21245 (cit. on pp. 97, 108).

[246] A. L. Yuille. "Energy functions for early vision and analog networks". In: *Biological Cybernetics* 61.2 (June 1989), pp. 115–123. ISSN: 1432-0770. DOI: 10.1007/bf00204595 (cit. on p. 166).

[247] Martin Zach, Florian Knoll, and Thomas Pock. "Stable Deep MRI Reconstruction Using Generative Priors". In: *IEEE Transactions on Medical Imaging* 42.12 (2023), pp. 3817–3832 (cit. on p. 94).

[248] Martin Zach, Erich Kobler, and Thomas Pock. "Computed Tomography Reconstruction Using Generative Energy-Based Priors". In: *Proc. of the OAGM Workshop 2021*. Graz: Verlag der Technischen Universität Graz, Dec. 2021, pp. 52–58 (cit. on pp. 94, 124).

[249] Martin Zach et al. "Explicit Diffusion of Gaussian Mixture Model Based Image Priors". In: *Proc. of the International Conference on Scale Space and Variational Methods in Computer Vision*. Cham: Springer International Publishing, 2023, pp. 3–15. ISBN: 978-3-031-31975-4 (cit. on pp. 138, 155, 166).

[250] Martin Zach et al. "Product of Gaussian Mixture Diffusion Models". In: *Journal of Mathematical Imaging and Vision* (Mar. 2024). ISSN: 1573-7683. DOI: 10.1007/s10851-024-01180-3 (cit. on pp. 84, 88, 138, 151).

[251] Jure Zbontar et al. "fastMRI: An Open Dataset and Benchmarks for Accelerated MRI". In: 2018 (cit. on pp. 61, 74, 77, 94, 99, 100, 103, 111, 116).

[252] Gushan Zeng et al. "A review on deep learning MRI reconstruction without fully sampled k-space". In: *BMC Med. Imag.* 21.1 (Dec. 2021). DOI: 10.1186/s12880-021-00727-9 (cit. on p. 98).

[253] Bo Zhou and S Kevin Zhou. "DuDoRNet: Learning a Dual-Domain Recurrent Network for Fast MRI Reconstruction with Deep T1 Prior". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 4273–4282 (cit. on pp. 94, 99).

[254] Bo Zhu et al. "Image Reconstruction by Domain-Transform Manifold Learning". In: *Nature* 555.7697 (Mar. 2018), pp. 487–492. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/nature25988 (cit. on pp. 94, 96, 99, 100).

[255] Song Chun Zhu and D. Mumford. "Prior learning and Gibbs reaction-diffusion". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.11 (1997), pp. 1236–1250. DOI: 10.1109/34.632983 (cit. on pp. 7, 80, 87, 137, 180).

[256] Song Chun Zhu, Ying Nian Wu, and David Mumford. "Minimax Entropy Principle and Its Application to Texture Modeling". In: *Neural Computation* 9.8 (1997), pp. 1627–1660. DOI: 10.1162/neco.1997.9.8.1627 (cit. on pp. 7, 75, 86, 87, 145).

[257] Song Chun Zhu, Yingnian Wu, and David Mumford. "Filters, Random Fields and Maximum Entropy (FRAME): Towards a Unified Theory for Texture Modeling". In: *International Journal of Computer Vision* 27.2 (1998), pp. 107–126 (cit. on pp. 7, 75, 80, 81, 86, 87, 105, 106, 180).

[258] Juntang Zhuang et al. "AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients". In: *Proc. of the Conferene on Neural Information Processing Systems* 33 (2020) (cit. on pp. 59, 60, 154).

[259] Dennis Zill. *A first course in differential equations with modeling applications*. en. 12th ed. Florence, KY: Brooks/Cole, June 2023 (cit. on p. 33).

[260] Daniel Zoran and Yair Weiss. "From learning models of natural image patches to whole image restoration". In: *Proc. of the International Conference on Computer Vision*. Barcelona, Spain, Nov. 2011, pp. 479–486. ISBN: 978-1-4577-1102-2 978-1-4577-1101-5 978-1-4577-1100-8 (cit. on pp. 81, 87, 151, 156, 164–166, 178, 180).