UNIVERSITÄT ZU LÜBECK

From the Institute of Mathematics and Image Computing
of the University of Lübeck
Director: Prof. Dr. Jan Modersitzki

# Functional Lifting: Nonlinear Inverse Scale Space Iterations and Neural Fields

Dissertation
for Fulfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck

from the Department of Computer Sciences and Technical
Engineering

Submitted by
Danielle Bednarski
from Oldenburg

Lübeck, 2024

First referee:
Second referee:
Date of oral examination:
Approved for printing:

# Abstract

Variational models lie at the core of formulating and understanding problems in modern mathematical image processing. Suitable mathematical models that accurately describes the given problem often have undesirable properties, in particular non-convexity. This thesis is built around a *functional lifting* strategy which embeds non-convex problems into a larger space, so that the lifted problem is convex and that global minimizers of the lifted problem can be mapped to global minimizers of the original problem. The contributions of this thesis are threefold.

We give a thorough introduction to the *calibration-based lifting* approach. In the literature, this approach is described in a $W^{1,1}$ setting. A first contribution of this thesis is the compilation of measure theoretic results in order to extend the approach and related theory to the BV setting. In addition, we also discuss possible discretization and optimization approaches found in the literature.

Calibration-based lifting is then used in order to extend the use case of *inverse scale space iterations*. These iterations are related to *non-linear spectral representation* and *filtering* techniques, which have proven to give impressive results in different imaging applications. While existing theory is built around the convex *deconvolution* or *denoising* problem, our goal is to generalize the results to variational problems with arbitrary, possibly non-convex data term and *total variation* regularizer.

Finally, we propose a *neural fields* based stochastic optimization approach for solving variational problems with non-convex data term and total variation regularizer. By combining calibration-based lifting with the powerful neural fields computational framework, we aim to present a novel stochastic optimization strategy for this difficult class of variational problems.

# Zusammenfassung

Variationsprobleme spielen eine zentrale Rolle in der Formulierung und dem Verständnis von Problemen in der modernen mathematischen Bildverarbeitung. Mathematische Modelle, die eine gegebene Problemstellung präzise beschreiben, haben oftmals unerwünschte Eigenschaften, so wie beispielsweise Nichtkonvexität. In ihrem Kern beschäftigt sich diese Arbeit mit einem funktionalen Liftingverfahren, welches nicht-konvexe Probleme in einen anderen Raum einbettet, sodass die geliftete Formulierung des Problems konvex ist und globale Minimierer des gelifteten Problems mit globalen Minimierern des ursprünglichen Problems in Zusammenhang stehen. Der Beitrag dieser Arbeit lässt sich in drei Teile strukturieren.

Zunächst wird eine ausführliche Einführung in ein kalibrierungsbasiertes Liftingverfahren gegeben. In der Literatur ist dieser Ansatz für Variationsprobleme mit $W^{1,1}$ Lösungsraum beschrieben. Der erste Beitrag dieser Arbeit besteht in der Erweiterung des kalibrierungsbasierten Liftingverfahrens auf Variationsprobleme mit BV Lösungsraum. Zusätzlich werden verschiedene Diskretisierungs- und Optimierungsansätze aus der Literatur erörtert und verglichen.

Das kalibrierungsbasierte Liftingverfahren wird anschließend verwendet, um Iterationen über den *inversen Skalenraum* für neue Problemklassen zu definieren. Diese Iterationen sind eng verwandt mit *nichtlinearer spektraler Repräsentation* und *Filteransätzen*, die bereits eindrucksvolle Ergebnisse in Bildgebungsanwendungen erzielt haben. Während sich die Theorie bisher ausschließlich mit konvexen Entfaltungs- und Entrauschungsproblemen beschäftigt hat, wird der Anwendungsfall in dieser Arbeit auf TV-regularisierte Variationsprobleme mit möglicherweise nichtkonvexem Datenterm erweitert.

Abschließend wird ein stochastischer Optimierungsansatz basierend auf *neuronalen Feldern* für TV-regularisierte Variationsprobleme mit möglicherweise nichtkonvexem Datenterm vorgeschlagen. Die Kombination des kalibrierungsbasierten Liftingverfahrens mit rechnerisch leistungsstarken neuronalen Feldern hat das Potenzial für einen effizienten Optimierungsansatz für diese schwierige Klasse an Variationsproblemen.

# Acknowledgements

# Notation and Abbreviations

## Abbreviations

| | |
|---|---|
| ANN | Artificial Neural Nets |
| BI | Bregman Iteration |
| BV | Functions of Bounded Variation |
| CBL | Calibration-based lifting |
| EmNeF | Neural Fields for Embedded Variational Problems |
| GAN | Generative Adversarial Network |
| GD | Gradient Descent |
| ISS | Inverse Scale Space (flow) |
| LBI | Lifted Bregman Iteration |
| LSTM | Long Short-Term Memory |
| NeRF | Neural Radiance Fields |
| NF | Neural Field |
| PDHG | Primal-Dual Hybrid Gradient |
| PINN | Physic Informed Neural Networks |
| RFF | Random Fourier Features |
| ROF | Rudin-Osher-Fatemi ($L^2$-$L^1$) Model for Image Restoration |
| SBV | Special Functions of Bounded Variation |
| SGD | Stochastic Gradient Descent |
| SS | Scale Space (flow) |
| TV | Total Variation (seminorm or regularizer) |

## Sets

| | |
|---|---|
| $\mathscr{B}_r(x)$ | Ball centred around $x \in \mathbb{R}^n$ with radius $r > 0$, i.e., $\{y \in \mathbb{R}^n \mid \|x - y\| \leq r\}$ |
| $\mathrm{BV}(\mathbb{R}^n; \mathbb{R}^d)$ | $\mathbb{R}^d$-valued functions of bounded variation |
| $C(S)$ | Continuous selections of a set-valued mapping $S$ |
| $C(\mathbb{R}^n; \mathbb{R}^d)$ | Same as $C^1(\mathbb{R}^n; \mathbb{R}^d)$ |
| $C^k(\mathbb{R}^n; \mathbb{R}^d)$ | $k$-times continuously differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^d$ |

| | |
|---|---|
| $C_c^k(\mathbb{R}^n; \mathbb{R}^d)$ | $C^k(\mathbb{R}^n; \mathbb{R}^d)$ functions with compact support |
| $C_0^k(\mathbb{R}^n; \mathbb{R}^d)$ | Closure of $C_c^k(\mathbb{R}^n; \mathbb{R}^d)$ wit respect to the $\|\cdot\|_\infty$-norm |
| conv $A$ | Convex hull of a set $A$ |
| $\partial^* E$ | Measure theoretic (also essential) boundary of a set $E$ |
| $E^t$ | Points of a set $E$ with density $t$. Especially, $E^0$ and $E^1$ denote the measure theoretic exterior and interior of $E$ |
| $\mathscr{F}\mathscr{E}$ | Reduced boundary of a set $E$ |
| $J_u$ | Set of approximate jump points of a function $u \in L_{\text{loc}}^1$ |
| $L^p(\Omega; \mathbb{R})$ | $L^p$-integrable functions, i.e. $\left\{ f : \Omega \to \mathbb{R} : \int_\Omega |f(x)|^p \, \mathrm{d}x < \infty \right\}$ |
| $L_{\text{loc}}^p(\Omega; \mathbb{R})$ | Locally $L^p$-integrable functions, i.e. $\left\{ f : \Omega \to \mathbb{R} : \int_A |f(x)|^p \, \mathrm{d}x < \infty, \forall A \subset \Omega \text{ compact} \right\}$ |
| $\mathscr{M}^+(\mathbb{R}^n)$ | Positive Radon measure on $\mathbb{R}^n$ |
| $\mathscr{M}_{\text{loc}}(\mathbb{R}^n; \mathbb{R}^m)$ | $\mathbb{R}^m$-valued Radon measure on $\mathbb{R}^n$ |
| $\mathscr{M}(\mathbb{R}^n; \mathbb{R}^m)$ | Finite $\mathbb{R}^m$-valued Radon measure on $\mathbb{R}^n$ |
| $\mathscr{N}(x)$ | Neighbourhood of $x$ |
| $\nu_E$ | Generalized inner normal of a set $E$ |
| $P(E; \Omega)$ | Perimeter of the set $E$ in $\Omega$ |
| $\overline{\mathbb{R}}$ | Extended real line $\mathbb{R} \cup \{-\infty, +\infty\}$ |
| $\mathbb{R}^n$ | $n$-dimensional Euclidean space |
| $\mathscr{S}_r(x)$ | $\mathbb{R}^{n-1}$-dimensional sphere centred around $x \in \mathbb{R}^n$ with radius $r > 0$, i.e. $\mathscr{S}_r(x) := \{ y \in \mathbb{R}^n \mid \|x - y\| = r \}$ |
| SBV$(\mathbb{R}^n; \mathbb{R}^d)$ | $\mathbb{R}^d$-valued special functions of bounded variation |
| $S_u$ | Approximate discontinuity set of a function $u \in L_{\text{loc}}^1$ |
| $V(u, \Omega)$ | Variation of a function $u$ in $\Omega$ |

## Operators and Functions

| | |
|---|---|
| $\mathbf{1}_u(x, t)$ | Indicator of the subgraph of $u$: $\mathbf{1}_u(x, t) = 1$, if $u(x) > t$, else 0. |
| $1_{\{v>s\}}(x, t)$ | Characteristic function: $1_{\{v>s\}}(x, t) = 1$ if $v(x, t) > s$, else 0. |
| conv $f$ | convex hull of a function $f$, see Sec. 2.1 |
| dom $f$ | effective domain of a function: $\{x : f(x) < +\infty\}$ |
| $\partial f$ | Subdifferential of convex function $f : \mathbb{R}^n \to \mathbb{R}$ |
| $\delta_C(x)$ | Indicator function of a set: $\delta_C(x) = 0$, if $x \in C$, else $+\infty$ |
| $Du$ | Distributional derivative of a function $u$ |

| | |
|---|---|
| $f^*(x)$ | Fenchel conjugate of $f$, see Def. 2.1.1 |
| $\tilde{f}(x, \lambda)$ | Perspective function of $f$, see Def. 2.1.2 |
| $f^\infty(x)$ | Recession function of $f$, see Def. 2.1.3 |
| $\Gamma_u$ | measure theoretic boundary of the subgraph of a function $u$ |
| graph $u$ | Complete graph of a function $u$ |
| hyp | Subgraph of $u$ |
| $\fint_\Omega f \, d\mu$ | Mean value of $f$ in $\Omega$, i.e. $\frac{1}{\mu(\Omega)} \int_\Omega f \, d\mu$ |
| $\mathrm{lev}_{>s}\, u$ | Superlevel sets of a function $u$ |
| $\nu_u$ | Generalized inner unit normal of the subgraph of a function $u$ |
| $\sigma_C(x)$ | Support function of set $C$, i.e. $\sigma_C(x) = \sup_{y \in C} \langle y, x \rangle$ |
| $\mathrm{supp}\, f$ | Support of f: Closure of $\{x : f(x) \neq 0\}$ |

## Measures and Distributions

| | |
|---|---|
| $A \llcorner B$ | Restriction of a set $A$ to $B$ |
| $d\mathcal{H}^n$ | $n$-dimensional Hausdorff measure |
| $\mathcal{L}^n$ | $n$-dimensional Lebesgue measure |
| $\mathcal{N}(\mu, \sigma^2)$ | Normal distribution with mean $\mu$ and variance $\sigma^2$ |

# Contents

# Introduction 1

Mathematical imaging is a fascinating field which not only finds application in diverse areas of life and science [AK06] (see Fig. 1.1) but also brings together various mathematical fields of studies such as measure theory, analysis and numerical optimization [BL11]. The imaging pipeline typically consists of multiple steps: Acquiring (discrete) measurements of the data, modelling the real-life problem as a PDE or a variational problem, discussing the existence and uniqueness of minimizers, designing discretization schemes, and developing numerical optimization strategies or learning frameworks in order to find solutions.

The results presented in this work are concerned with different aspects of the imaging pipeline starting with a variational problem (re-)formulation, an iterative optimization strategy which also allows to analyse the solution's decomposition into scales, and a contemporary learning framework. As such, we will use notations and results from measure theory [AFP00; ABM14], convex and variational analysis [ET99; Roc70; RW09], numerical optimization [NW99] and learning [BN06].

The combining element in this work is the *calibration-based lifting* strategy (CBL) [ABD03; Poc+10; MC17]. It is a *lifting* (also *embedding*) strategy, which allows to reformulate non-convex variational problems in a "higher-dimensional" space, such that the lifted formulation is convex and global minimizers of the embedded, convex energy can be mapped to global minimizers of the original, non-convex energy. CBL is a very powerful technique, as convexity greatly simplifies the numerical optimization and, therefore, extents the practical applicability of certain non-convex variational problems.

The first goal of this thesis is to give a holistic overview and introduction to the calibration-based lifting approach. We recall and discuss the derivation of the CBL method, theoretical results, as well as discretization and optimization aspects found in the literature of the imaging community. We furthermore extend certain results and proofs concerning the CBL approach to a broader class of variational problems by relying on a compilation of measure theoretic results. This lays the groundwork for the next two goals:

> ▶ *Inverse scale space iterations* and related *nonlinear spectral representation* or *filtering* techniques have proven to be very successful for different imaging tasks such as image fusion and manipulation [HG18]. Yet, to the best of our knowledge,

all results in the literature are build around the convex *deconvolution/ denoising* problem. We use the calibration-based lifting strategy in order to extend the use case of these iterations to a broader class of possibly non-convex variational problems. To the best of our knowledge, we were the first to propose such a unified extension in [BL21; BL22].

▶ Non-convex imaging problems pose a great numerical challenge. We propose a *neural fields* based learning approach for solving variational problems with non-convex data term and *total variation* regularizer. By combining the theory of the calibration-based lifting strategy with a computationally powerful learning approach, we aim to present a novel and general optimization strategy for this difficult class of variational problems. Our so-called EmNeF approach first presented in [BL23] can be understood as an alternative to classical optimization of CBL lifted problems with the *primal-dual hybrid gradient* algorithm or as a method which allows to incorporate more physical knowledge into learning frameworks.

The remainder of this chapter establishes the general setting and notation, explains the concept of functional lifting and, lastly, gives an introduction to the main fields of our interest in order to motivate our findings.

## 1.1 Problem Setting

Imaging generally starts with measurements of some visual data obtained through imaging devices – such as cellphone cameras, polaroid pictures, MRI scans, microscopic or satellite images. Recovering the unknown image from measurements is an often ill-posed inverse problem, which requires to incorporate prior knowledge concerning the solution. A particularly common approach – which naturally occurs when formulating the problem in a statistical sense and deriving a *maximum-a-posteriori* (MAP) estimator – is the formulation of a (continuous) variational problem consisting of a *data term* and a *regularizer* [BL11].

In this work, we consider variational minimization problems of the form

$$\inf_{u \in U} F(u), \tag{1.1}$$

where $U$ is a suitable feasible set with elements $u : \Omega \to \Gamma$ which map from the open and bounded image *domain* $\Omega \subset \mathbb{R}^n$ to the image *range*[1] $\Gamma \subset \mathbb{R}^d$. The objective functional $F$ is often of integral form and may depend on (possibly distributional) derivatives of $u$,

1: Due to historical reasons, we refer to the discretized image range $\Gamma^h$ as the *label space*.

**(a)** Reference $R$



**(b)** Template $T$ and map $\varphi$



**(c)** Difference $R - T \circ \varphi$



**(d)** 1st input image $I_1$



**(e)** 2nd input image $I_2$



**(f)** Depth map
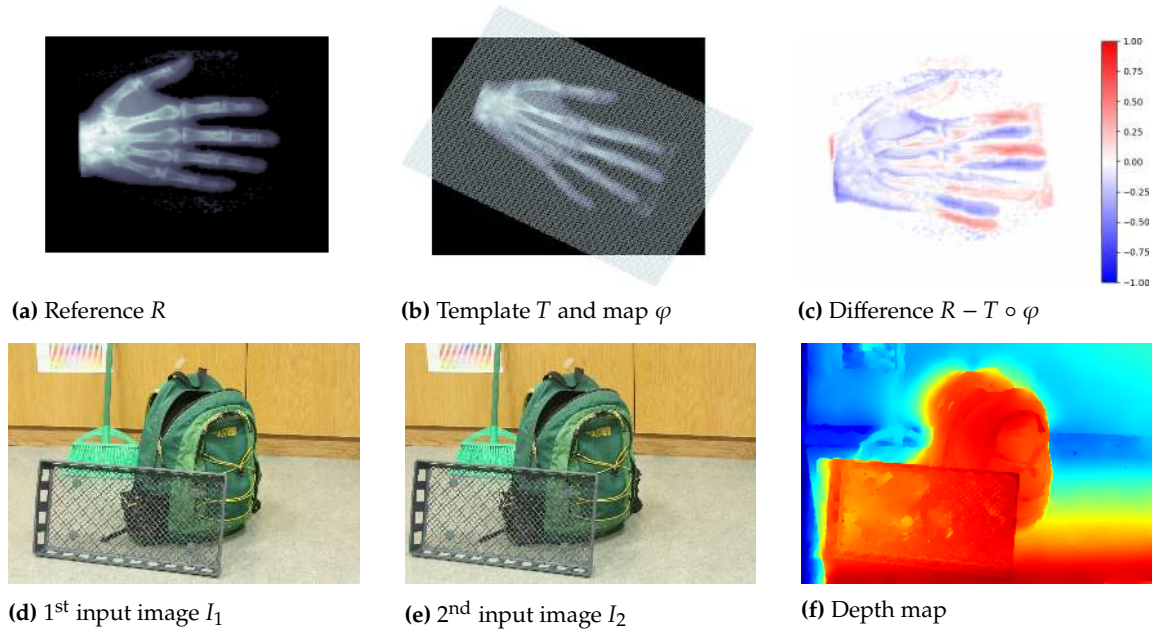
**Figure 1.1:** Mathematical imaging finds application in a variety of fields and for many different tasks. Here are just two examples. The **top row** shows an example from medical image registration, where a transformation $\varphi$ is sought which maps a template image $T$ onto a reference image $R$ such that $T(\varphi(x)) \sim R(x)$. Input images from [Ami94]. The **bottom row** depicts a stereo matching example, where a depth map $u$ is sought given two pre-aligned images $I_1, I_2$ of the same scene. The images are assumed to be pre-aligned such that the epipolar lines align and $I_1(x_1, x_2) \approx I_2(x_1, x_2 - u(x_1, y_2))$. Input images "Backpack" from [Sch+14].

as in the generic and often-used model

$$F(u) := \int_\Omega f(x, u(x), \nabla u(x)) \, dx. \tag{1.2}$$

We also consider the special case where the energy is separable into a *data term $D$*, which is small whenever the candidate $u$ is compatible with the measurement or input data, and a *regularizer $J$*, which allows to incorporate prior knowledge about the solution, such as desired smoothness. In this case we write:

$$F(u) = \underbrace{\int_\Omega \rho(x, u(x)) \, dx}_{D(u)} + \underbrace{\int_\Omega \eta(x, \nabla u(x)) \, dx}_{J(u)}. \tag{1.3}$$

**Total Variation.** A typical regularizer, which we will also mainly be concerned with in this thesis, is the *total variation*

$$\mathrm{TV}(u) := \int_\Omega d|Du|, \tag{1.4}$$

Here, $Du$ denotes the *distributional derivative* of $u$. The total variation is well-defined for piece-wise smooth functions with jump discontinuities and, in the special case of smooth functions, reduces

to $TV(u) = \int_\Omega \|\nabla u(x)\| \, dx$. The typically associated feasible set consists of functions of *bounded variation*

$$\mathrm{BV}(\Omega; \mathbb{R}) := \{u \in L^1(\Omega; \mathbb{R}) \mid TV(u) < \infty\}. \qquad (1.5)$$

The BV space is especially suitable when working with images, as it permits jump discontinuities, which allows to model objects with sharp edges.

Why is the total variation of special interest to our goals and contributions?

▸ The total variation regularizer is convex, 1-homogeneous and lower-semicontinuous in $\mathrm{BV}(\Omega; \mathbb{R})$ [AFP00, Prop. 3.6]. Additionally, it fulfils the *coarea formula* [AFP00, Thm. 3.40]. All these properties allow to prove a *tightness result* in the calibration-based lifting approach: it holds $\inf_{v \in \mathscr{C}} \mathscr{F}(v) = \inf_{u \in \mathcal{U}} F(u)$, where $\mathscr{F}$ denotes the lifted energy and $\mathscr{C}$ the lifted feasible set. This tightness result is central for the argument made in Chpt. 3 that minimizing the lifted convex energy is equivalent to minimizing the original energy.

▸ As the total variation is convex and 1-homogeneous, it naturally fulfills the assumptions typically required in the theory of nonlinear spectral representation and filtering [Bur+16] which will be of interest in Chpt. 4.

Let us now consider two concrete examples of variational problems with TV regularizer and explain their relevance to this work.

**Denoising.**   Assume a given image $f \in L^2(\Omega, \mathbb{R}^k)$ is corrupted by (possibly normally distributed) noise, and the true, noise-free image $\tilde{u}$ is to be found. An established method for estimating a solution $u^* \approx \tilde{u}$ is the Rudin-Osher-Fatemi (ROF) denoising model [ROF92]

$$\inf_{u \in \mathrm{BV}(\Omega; \mathbb{R}^k)} \left\{ \frac{1}{2} \int_\Omega \|u(x) - f(x)\|_2^2 \, dx + \lambda \, TV(u) \right\}. \qquad (1.6)$$

By controlling the variation of the solution with the TV regularizer, small-scale features in the original image $f$ can be suppressed in the solution $u^*$. Another interpretation is that the solution $u^*$ is a nonlinear interpolate between the original $L^2$ image $f$ and its median $\bar{f}$ (a BV function), with weighting set by the regularization parameter $\lambda \in \mathbb{R}$.

Depending on the choice of the regularization parameter $\lambda$, different features of $f$ are moved into the noise component $u^* - f$. Therefore, the regularizer introduces a mathematical notion of *coherent information* and *scale* to image data. This notion allows to

define nonlinear scale space representation and filtering techniques for images [Gil13; Gil14; Bur+16].

In the literature, nonlinear scale space methods evolve around the convex denoising (or similar deconvolution) problem (1.6). These methods give great results in image manipulation [HG18]. In Chpt. 4, we use the calibration-based lifting approach in order to transfer the idea of a nonlinear spectral representation and filtering to the solution of a broader class of variational problems – more precisely, problems with total variation regularizer and arbitrary, possibly non-convex data term. This allows us to make a first step towards extending the spectral theory to objects associated with images, such as *depth* or *segmentation maps*. In case of the denoising problem we can show that our extended method is in fact equivalent to the original one.

**Stereo Matching.**    In *stereo matching*, the goal is to estimate the depth of a scene from two or multiple input images. These images are taken from slightly different viewpoints. The depth can be estimated by calculating how much a pixel or area of interest shifts its position between the different images. A large shift indicates closeness, a small shift indicates larger distance to the camera. See Fig. 1.1 for an example.

In order to simplify the problem, preprocessing steps can be applied in order to ensure that the shift only involves one axis; the input images are *rectified*, such that the *epipolar lines* align [Sch+14]. Thus, the displacement of points between the two images is restricted to, e.g., the $x_2$–axis and can be modelled as a scalar function $u : \mathbb{R}^2 \to \mathbb{R}$. Given two input images $I_1$ and $I_2$, and denoting the measurement (interpolation) of $I_2$ on the shifted domain (grid) by $\tilde{I}_2(x) := I_2(x_1, x_2 - u(x))$, the data term for this problem can be formulated as

$$\rho(x, u(x)) := \int_{W(x)} \sum_{j=1,2} \max \left\{ d_j(I_1(y), \tilde{I}_2(y)), \tau \right\} \, \mathrm{d}y. \qquad (1.7)$$

Here, $W(x)$ denotes a patch around $x$, $\tau$ is a scalar threshold and $d_j$ is the absolute gradient difference defined as

$$d_j(I_1(y), \tilde{I}_2(y)) := |\partial_{x_j} I_1(y_1, y_2) - \partial_{x_j} I_2(y_1, y_2 - u(y))|. \qquad (1.8)$$

This problem is non-convex and nonlinear in $u$, which makes optimization rather difficult, as classical gradient-descent based optimization algorithms are likely to converge to local minima. This is where functional lifting strategies become relevant, as they allow

to reformulate a given variational problem in a higher-dimensional space, such that the higher-dimensional problem is convex.

In this work, we use stereo matching as a representative example for non-convex problems. In Chpt. 4 we use it in order to demonstrate that our newly introduced *generalized nonlinear inverse scale space iterations* indeed generalize the notion of nonlinear inverse scale spaces. In Chpt. 5 we also come back to this example in order to compare the performance of our newly introduced neural fields based solving strategy for non-convex problems with the one of the sate-of-the-art gradient-descent based optimization algorithm.

Having established our general problem setting whilst considering relevant examples, we now proceed with motivating the three main topics of this work.

## 1.2 Background and Motivation

### 1.2.1 Functional Lifting

During optimization, convexity of the variational model plays a vital role. There are very effective optimization algorithms for solving convex problems [NW99; BV04]. Non-convex problems pose greater difficulties – for example, the choice of a good initialization may not only affect computational time but also convergence to a global instead of a local minimizer [BV04]. Not all variational models in imaging are convex, since non-convex problems might model the image acquisition process or the given task at hand more precisely.

A particular strategy for dealing with non-convexity, which we will also be concerned with in this thesis, is to reformulate an originally non-convex problem

$$\inf_{u \in U} F(u) \tag{1.9}$$

as a higher-dimensional[2], convex problem

$$\inf_{v \in \mathscr{C}} \mathscr{F}(v). \tag{1.10}$$

In the imaging community, such reformulations are called *functional lifting*. A related topic is the *linear programming relaxation* in the discrete combinatorics community as well as the term *embedding*, which is typically used in the analysis community. The general concept of this functional lifting strategy is visualized in Fig. 1.2.

2: The term "higher-dimensional" stems from the discrete setting and is here, in the continuous setting, slightly abused. To be more precise, we should say that the originally non-convex energy $F$ is replaced by a convex energy $\mathscr{F}$, which is defined over a feasible set which contains functions with a "higher-dimensional" domain. As an example, the feasible set $U$ of functions $u : \Omega \to \Gamma$ might be replaced by the feasible set $\mathscr{C}$ of functions $v : \Omega \times \Gamma \to \mathbb{R}$.
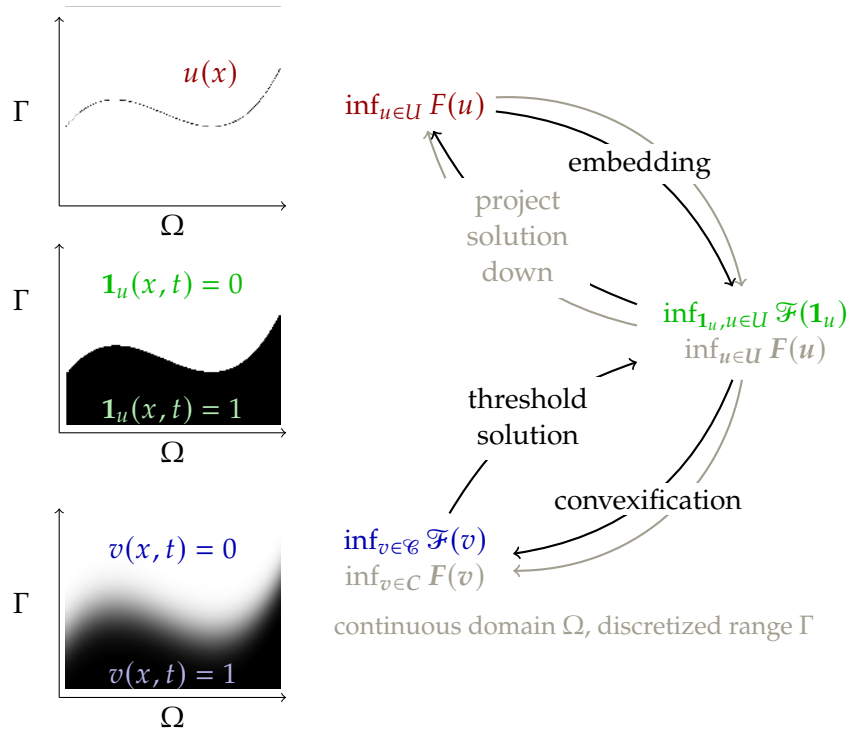
**Figure 1.2: Calibration-Based Lifting.** The idea of functional lifting is to reformulate non-convex variational problems over a new solution space, such that the new formulation is convex and that global minimizers of the latter can be projected onto global minimizers of the former. Even though functional lifting increases the dimensionality of the discrete problem, optimization is greatly simplified as convergence of gradient-descent based optimization algorithms to suboptimal local minimizers is ruled out.

In the calibration-based lifting (CBL) approach, the original problem $\inf_{u \in U} F(u)$ **(top)**, is reformulated as a minimization problem of the convex energy $\mathcal{F}$ over the non-convex set of characteristic functions of subgraphs **(middle)**. By formally replacing the non-convex feasible set with a convex superset $\mathcal{C}$ one reaches the fully convex problem $\inf_{v \in \mathcal{C}} \mathcal{F}(v)$ **(bottom)**. In the continuous setting, any global minimizer $v^* \in \mathcal{C}$ of $\mathcal{F}$ can be mapped to $\mathbf{1}_u^*$ via a pointwise thresholding formula. $\mathbf{1}_u^*$ then can uniquely be mapped to to a global minimizer $u^* \in U$ of $F$. The flowchart on the right shows the pipeline in the continuous (black) and discrete (gray) setting.

Desirable properties of functional lifting approaches are:

1. The lifted functional $\mathcal{F} : \mathcal{C} \rightarrow \mathbb{R}$ is a lower semicontinuous and coercive function. Under certain assumptions on $\mathcal{C}$, this guarantees the existence of a minimizer, see for example [Sch13, Thm. 13.3].

2. The lifted formulation is *tight*, which means that it holds

$$\inf_{v \in \mathcal{C}} \mathcal{F}(v) = \inf_{u \in U} F(u). \tag{1.11}$$

   No new artificial minima are introduced.

3. There is a "projection" that maps global minimizers $v^* \in \arg\min_{v \in \mathcal{C}} \mathcal{F}(v)$ of the lifted problem to global minimizers $u^* \in \arg\min_{u \in U} F(u)$ of the original problem.

If these properties are fulfilled, solving the convex problem (1.10) helps finding global minimizers of the original, non-convex problem (1.9).

Lifting strategies are very powerful, as they allow to solve originally non-convex variational problems numerically with the help of gradient-descent based optimization algorithms and without the risk of converging to non-global local minima. Over the years, many different strategies have been proposed – we give a short overview in the following subsection.

**Related Work.** The first lifting/relaxation strategies in the imaging community were concerned with fully discrete problems, that is (multi-)labelling problems on a discrete grid. The idea was to formulate the problems as *Markov Random Fields* (MRFs), that is as conditional probability models consisting of certain undirected graphs. Minimum energy configurations of certain MRFs can be computed in terms of a minimum cut problem [Ish03]. Some of the fields of application were segmentation [IG98b], image restoration [IG99], and stereo matching [IG98a]. See [Lel+13a] for a more comprehensive study and overview of related work in the fully discrete setting.

Subsequent research was concerned with (multi-) labelling problems on a continuous grid. These problems can be described as finding a function $u : \Omega \rightarrow \Gamma$ that assigns each point $x$ in a continuous domain $\Omega$ a label $u(x)$ from a discrete range $\Gamma$ while minimizing some functional $F$.

In their famous work [CSV00], Chan and Vese reformulate the minimal partition problem in terms of minimizing over characteristic functions of sets of finite perimeter. Later, in [CEN06], connecting minimal partitions and (binary) image denoising and using a thresholding theorem, the authors argued that the (non-convex) solution space of characteristic functions of sets with finite perimeter can be extended to a convex solution space, resulting in a convex formulation for the minimal partition problem.

Similar ideas were subsequently employed for multi-label problems [Poc+08; LS11; ZK12; CCP12] and vector-valued multi-label problems [SGC11; SCC12]. It was shown that the continuous nature of the lifting approaches reduces metrication errors, also known as grid bias, [Poc+08] and improve isotropy, see [Lel+13a] for an overview.

The most recent approaches focus on lifting fully continuous problems, where a function $u : \Omega \rightarrow \Gamma$ is sought, which assigns each point $x$ in a continuous domain $\Omega$ a value $u(x)$ from a continuous range $\Gamma$ while minimizing some functional $F$. The lifted, convex problem can be formulated in terms of complete graphs [Poc+10], measures [Vog20; Vog+20] or currents [MC19b; Möl20]. Discretizing these continuously formulated lifted energies can lead to very high-dimensional discrete problems, which can

be numerically challenging. Custom *sublabel-accurate* discretization schemes were introduced in order to contain this problem [MC17; Ye+22; Ye+23].

Many of the above-mentioned approches are designed explicitly for the first-order total variation regularizer. In [Lel+13b], the special case of manifold-valued total variation problems is considered. There are also functional lifting approaches for second-order regularizers [LL16; VL19; Vog20].

**Contribution.** In this work, we employ the calibration-based lifting approach [ABD03; Poc+10; MC17]. It is a fully continuous lifting approach for scalar problems which fulfils all the desirable properties listed on p. 6. In particular, the approach is tight, which means that it holds

$$\inf_{u \in U} F(u) = \inf_{v \in \mathscr{C}} \mathscr{F}(v), \qquad (1.12)$$

and (global) minimizers $u^* = \arg\min_{u \in U} F(u)$ can be mapped to (global) minimizers $v^* = \arg\min_{v \in \mathscr{C}} \mathscr{F}(v)$.

We extend central results and proofs in [Poc+10] from variational problems defined over $W^{1,1}$ functions to variational problems defined over BV functions by using a collocation of measure theoretic results. As subsequent contributions in this thesis heavily rely on the calibration-based lifting approach and its properties, we provide a self-contained derivation of the approach, a summary of its properties as well as a short discussion of numerical aspects.

### 1.2.2 Scale Spaces

Scale-space theory is concerned with multiscale representation of a signal or an image. The multiscale representation allows to construct filters based on scale, which are useful for a variety of tasks: noise removal, edge detection, feature extraction or feature enhancement [HG18]. There are many different ways of defining scale spaces [Koe84; SG01; Bur+16]– *the* scale space does not exist. In the following, we will introduce the classical Gaussian scale space and a more recent nonlinear scale space that will be the basis for part of this thesis.

**Gaussian Scale Space.** One of the early scale spaces for images – namely the *Gaussian scale space* – was introduced in [Koe84]. Starting with an input image $f : \Omega \to \mathbb{R}, \Omega \subset \mathbb{R}^n$, the idea is
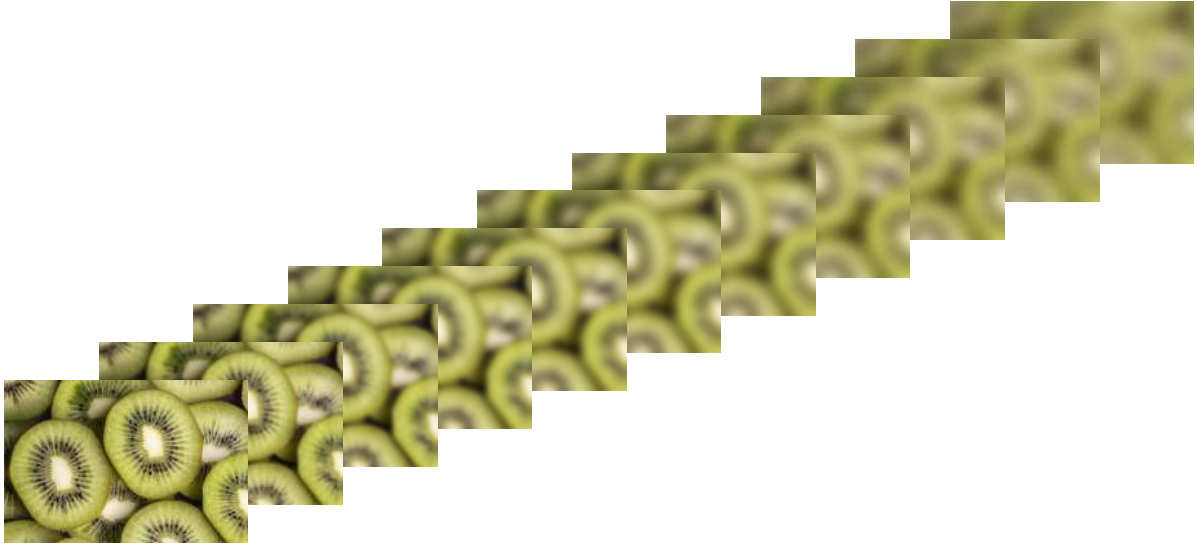
**Figure 1.3: Gaussian Scale Space.** Scale-space theory is concerned with multiscale representation of a signal or an image. One of the earliest scale spaces for images is the Gaussian scale space: It is obtained by convoluting a given image with a Gaussian kernel. In this example, the convolution of an image $f$ with a Gaussian kernel with variance $\sigma^2 = 0.1, 2.1...20.1$ **(left to right)** is shown. With growing variance, details in the image are progressively smoothed – which also effects edges in the image. With the help of convex, absolutely 1-homogeneous regularizers (such as total variation), scale spaces that preserve sharp edges can be defined – see Fig.1.4.

Input image by Freepik `https://www.freepik.com/free-photo/_13105039.htm`

to find a one-parameter family of images defined by some linear operator $L^t$, i.e., $\tilde{f} : \Omega \times \mathbb{R}^+ \to \mathbb{R}$,

$$\tilde{f}(x, t) = (L^t f)(x), \tag{1.13}$$

which fulfil constraints such as causality (features at coarse levels can be explained by features at fine levels), as well as homogeneity and isotropy (this can be understood as translation and rotational invariance in the spatial dimensions). The classical construction is to define such a one-parameter family $\tilde{f} : \Omega \times \mathbb{R}^+ \to \mathbb{R}$ as the solution of the heat equation, starting at the original image $f : \Omega \to \mathbb{R}$:

$$\frac{\partial}{\partial t} \tilde{f}(x, t) = \Delta \tilde{f}(x, t), \quad \tilde{f}(x, 0) = f(x). \tag{1.14}$$

Here $\Delta$ denotes the spatial domain Laplacian. The solution of this PDE is explicitly given as [Eva22, p. 46]

$$\tilde{f}(x, t) = (f * g_t)(x), \tag{1.15}$$

where the Gaussian kernel $g_t : \Omega \to \mathbb{R}$ is defined as

$$g_t(x) = \frac{1}{\left(\sqrt{4\pi t}\right)^n} \exp\left(\frac{-\|x\|_2^2}{4t}\right). \tag{1.16}$$
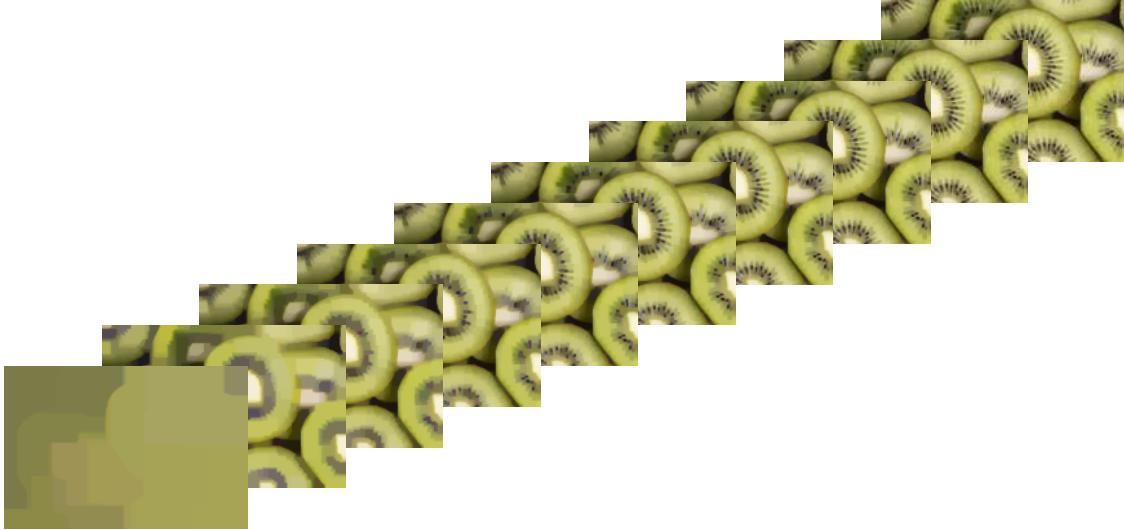
**Figure 1.4: TV-Based Inverse Scale Space.** Convex, absolutely 1-homogeneous regularizers such as the total variation can be used to define an inverse scale space. The most prominent difference to the Gaussian scale space – apart from the inverse behaviour – is that sharp edges are well preserved. This makes the TV-based inverse scale space interesting for image manipulation tasks such as scale-based feature enhancement or reduction; see also Fig. 1.5.

Input image by Freepik `https://www.freepik.com/free-photo/_13105039.htm`

In this context, the scale $t$ is equivalent to the variance $\sigma^2$ typically associated with the Gaussian kernel. For small $t$, $\tilde{f}$ is approximately the original input image. Due to the convolution with the Gaussian kernel, details are smoothed out with increasing $t$; see Fig. 1.3 for an example. Assume that we are interested in enhancing or filtering certain objects in the image, such as the black kiwi seeds. Although the seeds disappear in the Gaussian scale space for $t$ large enough, the approach does not appear to be well suited for the task: Scales with large $t$ are blurry and contain no sharp edges.

Let us recall the ROF denoising example (1.6). The TV regularizer allows sharp discontinuities in the solution. While the TV regularizer introduces a notion of coherent information, the regularization parameter introduces a notion of size. This motivates the following introduction of a scale space which evolves around convex, 1-homogeneous regularizers.

**Scale Spaces from Convex, 1-Homogeneous Regularizers.**  The existing theory of scale spaces for images from 1-homogeneous, convex regularizers is closely linked to variational problems

$$\inf_{u \in U} F(u), \quad F(u) = \frac{1}{2}\|f - u\|_2^2 + \lambda J(u), \quad (1.17)$$

where $f : \Omega \to \mathbb{R}$ is some input image, $\lambda > 0$ is a regularization parameter, and $J$ is a convex, absolutely 1-homogeneous regularizer; i.e., $J(\lambda u) = |\lambda| J(u)$. The data term $\frac{1}{2}\|f - u\|_2^2$ fits the solution to a

given input $f$ and the regularizer typically enforces smoothness and regularity.

The most intuitive way of generating a scale-space iteration is to iteratively solve (1.17) for increasing larger $\lambda_k$:

$$u_k \in \arg\min_{u \in U} \frac{1}{2}\|f - u\|_2^2 + \lambda_k J(u). \tag{1.18}$$

Interpreting $\lambda_k$ as time steps and passing informally to the continuous limit, this gives the following scale space flow for $t \in [0, T)$

$$\tilde{u}_{VM}(t) \in \arg\min_{u \in U} \frac{1}{2}\|f - u\|_2^2 + tJ(u). \tag{1.19}$$

Similar to the Gaussian scale space it holds $\tilde{u}_{VM}(0) = f$ under the assumption that $f \in U$. For finite $t < T$, the flow converges to the projection of $f$ onto the null space of the regularizer [Bur+16]. In the literature, this method is referred to as a *variational method* (VM) for generating a scale space flow.

Another approach of generating inverse scale-space iterations is to iteratively solve the minimization problem (1.17) for fixed $\lambda$ while replacing $f$ with the previous solution: For $u_0 := f$,

$$u_k \in \arg\min_{u \in U} \frac{1}{2}\|u_{k-1} - u\|_2^2 + \lambda J(u). \tag{1.20}$$

Passing informally to the continuous limit, this iteration leads to the *gradient flow* (GF) [Gil13]. Omitting the dependency on $x \in \Omega$ and assuming that $\tilde{u}_{GF}$ is sufficiently smooth, the associated PDE reads

$$\frac{\partial}{\partial t}\tilde{u}_{GF}(t) = -p_{GF}(t), \quad p_{GF}(t) \in \partial J(\tilde{u}_{GF}(t)), \quad \tilde{u}_{GF}(0) = f, \tag{1.21}$$

where $\partial J$ denotes the subdifferential of $J$. Similar to the (VM) setting, the flow converges to the projection of $f$ onto the null space of the regularizer for finite $t < T$ [Bur+16].

While we started with the above examples as they are the easiest to understand conceptually, we now shift the focus to an *inverse scale space* iteration/flow. An inverse scale space iteration can be achieved by solving the *Bregman iteration* [Osh+05; Bur+16]: Setting $p_0 \equiv 0$, the Bregman iteration is defined as

$$\text{For } k = 1, 2, \ldots \tag{1.22}$$

$$u_k = \arg\min_{u \in U} \frac{\lambda}{2}\|f - u\|_2^2 + J(u) - \langle p_{k-1}, u \rangle, \tag{1.23}$$

$$p_k \in \partial J(u_k). \tag{1.24}$$

**(a)** Original image　　　　　**(b)** Band-stop filtered　　　　　**(c)** Band-pass filtered

**Figure 1.5: Nonlinear Spectral Decompositions.** With the help of the inverse scale space flow a nonlinear spectral decomposition (1.27) is defined for images which then allows to define filters (1.28). The filters manipulate the nonlinear eigenfunctions of the regularizer which are contained in the input image according to their eigenvalue (scale). We here show the result of a total variation based band-stop **(b)** and band-pass **(c)** filter applied to the input image **(a)**. As indicator functions of certain convex sets are eigenfunctions of the 2D total variation regularizer [BCN02, Thm. 4] edges are well preserved by the filter.

Input image by Freepik `https://www.freepik.com/free-photo/_13105039.htm`

In case of the total variation regularizer, (1.24) can be replaced with the explicit form $p_k = p_{k-1} + \lambda(f - u_k) \in \partial J(u_k)$. Passing again informally to the continuous limit, omitting the dependency on $x \in \Omega$ and assuming that $p_{IS}$ is sufficiently smooth gives the *inverse scale space* flow (ISS) [Bur+16]:

$$\frac{\partial}{\partial s} p_{IS}(s) = f - \tilde{u}_{IS}(s), \quad p_{IS}(s) \in \partial J(\tilde{u}_{IS}(s)), \quad p(0) = 0. \quad (1.25)$$

In contrast to the scale space flows introduced above, now $\tilde{u}_{IS}(0)$ is the projection of $f$ onto the null space of the regularizer [Bur+16] – this "inverse" behavior explains the naming. An example with the total variation regularizer is shown in Fig. 1.4. Compared to the results of the Gaussian scale space in Fig. 1.3, the most prominent difference apart from the inverse behavior is that details in the different scales are less blurry and that edges appear to be sharper throughout the iteration – details appear in the flow based on their spatial size. Therefore, it can be argued that this scale space representation is inherently better suited for enhancing or extracting features of natural images.

**Nonlinear Spectral Representation and Filters.**　　If $f$ is a *nonlinear eigenfunction* of $J$ for the eigenvalue $\lambda$ (i.e., $\lambda f \in \partial J(f), \|f\| = 1$) with zero mean, it holds [Bur+16]

$$\tilde{u}_{IS}(s) = \begin{cases} 0, & \text{if } s \leq \lambda, \\ f, & \text{else.} \end{cases} \quad (1.26)$$

We see that $f$ appears precisely at $s = \lambda$ in the inverse scale space flow $\tilde{u}_{IS}$. This observation encouraged the definition of the

following nonlinear spectral decomposition of an arbitrary input image $f$ [Bur+16]:

$$f = \int_0^\infty \frac{\partial}{\partial s} \tilde{u}_{IS}(s) \, \mathrm{d}s. \tag{1.27}$$

Here, $\frac{\partial}{\partial s} \tilde{u}_{IS}(s) =: \mu_J^f(s)$ is a *data-dependent* vector-valued distribution called *weak spectral representation*. In analogy to the spectral decomposition of a bounded linear operator on a Hilbert space, one can think of $\mu_J^f$ as a decomposition of $f$ into nonlinear eigenfunctions of $J$ – but only those nonlinear eigenfunctions that are "contained" in the input image $f$. For a more thorough introduction, we refer to Sec. 2.3.

Features in the input image can be enhanced, reduced, or even removed by constructing a non-linear filter $\tilde{w}$ from a bounded, measurable map $w : \mathbb{R}_0^+ \to \mathbb{R}$ via

$$\tilde{w}(f) := \int_0^\infty w(s) \frac{\partial}{\partial s} \tilde{u}_{IS}(s) \, \mathrm{d}s. \tag{1.28}$$

An example is shown in Fig.1.5.

**Why Nonlinear Spectral Representation?**   Non-linear eigenfunctions are especially well studied for the TV regularizer. In [BCN02, Thm. 4] it is shown that indicator functions of certain bounded, convex sets $C \subset \mathbb{R}^2$ with finite perimeter are nonlinear eigenfunctions of the TV regularizer. This explains why the filter (1.28) can preserve edges well. Due to the definition of the weak spectral representation $\mu_J^f(s)$, the filter can furthermore be considered to be "data-driven", as $\mu_J^f(s)$ holds the information which nonlinear eigenfunctions of $J$ are contained in the input image $f$. Nonlinear spectral representation and the associated filters lead to visually compelling results [GMB16; HG18].

**Related Work and Contribution.**   Nonlinear spectral representation and related edge-preserving filters are not only of interest for images but also for related objects. Let us consider the example of image segmentation and the scale space of the segmentation maps.

The goal of (binary) image segmentation is to separate the domain $\Omega \subset \mathbb{R}^n$ of an image $f : \Omega \to \Gamma$ into non-overlapping regions $\Omega_1$ and $\Omega_2$ divided by a contour $C$ such that, e.g., the following energy

is minimized:

$$\arg \min_{\substack{C \subset \Omega, \\ c_1, c_2 \in \Gamma}} \int_{\Omega_1} (f(x) - c_1)^2 \, \mathrm{d}x + \int_{\Omega_2} (f(x) - c_2)^2 \, \mathrm{d}x$$

$$+ \lambda \mathcal{H}^{n-1}(C). \tag{1.29}$$

The contour $C$ can implicitly be represented by the zero level set of a Lipschitz function [CV01]. Representing the regions by a characteristic function $u$, one equivalently obtains

$$\arg \min_{\substack{u:\Omega \to \{0,1\} \\ c_1, c_2 \in \Gamma}} \int_{\Omega} (f(x) - c_1)^2 (1 - u(x))$$

$$+ (f(x) - c_2)^2 u(x) \, \mathrm{d}x + \lambda \, \mathrm{TV}(u). \tag{1.30}$$

In [Zeu+17], the authors extend this binary segmentation problem to a "multiscale" segmentation approach. With the help of the Bregman iteration, they generate an inverse scale space of segmentation maps in which objects are segmented according to their size and color. Using the notion of the weak spectral representation and related filters, they can then generate segmentation maps for different objects. In our kiwi example, this could mean a segmentation map for the kiwi slices, a separate segmentation map for black-colored kiwi seeds, and another segmentation map for light-colored kiwi seeds.

Considering the available literature, [Zeu+17] is the only publication that we are aware of that considers a setting other than the classical $L^2$ data term in the context of nonlinear spectral representation. To the best of our knowledge, the theory of nonlinear inverse scale spaces, weak spectral representation and related filters has also only been considered for images and the convex denoising/deconvolution problem. An interesting question is, whether the theory can be generalized to other image-related mathematical objects, such as the solution of variational problems

$$\inf_{u \in U} \{\lambda H(u) + J(u)\} \tag{1.31}$$

with convex, absolutely 1-homogeneous regularizer $J$ and *non-quadratic*, possibly even non-convex data term $H$. What is an inverse scale space of solutions of such problems? Can we define a nonlinear spectral decomposition of these solutions, e.g., of the depth map minimizing a stereo matching problem? In Chpt. 4 we make a contribution towards answering these questions by introducing *generalized inverse scale space iterations* or – for short – the *lifted Bregman iteration* (LBI).
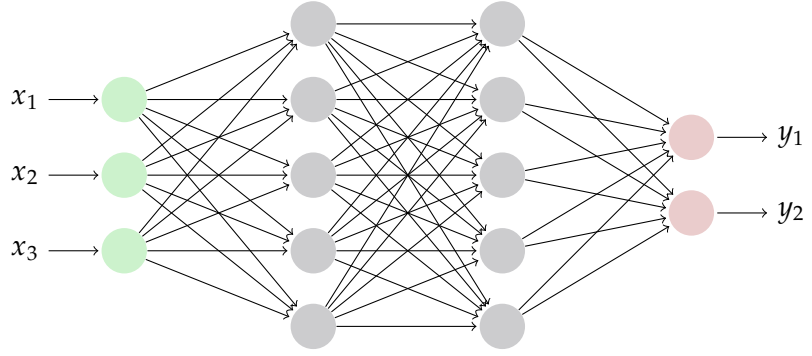
Reformulating (1.31) as

$$\inf_{v \in \mathscr{C}} \{\lambda \mathscr{H}(v) + \mathscr{J}(v)\} \tag{1.32}$$

using the theory of calibration-based lifting, we obtain a convex problem which we then apply the Bregman iteration to:

$$\text{For } k = 1, 2, \dots \tag{1.33}$$

$$v_k = \arg\min_{v \in \mathscr{C}} \lambda \mathscr{H}(v) + \mathscr{J}(v) - \langle \rho_{k-1}, v \rangle, \tag{1.34}$$

$$\rho_k \in \partial \mathscr{J}(v_k). \tag{1.35}$$

By investigating and comparing the subdifferential of the original and lifted total variation and linking elements of the former to the latter, we prove that the iterates $v_k$ obtained from the lifted Bregman iteration are in a certain sense equivalent to the ones from the original Bregman iteration in case of the convex ROF denoising problem. We also demonstrate the application of the lifted Bregman iteration to a non-convex problem and show experimentally that basic inverse scale space properties carry over to the non-convex setting.

### 1.2.3 Neural Fields and Learning

Over the past decades, learning frameworks have drastically risen in popularity. They have proven to perform extremely well on different imaging tasks such as image reconstruction, image restoration, image registration, image segmentation or object detection [LL19; Suz17; Zha+19]. Early learning approaches were mainly data-driven, which is also reflected in the availability of many huge databases, such as MNIST [LeC98], ImageNet [Den+09], COCO [Lin+14], CIFAR-10 and CIFAR-100 [KH+09] – each of them containing tens- to several hundred thousand images. This requirement for large amounts of training data becomes an issue, for example when the imaging process is expensive [RPK19], when privacy is an issue, or when data can only be measured sparsely due to technical

issues [Cai+21]. In recent years, hybrid data- and model-driven learning approaches, which incorporate physical knowledge of the problem in the loss function, have become increasingly popular. A prominent example is the recent development of *physics-informed neural networks* (PINNs) [Kar+21; Lu+21; Cuo+22] and *neural fields* (NFs) [Xie+22]. But what distinguishes PINNs and NFs from other learning approaches?

**Neural Networks.** A general neural network (see Fig. 1.6) can be viewed as a family of functions $F_\Theta : \mathbb{R}^n \to \mathbb{R}^d$ parameterized by a *weight vector* $\Theta$ and typically formed by a concatenation of *layers*. Each layer is an (often non-linear) function $f_i : \mathbb{R}^m \to \mathbb{R}$,

$$f_i(x; \theta_i) = \sigma_i \left( \sum_{j=1}^{m} a_{i,j} x_j - b_i \right), \tag{1.36}$$

where $\sigma_i$ is called *activation function* and $\theta_i = (a_{i,1}, ..., a_{i,m}, b_i)$ are the trainable *weights*. Concatenating all layers and writing $\Theta = (\theta_1, ..., \theta_p)$, a network can be represented by

$$F_\Theta(x) := F(x; \Theta) := f_p(...f_2(f_1(x; \theta_1); \theta_2); ...\theta_p). \tag{1.37}$$

The goal is to find the most suitable parameters for a given task by optimizing the parameters with respect to a suitable loss function.

A well-known problem in the context of neural networks is image classification: One is given a set of labelled images depicting instances of different classes (e.g., dog, cat, etc.) and the goal is to design and train a neural network which accepts a previously unknown image $x$ and outputs a scalar $F_\Theta(x)$ according to the class of the image. During the training process, *batches* of images are sampled from the data set, they are then passed through the network and the parameters of the network are updated according to a loss function such as the mean squared error between the respective expected and predicted classes.

Input and output spaces as well as loss functions are as diverse as the applications of neural networks. In this work we focus on *coordinate-based neural networks*.

**Coordinate-Based Neural Networks.** These networks accept low-dimensional coordinates, e.g., $x \in \Omega \subset \mathbb{R}^2$, as input and predict some physical value for the given coordinates.

*Physics-informed neural networks* (PINNs), for example, are used to find the solution to a partial differential equation (PDE). They accept

coordinates in the domain of the solution as input and are trained with the goal to output the value of the solution at these coordinates. The loss function is governed by the PDE, and in their simplest form the networks are trained for one specific problem instance and do not generalize to different initial or boundary values. There is a close connection to classical optimization: The neural networks can be understood as a nonlinear approximation/parameterization of the solution of the PDE, and the training process amounts to – usually stochastic – optimization of the parameters of this approximation.

In the imaging community, coordinate-based neural networks are typically referred to as *neural fields* (NFs). For illustration, consider the simple variational problem

$$\inf_{u \in U} F(u), \quad F(u) = \int_{\Omega} f(x, u(x)) \, \mathrm{d}x. \tag{1.38}$$

Instead of computing a minimizer $u^*$ using a classical optimization algorithm, we could design and train a neural network to learn an implicit, coordinate-free representation of the solution $u^*(x) \approx F(x; \Theta^*)$. This could be achieved by the following iterative training procedure:

1. Randomly select a *batch* $b_k = \{x_1, \dots, x_l \mid x_i \in \Omega\}$ of $l$ coordinates from the image domain.
2. Calculate the current loss value – in our example:

$$L(\Theta^{k-1}; b_k) = \frac{1}{l} \sum_{i=1}^{l} f(x_i, F(x_i; \Theta^{k-1})). \tag{1.39}$$

3. Update the parameters, using for example the *gradient descent* algorithm
$$\Theta^k = \Theta^{k-1} - \alpha \nabla_{\Theta} L(\Theta^{k-1}; b_k), \tag{1.40}$$

   where $\alpha > 0$ is the step size.

In effect, coordinate-based learning approaches constitute a way of – non-linearly! – parameterizing the function space of possible solutions and learning the parameters in a (stochastic) optimization process in order to find the minimizer of the objective function within this set. In addition to the nonlinear parameterization, these approaches typically rely on stochastic optimization as outlined above. Usually, the batch size $l$ is chosen rather small during training in order to speed up the second and third step in each iteration. After the training is completed, one can then choose an arbitrary fine grid on which the network is evaluated – if a grid representation is required.

Vanilla coordinate-based learning approaches *do not generalize* to a new problem; if the loss function depends on some input data, the network needs to be retrained in case of new input data. In this sense they can also be viewed as classic (stochastic) optimization. The learning and automatic differentiation frameworks open more possibilities, such as introducing *latent variables* that allow to learn the shape of the solution space also depending on the input image and other properties.

Neural fields currently celebrate a huge success in imaging. They are particularly popular in the form of neural radiance fields (NeRFs) [Mil+21], which allow to synthesize novel views of 3D scenes by learning their so-called radiance field, which predicts the light emission for every point in space and emission direction.

Among the model-driven, coordinate-based learning frameworks, we are aware of two examples that include embedding strategies in the formulation of the loss function [MC19a; Pal+22]. This helped to inspire our approach, which we detail below.

**Motivation and Contribution.** We consider variational problems of the form (1.3), which are separable into a (possibly non-convex) data term and TV regularizer (1.4), and employ the calibration-based lifting approach in order to reach a primal-dual problem of the form

$$\inf_{v \in \mathcal{C}} \sup_{\varphi \in \mathcal{K}} \left\{ \int_{\Omega \times \Gamma} \langle \varphi, \mathrm{d}Dv \rangle \right\}, \tag{1.41}$$

where $Dv$ denotes the distributional derivative of $v$, and where $v$ and $\varphi$ are defined over $\Omega \times \Gamma$. Typically, a minimizer is computed with the primal-dual gradient descent algorithm (PDHG) [Poc+09; Poc+10] using a so-called *sublabel-accurate* discretization scheme [Möl+15; MC17].

As neural fields have already proven to be quite successful in solving various imaging problems such as novel view synthesis [Mil+21] and shape reconstruction [Pal+22] (see also [Xie+22] for a comprehensive overview), we propose a neural fields based learning framework for solving the embedded primal-dual problem (1.41). The approach amounts to training (stochastically optimizing) multiple neural fields representing the primal and dual variable and results in a nonlinear approximation of the solutions $v^*$ and $\varphi^*$ over the continuous domain $\Omega \times \Gamma$.

## 1.3 Outline

**In chapter 2** we cover the required mathematical preliminaries. In particular, we recall notions and results concerning convex analysis, as well as the total variation regularizer and associated space of functions of bounded variation, which play a major role throughout the following chapters. Furthermore, we present an overview of spectral theory for convex 1-homogeneous functionals.

**In chapter 3** the calibration-based lifting approach for scalar variational problems is established. We aim to present a self-contained derivation of the approach and summary of some important properties while extending certain results found in the literature to a broader class of problems. While the derivation of the lifted formulation roots in the field of variational analysis, we also discuss discretization and optimization schemes for the lifted problem.

**In chapter 4** the calibration-based lifting approach is then used as a means to define a nonlinear inverse scale space iteration for scalar, possibly non-convex problems with total variation regularizer. We show that our approach directly relates to the existing approach in case of the convex denoising problem and, furthermore, expands the area of application to non-convex problems.

Related own publications:

► D. Bednarski and J. Lellmann. 'Inverse Scale Space Iterations for Non-convex Variational Problems Using Functional Lifting'. In: *Scale Space and Variational Methods in Computer Vision*. Ed. by A. Elmoataz, J. Fadili, Y. Quéau, J. Rabin, and L. Simon. Cham: Springer International Publishing, 2021, pp. 229–241. https://doi.org/10.1007/978-3-030-75549-2_19 **Winner of best student paper award at SSVM 2021.**

*Attribution:* The author of this thesis developed the theory, performed the computations and wrote the manuscript under supervision of J.L.

► D. Bednarski and J. Lellmann. 'Inverse Scale Space Iterations for Non-Convex Variational Problems: The Continuous and Discrete Case'. *Journal of Mathematical Imaging and Vision* **65**, 2022, 124–139. https://doi.org/10.1007/s10851-022-01125-8

*Attribution:* The author of this thesis developed the theory, performed the computations and wrote the manuscript under supervision of J.L.

**In chapter 5** we propose a learning strategy for solving scalar variational problems with TV regularizer: First, a given energy is lifted with the help of the calibration-based lifting. Then, multiple neural fields are stochastically optimized in order to reach a non-linear parametric approximation of the primal and dual minimizers. The proposed framework can easily be adjusted for different possibly non-convex data terms.

Related own publications:

► D. Bednarski and J. Lellmann. 'EmNeF: Neural Fields for Embedded Variational Problems in Imaging'. In: *Scale Space and Variational Methods in Computer Vision*. Ed. by L. Calatroni, M. Donatelli, S. Morigi, M. Prato, and M. Santacesaria. Cham: Springer International Publishing, 2023, pp. 137–148. https://doi.org/10.1007/978-3-031-31975-4_11

*Attribution:* The author of this thesis developed the theory, performed the computations and wrote the manuscript under supervision of J.L.

**Addendum** The following preprint was written during the PhD period but is not part of this thesis:

► T. Vogt, R. Haase, D. Bednarski, and J. Lellmann. 'On the Connection between Dynamical Optimal Transport and Functional Lifting'. In: *arXiv preprint arXiv:2007.02587* (2020) https://arxiv.org/abs/2007.02587

*Attribution:* T.V. developed the theory and implemented the approach. R.H. extended the theory and performed further computations. T.V., R.H., and the author of this thesis wrote the final manuscript. The project was carried out under supervision of J.L.

# Notation and Mathematical Preliminaries | 2

In this chapter, we recall various definitions and results from the fields of convex analysis and measure theory. These concepts are vital to understanding the continuous and discrete version of the calibration-based lifting approach and we, therefore, give them space in the main part of this work. Furthermore, we provide a short introduction to spectral theory as this motivates the generalized inverse scale space iteration which we propose later.

## 2.1 Convex Analysis

In the following, we consider functions $f : \mathbb{R}^n \to \overline{\mathbb{R}}$. Here, $\overline{\mathbb{R}} := \mathbb{R} \cup \{-\infty, +\infty\}$ denotes the extended real line. We denote by $\mathrm{dom} f$ the *effective domain* of a function, i.e., the set $\{x \in \mathbb{R}^n : f(x) < +\infty\}$ [RW09, p. 5]. A function $f$ is called *proper*, if $f(x) < \infty$ for at least one $x \in \mathbb{R}^n$ and $f(x) > -\infty$ for all $x \in \mathbb{R}^n$ [RW09, p. 5]. The *convex hull* $\mathrm{con} f$ of a function $f$ is the largest convex function majorized by $f$, i.e., $\mathrm{con} f(x) \leq f(x)$ for all $x \in \mathbb{R}^n$ [RW09, Prop. 2.31]. A function $f$ is called *lower semi-continuous* if for all $x \in \mathbb{R}^n$ it holds $f(x) \leq \liminf_{y \to x} f(y)$ [Roc70, p. 51]. The *lower semi-continuous hull* of a function $f$ is the largest lower semi-continuous function majorized by $f$ [Roc70, p. 52].

**Duality Result.** In the following, we introduce special functions from the theory of convex analysis, leading up to a duality result which will play a vital role in the calibration-based lifting approach.

---

**Definition 2.1.1** (Fenchel Conjugate [RW09, Chpt. 11.A]) *For a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, the* (Fenchel) *conjugate $f^* : \mathbb{R}^n \to \mathbb{R}$ is defined as*

$$f^*(m) := \sup_{x \in \mathbb{R}^n} \{\langle x, m \rangle - f(x)\}. \tag{2.1}$$

---

The conjugate $f^{**}$ of $f^*$ is called the *biconjugate*; see Fig. 2.1 for a 1D example. For the biconjugate it always holds $f^{**} \leq f$. In fact, if the convex hull $\mathrm{con} f$ is proper, then $f^{**}$ is the closure of $\mathrm{con} f$ [RW09, Thm. 11.1]. Many further interesting properties of the Fenchel conjugate can be found in [RW09, Chpt. 11.A.].
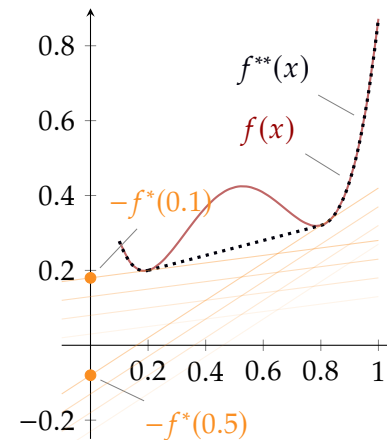


**Figure 2.1: Fenchel Conjugate.** The non-convex function $f : \mathbb{R} \to \mathbb{R}$ **(red line)** is approximated from below by affine functions $y_i = \langle m_i, x \rangle + t_i$ with slopes $m_i \in \mathbb{R}$ and intercepts $t_i \in \mathbb{R}$ **(orange lines)**. The largest intercept $t_i$ for which the affine function with slope $m_i$ lies completely below $f$ is the negative conjugate $-f^*(m_i)$ **(orange dots)**. The supremum over all such affine functions is the biconjugate $f^{**}(x) = \sup_{m_i \in \mathbb{R}} \langle m_i, x \rangle - f^*(m_i)$ **(blue dotted line)**.
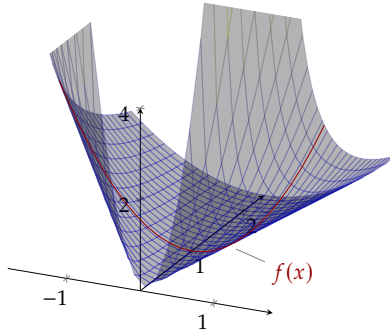
**Figure 2.2: Perspective Function.**
The function $f(x) = x^2$ **(red)** and its perspective function $\tilde{f}(x, \lambda) = \frac{x^2}{\lambda}$ **(blue)**.

Later, we will be interested in sets $\{(m, t) | t \leq f^*(m)\} \subset \mathbb{R}^{n+1}$, or more precisely in the support functions of such sets. For some set $C \subset \mathbb{R}^n$, the *support function* $\sigma_C : \mathbb{R}^n \to \overline{\mathbb{R}}$ is defined as $\sigma_C(x) := \sup_{y \in C} \langle x, y \rangle$.

Let us recall the definition of the *perspective* and *recession* functions.

**Definition 2.1.2** (Perspective Function [HL96, p. 160]) *For a proper function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, the* perspective function $\tilde{f} : \mathbb{R}^n \times \mathbb{R} \to \overline{\mathbb{R}}$ *is defined as*

$$\tilde{f}(x, \lambda) := \begin{cases} \lambda f(\frac{x}{\lambda}), & \text{if } \lambda > 0, \\ +\infty, & \text{otherwise.} \end{cases} \tag{2.2}$$

Fig. 2.2 depicts the perspective function in a simple 1D setting. The perspective of a convex function in $\mathbb{R}^n$ is convex in $\mathbb{R}^{n+1}$ [HL96, Prop. 2.2.1].

**Definition 2.1.3** (Recession Function) *For a proper function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, the* recession function $f^\infty : \mathbb{R}^n \to \overline{\mathbb{R}}$ *is defined as*

$$f^\infty(x) := \lim_{\lambda \to +\infty} \frac{1}{\lambda} f(\lambda x). \tag{2.3}$$

If $f$ is proper, convex, and lower semi-continuous, then $f^\infty$ is also proper, convex, lower semi-continuous, and, furthermore, positively 1-homogeneous [Roc70, Thm. 8.5, Cor. 8.5.2, p. 35, p. 52]. Furthermore, the lower semi-continuous hull $h : \mathbb{R}^n \times \mathbb{R} \to \overline{\mathbb{R}}$ of the perspective function of $f$ is given by [Roc70, p. 67, p. 52]

$$h(x, \lambda) = \begin{cases} \lambda f\left(\frac{x}{\lambda}\right), & \text{if } \lambda > 0, \\ \lim_{\mu \to +\infty} \frac{1}{\mu} f(\mu x), & \text{if } \lambda = 0, \\ +\infty, & \text{if } \lambda > 0. \end{cases} \tag{2.4}$$

**Corollary 2.1.4** ([Roc70, Cor. 13.5.1]) *Let* $f : \mathbb{R}^n \rightarrow \mathbb{R}$ *be a proper, convex, and lower semi-continuous function. The lower semi-continuous hull of the perspective function of $f$ is the support function of the set*

$$K := \{(m, t) \mid t \leq -f^*(m)\} \subset \mathbb{R}^{n+1}. \tag{2.5}$$

*Specifically, with $h : \mathbb{R}^{n+1} \rightarrow \overline{\mathbb{R}}$ defined as in (2.4), it holds*

$$h(x, \lambda) = \sigma_K((x, \lambda)) = \sup_{(m,t) \in K} \left\langle \begin{pmatrix} x \\ \lambda \end{pmatrix}, \begin{pmatrix} m \\ t \end{pmatrix} \right\rangle. \tag{2.6}$$

This duality result will be of central interest in the calibration-based lifting approach. There, after careful considerations involving set-valued mappings and continuous selections, the (finite-dimensional) duality result is used to reformulate a primal variational problem in terms of a primal-dual variational problem.

**Convex, 1-Homogeneous Functionals.** One of the most important function classes considered in this work is the one of convex, 1-homogeneous regularizers, such as the total variation.

Let $V$ denote a locally convex vector space and $V^*$ its topological dual. From now on, we consider $F : V \rightarrow \overline{\mathbb{R}}$. $F$ is called *absolutely 1-homogeneous* if

$$F(\lambda u) = |\lambda| F(u) \text{ for all } \lambda \in \mathbb{R}, \tag{2.7}$$

and *positively 1-homogeneous* if

$$F(\lambda u) = \lambda F(u) \text{ for all } \lambda > 0. \tag{2.8}$$

Note that we do not require $F$ to be continuous. Therefore, we use a more general definition of differentiability:

**Definition 2.1.5** (Subdifferential [ET99, Def. 5.1]) *A function $F : V \rightarrow \overline{\mathbb{R}}$ is said to be subdifferentiable at $u \in V$ if it has a continuous affine minorant which is exact at $u$. The slope $p \in V^*$ of such a minorant is called subgradient of $F$ at $u$ and the set of subgradients at $u$ is called the subdifferential at $u$. To summarize, if $F$ is finite at $u$, the subdifferential is given by*

$$\partial F(u) = \{p \in V^* : F(u) + \langle v - u, p \rangle \leq F(v) \, \forall v \in V\}. \tag{2.9}$$

We now pass to functionals $J$ which are defined on some Banach space $\mathcal{U}$. The subdifferential gives rise to the following definition of *non-linear eigenfunctions* of some convex functional:

---

**Definition 2.1.6** (Non-linear eigenfunction) *Let $\mathcal{U}$ denote some Banach space and let $J : \mathcal{U} \to \overline{\mathbb{R}}$ be a convex functional. We call u a* non-linear eigenfunction *of J with* eigenvalue $\lambda$ *if*

$$\lambda u \in \partial J(u). \qquad (2.10)$$

---

In Chpt. 4 we are interested in a *spectral representation* of image related objects that heavily relies on the non-linear eigenfunctions of some convex, 1-homogeneous regularizer.

## 2.2 Functions of Bounded Variation

This section summarizes important properties of the total variation and the space of functions of (special) bounded variation. Results related to measure theory will play a vital role in the calibration based lifting approach presented in chapter 3, where the main idea is to reformulate variational problems over some function space as variational problems over the subgraphs of functions in the original solution space.

**General Setting and Notation.** Let $(X, \mathcal{A})$ denote a measure space. We call $\mu : \mathcal{A} \to \mathbb{R}^m$ a *(vector-valued) measure* if $\mu(\emptyset) = 0$ and $\mu$ is $\sigma$-additive, i.e., for any sequence $(A^k)$ of pairwise disjoint elements in $\mathcal{A}$ it holds $\mu\left(\bigcup_{k=0}^{\infty} A^k\right) = \sum_{k=0}^{\infty} \mu(A^k)$ [AFP00, Def. 1.4a)].

---

**Definition 2.2.1** (Total Variation (I) [AFP00, Def. 1.4b]) *Let $(X, \mathcal{A})$ denote a measure space and $\mu : \mathcal{A} \to \mathbb{R}^m$ a vector-valued measure. The* total variation *of $\mu$ is defined as*

$$|\mu|(A) := \sup_{(A^k)_{k\in\mathbb{N}} \in \mathcal{K}} \left\{ \sum_{k=0}^{\infty} \left\| \mu\left(A^k\right) \right\| \right\}, \qquad (2.11)$$

$$\mathcal{K} := \left\{ \left(A^k\right)_{k\in\mathbb{N}} \,\middle|\, A^k \in \mathcal{A} \text{ pairwise disjoint}, A = \bigcup_{k=0}^{\infty} A^k \right\},$$

*for every $A \in \mathcal{A}$.*

---

There is some ambiguity in the choice of the inner norm $\|\cdot\|$ on $\mathbb{R}^m$; we will usually use $\|\cdot\| = \|\cdot\|_2$ unless specified otherwise.

For a locally compact, separable metric space $X$ we denote by $\mathscr{B}(X)$ its Borel $\sigma$-algebra, i.e., the smallest $\sigma$-algebra in $X$ that contains all open subsets of $X$. Furthermore, we denote by $(X, \mathscr{B}(X))$ the associated measure space. Following [AFP00, Def.1.40], a positive measure on $(X, \mathscr{B}(X))$ is called *Borel measure*. If such a measure is finite on compact sets, it is called *positive Radon measure* $\mathscr{M}^+(X)$. By *vector-valued Radon measure* $\mathscr{M}_{\mathrm{loc}}(X; \mathbb{R}^m)$, we refer to functions defined on the relatively compact Borel subsets of $X$ which are measures on $(K, \mathscr{B}(K))$ for every compact $K \subseteq X$. If the function is a measure on $(X, \mathscr{B}(X))$, we call it a *finite vector-valued Radon measure* $\mathscr{M}(X; \mathbb{R}^m)$.

Let $X, Y$ denote metric spaces. A function $f : X \to Y$ is called a *Borel function*, if $f^{-1}(Z) \in \mathscr{B}(X)$ for every open set $Z \subset Y$ [AFP00, Def. 1.42].

Let $\Omega \subseteq \mathbb{R}^n$ be open. We denote by $C_c(\Omega; \mathbb{R}^m)$ the space of $\mathbb{R}^m$-valued continuous functions with compact support in $\Omega$ and by $C_0(\Omega; \mathbb{R}^m)$ the completion with respect to the norm

$$\|f\|_\infty := \inf_{C \in [0, \infty]} \{\|f(x)\| \le C \text{ for } \mu - \text{a.e. } x \in \Omega\}; \qquad (2.12)$$

see also [AFP00, p. 9, p. 25]. According to Riesz's Theorem [AFP00, Thm. 1.54 and Remark 1.57], the dual space of $C_0(\Omega; \mathbb{R}^m)$ is $\mathscr{M}(X; \mathbb{R}^m)$ with the dual pairing

$$\langle f, \mu \rangle := \sum_{h=1}^m \int_\Omega f_h \, \mathrm{d}\mu_h. \qquad (2.13)$$

A Lebesgue-measurable function $f : \Omega \to \mathbb{R}^m$ is called *locally integrable*, if for every compact set $K \subset \Omega$ it holds $\int_K \|f(x)\| \, \mathrm{d}x < \infty$. We write $f \in L^1_{\mathrm{loc}}(\Omega; \mathbb{R}^m)$.

**Functions of Bounded Variation.** In mathematical image processing, a common approach is to think of images as piecewise smooth functions with sharp discontinuities, as the latter are essential for modelling objects with sharp edges. A popular function space in this context is the space of *functions of bounded variation*.

**Definition 2.2.2** (Space of Functions of Bounded Variation – inspired by [AFP00, Def. 3.1]) *We define the space of* functions of bounded variation $BV(\Omega; \mathbb{R}^m)$ *as the set of all functions* $u \in L^1(\Omega; \mathbb{R}^m)$ *whose distributional Jacobian is representable as a finite vector-valued Radon measure* $Du \in \mathcal{M}(\Omega; \mathbb{R}^{m,n})$, *i.e., for all* $\varphi = (\varphi_1, \ldots, \varphi_m)^\top \in C_c(\Omega; \mathbb{R}^{m,n})$ *it holds*

$$\sum_{i=1}^m \int_\Omega u_i \operatorname{div} \varphi_i \, dx = -\sum_{i=1}^m \sum_{j=1}^n \int_\Omega \varphi_{i,j} \, dD_j u_i. \tag{2.14}$$

Next, we will define the *total variation* TV for vector-valued $L^1_{\mathrm{loc}}$ functions. A function $u \in L^1(\Omega; \mathbb{R}^m)$ is of *bounded variation*, if and only if its total variation $\mathrm{TV}(u)$ is finite; we write $u \in \mathrm{BV}(\Omega; \mathbb{R}^m)$ [AFP00, Prop. 3.6]. Similarly, a function $u \in L^1_{\mathrm{loc}}(\Omega; \mathbb{R}^m)$ is called *locally of bounded variation*, if its total variation is finite for all open sets $A \subsetneq \Omega$ with compact closure $\bar{A} \subset \Omega$; in this case we write $u \in \mathrm{BV}_{\mathrm{loc}}(\Omega; \mathbb{R}^m)$ [EG15, p. 167].

**Definition 2.2.3** (Total Variation (II) [AFP00, Def. 3.4]) *The* total variation *of a function* $u \in L^1_{loc}(\Omega; \mathbb{R}^m)$ *is defined as*

$$\mathrm{TV}(u) := \sup_{\varphi \in \mathcal{K}} \left\{ -\sum_{i=1}^m \int_\Omega u_i(x) \operatorname{div} \varphi_i(x) \, dx \right\}, \tag{2.15}$$

$$\mathcal{K} := \{\varphi \in C_c(\Omega; \mathbb{R}^{m,n}) \mid \|\varphi(x)\|_F \leq 1, \, \forall x \in \Omega\}. \tag{2.16}$$

The total variation is convex, positively 1-homogeneous, and lower semi-continuous in $\mathrm{BV}(\Omega; \mathbb{R})$ with respect to the $L^1_{\mathrm{loc}}$-topology [AFP00, Prop. 3.6]. The space $\left(\mathrm{BV}(\Omega; \mathbb{R}^m), \|\cdot\|_{\mathrm{BV}(\Omega; \mathbb{R}^m)}\right)$, with

$$\|u\|_{\mathrm{BV}(\Omega; \mathbb{R}^m)} := \|u\|_{L^1(\Omega; \mathbb{R}^m)} + TV(u), \tag{2.17}$$

is a Banach space [ABM14, Thm. 10.1.1].

The total variation of a function $u \in \mathrm{BV}(\Omega; \mathbb{R}^m)$ can also be expressed in terms of the total variation of its distributional Jacobian $Du \in \mathcal{M}(\Omega; \mathbb{R}^{m,n})$ [AFP00, Prop. 3.6]:

$$TV(u) = |Du|(\Omega) = \int_\Omega \mathrm{d}|Du|. \tag{2.18}$$

**Compactness Results.**   The topology induced by the $\|\cdot\|_{\mathrm{BV}}$ norm is often too strong and may not allow the existence of minimizing sequences. Therefore, one typically uses the following definition of weak*-convergence:

**Definition 2.2.4** (Weak*-Convergence [AFP00, Def. 3.11, Prop. 3.13]) *We say that a sequence $(u^k)_{k\in\mathbb{N}}$ with $u^k \in \mathrm{BV}(\Omega;\mathbb{R}^m)$ weak\*-converges to a function $u \in \mathrm{BV}(\Omega;\mathbb{R}^m)$ iff $u^k \to u$ in $L^1(\Omega;\mathbb{R}^m)$ and either of these conditions is fulfilled:*

a) *$Du^k$ weakly\* converges to $Du$ in $\Omega$, i.e.,*

$$\lim_{k\to\infty} \int_\Omega \varphi\, dDu^k = \int_\Omega \varphi\, dDu, \ \forall\varphi \in C_0(\Omega,\mathbb{R}^{m,n}), \quad (2.19)$$

b) *$(u^k)_{k\in\mathbb{N}}$ is bounded in $\mathrm{BV}(\Omega;\mathbb{R}^m)$, i.e.,*

$$\sup_{k\in\mathbb{N}} \left\{ \|u^k\|_{\mathrm{BV}} \right\} < \infty. \quad (2.20)$$

This definition allows the following compactness result:

**Theorem 2.2.5** (Compactness [AFP00, Thm. 3.23, Prop. 3.21])

▶ *Let $\Omega$ be open. Every sequence $(u^h)_{h\in\mathbb{N}}$, $u^h \in \mathrm{BV}_{loc}(\Omega;\mathbb{R}^m)$ for which*

$$\sup_{h\in\mathbb{N}} \left\{ \int_A \|u^h\|\, dx + |Du_h|(A) \right\} < \infty, \quad (2.21)$$

*for all open sets $A \subsetneq \Omega$ with compact closure $\bar{A} \subset \Omega$ admits a subsequence which converges in $L^1_{loc}(\Omega;\mathbb{R}^m)$ to $u \in \mathrm{BV}_{loc}(\Omega;\mathbb{R}^m)$.*

▶ *Let $\Omega$ be open with compact Lipschitz boundary. Every sequence $(u^h)_{h\in\mathbb{N}}$, $u^h \in \mathrm{BV}(\Omega;\mathbb{R}^m)$, that is bounded in $\mathrm{BV}(\Omega;\mathbb{R}^m)$ admits a subsequence that weakly\*-converges to some $u \in \mathrm{BV}(\Omega;\mathbb{R}^m)$.*

**Decomposition Results and Space of** SBV**.**   The measure $Du$ can uniquely be decomposed into three parts. Before looking at this decomposition, we define the *approximate discontinuity* and *jump sets* associated with functions in $L^1_{\mathrm{loc}}$, which are needed later on for the decomposition.

In the following definition we denote by $\mathscr{B}_r(x)$ a ball of radius $r$ centered around $x$ and by $\fint_C u(x)\, dx$ the mean value of $u$ on $C$.
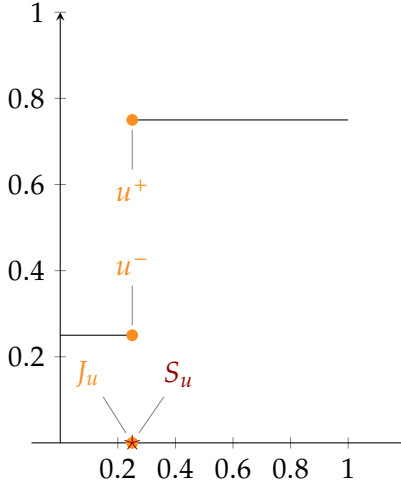
**Figure 2.3: Jump Points.** In this example, $u : (0,1) \to \mathbb{R}$ has one approximate jump point $J_u = \{0.25\}$. The function jumps from $u^- = 0.25$ to $u^+ = 0.75$. Furthermore, the function has no approximate limit at this point, i.e., also $S_u = \{0.25\}$.

**Definition 2.2.6** (Approximate Discontinuity, Jump Set and Differential [AFP00, Def. 3.63, Def. 3.67, Def. 3.70]) *Let $u \in L^1_{loc}(\Omega; \mathbb{R}^m)$.*

1. *We say that $u$ has an* approximate limit *at $x \in \Omega$, if there exists an $a \in \mathbb{R}^m$ such that*

$$\lim_{r \searrow 0} \fint_{\mathscr{B}_r(x)} |u(y) - a| dy = 0. \qquad (2.22)$$

*The* approximate discontinuity set $S_u$ *is the set of points that do* not *have this property.*

2. *We say that $u$ has an* approximate jump point *at $x \in \Omega$, if there exist $u^+, u^- \in \mathbb{R}^m$, $u^+ > u^-$, and $\nu_u \in \mathcal{S}^{n-1}$ such that*

$$\lim_{r \searrow 0} \fint_{\mathscr{B}_r^+(x,\nu_u)} |u(y) - u^+| dy = 0, \qquad (2.23)$$

$$\lim_{r \searrow 0} \fint_{\mathscr{B}_r^-(x,\nu_u)} |u(y) - u^-| dy = 0, \qquad (2.24)$$

*where*

$$\mathscr{B}_r^+(x, \nu_u) := \{y \in \mathscr{B}_r(x) | \langle y - x, \nu_u \rangle > 0\}, \qquad (2.25)$$
$$\mathscr{B}_r^-(x, \nu_u) := \{y \in \mathscr{B}_r(x) | \langle y - x, \nu_u \rangle < 0\}. \qquad (2.26)$$

*The* set of approximate jump points *is denoted by $J_u$.*

3. *We say $u$ is* approximately differentiable *at $x \in \Omega \backslash S_u$ if there exists an $m \times N$ matrix $L$ such that*

$$\lim_{r \searrow 0} \fint_{\mathscr{B}_r(x)} \frac{|u(y) - \tilde{u}(x) - L(y - x)|}{r} dy = 0, \qquad (2.27)$$

*where $\tilde{u}(x)$ denotes the approximate limit. The matrix $L$ is called* approximate differential *of $u$ at $x$.*

Fig. 2.3 shows a simple example. It can be shown that $S_u$ is an $\mathscr{L}^n$-negligible Borel set and $u|_{\Omega \backslash S_u}$ is a Borel function [AFP00, Prop. 3.64].

**Theorem 2.2.7** (Radon-Nikodým Decomposition of $Du$ [AFP00, Def. 1.24, Thm. 1.28, Thm. 3.83]) *The Radon-Nikodým decomposition of the distributional derivative $Du \in \mathcal{M}(\Omega; \mathbb{R}^{m,n})$ of a function $u \in L^1(\Omega; \mathbb{R}^m)$ with respect to $\mathscr{L}^n$ is given by*

$$Du = D^a u + D^s u, \tag{2.28}$$

*where $D^a u, D^s u$ are $\mathbb{R}^{m,n}$-valued measures. Furthermore, $D^a u$ is absolutely continuous with respect to $\mathscr{L}^n$ (write $D^a u \ll \mathscr{L}^n$) and $D^s u$ and $\mathscr{L}^n$ are mutually singular (write $D^s u \perp \mathscr{L}^n$). The density of the absolutely continuous part of $D^a u$ with respect to $\mathscr{L}^n$ is the approximate differential $\nabla u$, i.e.,*

$$D^a u = \nabla u \mathscr{L}^n \llcorner \Omega. \tag{2.29}$$

By this decomposition and the definition of BV, any $u \in \mathrm{BV}(\Omega; \mathbb{R}^m)$ with $D^s u = 0$ is also in $W^{1,1}(\Omega; \mathbb{R}^m)$. One can, furthermore, show that the Sobolev space $W^{1,1}(\Omega; \mathbb{R}^m)$ is in fact a subspace of $\mathrm{BV}(\Omega; \mathbb{R}^m)$: Denoting by $Gu$ the (weak) Jacobian of $u$ one can use $Gu\mathscr{L}^n =: Du$ as distributional Jacobian of $u$. $Du$ is of bounded variation, as $Gu \in L^1(\Omega; \mathbb{R}^m)$ by definition.

Now we can use the above introduced sets $S_u$ and $J_u$ to further decompose the singular part into $D^s u = D^j u + D^c u$, with [AFP00, Def. 3.91]

$$D^j u := D^s u \llcorner J_u \quad \text{and} \quad D^c u := D^s u \llcorner (\Omega \backslash S_u). \tag{2.30}$$

The jump part $D^j u$ is concentrated on the approximate jump points of $u$. Using the notation from Def. 2.2.6, it can be expressed as [AFP00, Thm. 3.77, p. 184]

$$D^j u = (u^+ - u^-) \otimes \nu_u \mathscr{H}^{n-1} \llcorner J_u. \tag{2.31}$$

**Definition 2.2.8** (Space of Special Functions of Bounded Variation [AFP00, p. 212]) *We say a function $u \in \mathrm{BV}(\Omega; \mathbb{R}^m)$ is a* special function of bounded variation *and write $u \in \mathrm{SBV}(\Omega; \mathbb{R}^m)$ if the Cantor part of its derivative $D^c u$ is zero.*

Following (2.18) and the above decomposition, the total variation can be rewritten as

$$TV(u) = |Du| = |D^a u| + |D^j u| + |D^c u|. \tag{2.32}$$

Let us close this paragraph by recalling the *Besicovitch derivation theorem*. The latter is closely related to the Radon-Nikodým decomposition and gives a concrete representation for the density of a vector-valued Radon measure with respect to a positive Radon measure. Later, we will use the Besicovitch derivation theorem in order to formulate the density of $Du$ with respect to $|Du|$.

**Theorem 2.2.9** (Besicovitch Derivation Theorem [AFP00, Thm. 2.22]) *Let $\Omega \subset \mathbb{R}^n$ be open and $\mu \in \mathcal{M}^+(\Omega; \mathbb{R})$, $\nu \in \mathcal{M}_{loc}(\Omega; \mathbb{R}^n)$ be Radon measures. Then for $\mu-$a.e. $x \in \text{supp } \mu$ the limit*

$$\nu_u(x) := \lim_{\rho \searrow 0} \frac{\nu(\mathcal{B}_\rho(x))}{\mu(\mathcal{B}_\rho(x))} \tag{2.33}$$

*exists in $\mathbb{R}^n$. Furthermore, the Radon-Nikodým decomposition of $\nu$ is given by $\nu = \nu_u \mu + \nu^s$ with $\nu^s := \nu \llcorner E$, where the set*

$$E := (\Omega \setminus \text{supp } \mu) \cup \left\{ x \in \text{supp } \mu : \lim_{\rho \to 0} \frac{|\nu|(\mathcal{B}_\rho(x))}{\mu(\mathcal{B}_\rho(x))} = \infty \right\} \tag{2.34}$$

*is $\mu-$negligible.*

**Sets of Finite Perimeter.** Characteristic functions of sets of finite perimeter are a special type of BV functions. Since they are heavily used in the calibration based lifting approach, let us summarize some related results.

**Definition 2.2.10** (Sets of Finite Perimeter [AFP00, Def. 3.35]) *Let $E \subseteq \mathbb{R}^n$ be $\mathcal{L}^n-$measurable. For any open set $\Omega \subseteq \mathbb{R}^n$ the* perimeter *of $E$ in $\Omega$ is defined as*

$$P(E; \Omega) := \sup_{\varphi \in \mathcal{K}} \left\{ \int_E \text{div } \varphi \, dx \right\}, \tag{2.35}$$

$$\mathcal{K} := \left\{ \varphi \in C_c^1(\Omega; \mathbb{R}^n), \|\varphi\|_\infty \leq 1 \right\}. \tag{2.36}$$

*This means that the perimeter of $E$ in $\Omega$ is the variation of the characteristic function $1_E$ in $\Omega$.*
*If $P(E; \Omega) < \infty$, we say that $E$ is of* finite perimeter*. If $P(E, \Omega') < \infty$ for all open subsets $\Omega'$ with compact closure $\bar{\Omega}' \subset \Omega$, we say that $E$ is of* locally finite perimeter*.*

In fact, all sets $E$ with $C^1$ boundary (see e.g. [Eva22, Chpt. C.1]) in $\Omega$ and $\mathcal{H}^{n-1}-$ negligible set $\Omega \cap \partial E$ are sets of finite perimeter. This can easily be shown using the divergence theorem: The integral in

(2.35) can be reformulated as

$$\int_E \operatorname{div} \varphi \, \mathrm{d}x = -\int_{\partial E \cap \Omega} \langle \nu_E, \varphi \rangle \, \mathrm{d}\mathcal{H}^{n-1} \qquad (2.37)$$

if both integrals exist. Here, $\nu_E$ denotes the inner unit normal of $E$. For these sets it holds in particular $P(E; \Omega) = \mathcal{H}^{n-1}(\Omega \cap \partial E)$ [AFP00, Prop. 3.62]; Intuitively, one has to find a sequence of $\varphi_n$ such that $\varphi(x) = \nu_E(x)$ for almost all $x \in \partial E \cap \Omega$.

For a set $E$ with finite perimeter in $\Omega$ it holds that $1_E \in \mathrm{BV}_{\mathrm{loc}}(\Omega; \mathbb{R})$ [AFP00, p. 143]. This helps proving the following proposition:

**Theorem 2.2.11** (Properties [AFP00, Prop. 3.36]) *For a set $E \subseteq \mathbb{R}^n$ with finite perimeter in $\Omega \subseteq \mathbb{R}^n$, the distributional derivative $D1_E$ is an $\mathbb{R}^n$−valued finite Radon measure in $\Omega$, i.e., $D1_E \in \mathcal{M}(\Omega; \mathbb{R}^n)$. Furthermore, the perimeter of $E$ in $\Omega$ is the variation of the distributional derivative of $1_E$ in $\Omega$, i.e., $P(E, \Omega) = |D1_E|(\Omega)$. The following generalized divergence theorem* holds

$$\int_E \operatorname{div} \varphi \, dx = -\int_\Omega \langle \nu_E, \varphi \rangle \, d|D1_E|, \quad \forall \varphi \in C_c^1(\Omega; \mathbb{R}^n), \quad (2.38)$$

*where $D1_E = \nu_E |D1_E|$ is the polar decomposition of $D1_E$.*

We can now recall the *coarea formula in* BV, which will also be of central interest in the calibration-based lifting approach.

**Theorem 2.2.12** (Coarea Formula in BV [AFP00, Thm. 3.40]) *For any open set $\Omega \subseteq \mathbb{R}^n$ and $u \in L_{loc}^1(\Omega; \mathbb{R})$ it holds*

$$V(u, \Omega) = \int_{-\infty}^{+\infty} P(\{x \in \Omega : u(x) > t\}, \Omega) \, dt. \qquad (2.39)$$

*If, furthermore, $B \subseteq \Omega$ is a Borel set and $u \in \mathrm{BV}(\Omega; \mathbb{R})$, then $\{u > t\}$ has finite perimeter in $\Omega$ for a.e. $t \in \mathbb{R}$ and it holds*

$$Du(B) = \int_{-\infty}^{+\infty} D1_{u>t}(B) \, dt, \qquad (2.40)$$

$$|Du|(B) = \int_{-\infty}^{+\infty} |D1_{u>t}|(B) \, dt. \qquad (2.41)$$

One of the intermediate steps in calibration-based lifting consists in integrating an objective function over the *measure-theoretic boundary* of subgraphs. Let us, therefore, introduce the following definitions:

**Definition 2.2.13** (Measure-theoretic interior, exterior and boundary of a set [AFP00][Def. 3.60]) *Let $E \subseteq \mathbb{R}^n$ be an $\mathscr{L}^n$-measurable set. The* points of density $t$ *are described by the sets $E^t \subseteq \mathbb{R}^n$, where*

$$E^t := \left\{ x \in \mathbb{R}^n : \lim_{\rho \searrow 0} \frac{|E \cap \mathscr{B}_\rho(x)|}{|\mathscr{B}_\rho(x)|} = t \right\}. \qquad (2.42)$$

*The sets $E^1$ and $E^0$ can be considered the* measure-theoretic interior *and* exterior *of the set $E$. This motivates the following definition of the* measure-theoretic *or* essential boundary $\partial^* E \subseteq \mathbb{R}^n$:

$$\partial^* E := \mathbb{R}^n \backslash (E^0 \cup E^1). \qquad (2.43)$$
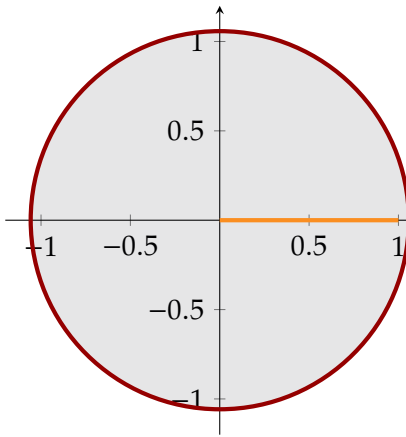


**Figure 2.4: Boundary.** The topological boundary of the set $E$ is given by $\partial E = \mathcal{S}_1(0) \cup F$, where $S$ is drawn in red and $F$ in orange. The measure-theoretic boundary, however, is given by $\partial^* E = \mathcal{S}_1(0)$, i.e., the red circle.

The measure-theoretic boundary may differ from the topological boundary. Consider for example the set $E \subset \mathbb{R}^2$ in Fig. 2.4, $E := \mathscr{B}_1(0) \backslash F, F := \{(x, 0) \in \mathbb{R}^2 : 0 \leq x \leq 1\}$. The measure-theoretic boundary is given by the sphere $\partial^* E = \mathcal{S}_1(0)$, the topological boundary by $\partial E = \mathcal{S}_1(0) \cup F$.

Consider an $\mathscr{L}^n$-measurable set $E \subseteq \mathbb{R}^n$ and the characterstic function $u = 1_E$. Then the following connection holds between the concepts introduced in Def. 2.2.6 and Def. 2.2.13 [AFP00, Ex. 3.68]: The approximate discontinuity set $S_u$ is the measure-theoretic boundary $\partial^* E$ and the jump set $J_u$ is a subset of $E^{1/2}$, i.e., the points of density 0.5.

As the final ingredient for the calibration-based lifting approach, we need to define the *inner unit normal* of sets of finite perimeter, which is typically defined over the *reduced boundary $FE$*:

**Definition 2.2.14** (Reduced boundary and inner unit normal [AFP00][Def. 3.54]) *Let $E \subseteq \mathbb{R}^n$ be an $L^n$-measurable set and $\Omega \subseteq \mathbb{R}^n$ the largest open set such that $E$ is locally of finite perimeter in $\Omega$. The* reduced boundary *is defined as*

$$\mathscr{F}\mathscr{E} := \{x \in \text{supp} \, |D\mathbf{1}_E| \cap \Omega : |\nu_E| = 1\} \qquad (2.44)$$

*where $\nu_E : \mathscr{F}\mathscr{E} \to \mathcal{S}^{n-1}$ is the* generalized inner normal *to $E$ defined as*

$$\nu_E := \lim_{\rho \to 0} \frac{D\mathbf{1}_E(\mathscr{B}_\rho(x))}{|D\mathbf{1}_E|(\mathscr{B}_\rho(x))}. \qquad (2.45)$$

An important result of Federer [AFP00, Thm. 3.61] shows that the reduced boundary $\mathscr{F}\mathscr{E}$ and the measure-theoretic boundary $\partial^* E$

coincide up to a $\mathcal{H}^{n-1}$-negligible set. This later allows us to use the definition of the inner unit normal when integrating over the approximate jump points $J_u$ against $\mathcal{H}^{n-1}$, even though the inner unit normal might not exist on the $\mathcal{H}^{n-1}$-negligible set $\partial^* E \backslash \mathcal{F} \mathcal{E}$.

## 2.3 Spectral Theory

In this section we recall definitions and results from the *spectral theory* for square matrices and bounded, self-adjoint operators on separable Hilbert spaces. This serves the purpose of motivating a less well-known spectral theory in the context of images. The latter allows to enhance/remove features in a given image with respect to a notion of scale introduced by some convex, 1-homogeneous functional (regularizer); The considered features are non-linear eigenfunctions of the regularizer and their scale is given by the eigenvalue of the non-linear eigenfunction. For the here admissible 2D total variation regularizer it has been shown that indicator functions of certain bounded, convex sets with finite perimiter are non-linear eigenfunctions of TV [BCN02, Thm. 4]. As the spectral theory for images evolves around these eigenfunctions it allows the definition of filters which handle sharp edges well.

In practice, the spectral theory for images is closely linked to the *Bregman iteration* on variational problems with a convex $L^2$ data term. In Chpt. 4 we propose an extension of the Bregman iteration to a broader class of variational problems, with the overall goal being an extension of the related spectral theory for images. While Chpt. 4 focuses on the definition and analysis of an extended Bregman iteration and does not explicitly provide an extension of the spectral theory, the following mathematical preliminaries are crucial for motivating our new approach.

### 2.3.1 Square Matrices

Spectral theory is well-known in the context of matrices. We say that a square matrix $A \in \mathbb{R}^{n \times n}$ has an *eigenvector $v$* to the *eigenvalue $\lambda$*, if

$$Av = \lambda v. \tag{2.46}$$

In case of a diagonalizable, quadratic matrix $A \in \mathbb{R}^{n,n}$ with $n$ distinct eigenvalues, we can write the matrix as a factorization in terms of its eigenvalues and -vectors:

$$A = VDV^\top. \tag{2.47}$$

Here, $V = (v_1|...|v_n)$ holds the eigenvectors $v_i$ of $A$ in its columns and $D = \mathrm{diag}(\lambda_1, ..., \lambda_n)$ is a diagonal matrix that holds the eigenvalues $\lambda_i$ of $A$. Alternatively, we can write

$$A = \sum_{i=1}^{n} \lambda_i v_i v_i^\top. \tag{2.48}$$

Note that $P_i := v_i v_i^\top$ are orthogonal projections onto the eigenspaces.

In this section we want to recall the extension of the spectral theory to bounded, self-adjoint operators on Hilbert spaces. This later motivates the definition of a *weak data-dependent spectral frequency representation*, which allows the decomposition of images $f$ in some Banach space $B$ with respect to a notion of scale induced by a convex, absolute 1-homogeneous regularizer $J$.

### 2.3.2 Bounded, Self-Adjoint Operators

In the following, let $H$ denote a separable Hilbert space and $B(H)$ denote the Banach space of bounded operators on $H$ with respect to the operator norm $\|A\| := \sup_{x \in H \setminus \{0\}} \frac{\|Ax\|}{\|x\|}$. An operator is called *self-adjoint* if $\langle Au, v \rangle = \langle u, Av \rangle$.

> **Definition 2.3.1** (Resolvent and Spectrum [Hal13, Def. 7.4]) *For $A \in B(H)$, the* resolvent set *of $A$, denoted $\rho(A)$, is the set of all $\lambda \in \mathbb{C}$ such that the operator $(A - \lambda I)$ has a bounded inverse. If $\lambda \in \rho(A)$, the operator $(A - \lambda I)^{-1}$ is called* resolvent *of $A$ at $\lambda$. The complement of $\rho(A)$ in $\mathbb{C}$ is called the* spectrum *of $A$, denoted by $\sigma(A)$.*

For self-adjoint operators, the spectrum of $A$ can be explained more precisely with the following proposition.

> **Proposition 2.3.2** (Spectrum – based on [Hal13, Prop. 7.5, Prop. 7.7]) *For self-adjoint $A \in B(H)$, it holds:*
>
> ▶ *The spectrum of $A$ is a closed, bounded, and non-empty subset of $\mathbb{R}$.*
> ▶ *A real number $\lambda \in \mathbb{R}$ is in the spectrum of $A$ if and only if there is a sequence of non-zero vectors $(u_n) \in H$ such that*
>
> $$\lim_{n \to \infty} \frac{\|Au_n - \lambda u_n\|}{\|u_n\|} = 0. \tag{2.49}$$

The second condition means that $\lambda \in \mathbb{R}$ is in the spectrum of $A$ if and only if there is a sequence $u_n$ so that the equality $Au_n = \lambda u_n$

that characterizes classical eigenvalues is asymptotically fulfilled. We will illustrate this in a small example.

**An Example. [Hal13, Chpt. 6.1, Example 7.9]**   Positive, self-adjoint operators $A$ on Hilbert space might not have eigenvectors in the finite-dimensional sense. Consider, e.g., the multiplication $A$ on $L^2([0,1])$:

$$(Au)(x) := xu(x). \tag{2.50}$$

This operator is clearly self-adjoint: For every $u, v \in L^2([0,1])$, it holds

$$\langle Au, v \rangle = \int_0^1 xu(x)v(x)\,\mathrm{d}x = \langle u, Av \rangle. \tag{2.51}$$

An eigenvector $w$ to the eigenvalue $\lambda \in \mathbb{R}$ would have to satisfy $\lambda w(x) = (Aw)(x) = xw(x)$. This directly implies that a possible eigenvector $w$ can only be supported on $x = \lambda$, which is a set of zero measure. Therefore, we cannot find an eigenvector in $L^2([0,1])$. However, extending the set of possible eigenvectors to the set of distributions, we find that for the shifted delta-distribution $\delta$ it holds

$$\big(A\delta(\cdot - \lambda)\big)(x) = x\delta(x - \lambda) = \lambda\delta(\cdot - \lambda)(x). \tag{2.52}$$

Therefore, the distributions $w = \delta(\cdot - \lambda)$ can be thought of as *generalized eigenvectors* of the positive, self-adjoint operator $A$.

Recalling Prop. 2.3.2, let us now determine the spectrum of $A$. Leaning on the generalized eigenvector, we define the sequence $(u_n)$ as

$$u_n(x) := \begin{cases} 1, & \text{if } x \in [\lambda - \frac{1}{n}, \lambda + \frac{1}{n}] \subset [0,1], \\ 0, & \text{else.} \end{cases} \tag{2.53}$$

The elements of the sequence $(u_n)$ are contained in $L^2([0,1])$:

$$\int_0^1 |u_n(x)|^2\,\mathrm{d}x \leq \int_{\lambda - \frac{1}{n}}^{\lambda + \frac{1}{n}} 1\,\mathrm{d}x = \frac{2}{n}, \tag{2.54}$$

where $\frac{2}{n} < \infty$ for all $n \in \mathbb{N}$. Plugging the sequence into (2.49) and assuming for simplicity that $n$ is large enough such that $[\lambda - \frac{1}{n}, \lambda + \frac{1}{n}] \subset [0,1]$, we get, for any $\lambda \in (0,1)$,

$$\frac{\|Au_n - \lambda u_n\|}{\|u_n\|} = \frac{\sqrt{\int_{\lambda - \frac{1}{n}}^{\lambda + \frac{1}{n}} (x - \lambda)^2\,\mathrm{d}x}}{\sqrt{\frac{2}{n}}} = \frac{\sqrt{\frac{2}{3}\left(\frac{1}{n}\right)^3}}{\sqrt{\frac{2}{n}}} = \sqrt{\frac{1}{3n^2}} \xrightarrow{n\to\infty} 0.$$

With slightly different definitions of $u_n$, one can equally show that the extremal value $0$ and $1$ are also in the spectrum of $A$.

### 2.3.2.1 Spectral Decomposition

Similar to the eigendecomposition (2.48), a more general *spectral decomposition* can be defined in the context of self-adjoint, bounded operators. For this, we first need to introduce *orthogonal projections* and *projection-valued measures*.

> **Proposition 2.3.3** (Orthogonal Projection [Hal13, Prop A.57]) *For any closed subspace $V \subset H$ there exists a unique bounded operator $P$, called the* orthogonal projection *onto $V$, such that $P = I$ on $V$, and $P = 0$ on the orthogonal complement $V^{\perp}$.*

> **Definition 2.3.4** (Projection-valued measure [Hal13, Def. 7.10]) *Let $X$ be a set and let $\Omega$ be a $\sigma$-algebra in $X$. A map $\mu : \Omega \rightarrow B(H)$ is called a* projection-valued measure *if the following properties are satisfied.*
>
> 1. *For each $E \subset \Omega$, $\mu(E)$ is an orthogonal projection.*
> 2. *$\mu(\emptyset) = 0$ and $\mu(X) = I$.*
> 3. *If $E_i \subset \Omega, i = 1, 2, 3...$, are disjoint, then for all $v \in H$, we have*
> $$\mu\left(\bigcup_{i=1}^{\infty} E_i\right) v = \sum_{i=1}^{\infty} \mu(E_i)v, \qquad (2.55)$$
> *where the convergence of the sum is in the norm topology on $H$.*
> 4. *For all $E_1, E_2 \in \Omega$, we have $\mu(E_1 \cap E_2) = \mu(E_1)\mu(E_2)$.*

In the matrix setting, $X$ is the set of eigenvalues $\{\lambda_1, ..., \lambda_n\}$ of a matrix $A \in \mathbb{R}^{n,n}$, and the projection-valued measure orthogonally projects on the subspace spanned by the eigenvectors for the eigenvalues in $E$: For example, for $E = \{\lambda_i, \lambda_j\}$ the map would be given by $\mu(E) = v_i v_i^{\top} + v_j v_j^{\top}$, where $v_i$ and $v_j$ are the respective eigenvectors.

This leads us to the following spectral theorem, which is the infinite-dimensional extension of the spectral matrix decomposition (2.48):

**Theorem 2.3.5** (**Spectral Theorem** [Hal13, Thm. 7.12]) *Let $A \in B(H)$ denote a self-adjoint operator. Then there exists a unique projection-valued measure $\mu^A$ on the Borel $\sigma$-algebra in $\sigma(A)$ with values in projections on $H$, such that*

$$A = \int_{\sigma(A)} \lambda d\mu^A(\lambda). \tag{2.56}$$

Eq. (2.56) is referred to as the *spectral decomposition* of $A$. Applying the operator $A$ to some $u \in H$ thus can be written as

$$Au = \int_{\sigma(A)} \lambda d\mu^A(\lambda) \cdot u \tag{2.57}$$

The *data-dependent spectral measure* $\mu^A(\lambda) \cdot u$ is of special interest. Recall the example given on p. 37: For an "almost" eigenvector $u$ of $A$ with "almost" eigenvalue $\lambda$ it holds

$$\mu^A(\cdot) \cdot u = u\delta(\cdot - \lambda). \tag{2.58}$$

Let us define, for some bounded, measurable map $w : \sigma(A) \to \mathbb{R}$, a new operator $\tilde{w}$ [Hal13, Def. 7.13] as

$$\tilde{w}(A) := \int_{\sigma(A)} w(\lambda) d\mu^A(\lambda). \tag{2.59}$$

Applying this new operator $\tilde{w}(A)$ to some $u \in H$, i.e.,

$$\tilde{w}(A)u = \int_{\sigma(A)} w(\lambda) d\mu^A(\lambda) \cdot u, \tag{2.60}$$

allows to enhance or reduce components of the data-dependent spectral measure according to the associated $\lambda$.

### 2.3.3 Convex, 1-Homogeneous Functionals

In [Bur+16], the authors extend the notion of a data-dependent spectral decomposition of an operator $A$ to the non-linear setting. Their goal is to decompose images $f \in L^2(\Omega; \mathbb{R}^k)$ with respect to a notion of scale induced by some convex, absolutely 1-homogeneous regularizer $J$.

In the following, let $J$ be a convex, lower semi-continuous, absolutely 1-homogeneous regularizer. The following definition is of special interest, as it introduces a new notion of scales:

**Definition 2.3.6** (Non-linear eigenfunctions) *We call $u$ a non-linear eigenfunction of $J$ to the eigenvalue $\lambda \in \mathbb{R}$, if $\|u\| = 1$ and $\lambda u \in \partial J(u)$.*

In analogy to the data-dependent spectral measure $\mu^A(\lambda) \cdot u$ in (2.57), the following *weak data-dependent spectral representation* can be defined:

**Definition 2.3.7** (Weak spectral representation [Bur+16, Def. 1.4])
*Let $B$ be a Banach space. A map from $f \in B$ to a data-dependent vector-valued distribution $\mu_J^f(s)$ on $B$ is called a* weak spectral representation *with respect to the convex functional $J$ if the following properties are satisfied:*

1. *For a nonlinear eigenfunction $f$ of $J$ with eigenvalue $\lambda$,*

    a) *the* weak spectral (frequency) representation *is given by*
    $$\mu_J^f(s) = f\,\delta(s - \lambda). \qquad (2.61)$$

    b) *the* weak spectral (wavelength) representation *is given by*
    $$\tilde{\mu}_J^f(s) = f\,\delta(s - \frac{1}{\lambda}). \qquad (2.62)$$

2. *Any input data $f \in U$ can be reconstructed by*

$$f = \int_0^\infty \mu_J^f(s)\,ds = \int_0^\infty \tilde{\mu}_J^f(t)\,dt. \qquad (2.63)$$

Informally, $\mu_J^f(\lambda)$ describes a projection onto non-linear eigenfunctions of $J$ with eigenvalue $\lambda$. Similar to the definition in (2.59), we can turn a bounded, measurable map $w : \mathbb{R}^+ \to \mathbb{R}$ into a non-linear filter $\tilde{w}$ via

$$\tilde{w}(f) := \int_0^\infty w(s)\mu_J^f(s)\,\mathrm{d}s. \qquad (2.64)$$

Using this definition, features in $f$ can be enhanced, reduced or even removed according to their scale. Here, a feature is a nonlinear eigenfunction of $J$ and its scale is given by the eigenvalue $\lambda$. How such a data-dependent weak spectral frequency representation can be defined is topic of Chpt. 4.

# Calibration-Based Lifting | 3

Solving imaging problems numerically is especially challenging when the underlying mathematical problem is non-convex. In case of certain non-convex problem classes, it can thus be beneficial to reformulate a given problem in order to arrive at a formulation that is easier to handle numerically. Ignoring the exact function spaces for now, our strategy will be as follows: A non-convex problem $\inf_{u:\Omega\to\Gamma} F(u)$ is replaced by an (after discretization) numerically less challenging convex formulation $\inf_{v:\Omega\times\Gamma\to\mathbb{R}} \mathcal{F}(v)$. In the imaging community, these approaches are referred to as *functional lifting*. In reference to the higher-dimensional domain $\Omega\times\Gamma$ of the functions in the solution space, the second formulation is referred to as the *higher-dimensional* or *lifted* problem.

Many different functional lifting strategies can be found in the literature. In order to better understand the context of the *calibration-based lifting* (CBL) approach that we will focus on, let us discuss some of the conceptual differences between the different approaches: Early approaches such as [Ish03] considered lifting/relaxation of discrete problems, i.e., (multi-)labelling problems on a discrete grid $\inf_{u^h:\Omega^h\to\Gamma^h} F^h(u^h)$. Later approaches such as [Möl+15; Lau+16] considered a semi-discrete setting, i.e., (multi-)labelling problems on a continuous domain $\inf_{u:\Omega\to\Gamma^h} F(u)$. Some of the latest approaches such as [MC19a; Vog20] consider lifting of the fully continuous problem $\inf_{u:\Omega\to\Gamma} F(u)$. Some lifting approaches are only applicable to scalar problems $\Gamma\subset\mathbb{R}$ [Poc+10; Möl+15], while others are designed for vectorial $\Gamma\subset\mathbb{R}^m$ [Lau+16; Vog20; MC19b] or manifold-valued [Lel+13b] problems. **Calibration-based lifting is a fully continuous lifting approach for scalar problems with first-order terms $Du$. In particular, it is well suited for variational problems with total variation regularizer, which play a vital role in imaging.**

Formulating a suitable lifting strategy can be challenging, as the lifted problem has to meet several requirements. First, it has to be theoretically sound and ideally *tight* in the sense that no new global minima are introduced in the lifted formulation. Second, the lifted formulation must be computationally tractable. Third, minimizers of the lifted problem have to be mappable to their counterpart in the original solution space; If this is not the case, the approach can only be used to verify and not to find a minimizer of the original problem. **Calibration-based lifting is tight and minimizers of the lifted problem can be mapped to minimizers of the original problem via a thresholding theorem. It is, furthermore, supported by**

In this chapter we are concerned with the calibration-based lifting strategy. We show a detailed derivation of the lifted problem formulation and related results concerning its tightness. We extend proofs and arguments shown in [Poc+10] from $u \in W^{1,1}(\Omega; \Gamma)$ to $u \in \mathrm{BV}(\Omega; \Gamma)$. While this chapter is rather theoretic, the main results will prove to be an important tool in the following application-oriented chapters: In Chpt. 4 we will use CBL in order to extend the notion of a *non-linear scale space iteration* to a broader class of variational problems, and in Chpt. 5 we will use CBL in the context of a *neural field-based optimization* strategy. The aim of this chapter is to present a coherent, self-contained and multi-layered view on this topic as a basis for the following chapters.

## 3.1 Preliminaries

As already mentioned, calibration-based lifting is applicable to scalar problems with first-order terms $Du$. Throughout this chapter, we therefore consider problems of the form [1]

$$\inf_{u \in \mathrm{BV}(\Omega; \Gamma)} F(u), \quad F(u) := \int_\Omega f(x, u, \mathrm{d}Du), \tag{3.1}$$

where $\Omega \subset \mathbb{R}^n$ is open and bounded, $\Gamma = [\Gamma_{\min}, \Gamma_{\max}] \subset \mathbb{R}$ and $Du$ denotes the distributional derivative of $u$. Furthermore, we impose the following requirements on $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$:

(A1) $f$ is convex in the last argument, i.e., $p \mapsto f(x, t, p)$ is convex in $\mathbb{R}^n$ for all $(x, t) \in \Omega \times \mathbb{R}$.

(A2) There exists a Borel set $B \subset \Omega \times \mathbb{R}$, $\mathcal{H}^n((\Omega \times \mathbb{R}) \backslash B) = 0$ such that $f$ is lower semicontinuous at every point of $B \times \mathbb{R}^n$.

(A3) $f$ is locally bounded in the first two arguments, i.e., for any $(x_0, t_0) \in \Omega \times \mathbb{R}$ there is a neighbourhood $A$ and some constant $c > 0$ such that $|f(x, t, 0)| \le c$ for all $(x, t) \in A$.

(A4) $f$ is coercive in the last argument, in the sense that $\exists c > 0$, $d \ge 0$ for which $c\|p\| - d \le f(x, t, p)$ for all $(x, t) \in \Omega \times \mathbb{R}$.

Note that the calibration-based lifting approach in fact only requires the assumptions (A1)-(A3). If, however, the requirements (A1)-(A4) are fulfilled, then it can be shown that $u \mapsto F(u)$ is lower semicontinuous on $\mathrm{BV}_{\mathrm{loc}}(\Omega; \Gamma)$ for the $L^1_{\mathrm{loc}}(\Omega; \Gamma)$-topology [Mas79, Thm. 3.1]. Together with the compactness in BV [AFP00, Thm. 3.23] this is an important ingredient for proving existence of minimizers of (3.1).

1: Here and in the following we slightly abuse the notation. The precise formulation of the problem heavily relies on the notation introduced in Def. 2.2.6:

$$F(u) = \int_\Omega f(x, u, \nabla u)\, \mathrm{d}x +$$

$$\int_{\Omega \backslash J_u} f^\infty \left(x, u^+, \frac{\mathrm{d}D^c u}{\mathrm{d}|D^c u|}\right) \mathrm{d}|D^c u| +$$

$$\int_{J_u} \int_{u^-(x)}^{u^+(x)} \frac{f^\infty \left(x, t, \frac{\mathrm{d}Du}{\mathrm{d}|Du|}\right)}{u^+(x) - u^-(x)}\, \mathrm{d}t\, \mathrm{d}|Du|,$$

where

$$f^\infty(x, t, p) := \lim_{s \to +\infty} \frac{1}{s} f(x, t, sp).$$

See also [Mas79, p. 390].

**Special Case: Separable Data Term and Regularizer.**   Let us consider a short example in order to illustrate that the requirements are reasonable for many applications. In imaging, the third argument of $f$ – the first-order distributional derivative $Du$ – is often part of the regularizer and might not appear in the data term. Let us assume that the data term and regularizer are separable in the sense that there exist $\rho : \Omega \times \mathbb{R} \to \mathbb{R}$ and $\eta : \mathbb{R}^d \to \mathbb{R}$ such that

$$f(x,t,p) = \rho(x,t) + \eta(p). \tag{3.2}$$

Together, (A3) and (A4) imply that the data term is proper and locally bounded; thus the regularizer is convex and coercive by (A1) and (A4). Properness and locally boundedness of the data term are in general reasonable assumptions and, for example, the popular total variation regularizer is both convex and coercive.

**Historically,**   the *calibration-based lifting* approach traces back to the *calibration method*. The latter is a well-established minimality criterion for the minimal surface problem[2] [Mor16, Chpt. 6], but was also used, in the context of the Mumford-Shah functional and (vector-valued) free-discontinuity problems [ABD03; Mor02]. In [Cha01], the existence of *calibrations* for certain scalar variational problems is shown. Thanks to this existence result, the authors argue that the calibration method can also be employed in order to reformulate non-convex problems as convex problems in a higher-dimensional space as a means to compute (previously unknown) minimizers. In [Poc+10], the calibration-based lifting is introduced to the imaging community and discretization schemes are analyzed in [Möl+15; MC17].

[2]: The minimality criterion states, that if a differential form exists which fulfils certain assumptions and which interacts with a given surface in a special way then the given surface is area-minimizing among all surfaces with the same boundary. The differential form is called the *calibration* and one says that the differential form *calibrates* the surface.

**The Idea**   of the calibration-based lifting approach is to reformulate the general energy (3.1) by considering indicator functions of the subgraph of possible solutions:

$$\mathcal{U} := \{\mathbf{1}_u, u \in U\}, \quad \mathbf{1}_u(x,t) := \begin{cases} 1, & u(x) \geq t, \\ 0, & u(x) < t. \end{cases} \tag{3.3}$$

The integrand $f$ in (3.1) is replaced by an interfacial energy $h$, which is integrated over the measure theoretic-boundary of the subgraph of $u$. This leads to the lifted (primal) energy $\mathcal{F}_p$ for which the following connection to the original energy $F$ holds:

$$\mathcal{F}_p(\mathbf{1}_u) = F(u), \ u \in U. \tag{3.4}$$

As the lifted solution space $\mathcal{U}$ is non-convex, the second step consists of relaxing the solution space to a convex superset $\mathcal{C} \supset \mathcal{U}$.

Under the assumption (A1) one can show that this relaxation is tight, which means that it holds

$$\inf_{v \in \mathscr{C}} \mathscr{F}_p(v) = \inf_{u \in \mathcal{U}} \mathscr{F}_p(u) = \inf_{u \in U} F(u), \qquad (3.5)$$

and that *non-integral* solutions – solutions $v^* \in \arg\min_{v \in \mathscr{C}} \mathscr{F}_p(v)$ for which $v^* \in \mathscr{C} \setminus \mathcal{U}$ – can be mapped to *integral* solutions $u^* \in \arg\min_{u \in \mathscr{C}} \mathscr{F}_p(v)$ for which $u^* \in \mathcal{U} \subset \mathscr{C}$. These integral solutions then point to solutions $u^* \in \arg\min_{u \in U} F(u)$. In a last step, duality is employed in order to reach a lifted primal-dual energy $\mathscr{F}$ for which

$$\mathscr{F}(v) = \mathscr{F}_p(v), \text{ for } v \in \mathscr{C}. \qquad (3.6)$$

holds. This primal-dual energy can be approximately minimized numerically using the PDHG algorithm. A summary of the complete line of argument of the CBL lifting approach can be found in Fig. 3.2.

The remainder of the chapter is organised as followed: In **Section 3.2** we derive the embedded primal-dual energy and state some central properties. Selected proofs are presented for the sake of cohesiveness. In **Section 3.3** we recall discretization schemes for the lifted problem. In **Section 3.4** we recall the PDHG algorithm, which can be used for solving the discretized problem.

## 3.2 Continuous Derivation

In the following, we will use various definitions and results concerned with functions of bounded variation and sets of finite perimeter. An introduction to the topic and to our notation can be found in Sec. 2.2.

In order to arrive at a lifted formulation of the problem, the original domain $\Omega$ and range $\Gamma$ are combined and used as the new domain $\Omega \times \Gamma$ for functions in the solution space. Functions $u \in \mathrm{BV}(\Omega; \Gamma)$ are identified with the following objects, which are also visualized in Fig. 3.1:

1. The *subgraph* of $u$ is defined as

$$\mathrm{hyp}\, u := \{(x, t) \in \Omega \times \mathbb{R} : u(x) \geq t\}. \qquad (3.7)$$

   For $u \in L^1(\Omega)$, the subgraph is *locally* of finite perimeter in $\Omega \times \mathbb{R}$, if and only if $u \in \mathrm{BV}(\Omega; \mathbb{R})$, see [Mir64, Thm. 1.10] or [AMP04, Prop. 4.2].
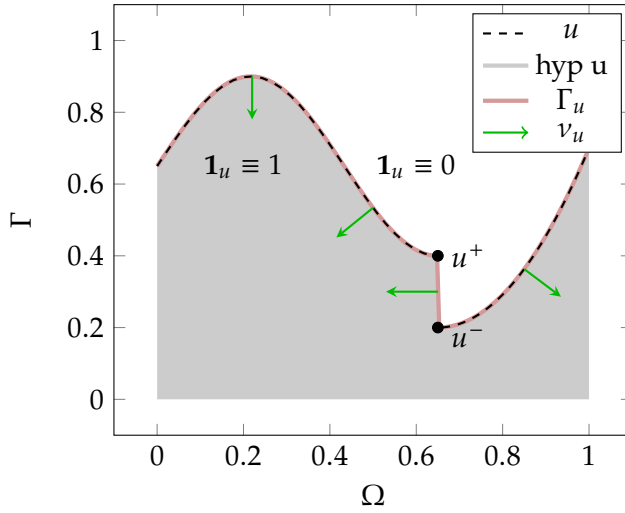
**Figure 3.1: Calibration-Based Lifting.** The idea of lifting approaches is to reformulate a given non-convex problem of the form (3.1) so that the new, lifted formulation is convex and global minimizers of the lifted formulation can be mapped to global minimizers of the original formulation.

In calibration-based lifting, instead of minimizing over functions $u \in \text{BV}(\Omega; \Gamma)$ **(black dotted line)**, the problem is reformulated in terms of the subgraph of $u$ **(gray area)**, its measure-theoretic boundary $\Gamma_u$ **(red line)** and the associated generalized inner unit normal $\nu_u$ **(green arrows)**, as well as the indicator function of the subgraph $\mathbf{1}_u$.

Furthermore, if $\mathcal{L}^n(\Omega) < +\infty$ it holds [AMP04, Prop. 4.2]

$$|Du|(\Omega) \leq P(\text{hyp } u; \Omega \times \mathbb{R}) \leq |Du|(\Omega) + \mathcal{L}^n(\Omega). \quad (3.8)$$

2. The *indicator function of the subgraph* $\mathbf{1}_u : \Omega \times \mathbb{R} \to \{0, 1\}$ is defined as

$$\mathbf{1}_u(x, t) := \begin{cases} 1, & \text{if } u(x) \geq t, \\ 0, & \text{else.} \end{cases} \quad (3.9)$$

If hyp $u$ is $\mathcal{L}^{n+1}$-measurable, its perimeter in $\Omega \times \mathbb{R}$ is defined as the total variation of the indicator function of the subgraph of $u$: $P(\text{hyp } u; \Omega \times \mathbb{R}) = TV(\mathbf{1}_u)$ [AFP00, Def. 3.35]. Together with [AMP04, Prop. 4.2], this means that if $\mathcal{L}^n(\Omega) < \infty$ then $u \in \text{BV}(\Omega; \Gamma)$ if and only if $\mathbf{1}_u \in \text{BV}(\Omega \times \mathbb{R}; \{0, 1\})$.

3. The *measure-theoretic boundary* $\Gamma_u$ of the subgraph is the subset $\Gamma_u \subset \Omega \times \Gamma$ defined as [AFP00, Def. 3.60]

$$\Gamma_u := \left\{ (x, t) \in \Omega \times \Gamma : \lim_{\rho \to 0} \frac{|\mathscr{B}_\rho(x, t) \cap \text{hyp } u|}{|\mathscr{B}_\rho(x, t)|} > 0, \right.$$
$$\left. \lim_{\rho \to 0} \frac{|\mathscr{B}_\rho(x, t) \setminus \text{hyp } u|}{|\mathscr{B}_\rho(x, t)|} > 0 \right\} \quad (3.10)$$

The measure-theoretic boundary of a set is described by the approximate discontinuity set of the indicator function of the set [AFP00, Ex. 3.68].

If the set is of finite perimeter one can argue using Federers' theorem [AFP00, Thm. 3.61] that (in our setting) $\mathcal{H}^n$-a.e. points of the measure theoretic boundary are in fact described by the jump set of the indicator function $\mathbf{1}_u$. In case of a continuous function $u$ with subgraph of finite perimeter, the

measure-theoretic boundary $\Gamma_u$ can thus be thought of as the complete graph

$$\text{graph } u := \{(x,t) \in \Omega \times \mathbb{R} : u(x) = t\}. \qquad (3.11)$$

If the function $u$ has a discontinuity at some point $\bar{x}$, the measure-theoretic boundary additionally comprises the connecting set

$$\text{graph } u \cup \{(\bar{x},t) : u^-(\bar{x}) \leq t \leq u^+(\bar{x})\}. \qquad (3.12)$$

4. The *generalized inner unit normal* $\nu_u : \Gamma_u \to \mathcal{S}^n$ of the subgraph of $u$ is defined as

$$\nu_{\Gamma_u}(x,t) := \lim_{\rho \to 0} \frac{D\mathbf{1}_u(\mathcal{B}_\rho(x,t))}{|D\mathbf{1}_u(\mathcal{B}_\rho(x,t))|}. \qquad (3.13)$$

The generalized inner unit normal exists on $\mathcal{H}^n$-a.e. point of the measure-theoretic boundary $\Gamma_u$; see [AFP00, Def. 3.54, Thm. 3.61].

An important result which brings the above introduced concepts together is the following representation of the distributional derivative of the indicator function of a subgraph: If the subgraph is of finite perimeter in $\Omega \times \mathbb{R}$, it holds

$$D\mathbf{1}_u = \nu_{\Gamma_u} \, d\mathcal{H}^n \llcorner \Gamma_u. \qquad (3.14)$$

This can be deduced from the following results stated in Sec. 2.2: If $\mathbf{1}_u \in \mathrm{BV}(\Omega \times \Gamma; \{0,1\})$, the Radon-Nikodým decomposition of $D\mathbf{1}_u$ with respect to $\mathcal{L}^{n+1}$ is given by

$$D\mathbf{1}_u = D^a\mathbf{1}_u + D^j\mathbf{1}_u + D^c\mathbf{1}_u. \qquad (3.15)$$

The indicator of the subgraph is a constant function with the exception of the singularities captured at the approximate discontinuity set $S_{\mathbf{1}_u}$. Therefore, the absolutely continuous part $D^a\mathbf{1}_u = D\mathbf{1}_u \llcorner (\Omega \setminus S_{\mathbf{1}_u}) = 0$ and the Cantor part $D^c = D^s\mathbf{1}_u \llcorner (\Omega \setminus S_{\mathbf{1}_u})$ vanish. According to Federer's Theorem [AFP00, Thm. 3.61], the measure theoretic boundary $\Gamma_u$, the set of approximate jump points $J_{\mathbf{1}_u}$ and the reduced boundary $\mathcal{FE}$ are the same up to an $\mathcal{H}^n$-negligible set as the subgraph is of finite perimeter by assumption; see also [AFP00, Example 3.68]. For any $(x,t) \in J_{\mathbf{1}_u}$ it holds $\mathbf{1}_u^-(x,t) = 0$ and $\mathbf{1}_u^+(x,t) = 1$. The generalized inner unit normal, which is defined on the reduced boundary, exists for $\mathcal{H}^n$-a.e. $(x,t) \in \Gamma_u$. The exact form of $D\mathbf{1}_u$ in (3.14) then follows with (2.31)

$$D\mathbf{1}_u = D^j\mathbf{1}_u \overset{(2.31)}{=} (\mathbf{1}_u^+ - \mathbf{1}_u^-) \otimes \nu_u \, d\mathcal{H}^n \llcorner J_u = \nu_{\Gamma_u} \, d\mathcal{H}^n \llcorner \Gamma_u. \quad (3.16)$$

The distributional derivative $D\mathbf{1}_u$ is an $(n+1)$-dimensional measure concentrated on $\Gamma_u$.

**Primal Formulation.** In this paragraph, we construct a convex energy $\mathscr{F}_p$ for which $\mathscr{F}_p(\mathbf{1}_u) \stackrel{!}{=} F(u)$ holds for any $u \in \mathrm{BV}(\Omega;\Gamma)$. As argued above, elements $u \in \mathrm{BV}(\Omega;\Gamma)$ have a one-to-one representation in terms of $\mathbf{1}_u \in \mathrm{BV}(\Omega \times \mathbb{R};\{0,1\})$. The set $\mathcal{U} := \{\mathbf{1}_u : u \in \mathrm{BV}(\Omega;\Gamma)\}$ is, however, non-convex. Therefore, we furthermore introduce a convex set $\mathcal{C} \supset \mathcal{U}$ and prove that global minimizers $v^* \in \arg\min_{v \in \mathcal{C}} \mathscr{F}_p(v)$ can be mapped to global minimizers $u^* \in \arg\min_{u \in \mathrm{BV}(\Omega;\Gamma)} F(u)$.

---

**Definition 3.2.1** (Primal Lifted Energy) *Let $\Omega \subset \mathbb{R}^n$ be open. For a Borel function $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \overline{\mathbb{R}}_+$ which fulfils (A1), we define the* perspective function $h : \Omega \times \mathbb{R} \times \mathbb{R}^{n+1} \to \overline{\mathbb{R}}$ *as*

$$
h(x,t,p) := \begin{cases} |p^t| f\left(x,t,\frac{p^x}{|p^t|}\right), & \text{if } p^t < 0, \\ \lim_{\lambda \to +\infty} \frac{1}{\lambda} f(x,t,\lambda p^x), & \text{if } p^t = 0, \\ +\infty, & \text{if } p^t > 0, \end{cases} \tag{3.17}
$$

*where the notation $p = (p^x, p^t) \in \mathbb{R}^n \times \mathbb{R}$ is used. Furthermore, we define the primal lifted energy $\mathscr{F}_p : \mathrm{BV}(\Omega \times \mathbb{R};\mathbb{R})$ as*

$$
\mathscr{F}_p(v) := \int_{\Omega \times \mathbb{R}} h\left(x,t,\frac{dDv}{d|Dv|}(x,t)\right) d|Dv|(x,t). \tag{3.18}
$$

---

It holds that $p \mapsto h(x,t,p)$ is positively 1-homogeneous (which follows directly from the definition of $h$) and convex (which can be seen by considering the epigraph, see [HL96, Prop. 2.2.1, p. 160]). Under the assumptions (A1)-(A3) $h$ is, furthermore, lower semicontinuous at each point $(x,t,p) \in B \times \mathbb{R}^{n+1}$ where $B \subset \Omega \times \mathbb{R}$ denotes a Borel set such that $\mathcal{H}^n((\Omega \times \mathbb{R}) \setminus B) = 0$. Let us shortly recall the arguments given in [Mas79, Proof of Thm. 3.1] to demonstrate where and why the assumptions (A1)-(A3) are necessary:

*Proof that h is lower semicontinuous.* Lower semicontinuity follows for $p^t < 0$ directly from (A2). Consider thus $(x_0, t_0, p_0) \subset B \times \mathbb{R}^{n+1}$ such that $p_0^t = 0$. Let $\epsilon > 0$; by the definition of $h$ there exists $\delta \in ]0, \epsilon[$ such that for all $p^t \in [-\delta, 0]$ it holds

$$
h(x_0, t_0, (p_0^x, 0)) - \epsilon < h(x_0, t_0, (p_0^x, p^t)). \tag{3.19}
$$

Since $h$ is lower semicontinuous at $(x_0, t_0, (p_0^x, -\delta))$, there exists a

neighbourhood $U$ such that for any $(x, t, (p^x, -\delta)) \in U$ it holds

$$h(x_0, t_0, (p_0^x, -\delta)) - \epsilon < h(x, t, (p^x, -\delta)). \qquad (3.20)$$

For $p^t \in ]-\delta, 0[$ and especially $|p^t|/|\delta| \in [0, 1]$ we rewrite $h(x, t, (p^x, -\delta))$ as

$$h(x, t, (p^x, -\delta)) = |\delta| f\left(x, t, \frac{p^x}{|\delta|}\right) \qquad (3.21)$$

$$= |\delta| f\left(x, t, \frac{|p^t|}{|\delta|} \frac{p^x}{|p^t|} + \left(1 - \frac{|p^t|}{|\delta|}\right) 0\right) \qquad (3.22)$$

and with (A1) we furthermore get

$$h(x, t, (p^x, -\delta)) \overset{(A1)}{\leq} |p^t| f\left(x, t, \frac{p^x}{|p^t|}\right) + (|\delta| - |p^t|) f(x, t, 0)$$

$$\leq h(x, t, (p^x, p^t)) + \epsilon f(x, t, 0). \qquad (3.23)$$

Together with (3.19)–(3.20) it holds for $p^t \in ] - \delta, 0]$, $\delta \in ]0, \epsilon[$, $(x, t, p^x)$ in the neighborhood of $(x_0, t_0, p_0^x)$ that

$$h(x_0, t_0, (p_0^x, 0)) - 2\epsilon - \epsilon f(x, t, 0) < h(x, t, (p^x, p^t)) \qquad (3.24)$$

Due to (A3) – the local boundedness of $f$ in the first two arguments – it follows that $h$ is lower semicontinuous at $(x_0, t_0, (p_0^x, 0))$. $\quad\square$

In the special case of an indicator function of a subgraph with finite perimeter equation (3.14) furthermore infers that the lifted primal energy (3.18) is an interfacial energy, which is defined over the measure theoretic boundary $\Gamma_u$ of the subgraph:

$$\mathcal{F}_p(\mathbf{1}_u) = \int_{\Gamma_u} h\left(x, t, \nu_{\Gamma_u}(x, t)\right) \, d\mathcal{H}^n(x, t). \qquad (3.25)$$

The following lemma is crucial as it links the original and lifted energy:

**Lemma 3.2.2** ([Mas79, Lemma 2.2]) *Let* $\Omega \subset \mathbb{R}^n$ *be open and bounded and let* $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \overline{\mathbb{R}_+}$ *be a Borel function which fulfils (A1). Let F be defined as in (3.1) and* $\mathcal{F}_p$ *as in Def. 3.2.1. For any* $u \in \mathrm{BV}(\Omega; \Gamma)$ *it holds*

$$F(u) = \mathcal{F}_p(\mathbf{1}_u). \qquad (3.26)$$

By Lemma 3.2.2, the global minimizers of

$$\inf_{v \in \mathcal{U}} \mathcal{F}_p(v), \quad \mathcal{U} := \{\mathbf{1}_u : u \in BV(\Omega; \Gamma)\}, \qquad (3.27)$$

are exactly the indicator functions of subgraphs of global minimizers of (3.1). While $\mathcal{F}_p$ is convex due to the convexity of $h$ with respect to its third parameter and linearity of $v \mapsto Dv$, problem (3.27) is still non-convex due to the non-convexity of $\mathcal{U}$. Thus, we consider the following relaxation of the solution space:

**Definition 3.2.3** (Relaxation) *Given a feasible set $U = BV(\Omega; \Gamma)$ with $\Gamma = [\Gamma_{\min}, \Gamma_{\max}] \subset \mathbb{R}$, the relaxed, lifted solution space is defined as*

$$\mathcal{C} := \{v \in BV(\Omega \times \mathbb{R}; [0,1]) : \qquad (3.28)$$

$$v(x,t) = 1, \ a.e. \ (x,t) \in \Omega \times \mathbb{R} \ where \ t \le \Gamma_{\min}, \quad (3.29)$$

$$v(x,t) = 0, \ a.e. \ (x,t) \in \Omega \times \mathbb{R} \ where \ t > \Gamma_{\max}\}. \quad (3.30)$$

For any $u \in BV(\Omega; \Gamma)$ where $\mathcal{L}^n(\Omega) < +\infty$, it holds that $\mathbf{1}_u \in BV(\Omega \times \mathbb{R}; \{0,1\})$ and thus $\mathbf{1}_u \in \mathcal{C}$. Furthermore, as $\mathcal{C}$ is convex, the problem

$$\inf_{v \in \mathcal{C}} \mathcal{F}_p(v) \qquad (3.31)$$

with $\mathcal{C}$ as in Def. 3.2.3 and $\mathcal{F}_p$ as in Def. 3.2.1 is fully convex.

After the relaxation of the solution space, two questions come to mind:

1. Is the relaxation tight, i.e., does $\inf_{v \in \mathcal{U}} \mathcal{F}_p(v) = \inf_{v \in \mathcal{C}} \mathcal{F}_p(v)$ hold or are new artificial global minima introduced?
2. Can *non-integral* global minimizers $v^* \in \arg\min_{v \in \mathcal{C}} \mathcal{F}_p(v)$ such that $v^* \in \mathcal{C} \backslash \mathcal{U}$ be mapped to global minimizers $u \in \arg\min_{u \in U} F(u)$?

Both questions are addressed in the following theorem.

**Theorem 3.2.4** (Thresholding) *Let $f$ be a Borel function fulfilling assumption (A1). Let $v^* \in \mathcal{C}$ be a global minimizer of*

$$\inf_{v \in \mathcal{C}} \mathcal{F}_p(v), \qquad (3.32)$$

*with $\mathcal{C}$ as in Def. 3.2.3 and $\mathcal{F}_p$ as in Def. 3.2.1. Then for $\mathcal{L}^1$-a.e. $s \in [0,1]$ the characteristic function[3] $1_{\{v^*>s\}} \in \mathcal{C}$ is also a global minimizer of (3.32). Furthermore, there is a $u \in U$ such that it holds $1_{\{v^*>s\}} = \mathbf{1}_u$.*

3: Similar to the indicator function of the subgraph we define

$$1_{\{v^*>s\}}(x,t) := \begin{cases} 1, & v^*(x,t) > s, \\ 0, & \text{else.} \end{cases}$$

As a reminder, recall

$$\mathbf{1}_u(x,t) := \begin{cases} 1, & u(x) \ge t, \\ 0, & \text{else.} \end{cases}$$

A proof will be provided below.

Let us remark that this theorem not only states that the relaxation is tight under assumption (A1), but also that any (even a non-integral) global minimizer of (3.31) can be mapped to a global minimizer of the original, non-convex problem (3.1): If there is a $u \in U$ such that $1_{\{v^* > s\}} = \mathbf{1}_u$, then it follows from Lemma 3.2.2 that $u$ is a global minimizer of (3.1).

In [Poc+10, Thm. 3.1], the thresholding result is already shown for $\inf_{u \in W^{1,1}} F(u)$. We here extend the theorem and proof to $\inf_{u \in BV(\Omega;\Gamma)} F(u)$.

*Proof.* Let $v^* \in \mathscr{C}$ be a global minimizer of (3.31).

First, we show that $\mathscr{F}_p$ as in Def. 3.2.1 satisfies a *generalized coarea formula*:

$$\mathscr{F}_p(v) = \int_{-\infty}^{+\infty} \mathscr{F}_p(1_{\{v > s\}}) \, \mathrm{d}s. \tag{3.33}$$

In the following, we denote by $\nu_v$ the Besicovitch derivative[4] of $Dv$ with respect to $|Dv|$. From Thm. 2.2.9 we directly get the existence of the Besicovitch derivative as well as the following Radon-Nikodým decomposition:

$$Dv = \nu_v |Dv| + Dv \llcorner E, \tag{3.34}$$

where

$$\nu_v(x,t) = \lim_{\rho \searrow 0} \frac{Dv(\mathscr{B}_\rho(x,t))}{|Dv|(\mathscr{B}_\rho(x,t))}, \tag{3.35}$$

$$E = \left\{ (x,t) \in \operatorname{supp} |Dv| : \lim_{\rho \to 0} \frac{|Dv|(\mathscr{B}_\rho(x,t))}{|Dv|(\mathscr{B}_\rho(x,t))} = \infty \right\}$$

$$\cup (\Omega \times \mathbb{R} \setminus \operatorname{supp} |Dv|). \tag{3.36}$$

It here holds that $Dv \llcorner E = 0$. Therefore, it follows

$$\mathscr{F}_p(v) = \int_{\Omega \times \mathbb{R}} h\left(x, t, \frac{\mathrm{d}Dv}{\mathrm{d}|Dv|}(x,t)\right) \mathrm{d}|Dv|(x,t) \tag{3.37}$$

$$= \int_{\Omega \times \mathbb{R}} h(x, t, \nu_v(x,t)) \, \mathrm{d}|Dv|(x,t). \tag{3.38}$$

As $v$ is a minimizer of (3.31), it holds $v \in BV(\Omega \times \mathbb{R}; [0,1])$. Thus we can use the coarea formula in BV [FR60, Thm. 1][AFP00, Thm. 3.40]: As $v \in BV(\Omega \times \mathbb{R}; [0,1])$, it follows that $\{v > s\}$ has finite perimeter in $\Omega \times \mathbb{R}$ for $\mathscr{L}^1$-almost all $s \in \mathbb{R}$ and $|Dv|(B) = \int_{-\infty}^{+\infty} |D1_{\{v>s\}}|(B) \, \mathrm{d}s$ for any Borel set $B \subseteq \Omega \times \mathbb{R}$. Consequently,

$$\mathscr{F}_p(v) = \int_{-\infty}^{+\infty} \int_{\Omega \times \mathbb{R}} h(x, t, \nu_v(x,t)) \, \mathrm{d}|D1_{\{v>s\}}|(x,t) \, \mathrm{d}s. \tag{3.39}$$

4: The definition of the Besicovitch derivative is given in Thm. 2.2.9.

Finally, by arguing that for $\mathcal{H}^n$-a.e. point on the boundary of $\{v > s\}$ it holds that $\nu_v(x, t) = \frac{D1_{\{v>s\}}}{|D1_{\{v>s\}}|}$, we get

$$\mathcal{F}_p(v) = \int_{-\infty}^{\infty} \int_{\Omega \times \mathbb{R}} h\left(x, t, \frac{D1_{\{v>s\}}}{|D1_{\{v>s\}}|}\right) \mathrm{d}|D1_{\{v>s\}}|(x,t)\, \mathrm{d}s \quad (3.40)$$

$$= \int_{-\infty}^{\infty} \mathcal{F}_p(1_{\{v>s\}})\, \mathrm{d}s. \quad (3.41)$$

This allows us to use the generalized coarea formula in order to show that $1_{\{v^*>s\}}$ is also a global minimizer of (3.31) for almost every $s \in [0, 1]$: For some minimizer $v^*$, we recall that the generalized coarea formula reads

$$\mathcal{F}_p(v^*) = \int_0^1 \mathcal{F}_p(1_{\{v^*>s\}})\, \mathrm{d}s. \quad (3.42)$$

As $v^*$ is a global minimizer of (3.31) and for every $s \in [0, 1]$ it holds $1_{\{v^*>s\}} \in \mathcal{C}$, the inequality $\mathcal{F}_p(v^*) \leq \mathcal{F}_p(1_{\{v^*>s\}})$ holds for any $s \in [0, 1]$. We show that in fact $\mathcal{F}_p(v^*) = \mathcal{F}_p(1_{\{v^*>s\}})$ for $\mathcal{L}^1$-a.e. $s \in [0, 1]$ by contradiction: Assume there is $\epsilon > 0$ and a subset $S \subset [0, 1]$ that is not $\mathcal{L}^1$-negligible such that

$$\mathcal{F}_p(v^*) + \epsilon \leq \mathcal{F}_p(1_{\{v^*>s'\}}) \quad (3.43)$$

for all $s' \in S$. Then

$$\mathcal{F}_p(v^*) \stackrel{(3.42)}{=} \int_0^1 \mathcal{F}_p(1_{\{v^*>s\}})\, \mathrm{d}s \geq \mathcal{F}_p(v^*) + \epsilon\mathcal{L}^1(S). \quad (3.44)$$

This is a contradiction; Therefore, $1_{\{v^*>s\}}$ is a minimizer of (3.31) for $\mathcal{L}^1$-a.e. $s \in [0, 1]$.

If $v^* \in \arg\min_{v \in \mathcal{C}} \mathcal{F}_p(v)$ and $\mathcal{F}_p(v^*) < +\infty$ then it must hold that $v^*$ is non-increasing with respect to the $\Gamma$-axis, i.e., $D_t v^*(B) \leq 0$ for all Borel sets $B \subseteq \Omega \times \mathbb{R}$, due to the definition of $h$ in (3.17). Consequently, there exists $u : \Omega \to \Gamma$ such that $1_{\{v^*>s\}} = \mathbf{1}_u$.

It remains to show that $u$ is of bounded variation in $\Omega$. Following the coarea formula in BV [AFP00, Thm. 3.40], $|\mathbf{1}_{\{v^*>s\}}|(\Omega \times \mathbb{R}) < +\infty$ for $\mathcal{L}^1$-a.e. $s \in [0, 1]$. As the subgraph hyp $u$ is a set of finite perimeter in $\Omega \times \mathbb{R}$, we can apply [AMP04, Prop. 4.2] and deduce that $u$ is of bounded variation in $\Omega$. $\qquad\square$

**Dual Formulation.** A useful transformation is to employ duality in order to reformulate the problem (3.31) as a maximum flux of (dual) vector fields through the measure-theoretic boundary of the primal variable. The resulting primal-dual formulation is linear

and solving it numerically with the PDHG algorithm only involves projections onto convex sets.

---

**Definition 3.2.5** (Dual formulation) *Let $\Omega \subset \mathbb{R}^n$ be open and $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \overline{\mathbb{R}}_+$ be a Borel function which fulfils (A1)-(A3). The lifted dual problem is defined as*

$$\inf_{v \in \mathscr{C}} \mathscr{F}(v), \tag{3.45}$$

*with $\mathscr{C}$ as in Def. 3.2.3 and*

$$\mathscr{F}(v) := \sup_{\varphi \in \mathscr{K}} \left\{ \int_{\Omega \times \mathbb{R}} \langle \varphi, dDv \rangle \right\}, \tag{3.46}$$

$$\mathscr{K} := \{(\varphi_x, \varphi_t) \in C_0(\Omega \times \mathbb{R}; \mathbb{R}^{n+1}) :$$
$$\varphi_t(x,t) \geq f^*(x,t,\varphi_x(x,t)), \ \forall (x,t) \in \Omega \times \mathbb{R}\}. \tag{3.47}$$

*The conjugate in (3.47) is to be understood w.r.t. the last variable.*

---

The following duality result shows that the primal lifted energy $\mathscr{F}_p$ and the dual lifted energy $\mathscr{F}$ agree on the relaxed solution space $\mathscr{C}$ given in Def. 3.2.3:

---

**Theorem 3.2.6** (Duality [Vog19, Thm. 6.5]) *Let $\Omega \subset \mathbb{R}^n$ be open and $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \overline{\mathbb{R}}_+$ be a Borel function which fulfils (A1)-(A3). Then, for every $v \in \mathscr{C}$ defined as in Def. 3.2.3, it holds*

$$\mathscr{F}_p(v) = \mathscr{F}(v), \tag{3.48}$$

*with $\mathscr{F}_p$ as in Def. 3.2.1 and $\mathscr{F}$ as in Def. 3.2.5.*

---

In [Poc+10, Thm. 3.2] it is shown that $\mathscr{F}(\mathbf{1}_u) = \mathscr{F}_p(\mathbf{1}_u)$ for $u \in W^{1,1}(\Omega; \Gamma)$ and the conjecture is made that the equality should also hold for $u \in \mathrm{BV}(\Omega; \Gamma)$. In [Vog20, Thm. 6.5] the duality result is proven. As the proof is interesting and illuminates the necessity of the assumptions, we here sketch the main line of argument, which relies among others on central results from [Mic56; Per18].

*Proof.* We define the set-valued mapping $S : \Omega \times \mathbb{R} \rightrightarrows \mathbb{R}^{n+1}$ as

$$S(x,t) := \left\{ (\psi_x, \psi_t) \in \mathbb{R}^{n+1} \,\middle|\, \psi_t \geq f^*(x,t,\psi_x) \right\}. \tag{3.49}$$

By $C(S)$ we denote the associated set of *continuous selections*:

$$C(S) := \{(\varphi_x, \varphi_t) \in C_0(\Omega \times \mathbb{R}; \mathbb{R}^{n+1}) : \tag{3.50}$$
$$(\varphi_x, \varphi_t)(x,t) \in S(x,t), \ \forall (x,t) \in \Omega \times \mathbb{R}\}. \tag{3.51}$$

The proof then consists of four steps.

<u>First</u>, we show that $S$ is

1. *non-empty*, i.e., $S(x,t) \neq \emptyset$ for each $(x,t) \in \Omega \times \mathbb{R}$,
2. *closed*, i.e., $S(x,t)$ is closed for each $(x,t) \in \Omega \times \mathbb{R}$,
3. *convex-valued*, i.e., $S(x,t)$ is convex for each $(x,t) \in \Omega \times \mathbb{R}$,
4. *inner semicontinuous*, i.e., the preimage $S^{-1}(Z) \subset \Omega \times \mathbb{R}$ is open for each open set $Z \subset \mathbb{R}^{n+1}$, where

$$S^{-1}(Z) := \{(x,t) \in \Omega \times \mathbb{R} : S(x,t) \cap Z \neq \emptyset\}. \qquad (3.52)$$

Using the definition of the Fenchel conjugate, we rewrite the set $S$ as

$$S(x,t) = \{(\psi_x, \psi_t) \in \mathbb{R}^{n+1} : \qquad\qquad\qquad (3.53)$$

$$f(x,t,p) \geq \langle \psi_x, p \rangle - \psi_t, \ \forall p \in \mathbb{R}^n\}. \quad (3.54)$$

In this alternative formulation, we see that $S$ is closed. According to (A4) there exist $c > 0$ and $d \geq 0$ such that $c\|p\| - d \leq f(x,t,p)$ for all $(x,t) \in \Omega \times \mathbb{R}$. It follows directly that $S$ is non-empty as $(0,d) \in S(x,t)$ for each $(x,t) \in \Omega \times \mathbb{R}$. Note that even without assumption (A4) it is possible to argue that $S$ is non-empty using (A1) and (A3).

For any $(\psi'_x, \psi'_t), (\psi''_x, \psi''_t) \in S(x,t)$, any $p \in \mathbb{R}^n$, and any $\lambda \in [0,1]$, we have

$$\langle \lambda \psi'_x + (1-\lambda)\psi''_x, p \rangle - \lambda \psi'_t - (1-\lambda)\psi''_t \qquad\qquad (3.55)$$

$$= \lambda \langle \psi'_x, p \rangle - \lambda \psi'_t + (1-\lambda) \langle \psi''_x, p \rangle - (1-\lambda)\psi''_t \qquad (3.56)$$

$$\leq \lambda f(x,t,p) + (1-\lambda)f(x,t,p) \qquad\qquad\qquad (3.57)$$

$$= f(x,t,p). \qquad\qquad\qquad\qquad\qquad (3.58)$$

Consequently, $S$ is also convex-valued.

Inner semicontinuity follows from the assumptions (A1)-(A3) and the following proposition:

**Proposition 3.2.7** *[Vog19, Prop. 6.11] Let $\Omega \times \mathbb{R} \subset \mathbb{R}^{n+1}$ be non-empty and assume that $f : \Omega \times \mathbb{R} \times \mathbb{R}^n$ is convex in the last variable as in (A1) and locally bounded in the first two variables as in (A3). Then $S$ defined as in (3.49) is inner semicontinuous if and only if $f$ is lower semicontinuous in all variables as in (A2).*

<u>Second</u>, we argue that $C(S) \neq \emptyset$. This can be shown with Michael's selection theorem:

**Theorem 3.2.8** (Michael's Selection Theorem [Mic56, Thm. 3.2″]) *Let $S : X \rightrightarrows Y$ be a set-valued mapping for a paracompact $T_1$-space $X$ and Banach space $Y$. If $S$ is non-empty, closed, convex-valued and inner semicontinuous, then $C(S) \neq \emptyset$.*

Here, it holds $X = \Omega \times \mathbb{R}$ and $Y = \mathbb{R}^{n+1}$. The latter is a Banach space and the former is a metric space and as such paracompact and $T_1$ [Sto48, Cor. 1, Thm. 2]. As $S$ as in (3.49) is non-empty, closed, convex-valued, and inner semicontinuous it follows that the set of continuous selections is non-empty.

<u>Third</u>, we show an integral representation for the support function

$$\sigma_{C(S)}(Dv) := \sup_{\varphi \in C(S)} \left\{ \int_{\Omega \times \mathbb{R}} \langle \varphi, \, \mathrm{d}Dv \rangle \right\}. \qquad (3.59)$$

This can be done using the following theorem:

**Theorem 3.2.9** (Integral Representation [Per18, Thm. 1]) *Let $X$ be a second-countable, locally compact Hausdorff space and let $S : X \rightrightarrows \mathbb{R}^{n+1}$ denote a closed, convex-valued mapping such that $C(S) \neq \emptyset$. Then the support function of $C(S)$ on $\mathcal{M}(X, \mathbb{R}^{n+1})$ has the following integral representation*

$$\sigma_{C(S)}(y) = \int_X \sigma_{S(x)} \left( \frac{\mathrm{d}\theta}{\mathrm{d}|\theta|}(x) \right) \mathrm{d}|\theta|(x) \qquad (3.60)$$

*if and only if $S$ is inner semicontinuous.*

In our setting, we have $X = \Omega \times \mathbb{R}$. As $\Omega \times \mathbb{R}$ is a subset of $\mathbb{R}^{n+1}$, it is a second-countable [Wil04, Def. 16.1], locally compact [Wil04, Def. 18.1] Hausdorff ($T_2$) space [Wil04, Def. 13.5]. We have already established that $S$ as in (3.49) is closed and convex-valued and $C(S)$ is non-empty. Thus, we can use the integral representation theorem and express (3.59) as

$$\sigma_{C(S)}(Dv) = \int_{\Omega \times \mathbb{R}} \sigma_{S(x,t)} \left( \frac{\mathrm{d}Dv}{\mathrm{d}|Dv|}(x,t) \right) \mathrm{d}|Dv|(x,t). \qquad (3.61)$$

This means that the support function of a set of functions (3.59) has an integral representation as a point-wise support function of a set of $\mathbb{R}^{n+1}$ vectors.

Fourth, formally negating the last component of $\varphi \in S(x, t)$ and $Dv$, we can directly apply the finite-dimensional result in Cor. 2.1.4 which gives

$$\sigma_{S(x,t)}\left(\frac{\mathrm{d}Dv}{\mathrm{d}|Dv|}(x, t)\right) = h\left(x, t, \frac{\mathrm{d}Dv}{\mathrm{d}|Dv|}(x, t)\right) \qquad (3.62)$$

and thus it follows $\mathcal{F}(v) = \mathcal{F}_p(v)$. $\qquad\qquad\square$

Let us now consider important properties of the lifted dual problem.

**Lemma 3.2.10** (Properties of $\mathcal{F}$) *Let $\Omega \subset \mathbb{R}^n$ be open and bounded. Let $f : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \overline{\mathbb{R}}_+$ be a Borel function fulfilling (A1)-(A3) and let $\mathcal{F}$ be defined as in Def. 3.2.5. Then it holds that $v \mapsto \mathcal{F}(v)$ is convex and lower semicontinuous on $\mathrm{BV}(\Omega \times \mathbb{R}; [0, 1])$ with respect to weak*-convergence.*

*Proof.* Recall that the space of finite Radon measures, which we denote by $\mathcal{M}(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$, is the dual space of $C_0(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$ [AFP00, Thm. 1.54, Remark 1.57]. Since $\mathcal{K} \subset C_0(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$, it follows that the functional $\tilde{\mathcal{F}} : \mathcal{M}(\Omega \times \mathbb{R}; \mathbb{R}^{n+1}) \to \mathbb{R}$ defined as

$$\tilde{\mathcal{F}}(Dv) = \sup_{\varphi \in \mathcal{K}} \int_{\Omega \times \mathbb{R}} \langle \varphi, \mathrm{d}Dv \rangle \qquad (3.63)$$

is the pointwise supremum over linear, weakly* continuous functions. This implies that $\tilde{\mathcal{F}}$ is convex and lower semicontinuous in $\mathcal{M}(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$ with respect to weak*-convergence in measure. Since $v \mapsto Dv$ is linear it follows that $v \mapsto \mathcal{F}(v)$ is convex.

Assume a sequence $(v_n)_{n \in \mathbb{N}} \subset \mathrm{BV}(\Omega \times \mathbb{R}; [0, 1])$ is given that weakly* converges to $v$ in BV, i.e., it holds that [AFP00, Def. 3.11]

- $v_n$ converges to $v$ in $L^1(\Omega \times \mathbb{R})$
- $Dv_n$ weakly* converges to $Dv$ in measure.

From the lower semicontinuity of $\tilde{\mathcal{F}}$ in $\mathcal{M}(\Omega \times \mathbb{R}; \mathbb{R}^{n+1})$ with respect to weak*-convergence in measure it directly follows that $\mathcal{F}$ is lower semicontinuous on $\mathrm{BV}(\Omega \times \mathbb{R}; [0, 1])$ with respect to weak* convergence in BV:

$$\lim_{n \to \infty} \mathcal{F}(v_n) = \lim_{n \to \infty} \tilde{\mathcal{F}}(Dv_n) \geq \tilde{\mathcal{F}}(Dv) = \mathcal{F}(v). \qquad (3.64)$$
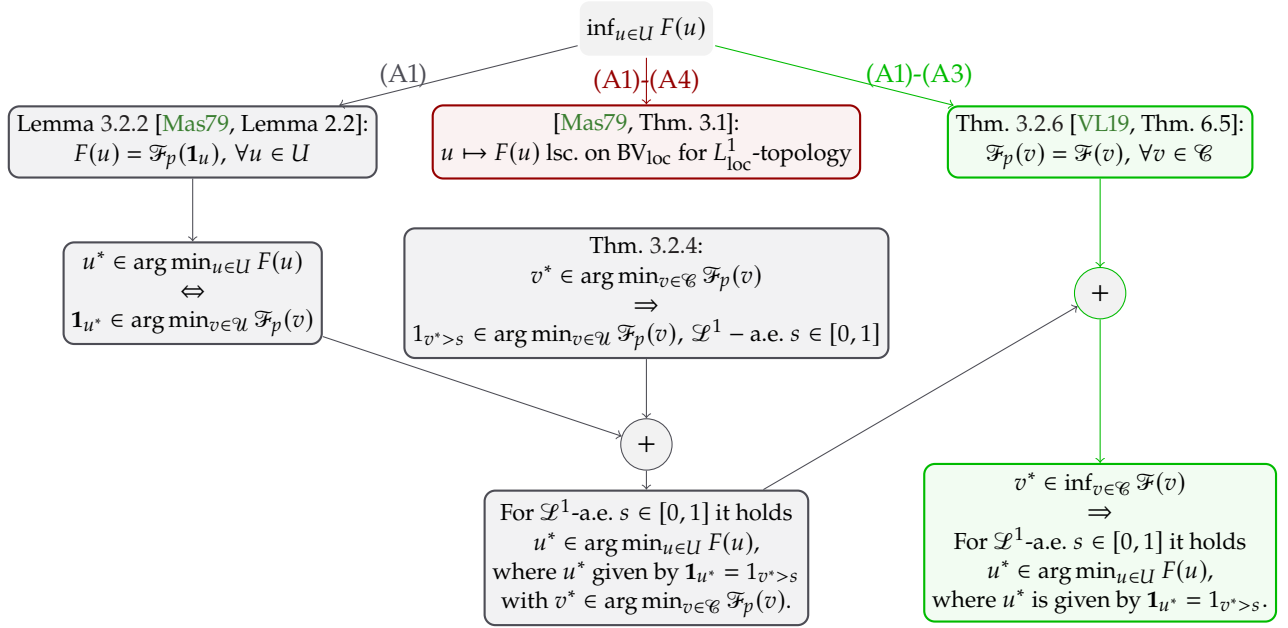
$\qquad\qquad\square$

**Figure 3.2: Overview of the Theoretical Results.** This flowchart visualizes the line of argument and also highlights which assumptions are needed for which results. **In red (top middle)**, we have the lower semicontinuity of the original problem on $BV_{\text{loc}}$ with respect to the $L^1_{\text{loc}}$-topology. Together with the compactness result in BV this is an important ingredient for proving existence of minimizers. **In gray (left)**, it is shown that the equivalence between the original and (primal) lifted problem over the non-convex solution space $\mathcal{U}$ has the mildest assumptions, namely (A1). **In gray (middle)**, the most exciting result and line of argument is shown, which also only requires assumption (A1): the tightness of the primal lifted problem over the convex solution space $\mathcal{C}$ as well as the existence of the thresholding formula. **In green (right)**, it is shown that the equivalence between the primal and dual lifted problem over the convex solution space $\mathcal{C}$ holds under assumptions (A1)-(A3). Together, with the tightness and thresholding results for the primal lifted problem this implies that solving the high-dimensional fully convex dual problem allows to find (global) minimizers of the originally non-convex problem.

**Summary.** We have shown that any variational problems of the form (3.1) fulfilling (A1)-(A4) can be lifted into a higher-dimensional, convex problem as in Def. 3.2.5. According to Thm. 3.2.6 and Thm. 3.2.4, the lifted energy is tight and any global minimizer $v^* \in \arg\min_{v \in \mathcal{C}} \mathcal{F}(v)$ of the lifted problem can be mapped to a global minimizer of the original problem. In Fig. 3.2 we give a short overview over the theoretical results, line of argument and needed assumptions.

**A Note on the TV Regularizer.** Assume that the integrand $f : \Omega \times \Gamma \times \mathbb{R}^d \to \mathbb{R}$ can be split into a data term and a regularizer as follows

$$f(x, t, p) = \rho(x, t) + \eta(p). \tag{3.65}$$

For a positive, 1-homogeneous, convex regularizer – such as TV – the combined constraint set (3.47) can then be split into two separate constraint sets: First we note that the constraint in (3.47), which involves calculating the Fenchel conjugate with respect to

the third variable, can be rewritten as

$$\varphi_t(x,t) \geq -\rho(x,t) + \eta^*(\varphi_x(x,t)). \qquad (3.66)$$

This can be seen by applying the definition of the Fenchel conjugate pointwise:

$$f^*(x,t,p) = \sup_q \{\langle p,q \rangle - f(x,t,q)\} \qquad (3.67)$$

$$= \sup_q \{\langle p,q \rangle - \rho(x,t) - \eta(q)\} \qquad (3.68)$$

$$= -\rho(x,t) + \sup_q \{\langle p,q \rangle - \eta(q)\} \qquad (3.69)$$

$$\qquad (3.70)$$

As $\eta$ is assumed to be convex and 1-homogeneous, it follows that $\eta^*$ is an indicator function with range $\{0, +\infty\}$ [RW09, Exm. 11.4]. Therefore, the dual formulation (3.46)–(3.47) can equivalently be written as

$$\mathcal{F}(v) := \sup_{\substack{\varphi_x \in \mathcal{K}_x \\ \varphi_t \in \mathcal{K}_t}} \left\{ \int_{\Omega \times \mathbb{R}} \langle (\varphi_x, \varphi_t)^\top, \mathrm{d}Dv \rangle \right\}, \quad (3.71)$$

$$\mathcal{K}_x := \{\varphi_x \in C_0(\Omega \times \mathbb{R}; \mathbb{R}^n) : \varphi_x(x,t) \in \mathrm{dom}\,\eta^*, \forall(x,t)\}, \quad (3.72)$$

$$\mathcal{K}_t := \{\varphi_t \in C_0(\Omega \times \mathbb{R}; \mathbb{R}) : -\varphi_t(x,t) \leq \rho(x,t), \forall(x,t)\}. \quad (3.73)$$

As we will discuss in Sec. 3.4, this problem can be solved using the PDHG algorithm which involves projections onto the constraint sets (3.72)–(3.73).

## 3.3 Discretization

Practical implementations of the calibration-based lifted problem in Def. 3.2.5 come with one challenge in particular: during lifting, the originally continuous range $\Gamma$ becomes part of the domain $\Omega \times \mathbb{R}$, which can greatly affect the dimensionality of the discretized problem. In this section we discuss numerical solutions, concentrating on the special case where the integrand $f : \Omega \times \Gamma \times \mathbb{R}^d \to \overline{\mathbb{R}}$ of the original problem can be split into

$$f(x,t,p) = \rho(x,t) + \eta(p), \qquad (3.74)$$

such that $\rho : \Omega \times \Gamma \to \overline{\mathbb{R}}$ denotes the integrand of a possibly non-convex data term and $\eta : \mathbb{R}^n \to \overline{\mathbb{R}}$ the integrand of a positive, 1-homogeneous, convex regularizer. In this case, the calibration-based lifted problem is of the form (3.71)–(3.73); note that the

information of the original data term is completely and exclusively included in the constraint set $\mathcal{K}_t$ (3.73). The goal of this section is to explain the numerical problem which arises from making the range $\Gamma$ part of the domain $\Omega \times \mathbb{R}$ and to discuss several discretization schemes. This section does not contain own theoretical contributions but summarizes and discusses results found in [Poc+09; Poc+10; Möl+15; MC17].

### 3.3.1 Finite Differences with Pointwise Constraints

Let us first consider a simple finite-differences approach in both $\Omega$ and $\Gamma$. We start by introducing a regular grid $\Omega^h \times \Gamma^h$ of size $m_1 \times .. \times m_n \times l$, discretizing both $\Omega$ and $\Gamma$ in the same way, and using a finite-differences approximation for the differential operatur $D$ in (3.71). Enforcing the constraint set (3.73) on each point of the grid results in

$$\mathcal{K}_t^h := \{\varphi_t^h \in \mathbb{R}^{N_x, l} : -\left(\varphi_t^h\right)_{i,j} \leq \rho(x_i, \gamma_j),$$
$$\forall (x_i, \gamma_j) \in \Omega^h \times \Gamma^h\}. \qquad (3.75)$$

The key observation using this approach is that enforcing the constraints only on the grid discards the values of the data term on all non-grid points, i.e., on all points not in the chosen set of *labels* $\Gamma^h$. This effectively turns the problem, which is originally formulated on a continuous range, into a labeling problem with a finite set of labels. A visualization can be found in Fig. 3.5a.

Such a finite difference approach is used in [Poc+09; Poc+10]. Implementing and solving the lifted problem with this discretization is straightforward and involves only simple projections that can be efficiently computed and will further be specified in (3.124)–(3.125). However, the results suffer from a comparatively strong *label-bias*[MC17]: solutions are drawn towards the discrete set $\Gamma^h$ due to the loss of information on the data term in between the chosen labels; see Fig. 3.3 for an example of the resulting visual effect. In order to achieve numerically and visually pleasing results, the number of labels needs to be quite large, which, however, increases memory usage and runtime.
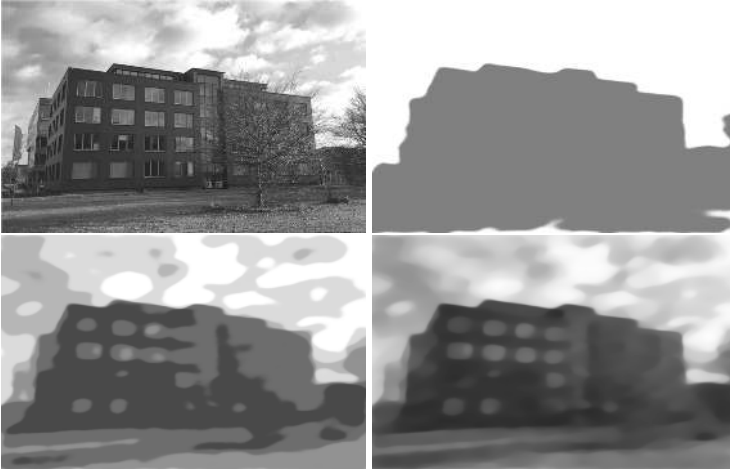
**Figure 3.3: Discretization Artifacts.** In order to illustrate some potential pitfalls encountered during the numerical optimization of the lifted energy, we consider an $L^2 - TV$ denoising problem. The energy is lifted with the CBL approach and discretized using a finite differences approach with 2, 7, and 100 labels **(top right to bottom right)**. A solution for the input image **(top left)** and regularization parameter $\lambda = 0.4$ is then computed with the PDHG algorithm. Especially the solution for 2 labels shows the strong label bias of this discretization scheme and the resulting cartoonish artifacts.

### 3.3.2 Approximation of Label-Continuous Formulation

In [MC17] the authors consider finite-element approximations of $v, \varphi_x$ and $\varphi_t$ with respect to the $\Gamma$-axis and discuss the impact of the choice of the used basis functions. In terms of the finite-element coefficients, they reach multi-label approximations of the label-continuous calibration-based lifted energy and demonstrate that these approximations are related to other known semi-discrete (continuous domain $\Omega$, discrete range $\Gamma$) relaxation approaches. We here highlight certain results shown in [MC17]. This serves the purpose of motivating the choice of the *sublabel-accurate* discretization approach which we use in experiments in later chapters. As the overview is kept rather short and does not focus on mathematical details, we refer the interested reader to [MC17].

**Grid and Basis Functions.** We follow [MC17] and introduce both a nodal and a cell-centered grid for the $\Gamma$ axis: For the nodal grid, the $\Gamma = [\gamma_1, \gamma_l]$ axis is split into $l-1$ non-overlapping, equally-sized intervals $\{\Gamma_i\}_{i=1,\dots,l-1}$, where $\Gamma_i = [\gamma_i, \gamma_{i+1}]$. The grid spacing is denoted by $h = \gamma_{i+1} - \gamma_i$. Furthermore, the grid is padded with $\gamma_0 := \gamma_1 - h$ and $\gamma_{l+1} := \gamma_l + h$. The associated cell-centered grid is denoted by $\{\gamma_i^*\}_{i=0,\dots,l}$, where $\gamma_i^* = \frac{\gamma_i + \gamma_{i+1}}{2}$.

In [MC17] the following constant and linear basis functions are introduced (see also Fig. 3.4 for a visualization): Constant basis functions centered around the cell-centered grid points

$$\Phi_i^0(t) := \begin{cases} 1, & \text{if } t \in [\gamma_i, \gamma_{i+1}], \\ 0, & \text{else,} \end{cases} \qquad (3.76)$$
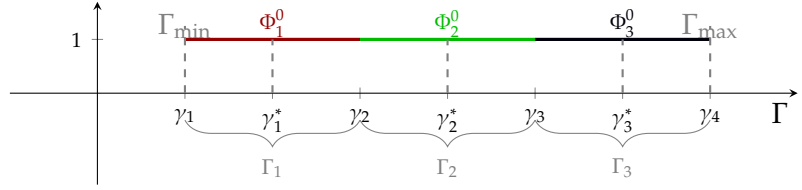
**(a)** Piecewise constant basis functions on the cell-centered grid, see also definition of $\mathbf{\Phi}^0_i$ in (3.76).
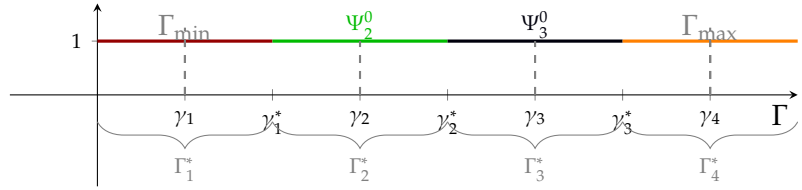


**(b)** Piecewise constant basis functions on the nodal grid, see also definition of $\mathbf{\Psi}^0_i$ in (3.77).

**Figure 3.4: Finite-element approximation.** Practical implementations of the calibration-based lifted problem in Def. 3.2.5 involve a discretization of the continuous range $\Gamma$. We here illustrate basis functions used for a finite-element approximation of $v, \varphi_x$, and $\varphi_t$ with respect to the $\Gamma$-axis. Depending on each choice of basis function, inserting the finite-element approximations of $v, \varphi_x$, and $\varphi_t$ into the original CBL energy preserves more or less information of the data term (see Fig. 3.5).
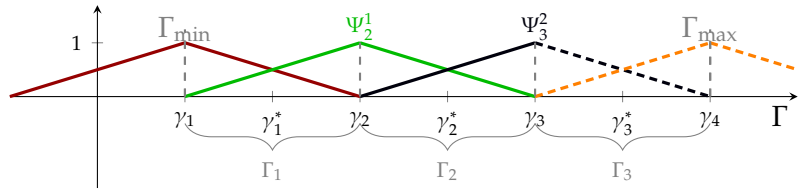


**(c)** Piecewise linear basis functions on the nodal grid, see also definition of $\mathbf{\Psi}^1_i$ in (3.78).

constant basis functions centered around the nodal grid points

$$\mathbf{\Psi}^0_i(t) := \begin{cases} 1, & \text{if } t \in [\gamma^*_i, \gamma^*_{i+1}], \\ 0, & \text{else,} \end{cases} \tag{3.77}$$

as well as linear basis functions centered around the nodal grid points

$$\mathbf{\Psi}^1_i(t) := \begin{cases} \frac{1}{h}(t - \gamma_{i-1}), & \text{if } t \in [\gamma_{i-1}, \gamma_i], \\ \frac{1}{h}(\gamma_{i+1} - t), & \text{if } t \in [\gamma_i, \gamma_{i+1}], \\ 0, & \text{else.} \end{cases} \tag{3.78}$$
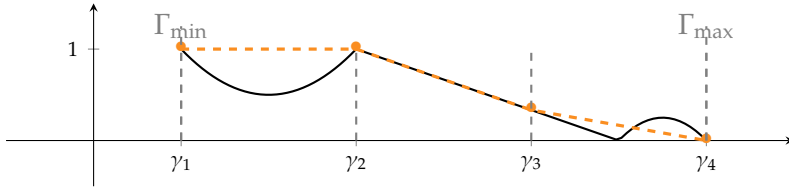
**(a) Finite Differences.** Using finite differences for discretizing the calibration-based lifted problem (3.71)–(3.73) results in a loss of information contained in the data term. Considering for fixed $x \in \Omega$ the data term $\rho(x, \cdot)$ **(black)**, the only information included in the discretized constraint $\mathcal{K}_t^h$ consists in the values of the data term at the chosen discretization points (labels) $\gamma_1, ..., \gamma_l \in \Gamma$ **(orange dots)**. In between the chosen labels the data term is approximated piecewise linearly. The coarser the discrete grid $\Gamma^h$, the more information is lost.
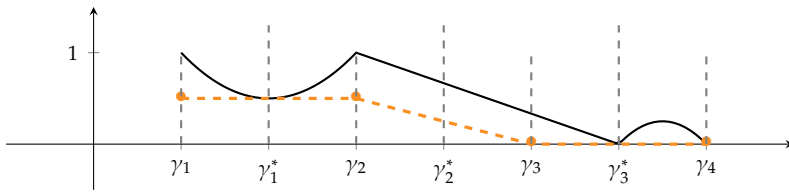


**(b) Min-Pool Discretization.** Consider a finite-element approximation of $\varphi_t$ using the basis function $\Psi_i^0$. Plugging this approximation into the constraint set $\mathcal{K}_t$ (3.73), we see that for fixed $x \in \Omega$ only the minimal value of the data term $\rho(x, \cdot)$ on each interval $[\gamma_{i-1}^*, \gamma_i^*]$ is known after this discretization **(orange dots)**. A further finite-element approximation of $v$ and $\varphi_x$ using the basis functions $\Phi_i^0$ and discretization of the energy (3.71)–(3.72) shows that the data term is approximated piecewise linearly in between the chosen labels $\gamma_i$.



**(c) Sublabel-Accurate Discretization.** While the primal and dual variables $v, \varphi_x$ are approximated in a piecewise constant way using the basis functions $\Phi_i^0$, the dual variable $\varphi_t$ is approximated using the basis functions $\Psi_i^1$. This discretization corresponds to a piecewise convex approximation of t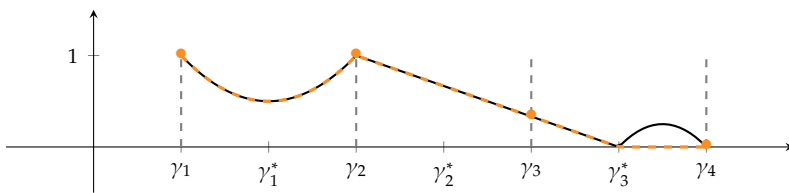he continuous data term in between the nodal grid points. Compared to the other discretization approaches, more information of the data term is preserved. This also has a positive impact on the numerical results: fewer labels are required in order to achieve acccurate results. The numerical solutions do not exhibit the substantial label bias artifacts that are encountered in the finite-differences setting (see Fig. 3.3).

**Figure 3.5: Discretization.** Three possible discretization schemes for the calibration-based lifted energy (3.71)–(3.73). Numerically solving the problem requires a discretization of the range $\Gamma$. Since we are typically interested in solutions with continuous range $\Gamma$ it is of special interest how much information of the original data term (which is formulated for continuous $\Gamma$) is preserved using different discretization schemes.

**Variables $v$ and $\varphi_x$.** In the following, the primal variable $v$ and the dual variable $\varphi_x$ are approximated using the $\Phi_i^0$ basis functions:

$$v(x,t) \approx \sum_{i=1}^{l} \hat{v}(x,i)\Phi_i^0(t), \qquad (3.79)$$

$$\varphi_x(x,t) \approx \sum_{i=1}^{l} \hat{\varphi}_x(x,i)\Phi_i^0(t), \qquad (3.80)$$

where $\hat{v} : \Omega \times \{1,...,l\} \to \mathbb{R}$ and $\hat{\varphi}_x : \Omega \times \{1,...,l\} \to \mathbb{R}^n$ denote some coefficient functions.

**Min-Pool Discretization.** Let $\varphi_t$ be approximated using the $\Psi_i^0$ basis functions:

$$\varphi_t(x,t) \approx \sum_{i=1}^{l} \hat{\varphi}_t(x,i)\Psi_i^0(t), \qquad (3.81)$$

where $\hat{\varphi}_t : \Omega \times \{1,...,l\} \to \mathbb{R}$ denotes the coefficient function. Consider again the constraint set $\mathcal{K}_t$. In order for

$$\sum_{i=1}^{l} \hat{\varphi}_t(x,i)\Psi_i^0(t) \approx -\varphi_t(x,t) \le \rho(x,t) \qquad (3.82)$$

to be fulfilled for all $(x,t) \in \Omega \times \Gamma$, the coefficient function must fulfill

$$\hat{\varphi}_t(x,i) \le \min_{\gamma \in [\gamma_{i-1}^*, \gamma_i^*]} \rho(x,\gamma) \qquad (3.83)$$

for all $(x,i) \in \Omega \times \{2,...,l-1\}$ and, similarly,

$$\hat{\varphi}_t(x,1) \le \min_{\gamma \in [\gamma_1, \gamma_1^*]} \rho(x,\gamma), \qquad (3.84)$$

$$\hat{\varphi}_t(x,l) \le \min_{\gamma \in [\gamma_{l-1}^*, \gamma_l]} \rho(x,\gamma). \qquad (3.85)$$

Consequently, the only information included in the discretized constraint set $\mathcal{K}_t$ is the minimal value of the data term $\rho(x,\cdot)$ on each interval $[\gamma_{i-1}^*, \gamma_i^*]$. Combined with the finite-element approximation of $v$ and $\varphi_x$, this leads to a piecewise linear approximation of the data term $\rho(x,\cdot)$ in between the chosen labels – for details we refer the reader to [MC17, Prop. 2, Fig. 4]. The approximation of the data term is visualized in Fig. 3.5b.

**Sublabel-Accurate Discretization.** Let $\varphi_t$ be approximated using the $\Psi_i^1$ basis functions:

$$\varphi_t(x,t) \approx \sum_{i=1}^{l} \hat{\varphi}_t(x,i)\Psi_i^1(t), \qquad (3.86)$$

where $\hat{\varphi}_t : \Omega \times \{1, ..., l\} \to \mathbb{R}$ denotes again the coefficient function. Consider again the constraint set $\mathcal{K}_t$. In order for

$$-\sum_{i=1}^{l} \hat{\varphi}_t(x,i)\Psi_i^1(t) \approx -\varphi_t(x,t) \leq \rho(x,t) \qquad (3.87)$$

to be fulfilled for all $(x,t) \in \Omega \times \Gamma$, the coefficient function must fulfill

$$\frac{\hat{\varphi}_t(x,i)\gamma_{i+1} - \hat{\varphi}_t(x,i+1)\gamma_i}{h} \geq \rho_i^* \left( \frac{\hat{\varphi}_t(x,i) - \hat{\varphi}_t(x,i+1)}{h} \right)$$
$$(3.88)$$

for any $(x,i) \in \Omega \times \{1, ..., l\}$, where

$$\rho_i(x,t) := \rho(x,t) + \delta_{\Gamma_i}(t), \quad \delta_{\Gamma_i}(t) := \begin{cases} 0, & \text{if } t \in \Gamma, \\ +\infty, & \text{else.} \end{cases} \qquad (3.89)$$

*Derivation of* (3.88). We consider the constraint on each interval $\Gamma_i$ separately and insert the finite element approximation. This gives

$$\inf_{t\in\Gamma_i} \left\{ \sum_{j=i}^{i+1} \hat{\varphi}_t(x,j)\Psi_j^1(t) + \rho(x,t) \right\} \geq 0 \quad \Leftrightarrow \quad (3.90)$$

$$\inf_{t\in\Gamma_i} \left\{ \hat{\varphi}_t(x,i)\frac{\gamma_{i+1} - t}{h} + \hat{\varphi}_t(x,i+1)\frac{t - \gamma_i}{h} + \rho(x,t) \right\} \geq 0. \quad (3.91)$$

Introducing the notation

$$a(i) := \frac{\hat{\varphi}_t(x,i)\gamma_{i+1} - \hat{\varphi}_t(x,i+1)\gamma_i}{h}, \qquad (3.92)$$

$$r(i) := \frac{(\hat{\varphi}_t(x,i) - \hat{\varphi}_t(x,i+1))}{h}, \qquad (3.93)$$

we can bring (3.91) into the form of a Fenchel conjugate:

$$\inf_{t\in\Gamma_i} \{a(i) - \langle r(i),t \rangle + \rho(x,t)\} \geq 0 \qquad (3.94)$$

$$\Leftrightarrow a(i) - \sup_{t\in\mathbb{R}} \{\langle r(i),t \rangle - \rho_i(x,t)\} \geq 0 \qquad (3.95)$$

$$\Leftrightarrow a(i) - \rho_i^*(r(i)) \geq 0. \qquad (3.96)$$

Combined with the finite-element approximation of $v$ and $\varphi_x$, this leads effectively to a piecewise convex approximation of the data term $\rho(x, \cdot)$ in between the chosen labels, see Fig. 3.5c. The resulting "sublabel-accurate" semi-discrete energy was first proposed as a lifting strategy in a semi-discrete setting (continuous $\Omega$, discrete $\Gamma$) [Möl+15]; As the derivation in [Möl+15] makes the piecewise convex approximation of the data term better understandable and we rely on the related results and notations in Chpt. 4, we present a summary in Sec. 3.3.3. The equivalence to the above-discussed finite-element approximation of the fully continuous calibration-based lifted energy is shown in [MC17, Prop. 4].

Due to the sublabel-accurate approximation of the data term, the sublabel-accurate discretization scheme is generally superior to the other two discussed discretization schemes: Fewer labels are needed in order to achieve numerically and visually pleasing results. In [Möl+15], the authors demonstrate that sublabel-accurate results achieved with $l = 32$ labels are on par with finite-difference results achieved with $l = 270$ labels. Therefore, the memory requirement and overall runtime is lower compared to other approaches.

However, as demonstrated in [Möl+15, Sec. 4], the implementation is more involved: During optimization, projections onto the epigraph of the piecewise[5] Fenchel conjugate of the data term are required – an operation which is not always easy to compute. Besides this practical hurdle, there is also a theoretical pitfall: While global minimizers of the original problem can be mapped to global minimizers of the lifted problem, there are *non-integral* global minimizers in the lifted setting which cannot be mapped to global minimizers in the original setting.

5: in between the labels

### 3.3.3 Sublabel-Accurate Lifting

For reference, we provide a short summary of the lifting approach with sublabel-accurate lifting for a TV regularized problem as presented in [Möl+15]. The derivation of the sublabel-accurate lifting approach neither directly builds on the calibration-based lifting formulation in Sec. 3.2 nor the discretization and approximation described in Sec. 3.3, but offers intuition on why the data term is effectively approximated in a piecewise convex way as shown in Fig. 3.5c.

Even though the line of argument differs from the one given in the previous subsections, the final semi-discrete problem is equivalent to the one which is obtained using calibration-based lifting in combination with a finite-element approximation of the variables $v$, $\varphi_x$ and $\varphi_t$ as described on p. 63, see [MC17, Prop. 4]. The notation is borrowed from [Möl+15] and will later be used in Chpt. 4.

In the following, we again assume that the energy $F$ can be split into a data term $H$ and total variation regularizer TV such that

$$F(u) := \underbrace{\int_\Omega \rho(x, u(x))\, dx}_{H(u)} + \underbrace{\int_\Omega d|Du|}_{\text{TV}(u)} \qquad (3.97)$$

and consider the problem

$$\inf_{u:\Omega\to\Gamma} F(u). \qquad (3.98)$$

In the previous section, we used calibration-based lifting in order to find a convex representation of this continuous(-domain) problem. In this section, the first step is to discretize the range $\Gamma$ using $l$ discretization points, also called *labels*, and using the discrete, nodal grid $\Gamma^h$ as introduced on p. 59. The second step is to identify scalar functions $u : \Omega \to \Gamma$ with vectorial functions $\boldsymbol{u} : \Omega \to \mathbb{R}^{l-1}$ using a pointwise argument. Subsequently, (3.98) is reformulated as

$$\inf_{\boldsymbol{u}:\Omega\to\mathbb{R}^{l-1}} \boldsymbol{F}(\boldsymbol{u}); \qquad (3.99)$$

in this form the problem is then convexified. The following arguments are formal and pointwise in terms of the integrands $\rho$ and $\eta$ for fixed $x \in \Omega$, dropping the dependency on $x$ and disregarding any regularity requirements.

From here on, bold symbols typically denote the semi-discretized lifted setting.

**Lifting the Solution Space [Möl+15, Chpt. 3].** Using the nodal grid $\Gamma^h$ introduced in Sec. 3.3, any scalar value in $\gamma_i^\alpha \in \Gamma$ can be written as a linear combination

$$\gamma_i^\alpha = \gamma_i + \alpha(\gamma_{i+1} - \gamma_i), \qquad (3.100)$$

where $i \in \{1, 2, ..., l-1\}$ such that $\gamma_i \leq \gamma_i^\alpha \leq \gamma_{i+1}$ and $\alpha \in [0, 1]$. This motivates the following lifted (vectorial) representation $\boldsymbol{1}_i^\alpha$ of an originally scalar value $\gamma_i^\alpha \in \Gamma$:

$$\boldsymbol{1}_i^\alpha := \alpha\boldsymbol{1}_i + (1 - \alpha)\boldsymbol{1}_{i-1} \in \mathbb{R}^{l-1}. \qquad (3.101)$$

Here, $\boldsymbol{1}_i \in \mathbb{R}^{l-1}$ is the vector of $i$ ones followed by zeroes[6].

The *non-convex* lifted label space is defined as the set

$$\boldsymbol{\Gamma} := \{\boldsymbol{1}_i^\alpha \in \mathbb{R}^{l-1} | i \in \{1, 2, ..., l\}, \alpha \in [0, 1]\}. \qquad (3.102)$$

Functions $u : \Omega \to \Gamma$ can thus be linked to vectorial functions

6: Consequently, the lifted vectors are of the form

$$\boldsymbol{1}_i^\alpha = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ \alpha \\ 0 \\ \vdots \\ 0 \end{pmatrix} \leftarrow i^{th} \text{ row}$$

$u : \Omega \rightarrow [0,1]^{l-1}$. These vectorial functions are called *(sublabel-) integral*, if $\boldsymbol{u}(x) = \mathbf{1}_i^{\alpha} \in \Gamma$ for (almost) every $x$. These sublabel-integral functions can be mapped to scalar-valued functions $u : \Omega \rightarrow \Gamma$ by using the (pointwise) transformation

$$u(x) = \gamma_1 + \sum_{i=1}^{l-1} \boldsymbol{u}^i(x)(\gamma_{i+1} - \gamma_i). \tag{3.103}$$

Having introduced a lifted representation for the solution space, the energy needs to be reformulated as well. The reformulation can be constructed separately for the data term and for the regularizer.

**Lifting of the Data Term [Möl+15, Chpt. 3.1].** For the – possibly non-convex – data term, the lifted representation is defined pointwise for fixed $x$ as $\rho(x, \cdot) : \mathbb{R}^{l-1} \mapsto \overline{\mathbb{R}}$,

$$\rho(x, \boldsymbol{u}(x)) := \inf_{\substack{i \in \{1, \ldots, l-1\}, \\ \alpha \in [0,1]}} \left\{ \rho(x, \gamma_i^{\alpha}) + \delta_{\mathbf{1}_i^{\alpha}}(\boldsymbol{u}(x)) \right\}. \tag{3.104}$$

With this definition, $\rho(x, \mathbf{1}_i^{\alpha}) = \rho(x, \gamma_i^{\alpha})$, see also Fig. 3.6a–b. Note that this is still a non-convex term, as $\rho$ only assumes finite values on the non-convex set $\Gamma$ and is infinite everywhere else. In order to arrive at a convex expression, the definition of the Fenchel conjugate is applied twice (under the integral) with respect to the second variable. This gives a relaxed and convex integrand which is used to define the following convex data term:

$$H(\boldsymbol{u}) = \int_{\Omega} \rho^{**}(x, \boldsymbol{u}(x)) dx. \tag{3.105}$$

This results in a convex expression in the lifted setting, see Fig. 3.6d, and in a piecewise convex approximation of the original data term in between the chosen labels, see Fig. 3.6c as well as Fig. 3.5c. An explicit expression for $\rho^{**}$ is given in [Möl+15, Prop. 1].

**Lifting of the Total Variation Regularizer [Möl+15, Chpt. 3.2].** Lastly, one can establish a lifted representation of the (isotropic) total variation regularizer, which builds on the theory developed in the context of lifting methods for multiclass labeling approaches [LS11; CCP12]. The lifted – and non-convex – integrand $\eta : \mathbb{R}^{(l-1) \times n} \mapsto \overline{\mathbb{R}}$ is defined as

$$\eta(\boldsymbol{g}) := \inf_{\substack{1 \leq i \leq j \leq l-1, \\ \alpha, \beta \in [0,1] \\ v \in \mathbb{R}^n}} \left\{ |\gamma_i^{\alpha} - \gamma_j^{\beta}| \cdot \|v\|_2 + \delta_{\left(\mathbf{1}_i^{\alpha} - \mathbf{1}_j^{\beta}\right) v^{\top}}(\boldsymbol{g}) \right\}. \tag{3.106}$$

**(a)** Original data term $\rho(x,\cdot)$



**(b)** Lifted data term $\rho(x,\cdot)$



**(c)** Approximation of data term $\rho(x,\cdot)$



**(d)** Lifted data term $\rho^{**}(x,\cdot)$
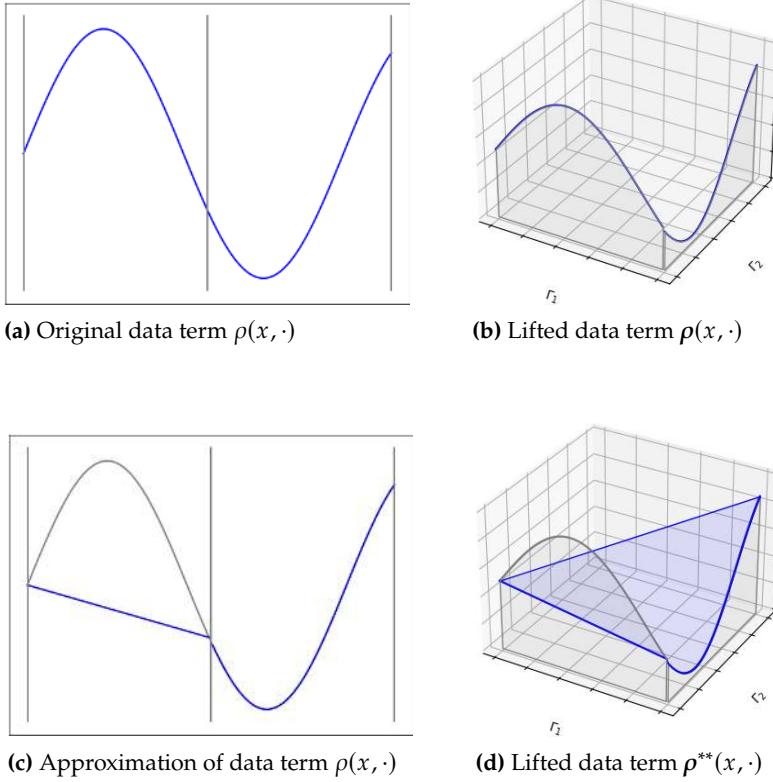
**Figure 3.6:** We consider the integrand of a non-convex data term $\rho(x,\cdot)$ for fixed, arbitrary $x \in \Omega$, dropping the $x$ for simplicity.

In **Fig. (a)**, the original integrand of the data term $\rho : \Gamma \to \mathbb{R}$ is shown in blue. Discretizing the range $\Gamma$ (x-axis) by introducing the three labels indicated by the vertical gray lines leads to the the high-dimensional, yet non-convex, integrand $\rho : \mathbb{R}^2 \to \mathbb{R}$ marked blue in **Fig. (b)**.

By calculating the biconjugate $\rho$ of the lifted data term, we obtain the convex, lifted representation of the integrand $\rho^{**} : \mathbb{R}^2 \to \mathbb{R}$ shown in blue in **Fig. (d)**. Looking at the $\Gamma_1$ and $\Gamma_2$ axis, it is apparent that the original integrand is approximated in a piecewise convex way in between the chosen labels, see also the blue line in **Fig. (c)** and orange dashed line in Fig. 3.5c

Applying the Legendre-Fenchel conjugate twice results in a relaxed – and convex – regularization term:

$$TV(u) := \int_\Omega \eta^{**}(Du), \tag{3.107}$$

where $Du$ is the distributional derivative in the form of a Radon measure. For the isotropic $L^2$-TV, it can be shown that, for $g \in \mathbb{R}^{(l-1)\times n}$ with rows $g_i, i = 1, ..., l-1$,

$$\eta^{**}(g) = \sup_{q \in K_{\text{iso}}} \{\langle q, g \rangle\}, \tag{3.108}$$

$$K_{\text{iso}} = \left\{ q \in \mathbb{R}^{(l-1)\times n} \Big| \|q_i\|_2 \leq \gamma_{i+1} - \gamma_i, \forall i = 1, ..., l-1 \right\}. \tag{3.109}$$

For more details we refer to [Möl+15, Prop. 4] and [CCP12].

While the authors of [Möl+15] only consider the isotropic $L^2$-TV, we here want to additionally consider the *anisotropic $L^1$-TV*, which is defined as $TV_{\text{an}}(u) := \int_\Omega \|Du\|_1$. We define the lifted *anisotropic $L^1$-TV* as:

$$\eta^{**}(g) = \sup_{q \in K_{\text{an}}} \{\langle q, g \rangle\}, \tag{3.110}$$

$$K_{\text{an}} = \bigcap_{j=1,...,n} \left\{ q \Big| \|q_{i,j}\|_2 \leq \gamma_{i+1} - \gamma_i, \forall i = 1, ..., l-1 \right\}. \tag{3.111}$$

The anisotropic $L^1$-TV will be of special interest in Chpt. 4.

*Motivation for the definition of $\boldsymbol{K}_{an}$.* With the same strategy as in the isotropic case (3.109), using the fact that $\| \cdot \|_\infty$ is the dual norm of $\| \cdot \|_1$, one obtains

$$\boldsymbol{K}_{\mathrm{an}} = \left\{ \boldsymbol{q} \in \mathbb{R}^{(l-1)\times n} \,\middle|\, \|\boldsymbol{q}_i\|_\infty \leq \gamma_{i+1} - \gamma_i, \forall i = 1, ..., l-1 \right\}. \tag{3.112}$$

For arbitrary, fixed $\boldsymbol{q} \in \mathbb{R}^{(l-1)\times n}$, it holds for any $1 = 1, ..., l-1$ that

$$\begin{aligned} \|\boldsymbol{q}_i\|_\infty \;\; &\leq \;\; \gamma_{i+1} - \gamma_i, & \text{(3.113)} \\ \Leftrightarrow \;\; \max_{j=1,...,d} |\boldsymbol{q}_{i,j}| \;\; &\leq \;\; \gamma_{i+1} - \gamma_i, & \text{(3.114)} \\ \Leftrightarrow \;\; \|\boldsymbol{q}_{i,j}\|_2 \;\; &\leq \;\; \gamma_{i+1} - \gamma_i, \quad \forall j = 1, ..., n. & \text{(3.115)} \end{aligned}$$

This motivates the definition of $\boldsymbol{K}_{\mathrm{an}}$ in (3.111) $\qquad\qquad\square$

**Lifted Problem.** Putting everything together, the sublabel-accurate lifted formulation of (3.98) is

$$\inf_{u:\Omega\to\mathbb{R}^{l-1}} \left\{ \int_\Omega \rho^{**}(x, \boldsymbol{u}(x))dx + \int_\Omega \eta^{**}(D\boldsymbol{u}) \right\}. \tag{3.116}$$

In [MC17, Prop. 4], it is shown that this formulation is equal to the one obtained by applying the sublabel-accurate discretization approach on the calibration-based lifted energy as described on p. 63.

## 3.4 Optimization

Typically, the lifted problem (3.71)-(3.73) is solved using the *primal-dual hybrid gradient* (PDHG) algorithm [Poc+09; Poc+10; Möl+15]. The PDHG algorithm is applicable to a broad class of variational problems of the following form:

$$\min_{v\in C} \max_{\varphi\in K} \{\langle Av, \varphi \rangle + G(v) - F^*(\varphi)\}, \tag{3.117}$$

where $G$ and $F^*$ are convex functions mapping to the extended real line, $A$ denotes some linear operator, and $C$ and $K$ are closed, convex sets.

*Backward gradient step.* Recall that a classical *gradient descent step* for a differentiable function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is defined as

$$x^{k+1} = x^k - \alpha\nabla f(x^k), \tag{3.118}$$

where $\alpha > 0$ is the *step size.* This can informally be thought of as a forward-Euler discretization of the "gradient flow" PDE

$$\frac{\partial}{\partial t} x = -\nabla f(x). \qquad (3.119)$$

For a convex, non-smooth function $f$, the PDE formally generalizes to

$$\frac{\partial}{\partial t} x \in \partial f(x). \qquad (3.120)$$

Using a backward Euler discretization, the PDE leads to the following update rule, which is called a *backward gradient step*:

$$x^{k+1} \in (I + \alpha \partial f)^{-1}(x^k), \qquad (3.121)$$

where $(I + \alpha \partial f)^{-1}$ denotes the preimage of the set-valued mapping $I + \alpha \partial f$, see also (3.52). If $f$ is proper, lower semicontinuous, and convex, the backward gradient step is single-valued and can be computed as the solution to the *proximal step*:

$$\text{prox}_{\alpha f}(x^k) = \arg \min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} \|x - x^k\|_2^2 + \alpha f(x) \right\}. \qquad (3.122)$$

See also [Eck89, Chpt. 3.2.3].

**PDHG Algorithm.** The idea of the PDHG algorithm is to alternatingly perform a backward gradient step on the primal and dual variables whilst projecting onto the set of feasible points. In its simplest form, the algorithm is described in Alg. 1:

---
**Algorithm 1:** PDHG Algorithm

---
1 **Initialize:**

2 $\quad \tau, \sigma > 0, \quad \theta \in [0,1], \quad (v^0, \varphi^0) \in C \times K, \quad \bar{v}^0 \leftarrow v^0.$

3 **For** $n \geq 0$

4 $\quad \varphi^{n+1} \leftarrow (I + \sigma \partial F^*)^{-1}(\varphi^n + \sigma A \bar{v}^n)$

5 $\quad v^{n+1} \leftarrow (I + \tau \partial G)^{-1}(v^n - \tau A^* \varphi^{n+1})$

6 $\quad \bar{v}^{n+1} \leftarrow v^{n+1} + \theta(v^{n+1} - v^n)$

---

In many imaging applications, the *resolvent operator* $(I + \sigma \partial F)^{-1}$ has a closed form representation (see examples in [CP11]). This is also the case for the CBL problem. In the following we will exemplarily work with the finite-difference discretization approach.

Consider the discretized version of the CBL problem in Def. 3.2.5:

$$\inf_{v \in C_h} \sup_{\varphi \in K_h} \sum_{i,j} \varphi(x_i, t_j)^\top D_h v(x_i, t_j), \qquad (3.123)$$

where $C_h$, $K_h$ denote the discretized version of the constraint sets $\mathscr{C}$, $\mathscr{K}$, and $D_h$ denotes the discretized version of the distributional derivative. Furthermore, we denote the adjoint of $D_h$ by $D_h^*$ and the projection onto a closed, convex set $C$ by $\mathrm{Proj}_C$. As $F^* \equiv \delta_{C_h}$, $G \equiv \delta_{K_h}$, and $A = D_h$, the PDHG algorithm boils down to Alg. 2:

---

**Algorithm 2:** PDHG Algorithm for CBL

---

1 **Initialize:**
2   $\tau, \sigma > 0, \quad \theta \in [0,1], \quad (v^0, \varphi^0) \in C_h \times K_h, \quad \bar{v}^0 \leftarrow v^0.$
3 **For** $n \geq 0$
4     $\varphi^{n+1} \leftarrow \mathrm{Proj}_{K_h}(\varphi^n + \sigma D_h \bar{v}^n)$
5     $v^{n+1} \leftarrow \mathrm{Proj}_{C_h}(v^n - \tau D_h^* \varphi^{n+1})$
6     $\bar{v}^{n+1} \leftarrow v^{n+1} + \theta(v^{n+1} - v^n)$

---

In case of the finite-difference discretization approach, computing the projections onto $C_h$ and $K_h$ is straightforward and has an explicit form: The projection onto $C_h$ is achieved by pointwise clamping

$$\mathrm{Proj}_{C_h}(v) = \min\{1, \max\{0, v\}\}. \tag{3.124}$$

For the TV regularizer, the projection onto $K_h$ can, furthermore, be written as

$$\mathrm{Proj}_{K_h}((\varphi_x, \varphi_t)) = \left( \frac{\varphi_x}{\max\{1, \|\varphi_x\|\}}, \max\{-\rho, \varphi_t\} \right). \tag{3.125}$$

Together with Alg. 2, these projections allow to construct a practical algorithm for numerically minimizing the lifted dual problem in Def. 3.2.5. In this work, we use the implementation from [LM15].

**Related Methods and Results.** Theory concerning the PDHG algorithm is only of minor importance in this work. Therefore, we only mention some papers for the interested reader: The PDHG algorithm as we use it was first introduced by [CP11]. Preliminary primal-dual methods for the $L^2 - \mathrm{TV}$ denoising and Mumford-Shah problem can be found in [ZC08; Poc+09]. Connections to other algorithms such as *ADDMM*, *Douglas-Rachford* or *Split Inexact Uzawa* are established in [EZC10; CP11]. Convergence results for the vanilla PDHG algorithm can be found in [CP11], however faster convergence is possible using *preconditioning* [PC11], *adaptive step sizes* [Gol+13], and a good choice of proximal parameters [FB18]. Also worth mentioning is the *stochastic primal-dual hybrid gradient* (SPDHG) algorithm, in which the dual variable is only partially updated in each step [Cha+18; GDE21; AFC22], and an extension to nonlinear operators $A$ [Val14].

## 3.5 Conclusion and Outlook

In this chapter, we have introduced the calibration-based lifting approach building on the general line of argument in [Poc+10]. By carefully considering the subgraph of $BV(\Omega; \Gamma)$-functions as well as related measure-theoretic objects and by recalling central results from [Mas79] and [Vog20], we extended the lifting and duality result in [Poc+10, Thm. 3.2] from the $W^{1,1}(\Omega; \Gamma)$ to the $BV(\Omega; \Gamma)$ setting. Furthermore, we proved that the important thresholding theorem [Poc+10, Thm. 3.1] also carries over to the $BV(\Omega; \Gamma)$ setting. Additionally, we have discussed possible discretization and optimization approaches.

These results are the foundation and inspiration for the following two contributions in this work: Firstly, in Chpt. 4, we will use the lifted dual formulation (Def. 3.2.5) in order to extend the notion of inverse scale space iterations. The resulting generalized inverse scale space iterations are studied in a continuous and sublabel-accurate discretized (Sec. 3.3.3) setting. Secondly, in Chpt. 5, we introduce a neural fields-based learning approach as an alternative to the classical optimization approach using the PDHG algorithm as discussed in Sec. 3.4.

<h1 align="center">Lifted Bregman Iteration | 4</h1>

In this chapter, we propose *generalized inverse scale space iterations*, shortly referred to as *lifted Bregman iterations* (LBI), for variational problems of the form (see also sidenote on p.42)

$$\inf_{u \in BV(\Omega;\Gamma)} F(u), \quad F(u) := \int_\Omega f(x, u(x), \mathrm{d}Du). \qquad (4.1)$$

We assume that $\Omega \subset \mathbb{R}^n$ is open and bounded, $\Gamma \subset \mathbb{R}$ is scalar and compact with $\Gamma = [\Gamma_{\min}, \Gamma_{\max}]$, $f$ fulfills the assumptions (A1)-(A4) introduced on p. 42 and is in addition

(B1) *separable* in the sense that $f : \Omega \times \mathbb{R} \times \mathbb{R}^n$ can be written with the help of $\rho : \Omega \times \mathbb{R} \to \mathbb{R}$ and $\eta : \mathbb{R}^n \to \mathbb{R}$ as

$$f(x, t, p) = \rho(x, t) + \eta(p), \qquad (4.2)$$

(B2) *positively* 1-*homogeneous* in the last argument, i.e.,

$$\eta(\lambda p) = \lambda \eta(p) \quad \forall p \in \mathbb{R}^n, \lambda > 0. \qquad (4.3)$$

Assumption (B2) is necessary, as the existing theory of non-linear inverse scale spaces and weak spectral representation [Bur+15; Bur+16] is formulated in the context of such regularizers and our approach will build on this existing theory. In combination with assumption (B1), we can employ a special form of the calibration-based lifting (see p. 56).

While assumptions (A1)-(A4) and (B1)-(B2) limit the choice of admissible regularizers, they allow the use of the total variation regularizer, which plays a prominent role in mathematical imaging. In the following, we denote by $H : BV(\Omega; Gamma) \to \mathbb{R}$ the *data term*

$$H(u) = \int_\Omega \rho(x, u(x)) \, \mathrm{d}x, \qquad (4.4)$$

and use throughout our explanations and experiments the *total variation* regularizer $TV : BV(\Omega) \to \mathbb{R}$, i.e.,

$$TV(u) = \int_\Omega \eta(Du) = \int_\Omega \mathrm{d}|Du|. \qquad (4.5)$$

## 4.1 Motivation and Outline

In mathematical image computing, some of the most common tasks evolve around analyzing, manipulating, enhancing, or reducing

features of a certain scale. In such approaches, a fundamental building block is to decompose a given image into components of different scale.

Our work builds on the recent idea of *weak data-dependent spectral frequency representation* [Bur+16, Def. 1.4], where an image $f \in B$, with $B$ denoting some Banach space, is decomposed into a set of non-linear eigenfunctions of a convex, absolutely 1-homogeneous regularizer $J$. It is based on a generalization of the *spectral theorem* for linear operators [Hal13, Thm. 7.12]. For a general introduction to the topic we refer the reader to Sec. 2.3; here we only recall the central motivation and equations.

The weak data-dependent spectral frequency representation is a vector-valued distribution $\mu_J^f$ on $B$, such that in the special case where $f$ is a nonlinear eigenfunction of $J$ with eigenvalue $\lambda$, i.e., $\lambda f \in \partial J(f)$ with $\|f\| = 1$, it holds

$$\mu_J^f(s) = f\delta(s - \lambda), \quad s \in \mathbb{R}_0^+. \tag{4.6}$$

In the general case $f \in B$ it holds

$$f = \int_0^\infty \mu_J^f(s)\,\mathrm{d}s. \tag{4.7}$$

Features in the input image $f$ can be enhanced, reduced or even removed by constructing a non-linear filter $\tilde{w}$ from a bounded, measurable map $w : \mathbb{R}^+ \to \mathbb{R}$ to the image $f$ via the definition

$$\tilde{w}(f) = \int_0^\infty w(s)\mu_J^f(s)\,\mathrm{d}s. \tag{4.8}$$

In this context, details are eigenfunctions of the regularizer $J$ and the notion of size correlates with the eigenvalues.

The analysis of non-linear eigenfunctions of the here considered class of convex, absolutely 1-homogeneous regularizers has mainly focused on the TV regularizer. In the 2D scenario, the authors of [BCN02, Thm. 4] have shown that the indicator function of a bounded set $C \subset \mathbb{R}^2$ with finite perimeter is a nonlinear eigenfunction of the TV regularizer if and only if considered set $C$ is convex, its boundary $\partial C$ is of the class $C^{1,1}$ and the essential supremum of the *curvature* $\kappa$ is bounded by the ratio of the *perimeter P* and *area A* of the set:

$$\operatorname*{ess\,sup}_{q \in \partial C} \leq \frac{P(C)}{A(C)}. \tag{4.9}$$

Due to this, the filter (4.16) has the potential to gracefully cope with sharp edges, in contrast to – for example – classical filtering by Fourier coefficients.

There are multiple, in certain cases equivalent, methods of deriving such a representation $\mu_J^f$: The variational method (VM), the gradient flow (GF) and the inverse scale space (ISS) flow discussed in 1.2.2. In the remainder of the chapter, we discuss the ISS method [Bur+06; Bur+15; Bur+16; GMB16].

**Inverse Scale Space Flow and Weak Data-Dependent Spectral Frequency Representation.** We denote by $f \in L^2(\Omega; \mathbb{R}^k)$ an image and by $J : U \to \mathbb{R}$ a convex, 1-homogeneous functional, where $U$ is a suitable Banach space embedded into $L^2(\Omega; \mathbb{R}^k)$. Recall that the *inverse scale space flow* (ISS) is defined as

$$\frac{\partial}{\partial s} p(s) = f - \tilde{u}(s), \tag{4.10}$$

$$p(s) \in \partial J(\tilde{u}(s)), \tag{4.11}$$

$$p(0) = 0, \tag{4.12}$$

where $\tilde{u}, p : \mathbb{R}_0^+ \to U$, $p$ is differentiable, and $\partial J$ is the subdifferential of $J$. For nonlinear eigenfunctions $f$ with eigenvalue $\lambda$ of $J$, it holds [Bur+16, p. 1388]

$$\tilde{u}(s) = \begin{cases} 0, & \text{if } s \leq \lambda, \\ f, & \text{else.} \end{cases} \tag{4.13}$$

Consequently, nonlinear eigenfunctions $f$ of $J$ appear as (Dirac) delta peaks in the distributional derivative $\frac{\partial}{\partial s} \tilde{u}(\cdot, s) = f(\cdot) \delta_\lambda(s)$. This motivates

$$\mu_J^f(s) := \frac{\partial}{\partial s} \tilde{u}(\cdot, s) \tag{4.14}$$

as a definition of the weak data-dependent spectral frequency representation. In [Bur+16], the authors show that any image $f \in L^2(\Omega; \mathbb{R}^k)$ is perfectly reconstructed by

$$f = \int_0^\infty \mu_J^f(s) \, ds \tag{4.15}$$

Furthermore, features of a certain scale can be enhanced or removed by applying some bounded, measurable filter function $w : \mathbb{R}^+ \to \mathbb{R}$ via

$$w(f) = \int_0^\infty w(s) \mu_J^f(s) \, ds. \tag{4.16}$$

**Bregman Iteration.** The inverse scale space flow is closely linked to the *Bregman iteration* which was originally introduced as an iterative regularization method [Osh+05] for variational problems

of the form

$$\min_{u \in U} \{H(u; f) + J(u)\}. \tag{4.17}$$

Here, $U$ is an appropriate solution space, $f$ some input data, and both $H$ and $J$ are assumed to be non-negative and convex with respect to $u$. The Bregman iteration is defined as:

---
**Algorithm 3:** Bregman Iteration (BI)

---
1 Initialize $p_0 = 0$

2 Repeat for $k = 1, 2, \dots$

3      $u_k \in \arg\min_{u \in U} \{H(u; f) + J(u) - \langle p_{k-1}, u \rangle\}$

4      $p_k \in \partial J(u_k)$

---

Let us now consider the special case of the ROF denoising problem [ROF92], i.e., $f \in L^2(\Omega; \Gamma)$, $U = \mathrm{BV}(\Omega; \Gamma)$, $H(u; f) = \frac{\lambda}{2}\|f - u\|_{L^2}^2$ and $J(u) = \mathrm{TV}(u)$. While the choice $p_k \in \partial J(u_k)$ is not necessarily unique as the total variation is a non-smooth functional, the Bregman iteration can be made deterministic by the observation that the choice $p_k = p_{k-1} + \lambda(f - u_k) \in \partial J(u_k)$ is possible [Osh+05]. This leads to the following modified iteration:

---
**Algorithm 4:** Bregman Iteration – ROF

---
1 Initialize $p_0 = 0$

2 Repeat for $k = 1, 2, \dots$

3      $u_k \in \arg\min_{u \in \mathrm{BV}(\Omega; \Gamma)} \left\{ \frac{\lambda}{2}\|f - u\|_{L^2}^2 + \mathrm{TV}(u) - \langle p_{k-1}, u \rangle \right\}$

4      $p_k = p_{k-1} + \lambda(f - u_k)$

---

The optimality condition for the third line is

$$0 \in -\lambda(f - u_k) + \partial \mathrm{TV}(u_k) - p_{k-1}. \tag{4.18}$$

With the special subgradient $p_k := p_{k-1} + \lambda(f - u_k) \in \partial \mathrm{TV}(u_k)$, it holds

$$0 = -\lambda(f - u_k) + p_k - p_{k-1}. \tag{4.19}$$

This can be rewritten to

$$f - u_k = \frac{1}{\lambda}(p_k - p_{k-1}) \tag{4.20}$$

Interpreting $\lambda$ small enough as the "time" passed between iteration $k - 1$ and $k$, the right-hand-side resembles a finite backward difference. Informally replacing the discrete steps of the Bregman iteration with a continuous time variable $s \in [0, T)$, we arrive at

$$f - u(s) = \frac{\partial}{\partial s}p(s), \quad p(s) \in \partial J(u(s)), \quad p(0) = 0, \tag{4.21}$$

which is simply the inverse scale space flow equation [Bur+16].

**Summary.** To summarize, for the ROF problem, the Bregman iteration with specific choice of subgradient leads in the continuous limit to the inverse scale space flow. Due to (4.13), the latter allows to define a weak spectral representation (4.14) of the input image. This weak spectral representation is built around nonlinear eigenfunctions of the convex, absolutely 1-homogeneous regularizer. Notably, for the 2D total variation regularizer, indicator functions of certain bounded, convex sets with finite perimeter are in fact eigenfunctions [BCN02, Thm. 4]. Consequently, the application of filter functions as in (4.16) handles edges well.

**Open Questions.** So far, attempts at constructing inverse scale space flows, weak spectral representations, and related filters have been limited to $L^2$ data terms. However, many interesting variational problems in image analysis, such as depth estimation, have inherently non-convex data terms. It is therefore an interesting challenge to extend these tools to more general problems by applying a version of the Bregman iteration to problems (4.1) with arbitrary, possibly non-convex data term:

- ▶ Does the Bregman iteration generate an inverse scale space iteration of solutions for arbitrary, possibly non-convex data terms?
- ▶ Can we define a non-linear spectral decomposition of these solutions, e.g., a spectral decomposition of the depth map minimizing a stereo matching problem?

**Lifted Bregman Iteration.** In this chapter, we take a step towards answering these questions. The core idea is to combine the calibration-based lifting approach with the Bregman iteration, which results in a *lifted Bregman iteration* that can be used for variational problems with non-convex data term.

Accordingly, we first embed the original problem

$$\inf_{u \in U} F(u), \quad F(u) := H(u) + J(u) \tag{4.22}$$

using the theory of calibration-based lifting (Chpt. 3). Later, we will consider the continuous as well as the discrete setting; For now we write the new, convex problem as

$$\inf_{v \in \mathscr{C}} \mathscr{F}(v), \quad \mathscr{F}(v) := \mathscr{H}(v) + \mathscr{J}(v). \tag{4.23}$$

While conceptually the energy $F$ is lifted, the convexity and positive 1-homogeneity of the regularizer allow to consider the lifted data term $\mathscr{H}$ and lifted regularizer $\mathscr{J}$ separately (see also (3.71) – (3.73)). The lifted Bregman iteration is then defined as

---

**Algorithm 5:** Lifted Bregman Iteration (LBI)

---

1   Initialize $\tilde{p}_0 = 0$

2   Repeat for $k = 1, 2, \ldots$

3      $v_k \in \arg\min_{v \in \mathscr{C}} \{\mathscr{H}(v) + \mathscr{J}(v) - \langle \tilde{p}_{k-1}, v \rangle\}$

4      $\tilde{p}_k \in \partial \mathscr{J}(v_k)$

---

Importantly, each step of the lifted Bregman iteration relies on the choice of a subgradient of the *lifted* regularizer $\mathscr{J}$.

While the iteration is straightforward to define, it is not clear if the iterates have similar scale-space properties as in the original iteration, and if there is a relation to the iterates of the original, unlifted Bregman iteration (Alg. 3) in the convex ROF case.

The main theoretical contribution in this chapter is to answer the question if and under which circumstances the original and lifted Bregman iteration produce "equivalent" results. This requires a discussion of the subdifferential of the original and lifted total variation. Equivalence when applied to the ROF denoising problem can be understood as a plausibility check for the new method.

**Equivalence.**   The exact definition of "equivalence" is not trivial, since there is potential ambiguity in choosing the subgradient in the lifted setting. We say that Alg. 4 and Alg. 5 are equivalent for the ROF problem, if we can find a suitable subgradient $\tilde{p}_{k-1}$ in the lifted setting that relates to a subgradient $p_{k-1}$ in the original setting such that some $u_k$ solves the original Bregman iteration and the indicator of its subgraph $\mathbf{1}_{u_k}$ solves the lifted Bregman iteration. This means that under a specific choice of subgradients, both iterations produce the same sequence of images, one in the original, and one in the lifted representation.

**Contribution and Outline.**   In the following three sections, we discuss the equivalence of the original (Alg. 4) and lifted Bregman iteration (Alg. 5) for the ROF denoising problem in a fully continuous setting $\Omega \times \Gamma$ (Sec. 4.2) and with sublabel-accurate discretization (Sec. 4.3–Sec. 4.4). We formulate a sufficient condition on the subgradients used in Alg. 4 and Alg. 5 such that the latter are equivalent according to our definition. In Sec. 4.5, we verify these findings numerically by showing that Alg. 4 and Alg. 5 produce the same iterates when applied to the convex ROF denoising problem. In addition, we show numerical results of Alg. 5 for a non-convex stereo matching problem, which suggest that the inverse scale space behavior of the Bregman iteration might carry over to the non-convex settings.

Bregman iteration for **convex** problems.  Lifted Bregman iteration extends to **non-convex** data terms.

**Variational Problem**

$$\inf_{u \in U} F(u), \quad F(u) := H(u) + J(u),$$

Depending on the data term, the problem might be non-convex. The regularizer is non-negative, positively 1-homogeneous, convex.

Embedding

**Equivalent!**

**Lifted Variational Problem**

$$\inf_{v \in C} \mathcal{F}(v), \quad \mathcal{F}(v) := \mathcal{H}(v) + \mathcal{J}(v).$$

This is a *convex* problem. Minimizers of this higher-dimensional problem relate to minimizers of the original problem.

Bregman iteration for convex problem

Bregman iteration for possibly non-convex problem

**Bregman Iteration**
Set $\tilde{p}_0 = 0$ and repeat for $k = 1, \dots$

$$u_k \in \arg \min_u \{H(u) + J(u) - \langle p_{k-1}, u \rangle\},$$

$$p_k \in \partial J(u_k).$$

Subgradients $p_k$ are chosen in the subdifferential of the regularizer.

Choose $p_{k-1}$. Use lifted version as $\tilde{p}_{k-1}$

Backproject $v_k$

**Equivalent!**

**Equivalent?**

**Lifted Bregman Iteration**
Set $\tilde{p}_0 = 0$ and repeat for $k = 1, \dots$

$$v_k \in \arg \min_v \{\mathcal{H}(v) + \mathcal{J}(v) - \langle \tilde{p}_{k-1}, v \rangle\},$$

$$\tilde{p}_k \in \partial \mathcal{J}(v_k).$$

Subgradients $\tilde{p}_k$ are chosen in the subdifferential of the lifted regularizer.

Undo embedding

**Projected Solution**
Project solutions onto original space.

Trick for proving sufficient condition: For convex problems we can take subgradients from BI and use them in LBI **instead** of choosing subgradients in the lifted setting. We call the so-defined iteration LBI*.

**Figure 4.1: Overview.** In this chapter we will discuss three different variations of the Bregman iteration. In our terminology, the classical *Bregman Iteration* (BI) is applied to the original variational problem. The proposed *Lifted Bregman Iteration* (LBI) is applied to the calibration-based lifted problem. In particular, single steps of the LBI rely on subgradients of the *lifted* regularizer. For certain results we, furthermore, define a third variation of the Bregman iteration: We perform the calibration-based lifting on single steps of the BI for a *given* subgradient of the *original* regularizer. We call the later approach *Bregman Iteration, Lifted* (LBI*).

## 4.2 Equivalence in the Continuous Setting

In this section, we consider the ROF denoising problem and the calibration-based lifting approach in the function space with continuous range $\Gamma$.

As the Bregman iteration crucially depends on subgradients of the regularizer, we first discuss the subdifferential $\partial \mathrm{TV}$ of the total variation and the exact definition of the calibration-based lifted total variation $\mathscr{TV}$. Before moving on, we introduce a second variation on the Bregman iteration – the LBI*, which relies on (lifted) subgradients of the original regularizer. With the help of the LBI* we introduce and prove a sufficient condition on the subgradients $p_k$ and $\tilde{p}_{k-1}$ under which algorithms Alg. 4 and Alg. 5 are equivalent. We furthermore prove the existence of subgradients $p_k \in \partial \mathrm{TV}(u_k)$ and $\tilde{p}_{k-1} \in \partial \mathscr{TV}(\mathbf{1}_{u_k})$ which fulfil this sufficient condition.

### 4.2.1 Subdifferential of the Total Variation

The Bregman iteration crucially requires elements from the subdifferential $\partial J$ of the regularizer. Unfortunately, for the choice $J = \mathrm{TV}$, this requires to study elements from the dual space BV* which is not yet fully understood [AFP00, Remark 3.12] [Anz83; BH16]. In this section, we follow the approach of [BH16] and argue under a restriction to $L^p$ functions with $p > 1$.

We define the following simplifying assumption:

(C1) We restrict ourselves to the case $u \in W^{1,1}$ and $\Omega \subseteq \mathbb{R}^2$, which allows to embed $W^{1,1} \hookrightarrow L^2$ by the Sobolev embedding theorem [Alt92, Thm. 10.9]; consequently we can view the total variation as an operator $TV : L^2(\Omega) \to \mathbb{R}$. From this narrow viewpoint, the subgradients of TV are also $L^2$ functions, which – in general – is not true.

The assumptions allow us to formulate the major arguments in a function space setting in an intuitive way that also transfers to the discretized problem. At this point we do not know how to fully generalize the argument due to the difficulties associated with characterizing the dual space of BV and subdifferential of TV.

The total variation can be viewed – and is often defined – as a support function [AFP00, Def. 3.4]:

$$\mathrm{TV}(u) = \sigma_\Psi(u), \tag{4.24}$$

$$\Psi := \{-\operatorname{div} \psi \,|\, \psi \in C_c(\Omega; \mathbb{R}^n), \|\psi\|_\infty \le 1\}. \tag{4.25}$$

Using the relation $\sigma_C^* = \delta_{\mathrm{cl}\,C}$ for convex sets $C$, its Fenchel conjugate is [ET99, Def. I.4.1, Example I.4.3]

$$\mathrm{TV}^*(p) = \sigma_\Psi^*(p) = \delta_{\mathrm{cl}\,\Psi}(p). \qquad (4.26)$$

According to [ET99, Prop. I.5.1], $p \in \partial\,\mathrm{TV}(u)$ if and only if

$$\mathrm{TV}(u) = \langle u, p \rangle - \delta_{\mathrm{cl}\,\Psi}(p). \qquad (4.27)$$

The exact definition of $\mathrm{cl}\,\Psi$ requires to define the topology with respect to which the closure is to be understood. Under assumption (C1), the closure of $\Psi$ with respect to the $L^2$ norm is [BH16, proof of Prop. 7]

$$\mathrm{cl}\,\Psi = \left\{ -\operatorname{div}\psi \mid \psi \in W_0^2(\operatorname{div};\Omega),\ \|\psi\|_\infty \le 1 \right\}, \qquad (4.28)$$

where

$$W^2(\operatorname{div};\Omega) := \left\{ \psi \in L^2(\Omega, \mathbb{R}^n) \mid \operatorname{div}\psi \in L^2(\Omega) \right\}, \qquad (4.29)$$

$$\|\psi\|_{W^2(\operatorname{div})} := \|\psi\|_{L^2}^2 + \|\operatorname{div}\psi\|_{L^2}^2, \qquad (4.30)$$

$$W_0^2(\operatorname{div};\Omega) := \overline{C_c^\infty(\Omega;\mathbb{R}^n)}^{\|\cdot\|_{W^2(\operatorname{div})}}. \qquad (4.31)$$

and where we say that $\operatorname{div}\psi \in L^2(\Omega)$ if there exists a $w \in L^2(\Omega)$ such that

$$\int_\Omega \nabla v(x) \cdot \psi(x)\,\mathrm{d}x = -\int_\Omega v(x) w(x)\,\mathrm{d}x, \quad \forall v \in C_c^\infty. \qquad (4.32)$$

It follows that under assumption (C1), $u^* \in L^2$ is a subgradient of TV if and only if $u^* = -\operatorname{div}\psi$ for some $\psi \in W_0^2(\operatorname{div};\Omega)$ which satisfies $\|\psi\|_\infty \le 1$. Furthermore, it holds [BH16, Prop. 7]

$$\mathrm{TV}(u) = \langle u, -\operatorname{div}\psi \rangle_{L^2}. \qquad (4.33)$$

### 4.2.2 Lifted Total Variation

In Chpt. 3 we have introduced the calibration-based lifting approach: An energy $F$ as in (4.1) can be reformulated as a convex energy

$$\mathcal{F}_d(v) := \sup_{\substack{\varphi_x \in \mathcal{K}_x \\ \varphi_t \in \mathcal{K}_t}} \left\{ \int_{\Omega \times \mathbb{R}} \left\langle \begin{pmatrix} \varphi_x \\ \varphi_t \end{pmatrix}, \mathrm{d}Dv \right\rangle \right\}, \qquad (4.34)$$

$$\mathcal{K}_x := \{ \varphi_x \in C_0(\Omega \times \mathbb{R};\mathbb{R}^n) : \varphi_x(x,t) \in \operatorname{dom}\eta^*,\ \forall(x,t) \}, \qquad (4.35)$$

$$\mathcal{K}_t := \{ \varphi_t \in C_0(\Omega \times \mathbb{R};\mathbb{R}) : -\varphi_t(x,t) \le \rho(x,t),\ \forall(x,t) \}, \qquad (4.36)$$

which is minimized over

$$\mathscr{C} = \{v \in \mathrm{BV}(\Omega \times \mathbb{R}; [0,1]) : \tag{4.37}$$

$$v(x,t) = 1, \ \text{a.e. } (x,t) \in \Omega \times \mathbb{R} \text{ where } t \leq \Gamma_{\min}, \tag{4.38}$$

$$v(x,t) = 0, \ \text{a.e. } (x,t) \in \Omega \times \mathbb{R} \text{ where } t > \Gamma_{\max}\}. \tag{4.39}$$

In this chapter, we are mainly interested in the lifted representation of the total variation regularizer, which we denote $\mathscr{TV}$. For $H \equiv 0$ and $J = \mathrm{TV}$, it turns out that $\varphi_t \equiv 0$ is optimal in (4.34)-(4.36): $\varphi_t$ is non-negative due to (4.36) and $(Dv)_t$ is non-positive on all Borel subsets of $\Omega \times \mathbb{R}$ due to the primal-dual structure of the problem and the boundary constraints (4.38)–(4.39) on $v$. Consequently, we can reduce the lifted total variation to

$$\mathscr{TV}(v) = \sup_{\substack{\varphi_x \in C_0(\Omega \times \mathbb{R}; \mathbb{R}^n) \\ \|\varphi_x\|_\infty \leq 1}} \left\{ \int_{\Omega \times \mathbb{R}} \langle \varphi_x, \, \mathrm{d}(Dv)_x \rangle \right\}. \tag{4.40}$$

Following the duality result in Thm. 3.2.6 and Lem. 3.2.2, the following equality holds for any $u \in \mathrm{BV}(\Omega; \Gamma)$ and the indicator function of its subgraph $\mathbf{1}_u \in \mathrm{BV}(\Omega \times \mathbb{R}; \{0,1\})$:

$$\mathscr{TV}(\mathbf{1}_u) = \mathrm{TV}(u). \tag{4.41}$$

Having discussed the subdifferential of the original total variation and the exact form of the lifted total variation, we now move to the main contribution in which we compare the subdifferential of the former and the latter.

### 4.2.3 Bregman Iteration, Lifted

The lifted Bregman iteration Alg. 5 is based on a "lift-first" strategy: The energy is first embedded using calibration-based lifting, after which the Bregman iteration is applied to the CBL problem. The required subgradients are chosen in the subdifferential of the lifted regularizer. In order to motivate our findings in the subsequent sections, we here introduce a third variation on the Bregman iteration. We apply the theory of calibration-based lifting on single steps of the original Bregman iteration *after* choosing a subgradient in the subdifferential of the original regularizer; Alg. 6 is based on a "Bregman-first" strategy.

We define an augmented data term

$$\tilde{H}(u;p) := H(u) - \langle p, u \rangle \tag{4.42}$$

and consider the energy $F_{\mathrm{Breg}}(u) = \tilde{H}(u;p) + \mathrm{TV}(u)$. Writing the

calibration-based lifted energy as

$$\mathscr{F}_{\text{Breg}}(v) = \tilde{\mathscr{H}}(v; p_{k-1}) + \mathscr{T}\mathscr{V}(v), \tag{4.43}$$

we define the following iteration:

---

**Algorithm 6:** Bregman Iteration, Lifted (LBI*)

---

1   Initialize $p_0 = 0$

2   Repeat for $k = 1, 2, \dots$

3      $v_k \in \arg\min_{v \in \mathscr{C}} \left\{ \tilde{\mathscr{H}}(v; p_{k-1}) + \mathscr{T}\mathscr{V}(v) \right\}$

4      Select a threshold $s \in [0, 1]$ and define $u_k \in \mathrm{BV}(\Omega; \Gamma)$ as

$$u_k := \Gamma_{\min} + \int_{\Gamma_{\min}}^{\Gamma_{\max}} \mathbf{1}_{\{v_k > s\}}(\cdot, t)\, dt. \tag{4.44}$$

5      Choose $p_k \in \partial\,\mathrm{TV}(u_k)$

---

As we assume that $F_{\text{Breg}}$ fulfills the assumptions (A1)-(A4) on p. 42 as well as (B1)-(B2) on p. 73, it follows directly from the thresholding Thm. 3.2.4 that for any

$$v_k \in \arg\min_{v \in \mathscr{C}} \left\{ \tilde{\mathscr{H}}(v; p_{k-1}) + \mathscr{T}\mathscr{V}(v) \right\} \tag{4.45}$$

it holds that $\mathbf{1}_{\{v_k > s\}} \in \mathrm{BV}(\Omega \times \Gamma; \{0, 1\})$ is also a global minimizer of (4.45) for $\mathscr{L}^1$-a.e. $s \in [0, 1]$. Furthermore, there is a $u_k \in \mathrm{BV}(\Omega; \Gamma)$ such that $\mathbf{1}_{\{v_k > s\}} = \mathbf{1}_{u_k}$ and that

$$u_k \in \arg\min_{u \in \mathrm{BV}(\Omega; \Gamma)} \left\{ \tilde{H}(u; p_{k-1}) + \mathrm{TV}(u) \right\}. \tag{4.46}$$

This directly implies that in the special case of the ROF problem, Alg. 4 and Alg. 6 are equivalent by our terminology.

In order to make the bridge to the equivalence between Alg. 4 and Alg. 5, we analyze how the lifting step operates on $F_{\text{Breg}}$, paying close attention to the *given* subgradient $p_{k-1} \in \partial\,\mathrm{TV}(u_{k-1})$. In preparation for the next section, let us introduce $\mathscr{F}_{\text{Breg}}(v)$ explicitly as

$$\mathscr{F}_{\text{Breg}}(v) = \sup_{\substack{\varphi_x \in \mathscr{K}_x \\ \varphi_t \in \mathscr{K}_t}} \left\{ \int_{\Omega \times \mathbb{R}} \left\langle \begin{pmatrix} \varphi_x \\ \varphi_t \end{pmatrix}, \, dDv \right\rangle \right\}, \tag{4.47}$$

$$\mathscr{K}_x := \{ \varphi_x \in C_0(\Omega \times \mathbb{R}; \mathbb{R}^n) : \varphi_x(x, t) \in \mathrm{dom}\{\eta^*\}, \forall(x, t) \}, \tag{4.48}$$

$$\mathscr{K}_t := \{ \varphi_t \in C_0(\Omega \times \mathbb{R}; \mathbb{R}) : -\varphi_t(x, t) + t p_{k-1}(x) \le \rho(x, t),$$

$$\forall(x, t) \in \Omega \times \mathbb{R} \}. \tag{4.49}$$

The term $tp_{k-1}(x)$ in (4.49) comes from the linear Bregman term $\langle p_{k-1}, u \rangle$. Now, we substitute $\tilde{\varphi}_t(x,t) := \varphi_t(x,t) - tp_{k-1}(x)$ in (4.47)–(4.49) and rewrite the problem as

$$
\mathcal{F}_{\text{Breg}}(v) := \sup_{\substack{\varphi_x \in \mathcal{K}_x \\ \tilde{\varphi}_t \in \tilde{\mathcal{K}}_t}} \left\{ \int_{\Omega \times \mathbb{R}} \left\langle \begin{pmatrix} \varphi_x \\ \tilde{\varphi}_t \end{pmatrix}, \mathrm{d}Dv \right\rangle \right.
$$
$$
\left. - \int_\Omega \int_\mathbb{R} \langle p_{k-1}(x), v(x,t) \rangle \, \mathrm{d}t \, \mathrm{d}x \right\},
\tag{4.50}
$$

$$
\tilde{\mathcal{K}}_t := \{ \tilde{\varphi}_t \in C_0(\Omega \times \mathbb{R}; \mathbb{R}) :
$$
$$
- \tilde{\varphi}_t(x,t) \le \rho(x,t), \forall (x,t) \in \Omega \times \mathbb{R} \}.
\tag{4.51}
$$

This shows that $\mathcal{F}_{\text{Breg}}$ is the subtraction of $\mathcal{F}$ (lifted version of the original energy $F$) and the linear term $\int_\Omega \int_\mathbb{R} \langle p_{k-1}(x), v(x,t) \rangle \, \mathrm{d}t \, \mathrm{d}x$ (reminiscent of the linear Bregman term). This opens the question, whether for a *given* subgradient $p_{k-1} \in \partial \mathrm{TV}(u_{k-1})$ its *lifted representation* $\tilde{p}_{k-1}(x, \cdot) \equiv p_{k-1}(x)$ is a valid subgradient in the lifted setting, that is, whether $\tilde{p}_{k-1} \in \partial \mathcal{TV}(\mathbf{1}_{u_{k-1}})$ holds. If we can find a suitable subgradient in the lifted setting that relates to a subgradient in the original setting we can show that the Bregman iteration Alg. 4 and the lifted Bregman iteration Alg. 5 are equivalent.

### 4.2.4 Sufficient Condition for Equivalence

The following proposition shows that the Bregman iteration (Alg. 3) and the fully continuous formulation of the lifted Bregman iteration (Alg. 5) are equivalent as long as the subgradients used in either setting fulfil a certain condition. We have to assume uniqueness of the minimizer for the original problem, as is the case for strictly convex functionals such as ROF.

**Proposition 4.2.1** (Sufficient condition) *Assume that the energy* $F(u) = H(u) + \mathrm{TV}(u)$ *fulfills the conditions (A1)-(A4) on p. 42 and the conditions (B1)-(B2) on p. 73, and that the problems*

$$
u_k \in \arg \min_{u \in \mathrm{BV}(\Omega;\Gamma)} \{H(u) + \mathrm{TV}(u) - \langle p_{k-1}, u \rangle\}
\tag{4.52}
$$

1: Integral minimizers are minimizers $v_k \in \mathrm{BV}(\Omega \times \mathbb{R}; \{0,1\}) \subset \mathcal{C}$. By the thresholding Thm. 3.2.4, such a minimizer always exists as long as the original problem has at least one global minimizer. Furthermore, there exists some $u \in \mathrm{BV}(\Omega;\Gamma)$ such that $v_k = \mathbf{1}_u$.

*of the Bregman iteration in Alg. 3 have unique solutions* $u_k$. *For* $k = 1, 2, \ldots,$ *let* $v_k$ *denote* integral minimizers[1] *of the problems*

$$
v_k \in \arg \min_{v \in \mathcal{C}} \{ \mathcal{H}(v) + \mathcal{TV}(v) - \langle \tilde{p}_{k-1}, v \rangle \}
\tag{4.53}
$$

*of the lifted Bregman iteration in Alg. 5.*
*Assume that the previous iterates* $u_{k-1}$ *and* $v_{k-1}$ *fulfil* $v_{k-1} = \mathbf{1}_{u_{k-1}}$. *If*

the chosen subgradients $p_{k-1} \in \partial \mathrm{TV}(u_{k-1})$ and $\tilde{p}_{k-1} \in \partial \mathcal{TV}(v_{k-1})$
in either setting satisfy

$$\tilde{p}_{k-1}(\cdot, t) = p_{k-1}(\cdot) \quad \text{for a.e. } t \in \Gamma, \qquad (4.54)$$

then the next iterates $v_k$ of the lifted Bregman iteration are the indicator
functions of the subgraphs of the next iterates $u_k$ of the original Bregman
iteration: $v_k = \mathbf{1}_{u_k}$.

*Proof.* If $\tilde{p}_{k-1}(\cdot, t) = p_{k-1}(\cdot)$ for a.e. $t \in \Gamma$, as assumed, then the
second integral in (4.50) can be rewritten as

$$\int_{\Omega} \int_{\mathbb{R}} \langle p_{k-1}(x), v(x, t) \rangle \, \mathrm{d}t \, \mathrm{d}x = \int_{\Omega \times \mathbb{R}} \langle \tilde{p}_{k-1}, v \rangle \, \mathrm{d}(x, t). \quad (4.55)$$

This means that for this the specific choice of $p_{k-1}$ and $\tilde{p}_{k-1}$ the prob-
lem (4.53) boils down to (4.50). As (4.50) is the calibration-based
lifted formulation of (4.52) it follows that any integral minimizer
$v_k = \mathbf{1}_u$ of (4.53) is the indicator function of the subgraph of a
minimizer of (4.52). As we assumed that the solution of (4.52) is
unique, it follows $v_k = \mathbf{1}_{u_k}$. $\qquad \square$

### 4.2.5 Existence of Subgradients Fulfilling the Sufficient Condition

One question that remains is whether subgradients $\tilde{p}_{k-1}$ as required
in Prop. 1 actually exist. In this section, we show that, under the
assumption (C1) made in Sec. 4.2.1, this is the case for $J = \mathrm{TV}$.

**Lemma 4.2.2** (Existence) *Under the assumption (C1) on p. 80, for ev-
ery subgradient $p \in \partial \mathrm{TV}(u)$ there exists a subgradient $\tilde{p} \in \partial \mathcal{TV}(\mathbf{1}_u)$
satisfying*

$$\tilde{p}(\cdot, t) = p(\cdot) \quad \text{for a.e. } t \in \Gamma. \qquad (4.56)$$

*Proof.* For fixed $p \in \partial \mathrm{TV}(u) \subseteq BV^*(\Omega)$ we define $\tilde{p} \in BV^*(\Omega \times \Gamma)$ as
$\langle \tilde{p}, v \rangle := \int_{\Omega} \int_{\Gamma} \langle p, v(x, t) \rangle \, \mathrm{d}t \, \mathrm{d}x$. If $p$ can be written as a (Lebesgue)
density, this corresponds to setting $\tilde{p}$ to constant copies of $p$ along
the $\Gamma$ axis, i.e., $\tilde{p}(x, t) := p(x), \forall t \in \Gamma$.

Following the assumption (C1) on p. 80, we consider functions
$u \in W^{1,1}(\Omega; \Gamma)$ under the embedding $W^{1,1} \hookrightarrow L^2$ and $p \in L^2(\Omega)$.
Due to the boundedness of $\Gamma$ it then also holds that $\tilde{p} \in L^2(\Omega \times \Gamma)$.

Therefore, similar to [BH16, Prop. 7] and [ET99, Example I.4.3, Prop. I.5.1], $\tilde{p}$ is a subgradient of $\mathcal{TV}$ at $\mathbf{1}_u$ if and only if

$$\mathcal{TV}(\mathbf{1}_u) = \langle \tilde{p}, \mathbf{1}_u \rangle - \mathcal{TV}^*(\tilde{p}). \tag{4.57}$$

From section 4.2.2, we recall

$$\mathcal{TV}(v) = \sup_{z \in \Phi_x} \left\{ \int_{\Omega \times \Gamma} \langle z, v \rangle \, \mathrm{d}(x, t) \right\} = \sigma_{\Phi_x}(v), \tag{4.58}$$

$$\Phi_x := \{ - \operatorname{div}_x \varphi_x | \varphi_x \in C_0(\Omega \times \Gamma; \mathbb{R}^n), \|\varphi_x\|_\infty \leq 1 \} \tag{4.59}$$

and, therefore,

$$\mathcal{TV}^*(\tilde{p}) = \delta_{\mathrm{cl}\,\Phi_x}(\tilde{p}), \tag{4.60}$$

where the closure is taken with respect to the $L^2(\Omega \times \Gamma)$ norm. Therefore, if we can show that $\tilde{p} \in \mathrm{cl}\,\Phi_x$ and $\mathcal{TV}(\mathbf{1}_u) = \langle \tilde{p}, \mathbf{1}_u \rangle$, by (4.57), we know that $\tilde{p} \in \partial \mathcal{TV}(\mathbf{1}_u)$.

The fact that $\tilde{p} \in \mathrm{cl}\,\Phi_x$ follows directly from $p \in \mathrm{cl}\,\Psi$ with $\Psi$ as in (4.25); recall

$$\Psi := \{ - \operatorname{div} \psi | \psi \in C_c(\Omega; \mathbb{R}^n), \|\psi\|_\infty \leq 1 \} \tag{4.61}$$

For every sequence $(p_n)_{n \in \mathbb{N}} \in \Psi$ with $p_n \xrightarrow{L^2(\Omega)} p$ we have a sequence $\psi_n$ of $C_c(\Omega)$ functions with $\|\psi_n\|_\infty \leq 1$ and $p_n = - \operatorname{div} \psi_n$. Thus for

$$(\varphi_x)_n(x, t) := \psi_n(x), \tag{4.62}$$

it holds $- \operatorname{div}_x(\varphi_x)_n \in \Phi_x$ and

$$- \operatorname{div}_x(\varphi_x)_n(x, t) = - \operatorname{div} \psi_n(x) = p_n(x). \tag{4.63}$$

Thus $- \operatorname{div}_x(\varphi_x)_n(\cdot, t) \xrightarrow{L^2(\Omega)} p$ in for all $t \in \Gamma$. Due to the boundedness of $\Gamma$, this implies $- \operatorname{div}_x(\varphi_x)_n \xrightarrow{L^2(\Omega \times \Gamma)} \tilde{p}$, which shows $\tilde{p} \in \mathrm{cl}\,\Phi_x$ as desired.

In order to show the final missing piece, namely $\mathcal{TV}(\mathbf{1}_u) = \langle \tilde{p}, \mathbf{1}_u \rangle$, we use (4.41) to argue that

$$\mathcal{TV}(\mathbf{1}_u) \stackrel{(4.41)}{=} \mathrm{TV}(u) = \langle p, u \rangle = \int_\Omega p(x) u(x) \, \mathrm{d}x. \tag{4.64}$$

We continue by rewriting $u$ using a coarea-type formula

$$\mathcal{TV}(\mathbf{1}_u) = \ldots = \int_\Omega p(x) \int_\Gamma \mathbf{1}_u(x, t) \, \mathrm{d}t \, \mathrm{d}x = \langle \tilde{p}, \mathbf{1}_u \rangle. \tag{4.65}$$

This concludes the proof. $\qquad\qquad\square$

By defining $\tilde{p}$ based on $p$ as above, we have recovered under the assumption (C1) a subgradient of the lifted regularizer $\mathcal{TV}$ of the form required by Prop. 1.

Now we can infer from Prop. 1 and Lemma 4.2.2 that under assumption (C1) the Bregman iteration and lifted Bregman iteration are equivalent when applied to the ROF problem, which provides a convenient validation of our construction of the lifted Bregman iteration in the continuous setting.

## 4.3 Equivalence under Sublabel-Accurate Discretization

In the previous section, we argued for the function solution space $BV(\Omega \times \Gamma)$. While theoretically interesting, this leaves the question whether a similar equivalence holds after sublabel-accurate discretization of the problem, i.e., with continuous domain $\Omega$ and discrete range $\Gamma^h = [\gamma_1, ..., \gamma_l]$.

In this section, we follow a formal line of argument as is also common in the related literature, arguing mostly pointwise. However, the arguments could equally be understood in the spatially discrete setting with finite $\Omega^h$, where arguments are more straightforward. For readability, we consider a fixed $x \in \Omega$ and omit $x$ in the arguments.

### 4.3.1 Bregman Iteration, Lifted

Analogously to our argument in the continuous setting, we first consider the "Bregman-first" Alg. 6 – now in a semi-discretized sublabel-accurate setting. In the following, we will use the notation introduced in Sec. 3.3.3; in particular, bold letters will indicate the sublabel-accurate setting.

In Alg. 6, minimizers of

$$\boldsymbol{F}_{\text{Breg}}(\boldsymbol{u}) = \tilde{\boldsymbol{H}}(\boldsymbol{u}; p_{k-1}) + \boldsymbol{TV}(\boldsymbol{u}) \qquad (4.66)$$

are sought. Again, we are mainly interested in the term $\tilde{\boldsymbol{H}}(\boldsymbol{u}; p_{k-1})$ which represents the sublabel-accurate lifted version of

$$\tilde{H}(u, p_{k-1}) = H(u) - \langle u, p_{k-1} \rangle . \qquad (4.67)$$

The following proposition states that under certain assumptions the sublabel-accurate lifted sum of two terms is the sum of the sublabel-accurate lifted terms.

**Proposition 4.3.1** *Consider for $i \in \{1, 2\}$*

$$\inf_{u \in BV(\Omega;\Gamma)} F_i(u), \quad F_i(u) := \int_\Omega f_i(x, u, dDu), \qquad (4.68)$$

*which fulfill (A1)-(A4) on p. 42 and (B1)-(B2) on p. 73 such that*

$$f_i(x, t, q) = \rho_i(x, t) + \eta(q). \qquad (4.69)$$

*For some $p : \Omega \to \mathbb{R}$ we consider data terms whose integrands $\rho_1$ and $\rho_2$ are pointwise linked via*

$$\rho_2(x, u(x)) := \rho_1(x, u(x)) - h(x, t), \ \ h(x, t) := p(x)u(x). \ \ (4.70)$$

*Applying calibration-based lifting together with the sublabel-accurate discretization as discussed in Sec. 3.3.3 on the problems $F_i$ results in lifted formulations*

$$\boldsymbol{F}_i(\boldsymbol{u}) = \int_\Omega \rho_i^{**}(x, \boldsymbol{u}(x)) \, dx + \int_\Omega \boldsymbol{\eta}^{**}(D\boldsymbol{u}). \qquad (4.71)$$

*Then it holds pointwise*

$$\rho_2^{**}(x, \boldsymbol{u}(x)) = \rho_1^{**}(x, \boldsymbol{u}(x)) - \boldsymbol{h}^{**}(x, \boldsymbol{u}(x)) \qquad (4.72)$$

$$= \rho_1^{**}(x, \boldsymbol{u}(x)) - \langle p(x)\tilde{\boldsymbol{\gamma}}, \boldsymbol{u}(x) \rangle, \qquad (4.73)$$

*where $\boldsymbol{\rho}_1, \boldsymbol{\rho}_2, \boldsymbol{h} : \mathbb{R}^{l-1} \mapsto \overline{\mathbb{R}}$ are calculated using eq. (3.104), and $\tilde{\boldsymbol{\gamma}} \in \mathbb{R}^{l-1}$ is defined as*

$$\tilde{\boldsymbol{\gamma}} := \Big( \gamma_2 - \gamma_1, \quad \ldots, \quad \gamma_l - \gamma_{l-1} \Big)^\top. \qquad (4.74)$$

Prop. 4.3.1 shows that the sublabel-accurate lifted sum of two functions is equal to the sum of the sublabel-accurate lifted functions if one of the summands is linear. This is quite useful in general and in particular with respect to the (lifted) Bregman iteration.

*Proof.* We deduce the biconjugate of $\boldsymbol{\rho}_2$ step-by-step and show that the final expression implies the claimed equality. Here, $x \in \Omega$ is fixed and we drop it from the notation. According to (3.104), the lifted representation of $\rho_2$ is

$$\boldsymbol{\rho}_2(\mathbf{u}) = \inf_{\substack{i \in \{1, \ldots, l-1\}, \\ \alpha \in [0,1]}} \Big\{ \rho_2(\gamma_i^\alpha) + \delta_{\mathbf{1}_i^\alpha}(\mathbf{u}) \Big\}. \qquad (4.75)$$

We use the definition of the Fenchel conjugate and the definition

of $\rho_2$:

$$\rho_2^*(\mathbf{v}) = \sup_{\mathbf{u} \in \mathbb{R}^{l-1}} \left\{ \langle \mathbf{u}, \mathbf{v} \rangle - \rho_2(\mathbf{u}) \right\} \tag{4.76}$$

$$= \sup_{\mathbf{u} \in \mathbb{R}^{l-1}} \left\{ \langle \mathbf{u}, \mathbf{v} \rangle - \inf_{\substack{i \in \{1,\dots,l-1\}, \\ \alpha \in [0,1]}} \left\{ \rho_2(\gamma_i^\alpha) + \delta_{\mathbf{1}_i^\alpha}(\mathbf{u}) \right\} \right\} \tag{4.77}$$

$$= \sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{v} \rangle - \rho_2(\gamma_j^\beta) \right\}. \tag{4.78}$$

With the definition of $\rho_2$ in (4.70), this gives:

$$\rho_2^*(\mathbf{v}) = \sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{v} \rangle - \rho_1 \left( \gamma_j^\beta \right) + p\gamma_j^\beta \right\}. \tag{4.79}$$

Using $\tilde{\gamma}$ as in (4.74), we can express $p\gamma_j^\beta$ in terms of $\mathbf{1}_j^\beta$:

$$\rho_2^*(\mathbf{v}) = \sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{v} \rangle - \rho_1 \left( \gamma_j^\beta \right) + p\gamma_1 + \langle p\tilde{\gamma}, \mathbf{1}_j^\beta \rangle \right\} \tag{4.80}$$

$$= \sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{v} + p\tilde{\gamma} \rangle + p\gamma_1 - \rho_1 \left( \gamma_j^\beta \right) \right\}. \tag{4.81}$$

Next, we compute the biconjugate of $\rho$:

$$\rho_2^{**}(\mathbf{w}) = \sup_{\mathbf{v} \in \mathbb{R}^{l-1}} \left\{ \langle \mathbf{v}, \mathbf{w} \rangle - \rho_2^*(\mathbf{v}) \right\}. \tag{4.82}$$

By substituting $\mathbf{z} := \mathbf{v} + p\tilde{\gamma}$, we get

$$\rho_2^{**}(\mathbf{w}) = \sup_{\mathbf{z} \in \mathbb{R}^{l-1}} \left\{ \langle \mathbf{z} - p\tilde{\gamma}, \mathbf{w} \rangle - \right. \tag{4.83}$$

$$\sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{z} \rangle + p\gamma_1 - \rho_1 \left( \gamma_j^\beta \right) \right\} \right\} \tag{4.84}$$

$$= \sup_{\mathbf{z} \in \mathbb{R}^{l-1}} \left\{ \langle \mathbf{z}, \mathbf{w} \rangle - \sup_{\substack{j \in \{1,\dots,l-1\}, \\ \beta \in [0,1]}} \left\{ \langle \mathbf{1}_j^\beta, \mathbf{z} \rangle - \rho_1 \left( \gamma_j^\beta \right) \right\} \right\}$$

$$- \langle \mathbf{w}, p\tilde{\gamma} \rangle - p\gamma_1 \tag{4.85}$$

$$= \rho_1^{**}(\mathbf{w}) - \left( \langle \mathbf{w}, p\tilde{\gamma} \rangle + p\gamma_1 \right). \tag{4.86}$$

In [Möl+15, Prop. 2], the authors consider the special case of a piecewise (in between the chosen labels $\gamma_i$) linear data term $\sigma$ and derive the following explicit form for the sublabel-accurate

expression $\sigma^{**}$:

$$\sigma^{**}(u) = \begin{cases} \sigma(\gamma_1) + \langle u, s \rangle, & \text{if } u_i \geq u_{i+1}, u_i \in [0,1], \\ \infty, & \text{else,} \end{cases} \quad (4.87)$$

where

$$s_i = \sigma(\gamma_{i+1}) - \sigma(\gamma_i). \quad (4.88)$$

Using this result, we see that the negative term in (4.86) is the biconjugate of the lifted linear term $h$ for admissible $\mathbf{w}$:

$$\mathbf{h}^{**}(\mathbf{w}) = \langle \mathbf{w}, p\tilde{\gamma} \rangle + p\gamma_1. \quad (4.89)$$

This concludes the proof of Thm. 4.3.1. $\qquad\qquad\square$

A consequence of Prop. 4.3.1 is

$$\tilde{H}(u; p_{k-1}) = H(u) - \langle p_{k-1}\tilde{\gamma}, u \rangle - p_{k-1}\gamma_1. \quad (4.90)$$

In the following, and similar to the continuous setting, we will use this relation to argue that if the *lifted given* subgradient $p_{k-1}\tilde{\gamma}$ is a valid subgradient in the lifted setting of Alg. 5 with sublabel-accurate discretization, it means we can find a suitable subgradient in the lifted setting that relates to a subgradient in the original setting such that Alg. 3 and Alg. 5 with sublabel-accurate discretization are equivalent.

### 4.3.2 Sufficient Condition for Equivalence

We again assume that the original problem $\inf_u F(u)$ has a unique solution, as is the case for strictly convex functionals such as ROF and introduce a sufficient condition on the chosen subgradients in Alg. 3 and Alg. 5 with sublabel-accurate discretization, which guarantees that the algorithms are equivalent. This is the semi-discretized version of Prop. 1:

**Lemma 4.3.2** (Sufficient Condition) *Assume that the energy* $F(u) = H(u) + \text{TV}(u)$ *fulfills the conditions (A1)-(A4) on p. 42, the conditions (B1)-(B2) on p. 73, and that, furthermore, the problems*

$$u_k \in \arg\min_{u \in \text{BV}(\Omega;\Gamma)} \{H(u) + \text{TV}(u) - \langle p_{k-1}, u \rangle\} \quad (4.91)$$

*of the Bregman iteration in Alg. 3 have unique solutions* $u_k$. *Assume*

*that for all $k = 1, 2, \ldots$ the minimizers $\boldsymbol{u}_k$ of the problems*

$$\boldsymbol{u}_k \in \arg\min_{\boldsymbol{u} \in \mathrm{BV}(\Omega;\mathbb{R}^{l-1})} \left\{ H(\boldsymbol{u}) + TV(\boldsymbol{u}) - \left\langle \tilde{\boldsymbol{p}}_{k-1}, \boldsymbol{u} \right\rangle \right\} \qquad (4.92)$$

*of the lifted Bregman iteration in Alg. 5 with sublabel-accurate discretization are sublabel-integral, that is they satisfy $\boldsymbol{u}_k(x) \in \boldsymbol{\Gamma}$ with $\boldsymbol{\Gamma}$ defined as in (3.102).*

*If for every $k = 1, 2, \ldots$ and at every point $x \in \Omega$ the chosen subgradients $p_{k-1} \in \partial \mathrm{TV}(u_{k-1})$ and $\tilde{\boldsymbol{p}}_{k-1} \in \partial TV(\boldsymbol{u}_{k-1})$ satisfy*

$$\tilde{\boldsymbol{p}}_{k-1}(x) \quad = \quad p_{k-1}(x)\tilde{\gamma}, \qquad (4.93)$$

*where*

$$\tilde{\gamma} := \left( \gamma_2 - \gamma_1, \quad \ldots, \quad \gamma_l - \gamma_{l-1} \right)^{\top}, \qquad (4.94)$$

*then the iterates $\boldsymbol{u}_k$ of the lifted Bregman iteration with sublabel-accurate discretization can be mapped to the iterates $u_k$ of the Bregman iteration via the pointwise transformation (3.103):*

$$u_k(x) = \gamma_1 + \left\langle \boldsymbol{u}_k, \tilde{\gamma} \right\rangle. \qquad (4.95)$$

*Proof.* It follows directly from Prop. 4.3.1 and the conclusion in (4.90) that (4.92) for this specific choice of $\tilde{p}_{k-1}$ is the sublabel-accurate lifted version of (4.91) – minus the term $p\gamma_1$ in (4.90). As the latter does not depend on $u$ it is not relevant for the minimization. As we assume that the solutions $\boldsymbol{u}_k$ of (4.92) are sublabel-integral, we may apply the pointwise projection (4.95) which results according to the theory in Chpt. 3 in a global minimizer of (4.91). This concludes the proof as we assume that the latter is unique. $\square$

The proposition states a sufficient condition on the subgradients $\tilde{p}_k$ and $p_k$ that guarantees that the Bregman iteration Alg. 3 and the lifted Bregman iteration Alg. 5 with sublabel-accurate discretization are equivalent. Note that the sufficient condition only holds under the assumption that the original problem has unique solutions $u_k$ and that sublabel-integral solutions $\boldsymbol{u}_k$ to the lifted Bregman iteration are found. Both assumptions are fulfilled for the strictly convex ROF functional. Existence of subgradients fulfilling the sufficient condition will be discussed in the next section in a fully discrete setting.

## 4.4 Fully-Discretized Setting

In this section, we consider the spatially discretized problem on the grid $\Omega^h$ with grid spacing $h$. We discuss how subgradients $\tilde{p}_k \in \partial TV(u_k)$ can be chosen in practice – if the choice of the lifted subgradients was numerically impossible or too expensive it would be inadvisable to use the lifted Bregman iteration in practice.

Furthermore, we show that subgradients fulfilling the sufficient condition in Lemma 4.3.2 exist when working with the anisotropic total variation. We propose a pointwise transformation, which allows to transform an arbitrary subgradient $\tilde{p}_k \in \partial TV(u_k)$ of an integral solution $u_k \in \Gamma$ into the form required by the sufficient condition. For the anisotropic total variation, we furthermore show that the transformed subgradient is indeed a valid subgradient. This allows us to demonstrate the equivalence of the Bregman iteration in Alg. 3 and the lifted Bregman iteration in Alg. 5 numerically and is thus part of the sanity check for the proposed algorithm.

### 4.4.1 Finding a Subgradient

Let us first discuss how to find a subgradient $\tilde{p}_k$ for the lifted Bregman iteration in Alg. 5. While there is ambiguity in the choice, the primal-dual nature of the lifted problem itself offers a way of finding at least one subgradient.

The discretized, sublabel-accurate total variation is of the form (3.107)

$$TV^h(u^h) = \sum_{x^h \in \Omega^h} \eta^{**}((\nabla^h u^h)(x^h)), \qquad (4.96)$$

$$\eta^{**}(g^h) := \max_{q^h \in K} \left\{ \langle q^h, g^h \rangle \right\} \qquad (4.97)$$

with $K$ defined by (3.109) (isotropic TV) or (3.111) (anisotropic TV). Here, $\nabla^h$ denotes the discretized forward-difference operator and in the following we write $(\nabla^h)^*$ for its adjoint.

The function $\eta^{**}$ is proper, convex, lower semicontinuous and 1–homogeneous. Therefore, a subgradient of $\eta^{**}$ fulfills [RW09, Prop. 11.3, Example 11.4]:

$$p^h \in \partial \eta^{**}(g^h) \quad \Leftrightarrow \quad p^h \in \arg\max_{p^h \in K} \left\{ \langle p^h, g^h \rangle \right\} \qquad (4.98)$$

Due to the properness and lower semicontinuity of $\eta^{**}$, it additionally holds [RW09, Cor. 10.9]

$$\partial\left(\sum_{x^h \in \Omega^h} \eta^{**}((\nabla^h \boldsymbol{u}^h)(x^h))\right) \supset \sum_{x^h \in \Omega^h} \partial\eta^{**}((\nabla^h \boldsymbol{u}^h)(x^h)). \qquad (4.99)$$

Next, we argue that

$$\partial(\eta^{**} \circ \nabla^h)(\boldsymbol{u}^h(x^h)) = (\nabla^h)^* (\partial\eta^{**})((\nabla^h \boldsymbol{u}^h)(x^h)). \qquad (4.100)$$

According to [Roc70, Thm. 23.9] the equality holds, if $\nabla^h$ is linear and the range of $\nabla^h$ contains a point of the relative interior of the domain of $\eta^{**}$: 0 is in the range of $\nabla^h$ as well as in the domain of $\eta^{**}$ which we here denote by $C$. $C$ is non-empty and convex. According to [Roc70, Thm. 6.4], 0 is in the relative interior of $C$ if and only if for any $\boldsymbol{y}^h \in C$ there exists a $\alpha > 1$ such that $(1 - \alpha)\boldsymbol{y}^h + \alpha 0 \in C$; For arbitrary $\boldsymbol{y}^h \in C$ we calculate

$$\eta^{**}((1 - \alpha)\boldsymbol{y}^h + \alpha 0) = \max_{q^h \in K} \left\{ \left\langle q^h, (1 - \alpha)\boldsymbol{y}^h + \alpha 0 \right\rangle \right\} \qquad (4.101)$$

$$= (1 - \alpha) \max_{q^h \in K} \left\{ \left\langle q^h, \boldsymbol{y}^h \right\rangle \right\}. \qquad (4.102)$$

As we assumed that $\boldsymbol{y}^h$ is in the domain $C$, it follows that also $(1 - \alpha)\boldsymbol{y}^h + \alpha 0$ is in the domain $C$ for any $1 < \alpha < +\infty$. This concludes the proof of (4.100).

Now we have a convenient choice for $\tilde{\boldsymbol{p}} \in \partial TV^h(\boldsymbol{u}^h)$: Let $q^h(x^h)$ denote the maximizer of (4.97) for $x^h \in \Omega^h$. Then, using the pointwise definition $\boldsymbol{p}^h(x^h) := (\nabla^h)^* q^h(x^h)$, we obtain a subgradient $\boldsymbol{p}^h$ of $TV^h(\boldsymbol{u}^h)$. As it holds that

$$\left\langle ((\nabla^h)^* q^h)(x^h), \boldsymbol{u}^h(x^h) \right\rangle = \left\langle q^h, (\nabla^h \boldsymbol{u}^h)(x^h) \right\rangle, \qquad (4.103)$$

the lifted Bregman iteration Alg. 5 with sublabel-accurate discretization in a spatially discrete setting can be implemented as:

---

**Algorithm 7:** Lifted Bregman Iteration (LBI) – Discrete Setting

---

1  Initialize $q_0^h = 0$

2  Repeat for $k = 1, 2, \ldots$

3      Set $F_{\text{Breg}}^h(\boldsymbol{u}^h, q^h) = \sum_{x^h \in \Omega^h} \rho^{**}(x^h, \boldsymbol{u}^h(x^h))$

4                     $+ \left\langle q^h(x^h) - q_{k-1}^h(x^h), (\nabla^h \boldsymbol{u}^h)(x^h) \right\rangle$

5      Find a solution $(\boldsymbol{u}_k^h, q_k^h)$ of the saddle-point problem

6          $\min_{\boldsymbol{u}^h : \Omega^h \mapsto \mathbb{R}^{(l-1)}} \max_{q^h : \Omega^h \mapsto K^h} F_{\text{Breg}}^h(\boldsymbol{u}^h, q^h),$

7

---

Here, $\rho^{**}$ denotes the integrand of the lifted data term; see (3.104).

In order to demonstrate the equivalence of the Bregman iteration in Alg. (3) and the lifted Bregman iteration in Alg. 5 for the convex ROF problem, we next discuss how the above chosen subgradients $p_k^h = (\nabla^h)^* q_k^h$ can be mapped to subgradients $\tilde{p}_k^h = (\nabla^h)^* \tilde{q}_k^h$ which fulfill the sufficient condition in Lemma 4.3.2.

### 4.4.2 Subgradient Transformation

In Lemma 4.3.2 we formulated a constraint on the subgradients for which the Bregman iteration in Alg. 4 and the lifted Bregman iteration in Alg. 5 are equivalent when applied to the convex ROF denoising problem. While this property is not necessarily satisfied if the subgradient $p_k^h$ is chosen according to the previous paragraph, we will now show that any such chosen subgradient $p_k^h$ can be transformed into another subgradient that satisfies condition (4.93). This allows us to investigate the conditional equivalence of the two algorithms experimentally.

Consider a pointwise sublabel-integral (see p. 66) solution $u_k^h$ with subgradient $p_k^h := (\nabla^h)^* q_k^h \in \partial TV^h(u_k^h)$, where $q_k^h(x^h) \in K$ is a maximizer of (4.97) for every $x^h \in \Omega^h$. We define a pointwise transformation: For fixed $x^h \in \Omega^h$ and $u_k^h(x^h) = \mathbf{1}_i^\alpha$, let $(q_k^h(x^h))^i \in \mathbb{R}^n$ denote the $i$-th row of $q_k^h(x^h)$, corresponding to the $i$-th label as prescribed by $u_k^h(x^h) = \mathbf{1}_i^\alpha$. Both in the isotropic and anisotropic case the transformation

$$\tilde{q}_k^h(x^h) := \frac{(q_k^h(x^h))^i}{\gamma_{i+1} - \gamma_i} \tilde{\gamma} \qquad (4.104)$$

returns an element of the set $K_{\text{iso}}$ or $K_{\text{an}}$. In the anisotropic case we can furthermore show that $\tilde{q}_k^h$ also maximizes (4.96) and, therefore, the transformation returns a subgradient $\tilde{p}_k^h := \nabla^\top \tilde{q}_k^h \in \partial J^h(u_k^h)$ of the desired form (4.93). The restriction to the anisotropic case is unfortunate but necessary due to the fact that the coarea formula does not hold in the discretized case for the usual isotropic discretizations: There is no known discretization that is both isotropic and satisfies the coarea formula exactly, therefore, any arguments that rely on rounding/thresholding solutions are generally restricted to the anisotropic case.

**Proposition 4.4.1** (Projection Formula) *Consider the following anisotropic TV regularizer in the calibration-based lifted setting with sublabel-accurate discretization:*

$$TV^h(u^h) = \sum_{x^h \in \Omega^h} \eta^{**}((\nabla^h u^h)(x^h)) \tag{4.105}$$

$$\eta^{**}(g) = \sup_{q \in K_{an}} \{\langle q, g \rangle\}, \tag{4.106}$$

$$K_{an} = \bigcap_{j=1,\dots,n} \left\{ q \mid \|q_{i,j}\|_2 \leq \gamma_{i+1} - \gamma_i, \forall i = 1, \dots, l-1 \right\}. \tag{4.107}$$

*Assume that a sublabel-integral $u^h$ is given, i.e., it holds for each $x_m^h \in \Omega^h$ that $u^h(x_m^h) = \mathbf{1}_{i_m}^{\alpha_m} \in \Gamma$ with $\Gamma$ as in (3.102). Moreover, assume that $p^h = (\nabla^h)^* q^h$ is a subgradient of $TV^h(u^h)$. Then*

$$\tilde{p}^h := (\nabla^h)^* \tilde{q}^h \tilde{\gamma} \tag{4.108}$$

*with $\tilde{q}^h$ pointwise for each $x^h \in \Omega^h$ defined as*

$$\tilde{q}^h(x^h) := \frac{(q^h(x^h))^{i_m}}{\gamma_{i_m+1} - \gamma_{i_m}} \tag{4.109}$$

*is also a subgradient of $TV^h(u^h)$. Furthermore, $(\nabla^h)^* \tilde{q}^h$ is a subgradient in the unlifted case, i.e., $(\nabla^h)^* \tilde{q}^h \in \partial TV^h(u^h)$ for*

$$u^h(x^h) = \gamma_1 + \sum_{i=1}^{l-1} u_i^h(x^h)(\gamma_{i+1} - \gamma_i). \tag{4.110}$$

*Proof.* The proof consists of two parts. First, we show that $\tilde{p}^h$ in (4.108) is a valid subgradient in the lifted setting. Second, we show that $(\nabla^h)^* \tilde{q}^h$ in (4.108) is a valid subgradient in the unlifted setting.

In the anisotropic case the spatial dimensions in the constraint set $K$ (4.107) are uncoupled, therefore we assume without loss of generality that $n = 1$. Consider two neighboring points $x_m^h$ and $x_{m+1}^h$ with $u^h(x_m^h) = \mathbf{1}_i^\alpha$ and $u^h(x_{m+1}^h) = \mathbf{1}_j^\beta$. Applying the forward difference operator, we have

$$\nabla^h u^h(x_m^h)h = \begin{cases} (\mathbf{0}_{i-1}, 1 - \alpha, \mathbf{1}_{j-i-2}, \beta, \mathbf{0}_{(l-1)-j})^\top, & i < j, \\ (\mathbf{0}_{i-1}, \beta - \alpha, \mathbf{0}_{(l-1)-i})^\top, & i = j, \\ (\mathbf{0}_{j-1}, \beta - 1, -\mathbf{1}_{i-j-2}, -\alpha, \mathbf{0}_{(l-1)-j})^\top, & i > j. \end{cases} \tag{4.111}$$

From the previous considerations, we know that the subgradients

of $\boldsymbol{TV}^h(\boldsymbol{u}^h)$ are given by $((\nabla^h)^\top \boldsymbol{q}^h)(x^h)$, where

$$q^h(x^h) \in \arg\max_{q \in K_{\mathrm{an}}} \left\langle \boldsymbol{q}, (\nabla^h u^h)(x^h) \right\rangle. \qquad (4.112)$$

From the definition of $\boldsymbol{K}_{\mathrm{an}}$ and from (4.111) we conclude that

$$q^h(x_m^h) = \begin{cases} (*, \ \gamma_{i+1} - \gamma_i, \ ..., \ \gamma_{j+1} - \gamma_j, \ *)^\top, & i < j, \\ (*, \ \mathrm{sgn}(\beta - \alpha)(\gamma_{i+1} - \gamma_i), \ *)^\top, & i = j, \\ (*, \ \gamma_j - \gamma_{j+1}, \ ..., \ \gamma_i - \gamma_{i+1}, \ *)^\top, & i > j, \end{cases} \qquad (4.113)$$

where the elements marked with $*$ can be chosen arbitrarily as long as $q^h(x_m^h) \in \boldsymbol{K}_{\mathrm{an}}$. Due to this special form, the transformation in (4.109) results in $\tilde{q}^h(x^h) \in \{-1, +1\}$ and thus $\tilde{\boldsymbol{q}}^h := \tilde{q}^h \tilde{\boldsymbol{\gamma}} = \pm \tilde{\boldsymbol{\gamma}}$. Crucially, this transformed vector $\tilde{\boldsymbol{q}}^h$ is a special case of the form described in (4.113), therefore it is also dual-optimal in (4.112) and $\tilde{\boldsymbol{p}}^h = \nabla^\top \tilde{q}^h \tilde{\boldsymbol{\gamma}}$ is another subgradient in the lifted setting.

It remains to show that $(\nabla^h)^* \tilde{q}^h$ is a subgradient in the unlifted setting. To this end, we employ sublabel-accurate discretization with $l + 1 = 2$ labels. The "lifted" label space is $\boldsymbol{\Gamma} = [0, 1]$, independently of the actual $\Gamma \subset \mathbb{R}$; see [Möl+15, Prop. 3]. Then with $u^h(x_m^h) = \gamma_i^\alpha$ and $u^h(x_{m+1}^h) = \gamma_j^\beta$ (corresponding to $\boldsymbol{u}^h(x_m^h) = \boldsymbol{1}_i^\alpha$ and $\boldsymbol{u}^h(x_{m+1}^h) = \boldsymbol{1}_j^\beta$ from before), applying the forward difference operator we have

$$\nabla u^h(x_m^h) = \frac{1}{h}(\gamma_j^\beta - \gamma_i^\alpha). \qquad (4.114)$$

Therefore, dual maximizers of (4.112) are

$$q^h(x_m^h) = \mathrm{sgn}(\gamma_j^\beta - \gamma_i^\alpha)|\boldsymbol{\Gamma}| = \pm 1, \qquad (4.115)$$

where the algebraic signs coincide pointwise in the lifted and unlifted setting. Thus $p^h = (\nabla^*)^\top q^h = (\nabla^*)^\top \tilde{q}^h$ is a subgradient in the unlifted setting. $\qquad \square$

The projection in the proposition above allows us to demonstrate the equivalence of the Bregman iteration in Alg. 3 and the lifted Bregman iteration in Alg. 5 numerically as a sanity check an experimental validation of the proposed lifted Bregman iteration.

**(a)** ROF

**(b)** Stereo Matching, Umbrella [Sch+14]

**(c)** Stereo Matching, Backpack [Sch+14]

**Figure 4.2: Input data.** Input images for the ROF and stereo matching experiments.

## 4.5 Numerical Results

In this section, we numerically support our theoretical findings concerning the equivalence of the Bregman iteration in Alg. 4 and the lifted Bregman iteration in Alg. 5 when applied to the ROF denoising problem with subgradients that fulfill the sufficient condition in Lemma 4.3.2. Furthermore, we investigate a stereo-matching example which suggests that the lifted Bregman iteration for variational models with arbitrary data term may produce an inverse scale space of the solution in which nonlinear eigenfunctions of the regularizer appear in order of the magnitude of their eigenvalue.

We use the toolboxes `sublabel_relax` [LM16; Möl+15] and `prost` [LM15] which provide CUDA implementations of the sublabel-accurate discretization framework and PDHG algorithm.

The experiments were performed on an Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz, NVIDIA GeForce RTX 2070, Ubuntu 18.04.6 LTS, MATLAB R2020b, CUDA 10.2.

### 4.5.1 Convex Energy with Synthetic Data

In the first experiment we consider the ROF problem

$$\inf_{u \in \mathrm{BV}(\Omega;\Gamma)} F(u), \quad F(u) = \frac{\lambda}{2}\|u(x) - f(x)\|_{L^2}^2 + \mathrm{TV}(u), \quad (4.116)$$

where $\Omega = (0, 300) \times (0, 300)$, $\Gamma = [0, 1]$ and $\lambda = 0.05$. The input image $f$ is shown in Fig. 4.2a.

We compare the results of the Bregman iteration in Alg. 4 and the lifted Bregman iteration in Alg. 5 with sublabel-accurate discretization. The results are shown in Fig. 4.3 and will be discussed after the following technical details.

**Technical Details.** After lifting the energy (4.116) using the CBL approach, the domain $\Omega \times \Gamma$ was discretized as $\Omega^h \times \Gamma^h$, where $\Omega^h$ is a cell-centered grid with $m = 300 \times 300$ grid points and $\Gamma^h$ is

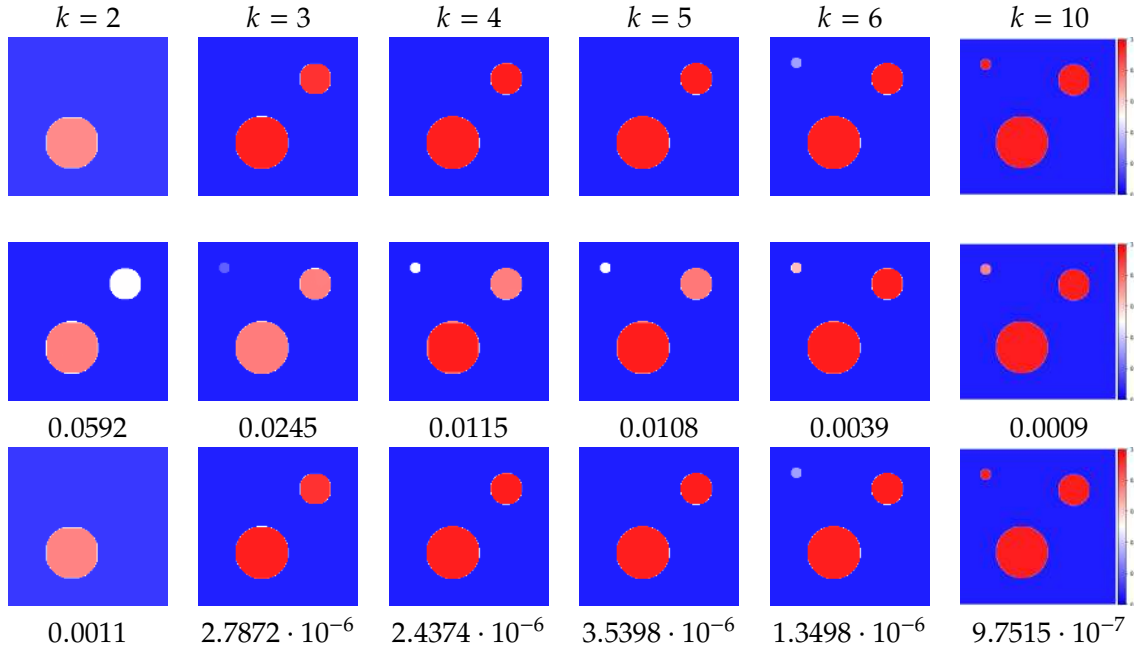| $k = 2$ | $k = 3$ | $k = 4$ | $k = 5$ | $k = 6$ | $k = 10$ |
|---|---|---|---|---|---|
| | | | | | |
| 0.0592 | 0.0245 | 0.0115 | 0.0108 | 0.0039 | 0.0009 |
| | | | | | |
| 0.0011 | $2.7872 \cdot 10^{-6}$ | $2.4374 \cdot 10^{-6}$ | $3.5398 \cdot 10^{-6}$ | $1.3498 \cdot 10^{-6}$ | $9.7515 \cdot 10^{-7}$ |

**Figure 4.3: Equivalence of Classical and Lifted Bregman on a Convex Problem.** On the *convex* ROF denoising problem with anisotropic TV, a plain implementation of the classical Bregman iteration as in Alg. 4 **(top row)** and a naïve implementation of the lifted generalization as in Alg. 6 with lifted subgradients chosen as described in Sec. 4.4.1 **(middle row)** show clear differences. If, however, the lifted subgradients are transformed as described in Sec. 4.4.2 such that they fulfill the sufficient condition in Lemma 4.3.2, the lifted iterates **(bottom row)** are visually indistinguishable from the classical iteration in this fully convex case. This is also quantitatively supported by the mean of the absolute pixelwise difference, e.g., $\frac{1}{|\Omega^h|} \sum_{x^h \in \Omega^h} |u_k^{l=2}(x^h) - u_k^{l=5}(x^h)|$, given under the images.

a nodal grid with $l$ grid points. The energy was then discretized using the sublabel-accurate discretization and solved using the PDHG algorithm.

For the implementation, we relied on the toolbox `sublabel_relax` [LM16; Möl+15] which provides an implementation of the ROF problem with CBL lifting and sublabel-accurate discretization. It uses finite forwards differences and Neumann boundary conditions for the total variation. Note, that we here used *anisotropic* total variation.

For the Bregman iteration we used $l = 2$ labels (Fig. 4.3, top row), in case of the lifted Bregman iteration we used $l = 5$ labels. We computed the lifted Bregman iteration twice: First, we chose the subgradients required by the linear Bregman term as described in Sec. 4.4.1 (Fig. 4.3, middle row). Second, we additionally transformed the chosen subgradients as described in Sec. 4.4.2 (Fig. 4.3, bottom row).

We solved the (lifted) Bregman iteration steps with the PDHG implementation in `prost` [LM15].

**Results.** The results shown in Fig. 4.3 clearly support our theory that Alg. 4 and Alg. 5 are equivalent when applied to the convex
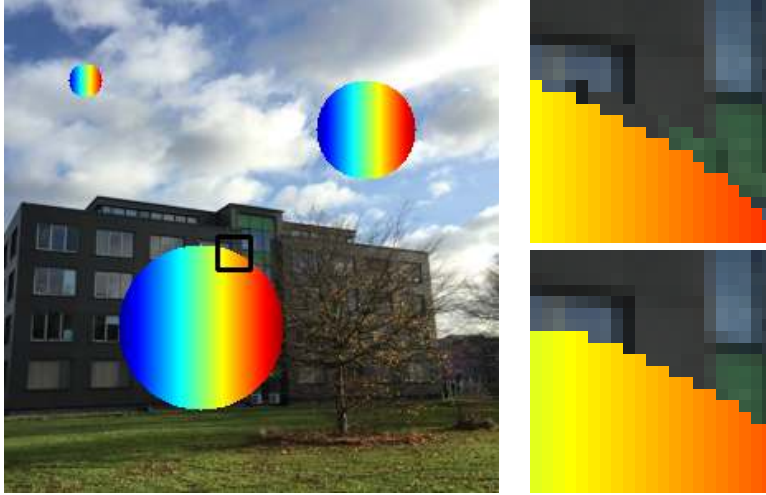
**Figure 4.4: Artificial Data for Stereo Matching.** The backgrounds of the input images $I_1$ and $I_2$ are chosen to be identical. Only within the three circles $I_1$ and $I_2$ differ. The information within the circles is shifted four pixels sideways; the circles themselves stay in place. The black square marks the area of the close-ups of $I_1$ **(top)** and $I_2$ **(bottom)**.

ROF problem when the subgradients fulfill the sufficient condition in Lemma 4.3.2: Once subgradients are transformed to fulfill the sufficient condition, the solutions of Alg. 4 and Alg. 5 are the same up to numerically negligible differences.

Using subgradients which do not fulfill the sufficient condition, we observe a slightly different behavior of the iterates $u_k$. The nonlinear eigenfunctions of the total variation appear to linearly increase in magnitude during the iteration instead of appearing at a precise iteration step.

### 4.5.2 Non-Convex Stereo Matching with Artificial Data

In the following example, we aim to empirically investigate whether properties of the Bregman iteration carry over to the lifted Bregman iteration for *arbitrary* (non-convex) data terms. We consider a relatively simple formulation of the stereo-matching problem:

$$\inf_{u \in BV(\Omega;\Gamma)} \{H(u) + \mathrm{TV}(u)\}, \tag{4.117}$$

$$H(u) = \lambda \int_\Omega h_\tau \left(|I_1(x_1, x_2) - I_2(x_1, x_2 - u(x))|\right) \, dx. \tag{4.118}$$

Here, $I_1$ and $I_2$ are two given input images and $h_\tau(\alpha) := \min\{\tau, \alpha\}$ is a threshholding function. In our experiments, we used $\lambda = \frac{1}{7}$ and $\tau = 0.1$. Note that the intensity-based data term is non-convex.

**Input Images.** A typical observation when using nonlinear scale space methods is that components in the solution corresponding to non-linear eigenfunctions of the regularizer appear at certain points in time depending on their eigenvalue.
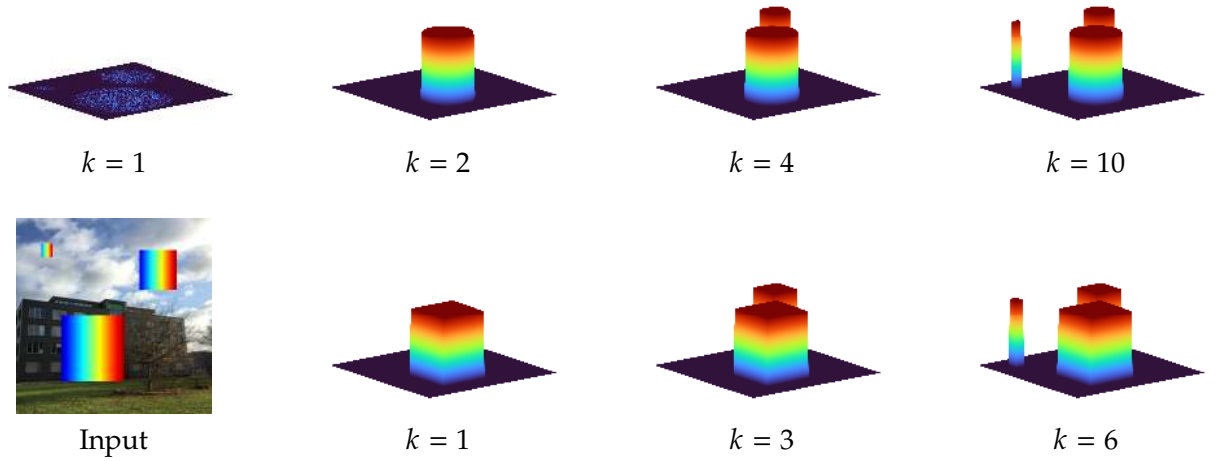
**Figure 4.5: Lifted Bregman on Stereo Matching Problem with Artificial Input Data.** We applied the lifted Bregman iteration in Alg. 5 to a non-convex stereo matching problem with data term (4.118). We evaluated the algorithm for an isotropic **(top)** and an anisotropic **(bottom)** implementation of the total variation. The input data for the isotropic setting is shown in Fig. 4.4; in the anisotropic setting we used square instead of circular cutouts **(bottom left)**. For this non-convex data term, the lifted Bregman iteration demonstrates a behavior reminiscent of the classical, convex (quadratic) inverse scale space flow: Components that correspond to nonlinear eigenfunctions of the total variation regularizer appear progressively in the depth map and components associated with larger eigenvalues appear at later iterations.

We thus constructed two artificial input images $I_1$ and $I_2$ such that the minimizer $\tilde{u}$ of the data term in (4.118) is the sum of eigenfunctions of the isotropic (anisotropic) TV, i.e., scalar multiples of indicator functions of circles (squares).

In the following we elaborate the isotropic setting. For non-overlapping circles $\mathscr{B}_{r_i}(m_i)$, $i = 1, ..., j < \infty$ with centers $m_i$ and radii $r_i$, we want a solution of the form

$$\tilde{u}(x) = \sum_{i=1}^{j} \mathbf{1}_{\mathscr{B}_{r_i}(m_i)}(x), \tag{4.119}$$

where

$$\mathbf{1}_{\mathscr{B}_{r_i}(m_i)}(x) = \begin{cases} 1, & \text{if } x \in \mathscr{B}_{r_i}(m_i), \\ 0, & \text{else.} \end{cases} \tag{4.120}$$

Fig. 4.4 shows the input data, which was designed accordingly; note that there is no displacement except inside the circles, where it is non-zero but constant.

**Technical Details.** After lifting the energy (4.118) using the CBL approach, the domain $\Omega \times \Gamma = (300, 300) \times [0, 10]$ was discretized as $\Omega^h \times \Gamma^h$, where $\Omega^h$ is a cell-centered grid with $m = 300 \times 300$ grid points and $\Gamma^h$ is a nodal grid with $l = 5$ grid points. The energy was then discretized using the sublabel-accurate discretization and solved using the PDHG algorithm.

For the implementation, we again relied on the github toolbox `sublabel_relax` [LM16; Möl+15] which provides an implementation of a slightly different stereo matching problem with CBL lifting and sublabel-accurate discretization. It uses finite forwards differences and Neumann boundary conditions for the total variation.

We solved the lifted Bregman iteration steps with the PDHG implementation given in `prost` [LM15].

**Results.**  In analogy to the convex ROF example in Fig. 4.3 and the theory of inverse scale space flow, we would expect the following property to hold for the lifted Bregman iteration: The solutions – here the depth maps of the artificial scene – returned in each iteration of the lifted Bregman iteration progressively incorporate the discs (eigenfunctions of isotropic TV) according to their radius (associated eigenvalue); the biggest disc should appear first, the smallest disc last.

Encouragingly, the experiments confirm these expectations numerically also for the non-convex case, see Fig. 4.5. This suggests that the lifted Bregman iteration could be useful for designing filters that decompose the solution of a variational problem with arbitrary data term into eigenfunctions of the regularizer.

### 4.5.3 Non-Convex Stereo Matching with Real-World Data

We also computed results for a stereo-matching problem with real-world data using a more elaborate formulation of the stereo-matching problem as in [Möl+15]

$$
\inf_{u\in\mathrm{BV}(\Omega;\Gamma)} \{H(u) + \mathrm{TV}(u)\}\,,
$$
$$
H(u) = \lambda \int_\Omega \int_{W(x)} \sum_{j=1,2} h_\tau(d_j(I_1(y), I_2(y)))\, \mathrm{d}y\, \mathrm{d}x. \tag{4.121}
$$

Again, $I_1$ and $I_2$ are two given input images and $h_\tau(\alpha) := \min\{\tau, \alpha\}$ is a truncation function. Furthermore, $W(x)$ denotes a patch around $x$ and $d_j$ is the absolute gradient difference

$$
d_j(I_1(y), I_2(y)) = \left| \frac{d}{dx_j} I_1(y_1, y_2) - \frac{d}{dx_j} I_2(y_1, y_2 - u(y)) \right|. \tag{4.122}
$$

In our experiments, we used $\tau = 0.1$. The input images are shown in Fig. 4.2b–4.2c.

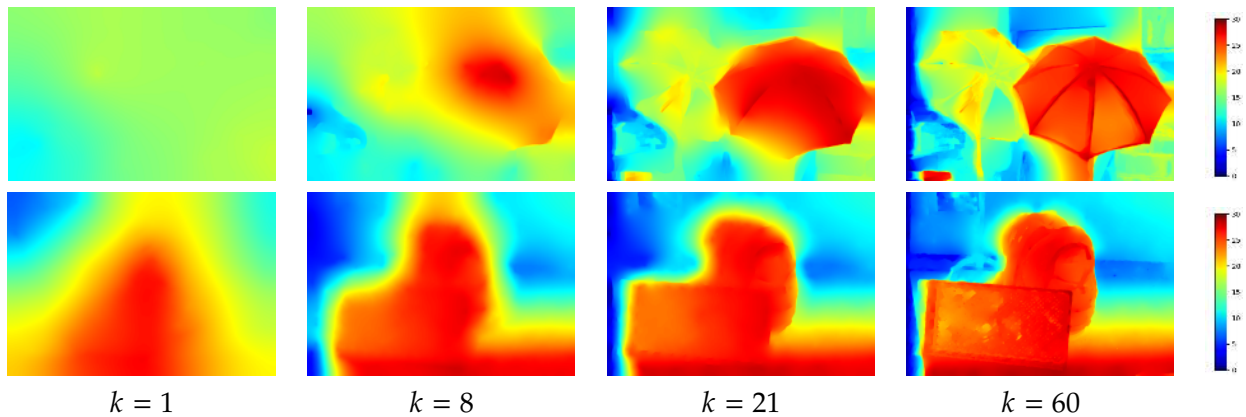$k = 1$      $k = 8$      $k = 21$      $k = 60$

**Figure 4.6: Lifted Bregman for Non-Convex Non-Linear Scale Space.** Shown are the results of the lifted Bregman iteration applied to a non-convex stereo matching problem with the data term described in (4.121) and total variation reqularizer. The input image pairs were taken from the "Umbrella" and "Backpack" instances from the Middlebury stereo datasets [Sch+14], see also Figs. 4.2b–4.2c. At $k = 1$, the solution is a coarse approximation of the depth map. As the iteration advances, details are progressively incorporated according to their scale. The results show a qualitative similarity to the inverse scale space flow associated to the Bregman iteration in case of the convex ROF denoising problem.

**Technical Details.** Again, we first lifted the energy (4.121) using the CBL approach. For the Umbrella (Backpack) input images the domain $\Omega \times \Gamma = (1480, 1008) \times [0, 30]$ ($\Omega \times \Gamma = (1470, 1008) \times [0, 30]$) was discretized as $\Omega^h \times \Gamma^h$, where $\Omega^h$ is a cell-centered grid with $m = 1480 \times 1008$ ($m = 1470 \times 1008$) grid points and $\Gamma^h$ is a nodal grid with $l = 5$ ($l = 5$) grid points. The energy was then discretized using the sublabel-accurate discretization and solved using the PDHG algorithm.

For the implementation, we again relied on the github toolbox `sublabel_relax` [LM16; Möl+15], which provides an implementation of the stereo matching problem with CBL lifting and sublabel-accurate discretization. It uses finite forwards differences and Neumann boundary conditions for the total variation. This time, we used isotropic TV regularizer and chose subgradients according to Sec. 4.4.1. We solved the lifted Bregman iteration steps with the PDHG implementation given in `prost` [LM15].

**Results.** The results can be seen in Fig. 4.6 (Umbrella: $\lambda = 0.1$; Backpack: $\lambda = 0.083$). Again, the evolution of the depth map throughout the iteration resembles an inverse scale space flow. The first solution is a smooth approximation of the depth proportions and as the iteration continues, finer structures are added.

### 4.5.4 Conclusion

We have proposed a combination of the Bregman iteration and the calibration based lifting approach with the aim of extending the

inverse scale space iteration to variational problems with convex, 1-homogeneous regularizer and possibly non-convex data term.

We showed that the lifted Bregman iteration is in fact an extension of the classic Bregman iteration: For the convex ROF denoising problem, the iterates of the two algorithms Alg. 4 and Alg. 5 agree if a sufficient condition on the subgradients is fulfilled.

Applied to nonconvex problems, numerical experiments show that the behavior of the solutions of the lifted Bregman iteration is very similar to what one expects in classical inverse scale space. Components that correspond to nonlinear eigenfunctions of the total variation regularizer appear respective to their eigenvalue over the course of the iteration.

This opens up a number of interesting theoretical questions: Does the lifted Bregman iteration in its continuous limit provably lead to an inverse scale space flow with similar properties as the classic inverse scale space flow (4.10)-(4.12)? Do nonlinear eigenfunctions of the regularizer appear at a scale of the flow determined by the respective eigenvalue as in (4.13)? If this was the case, one could define weak data-dependent spectral representations as in (4.14) and define edge-preserving filters as in (4.16) for the solution to a broad class of variation problems (4.1).

Such a theory is, however, mathematically challenging. Let us consider a single step of the lifted Bregman iteration:

$$v_k \in \arg\min_{v \in \mathcal{C}} \left\{ \lambda \mathcal{H}(v) + \mathcal{J}(v) - \langle \tilde{p}_{k-1}, v \rangle \right\}. \tag{4.123}$$

Assuming sufficient regularity of the energy such that additivity of the subdifferentials holds [Roc70, Thm. 23.8], the optimality condition reads

$$0 \in \lambda \partial \mathcal{H}(v_k) + \partial \mathcal{J}(v_k) - \tilde{p}_{k-1}. \tag{4.124}$$

Assume we have $\tilde{q}_k \in \partial \mathcal{H}(v_k)$ and $\tilde{p}_k \in \partial \mathcal{J}(v_k)$ such that the condition is fulfilled:

$$0 = \lambda \tilde{q}_k + \tilde{p}_k - \tilde{p}_{k-1} \tag{4.125}$$

$$\Leftrightarrow \quad -\tilde{q}_k = \frac{1}{\lambda}(\tilde{p}_k - \tilde{p}_{k-1}). \tag{4.126}$$

Interpreting $\lambda$ small enough again as the "time" passed between step $k - 1$ and $k$, replacing the discrete steps of the lifted Bregman iteration informally with a continuous time variable $s \in [0, T)$, and dropping the dependecy on $\Omega$ for better readability gives

$$-\tilde{q}(s) = \frac{\partial}{\partial s} \tilde{p}(s), \quad \tilde{q}(s) \in \partial \mathcal{H}(v(s)), \quad \tilde{p}(s) \in \partial \mathcal{J}(v(s)), \quad \tilde{p}(0) = 0.$$

Unfortunately, it is not clear how to find an explicit form for the solution $v$ for a general data term $H$. Thus, the question whether eigenfunctions of the regularizer appear at a scale of the flow given via the respective eigenvalue as in the convex setting (see (4.13)) currently remains open.

As the theory behind the application of the lifted Bregman iteration to variational problems with arbitrary, possibly non-convex data term is still to be fully developed, the here presented approach opens many interesting questions. Is it possible to define a spectral theory that is dependent on the input data, the data term and the regularizer of the problem at hand? Can we define an inverse scale space for/of the unknown solution to a data term, where the notion of scale is introduced via the eigenfunctions and eigenvalues of the regularizer? So far we do not have answers to these questions yet, which would constitute a larger step towards constructing non-linear decompositions, filters, and generation of a hierarchy of solutions to problems with non-convex data terms, with a precise nonlinear notion of scale.

# EmNeFs: Embedded Neural Fields | 5

In this chapter, we propose a learning-based framework for solving variational problems

$$\inf_{u \in BV(\Omega;\Gamma)} F(u), \quad F(u) := \int_\Omega f(x, u(x), \mathrm{d}Du), \qquad (5.1)$$

where $\Omega \subset \mathbb{R}^n$ is open and bounded, and $\Gamma \subset \mathbb{R}$ is closed; see also sidenote on p. 42. Furthermore, we assume that $f$ fulfills the assumptions (A1)–(A4) introduced on p. 42 and is in addition

(B1) *separable* in the sense that $f : \Omega \times \mathbb{R} \times \mathbb{R}^n$ can be written with the help of $\rho : \Omega \times \mathbb{R} \to \mathbb{R}$ and $\eta : \mathbb{R}^n \to \mathbb{R}$ as

$$f(x, t, p) = \rho(x, t) + \eta(p), \qquad (5.2)$$

(B2) *positively* 1-*homogeneous* in the last parameter, i.e.,

$$\eta(\lambda p) = \lambda \eta(p), \quad \forall p \in \mathbb{R}^n, \lambda > 0. \qquad (5.3)$$

Assumptions (B1)–(B2) will allow us to employ a special form of the calibration-based lifting (see p. 56) and to compare it to the state-of-the-art approach, which uses sublabel-accurate discretization. The framework proposed in this chapter could, however, be adjusted to problems (5.1) fulfilling only (A1)–(A4). In this case, the implementation of the constraints (5.9)–(5.10) would have to be replaced by an implementation of the constraint (3.47).

In the following, we denote by $D : BV(\Omega;\Gamma) \to \mathbb{R}$ the *data term*

$$D(u) = \int_\Omega \rho(x, u(x)) \, \mathrm{d}x, \qquad (5.4)$$

and use throughout our explanations and experiments the *total variation* regularizer TV : $BV(\Omega;\Gamma) \to \mathbb{R}$,

$$TV(u) = \int_\Omega \eta(Du). \qquad (5.5)$$

The TV regularizer is admissible as it is convex, positively 1-homogeneous and lower-semicontinuous in $BV(\Omega;\mathbb{R})$ with respect to the $L^1_{\mathrm{loc}}$-topology, see [AFP00, Prop. 3.6]. It is used exemplarily as it is common in variational imaging problems. Using the EmNeF approach in combination with a different regularizer which fulfills assumptions (B1)–(B2) only requires to change the implementation of the constraint (5.9).

## 5.1  Motivation and Outline

As outlined in the previous chapters, calibration-based lifting is a powerful theoretical framework, for reformulating non-convex variational problems of the form (5.1) as convex ones, and, since being introduced into the imaging community [Poc+09; Poc+10], related papers have won prizes at CVPR 2016 or SSVM 2021. However, applications of calibration-based lifting in the imaging community are hard to find, which might be due to the complex theory, numerically challenging implementation and computational run time which is not suited for real-time applications.

1:  Neural fields [Xie+22] are coordinate-based networks: they accept (often low-dimensional) coordinates as inputs and predict the value of a *field* at said coordinates – here *field* refers to a physical quantity defined over the input domain.

Neural fields (NFs)[1], on the other hand, celebrate a huge practical success and are widely used in a variety of applications [Xie+22; Mil+21; Koe+22]. They allure with their computational efficiency in representing functions.

Our aim in this chapter is to present a novel framework for solving variational problems (5.1) numerically by combining the strong theory of calibration-based lifting with the computational power of neural fields. Instead of tackling the calibration-based lifted formulation (5.1) with the PDHG algorithm (see Sec. 3.3.2 – Sec. 3.4), we propose a neural fields based approach which we call *embedded neural fields* (EmNeF).

**SA-PDHG Approach.**   The state-of-the-art is to discretize the lifted energy using the *sublabel-accurate* discretization scheme [Möl+15; MC17] discussed in Sec. 3.3–3.3.3 and to solve the discrete problem with the *primal-dual hybrid gradient* (PDHG) [Poc+09; Poc+10; Möl+15] algorithm. While this approach allows to find good solutions of variational problems (5.1), it also has several drawbacks.

The sublabel-accurate discretization leads to a rather intricate problem, which requires, among others, to calculate projections onto the piecewise (in between the the chosen discretization points of Γ (labels)) epigraph of the Fenchel conjugate of the data term. This projection is not always easy to compute and might require further approximation steps.

The *sublabel-accurate* discretization scheme, furthermore, relies on assumptions (B1)-(B2). If these assumptions are not fulfilled, another discretization scheme such as the finite-difference approach from [Poc+10] which we also discussed in Sec. 3.3 needs to be employed. This discretization leads to a discrete problem which is less involved to implement, but more expensive to solve, as it requires a finer discretization with respect to the Γ-axis.

**EmNeF Approach.** In the EmNeF approach, the goal is to "learn" a continuous representation of the lifted solution in terms of a neural field. While we use terminology and frameworks from the learning community, the EmNeF approach is not a classical learning approach as it does not generalize to different input data. It can instead be understood as stochastic optimization of a non-linear approximation.

Our method can be described in two steps, where the first step again consists of using the calibration-based lifting framework (Chpt. 3) in order to express problems of the form (5.1) as convex problems

$$\inf_{v \in \mathcal{C}} \sup_{\substack{\varphi_x \in \mathcal{K}_x \\ \varphi_t \in \mathcal{K}_t}} \mathcal{F}(v, \varphi_x, \varphi_t), \tag{5.6}$$

where

$$\mathcal{F}(v, \varphi_x, \varphi_t) := \int_{\Omega \times \mathbb{R}} \left\langle \begin{pmatrix} \varphi_x \\ \varphi_t \end{pmatrix}, \mathrm{d}Dv \right\rangle. \tag{5.7}$$

Developing a full approximation theory using neural fields is a long-term undertaking for the scientific community as a whole [HSW89; CMB00; SM17; Her+21] and we do not aim to do so here. In order to not give this impression, for the remainder of this chapter, we will argue informally without rigorously defining the function spaces. Therefore, we consider the primal constraint set

$$\mathcal{C} = \{v : \Omega \times \mathbb{R} \to [0, 1] : v(\cdot, t) = 1, \ t \leq \Gamma_{\min},$$
$$v(\cdot, t) = 0, \ t > \Gamma_{\max}\}, \tag{5.8}$$

as well as the dual constraint sets

$$\mathcal{K}_x := \{\varphi_x : \Omega \times \mathbb{R} \to \mathbb{R}^n \mid \varphi_x(x, t) \in \mathrm{dom}\,\eta^*, \ \forall(x, t)\}, \tag{5.9}$$

$$\mathcal{K}_t := \{\varphi_t : \Omega \times \mathbb{R} \to \mathbb{R} \mid -\varphi_t(x, t) \leq \rho(x, t), \ \forall(x, t)\}. \tag{5.10}$$

Again, $Dv$ represents the distributional derivative, and $\rho$ and $\eta$ denote the integrands of the data term and TV regularizer.

This leads us to the second step of the EmNeF approach, where we introduce three neural fields $F_{\Theta_v}^v, F_{\Theta_x}^{\varphi_x}$ and $F_{\Theta_t}^{\varphi_t}$ which accept coordinates $(x, t) \in \Omega \times \Gamma$ as input. The fields are to be understood as non-linear approximations of the primal and dual variables $v, \varphi_x, \varphi_t$. During "training", random points in the continuous domain $\Omega \times \Gamma$ are chosen and the parameters of the fields are iteratively updated such that the neural fields minimize/maximize the lifted energy (5.7). To this end, the energy (5.7) is incorporated in the loss functions according to which the parameters of the neural fields are updated. After the training (stochastic optimization of the parameters) is completed, each field can be understood as a

parameterized neural approximation of a primal/dual solution $v^*$, $\varphi_x^*$ and $\varphi_t^*$. The conceptual difference to the PDHG approach is that we obtain via *stochastic* optimization a *non-linear* neural approximation $\boldsymbol{F}_{\Theta_v}^v$ of the minimizer $v^*$ which is defined on the *continuous* domain $\Omega \times \Gamma$.

**Outline.**   In **Sec. 5.2**, we discuss some basic principles of machine learning from a high-level perspective and give a short introduction to the here used methods. In **Sec. 5.3**, we describe the architecture of the neural fields and demonstrate how the sets (5.8)–(5.10) can be incorporated as hard constraints into the design of the neural fields. We describe the training process as well as the inference and rendering steps. In **Sec. 5.4**, we compare the performance of the EmNeF approach to established approaches with respect to number of parameters, runtime and objective function value: In the non-convex case, we compare against a CUDA implementation [LM15] of the PDHG algorithm [Poc+09; CP11; Gol+13; FB18] using the sublabel-accurate discretization scheme [LM16; Möl+15]. In the convex case, we additionally compare against the established convex optimization toolbox CVXPY [Agr+18; DB16].

## 5.2  Introduction to the Learning Framework

In this section, we discuss the basic principles of machine learning required for the remainder of this chapter from a high-level perspective. For more in-depth discussions see for example [BN06; LBH15].

Neural networks are inspired by the biological nervous system. The human brain is believed to contain about 100 billion neurons which generate, receive, and conduct small electric signals, and as such process information [VBH16]. In the terminology of artificial neural networks, the neurons are analogous to the network's *nodes*, which are essentially non-linear function blocks. The *architecture* of the network describes which nodes are connected and how they interact with each other. One node can have multiple input and output connections which are equivalent to the biological structure of dendrites and axons. How exactly information is passed on between the nodes is modelled by the trainable parameters (weights) of the network.

**Neural Networks.** Mathematically, a neural network can be viewed as a family of functions $\boldsymbol{F}_\Theta : \mathbb{R}^n \to \mathbb{R}^d$, parameterized by a parameter vector $\Theta \in \mathbb{R}^p$. During the *training* process, this parameter

vector $\Theta$ is optimized to minimize a loss function $L(\Theta)$. The parameterized functions are typically composed from different *layers*. Each layer is a non-linear parameterized function $f_i : \mathbb{R}^{m_i} \to \mathbb{R}$,

$$f_i(x; \theta_i) = \sigma_i \left( \sum_{j=1}^{m_i} a_{i,j} x_j - b_i \right), \qquad (5.11)$$

where $\sigma_i$ is called *activation function* (see Tab. 5.1 for some examples) and $\theta_i = (a_{i,1}, \ldots, a_{i,m_i}, b_i)$ are the trainable *weights* (parameters of the network). Concatenating all layers and writing $\Theta = (\theta_1, \ldots, \theta_p)$, the full network is described by

$$\boldsymbol{F}_\Theta(x) := \boldsymbol{F}(x; \Theta) := f_p(\ldots f_2(f_1(x; \theta_1); \theta_2); \ldots \theta_p). \qquad (5.12)$$

**Some Examples.** Neither input nor output of a neural network are *per se* restricted to a specific data type and both are problem dependent. Let us consider three examples, which showcase the versatility of neural networks and emphasize the difference between *data-driven* approaches – which are trained on labeled data – and *model-driven* approaches – which incorporate (physical) knowledge about the solution:

1. One of the prototypical examples in learning is the problem of image *recognition and categorization*. A network is trained on a set of *labeled images* such that new, previously unseen instances belonging to the same group of classes can be labeled correctly.

   A famous example is the task of classifying handwritten digits in the MNIST dataset [LeC98]. The goal is to obtain a network that accepts an image of a single, handwritten digit as input and returns the digit as output. Such a network could be trained with the categorical cross-entropy loss function. Such data-driven approaches typically require a large amount of labeled data, since physical knowledge of the problem is missing. The MNIST dataset, for example, consists of 60,000 examples in the training set which amounts to 6,000 examples for each class.

2. Deep learning can also be used for medical applications such as *image registration*, see [HKY20] for an overview. The latter is the task of finding a deformation $\varphi$ which maps a template image $T$ into the coordinate system of a reference image $R$ so that $T(\varphi(x)) \approx R(x)$. The problem is also visualized in Fig. 1.1.

   For example, one can train a *convolutional neural network* (CNN) to predict the deformation $\varphi$ for given input images $T$ and $R$. The loss function can be an assembly of different penalty terms which either ensure the similarity

of $T(\varphi(x)) \sim R(x)$ or certain assumptions on $\varphi$. While the network is trained on a given data set, it also includes physical knowledge of the problem through the loss function and generalizes to previously unseen images of the same modality/type.

3. Neural networks can also be used for solving partial differential equations (PDEs), see [Cuo+22] for an overview. Consider for example the *heat equation*. In this scenario, a network could be trained to accept low-dimensional coordinate points from the integration domain as input and to predict the value of the solution at these points. The governing PDE, as well as the initial and boundary conditions are translated into a loss function optimization problem. The training process can be augmented with "data", for example if parts of the solution are already known, but is mainly model-driven through the governing PDE.

**Training.**   During the training process, one seeks to find "good" values for the parameters (also called *weights*) $\Theta$ with respect to a *loss function L*, which assesses the quality of the current weights. Consider for example the the *mean squared error* (MSE) in supervised training, where $k$ input samples $x = (x_1, ..., x_k)$, $x_i \in \mathbb{R}^n$ and their corresponding desired (ground truth) $y = (y_1, ..., y_k)$, $y_i \in \mathbb{R}^d$ are given:

$$L_{\mathrm{MSE}}(\Theta; x, y) := \frac{1}{k} \sum_{i=1}^{k} (F(x_i; \Theta) - y_i)^2 . \qquad (5.13)$$

After calculating the loss function for the current weights, the gradient of the loss function with respect to the weights is derived using automatic differentiation. The parameters $\Theta$ are then updated accordingly, in the most simple form using the classical *gradient descent* (GD) algorithm

$$\Theta^{k+1} = \Theta^k - \alpha^k \nabla_\Theta L(\Theta^k; x, y), \qquad (5.14)$$

where $\alpha^k$ is the (adaptive) *stepsize*. More widely used in practice is the *stochastic gradient descent* (SGD) algorithm, which updates the weights in (5.14) by approximating the gradient $\nabla_\Theta L$ in each iteration over a randomly chosen subset of $(x, y)$ [Bot12]. Its use is motivated by the high cost of running the backpropagation over the full training set. An overview over more refined stochastic optimization approaches such as Adam [KB14] can be found in [Rud16].

Even if the chosen loss function is convex with respect to the networks output, it is typically *non*-convex with respect to the networks parameters $\Theta$ due to the non-linearities $\sigma_i$ in the network.
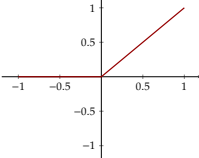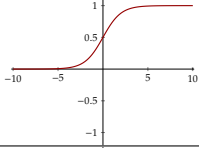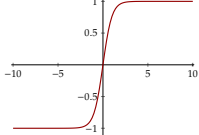
Therefore, the problem

$$\inf_{\Theta \in \mathbb{R}^q} L(\Theta; \boldsymbol{x}, \boldsymbol{y}) \tag{5.15}$$

is typically non-convex, which makes the formulation of convergence results for classical gradient-based optimization strategies tricky. Current research is concerned with the question of which characteristics make certain global minimizers more attracting than others for certain optimization algorithms [MMS21; CB20; Gun+18].

In the following subsections, we discuss three concepts that will be of particular relevance in the design of our approach:

▶ *Generative adversarial networks* (GANs): The term describes a competitive learning framework in which two networks are trained against each other. In the usual use case, one network (the *generator*) tries to create new images which are similar to images in a given training set, while the other network (the *discriminator*) tries to discriminate these newly generated images from original (non-generated) images. The setting can be viewed as a game, since the cost of each player (network) is influenced by the other player's parameters, but the players can only control their own parameters.
**In the EmNeF approach, we also optimize the parameters of multiple networks. While one network learns to minimize a given loss function, other networks learn to maximize it.**

▶ *Neural fields* (NF) and *physics-informed neural networks* (PINN): Both terms are closely related and describe a certain type of network which accepts typically low-dimensional coordinate points as input and has a PDE or variational model as a loss function. After the training process, the network can be understood as a continuous, parameterized solution of the PDE or variational model it was trained with. The inference (forward) step can be thought of as discretization/evaluation of the solution.
**In the EmNeF approach, we make use of the properties of neural fields in order to avoid the problems typically encountered during discretization and optimization of the calibration-based lifted energies.**

▶ *Positional encoding*: Positional encoding is a successful preprocessing step applied in the context of neural fields. It maps the low-dimensional input coordinates to a higher-dimensional space and speeds up learning solutions with high-frequency content. While thematically subordinate to neural fields, positional encoding is a central part of the EmNeF approach.

**Table 5.1: Activation Functions** are often non-linear and sometimes even non-smooth. Together with linear weighting, they allow networks to learn highly non-trivial tasks.

| Name | Function | Plot |
|---|---|---|
| ReLU | $\sigma(x) = \begin{cases} 0, & \text{if } x \leq 0, \\ x, & \text{else.} \end{cases}$ |  |
| Sigmoid | $\sigma(x) = \frac{1}{1+\exp(-x)}$ |  |
| Tanh | $\sigma(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$ |  |

## 5.2.1 Generative Adversarial Networks (GANs)

*Generative adversarial models* fall under the relative broad umbrella term of *generative modeling*. In the tutorial [Goo16], a soft introduction to generative modeling and the related taxonomy can be found. We here follow the tutorial and introduce some of the important concepts.

**Generative Modeling.** This describes the task of learning an – either implicit or explicit – approximation $p_{\text{model}}$ of the true probability distribution $p_{\text{data}}$ of a given data set. If the goal is an *explicit approximation* of the probability distribution, the trained model (network) can be understood as a parameterized representation of the approximated probability distribution. In case of an *implicit approximation*, the model provides an indirect way of interacting with the approximated probability distribution, e.g., by drawing samples from it. In this case, the network maps some latent variable (which can be in the same or in a different space then the training data) and returns a sample drawn from the approximated probability distribution (see "Generator" in Fig. 5.1).

**Generative Adversarial Models.** The goal of *generative adversarial networks* (GANs) [Goo+14; Goo+20; Gui+23] is to generate plausible new data after looking at training examples that is to learn the probability distribution implicitly.

Consider a real-life example: Both the art connoisseur and the art forger study famous art pieces in order to accumulate knowledge. The art forger uses the acquired knowledge in order to create new art pieces which resemble the work of some famous artist. The art connoisseur uses his knowledge to recognize pieces of some famous artist. The more attention the pieces of the art forger get

and the better educated the audience is, the higher is the incentive for the art forger to work on the plausibility of his work.

GANs use the concept of adversaries in a learning framework. Two networks are trained simultaneously. The *generator* $G_{\Theta_G} : \mathbb{R}^z \to \mathbb{R}^n$ is trained to generate new, plausible samples of the estimated probability distribution $p_{\text{model}}$. The *discriminator* $D_{\Theta_D} : \mathbb{R}^n \to [0, 1)$ acts as an antagonist which is trained to discriminate real from fake samples of the true probability distribution $p_{\text{data}}$ (see Fig.5.1). Mathematically, $D_{\Theta_D}$ and $G_{\Theta_G}$ are antagonistic players in a game; both networks influence each other, but each network has its own set of parameters.

Typically, the cost

$$-\frac{1}{2}\mathbb{E}_{x \sim p_{\text{data}}} \log(D_{\Theta_D}(x)) - \frac{1}{2}\mathbb{E}_z \log(1 - D_{\Theta_D}(G_{\Theta_G}(z))) \qquad (5.16)$$

is used for the discriminator, while different choices for the cost of the generator are explored in the literature, such as the negative cost of the discriminator or

$$\frac{1}{2}\mathbb{E}_z \log(1 - D_{\Theta_D}(G_{\Theta_G}(z))). \qquad (5.17)$$

The interested reader can find an overview and discussion of alternative choices in [Goo16; Gui+23].

During training, the parameters of the networks are updated alternately. Denoting the respective loss functions by $L^D$ and $L^G$, the (adaptive) step sizes by $\alpha^k$ and $\beta^k$, and the sampled batches by $x$ and $z$, this alternating update can be specified as:

$$\Theta_D^{k+1} := \Theta_D^{k+1} - \alpha^k \nabla_{\Theta_D} L^D(\Theta_D^k; \Theta_G^k, x, z), \qquad (5.18)$$

$$\Theta_G^{k+1} := \Theta_G^{k+1} - \beta^k \nabla_{\Theta_G} L^G(\Theta_G^k; \Theta_D^{k+1}, x, z). \qquad (5.19)$$

In the special case where the negative cost of the discriminator is used as cost for the generator ($L^G = -L^D$) this is conceptually similar to the PDHG algorithm: The parameters $\Theta_D$ and $\Theta_G$ are updated alternately in order to minimize/maximize the same loss function.

**Adaptation to our Work.**   Motivated by the success of such approaches, we design and train multiple neural networks, representing the primal and dual variables of the lifted energy (5.7), which are then optimized in a minimax fashion.

We are not the first to borrow ideas from the GAN approach in order to solve primal-dual energies. We would like to especially
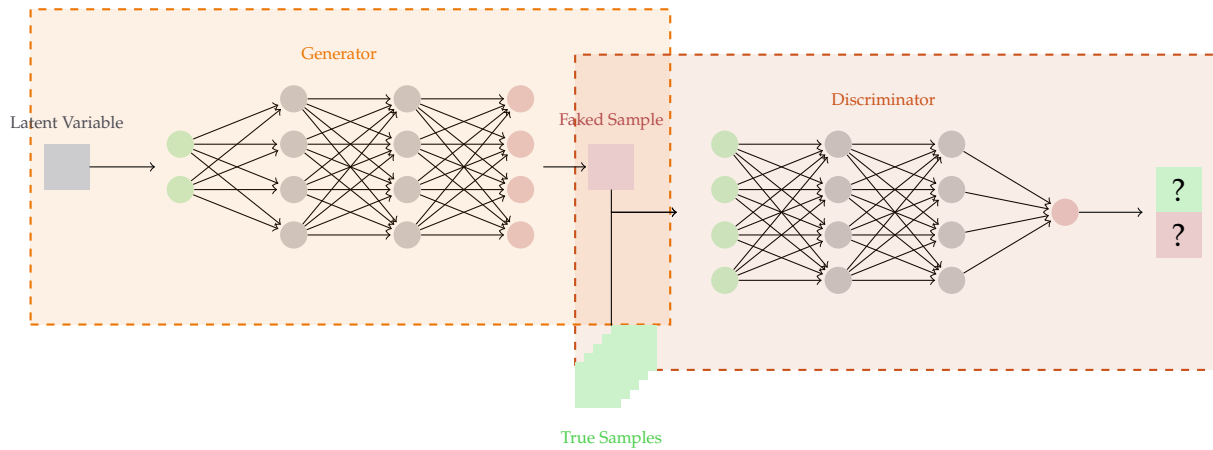
**Figure 5.1: Generative Adversarial Network.** The generator learns the distribution of the training data (green squares) and generates new samples (red squares). The discriminator learns to estimate the probability of whether a given image is a real or generated sample.

point out [MC19a], which addresses the problem of generative modeling – but instead of considering the input data to be samples from a probability distribution ("0-current"), the authors argue that perceiving them in terms of *k-currents* [Mor16, Chpt. 4] allows to incorporate available geometric information into the model: the given data can be oriented with respect to a latent variable which represents for example time (in case of a video sequence) or rotation (in case of handwritten digits). Using this current-based lifting approach, the authors formulate a primal-dual energy which they solve by training multiple networks in a GAN-inspired manner.

### 5.2.2　Coordinate-Based Neural Networks

As the name suggests, *coordinate-based neural networks* accept (often low-dimensional) coordinate points as input and predict some value at these points. Two recent and popular examples of coordinate-based neural networks are *physics-informed neural networks* (PINNs), which were first proposed in [RPK19], and *neural fields* (NFs), see [Xie+22] for a broad overview and introduction. While the term physics-informed neural network is used in the context of partial differential equations (PDEs), the term neural field is typically associated with imaging problems. Both frameworks, however, share a lot in common.

**Physics-Informed Neural Networks.**　PINNs combine deep learning frameworks with PDEs and constrain neural networks by including prior (physical) knowledge of the problem at hand. [Cai+21] demonstrate how PINNs can be used for solving ill-posed problems. As a classical example, consider the *heat equation* [Cai+21].

For initial condition $h$ and boundary condition $g$, the heat equation is given as:

$$\frac{\partial}{\partial t} u(x,t) = \Delta_x u(x,t), \quad t \in [0,T], \ x \in \Omega, \qquad (5.20)$$

$$u(x,0) = h(x), \quad x \in \Omega, \qquad (5.21)$$

$$u(x,t) = g(x), \quad t \in [0,T], \ x \in \partial\Omega. \qquad (5.22)$$

The goal is to train a neural network that accepts low-dimensional coordinates $(x,t) \in \bar{\Omega} \times [0,T]$ as input and returns the value of the solution $u$ at the given points. In a *fully model-driven* approach, the loss function is chosen such that it penalizes the residual of the governing PDE (5.20)

$$L^{PDE}(\Theta; \boldsymbol{x}, \boldsymbol{t}) := \frac{1}{k} \sum_{i=1}^{k} \left( \frac{\partial}{\partial t} F(x_i, t_i; \Theta) - \Delta_x F(x_i, t_i; \Theta) \right)^2. \quad (5.23)$$

The initial (5.21) and boundary (5.22) conditions can either be similarly included in the loss function as *soft constraints* or they can be enforced as *hard constraints* in the output layer of the network [Wu+23]. If the solution $u$ is known at certain interior points, this information can be included in the loss function as well by adding an additional term such as the mean squared error (MSE). In [Cai+21] it is shown that such a *partially data-driven* approach can lead to superior results compared to the fully model-driven approach.

The network is trained on *batches* of coordinate points. These batches can be sampled grid-free and randomly, however, it can be beneficial to give special attention to boundary points or to areas of interest [Wu+23]. A PINN, therefore, learns a parameterized *neural representation* of the solution of the PDE in a grid-free manner. Differential terms in the PDE can either be accounted for by sampling neighboring "grid" points with a fixed distance to the randomly sampled points, and using *finite differences* (FD) when calculating the loss function or by applying *automatic differentiation* (AD) frameworks [Cai+21]. While AD-reliant approaches prevail in literature, it has been observed experimentally that incorporating FD-terms in the loss function can lead to better results – although the theory behind the better results is still unknown [LDR22].

Once a PINN is trained, it can easily be evaluated with arbitrary resolution. This step is also called *inference*. Note, that the approach also comes with some caveats: Since the spatial points are randomly sampled during training, the probability of sampling the location of a singularity during the training process is zero. Also, it is important to sample a good ratio of boundary and inner points in order to avoid exploding gradients. And, last but not least, due to

non-convexity associated to most learning problems, convergence can be a problem.

**Neural Fields.**   The term *neural field* was coined by the imaging community and a summary of related terms and topics, as well as an extensive literature review about NFs can be found in [Xie+22]. A *field* here refers to a physical quantity that is defined over coordinates. *Neural fields* parameterize such a field. They are coordinate-based neural networks, which map a coordinate point to a scalar or vectorial quantity.

Neural fields have proven especially powerful for the task of novel view synthesis. For this task, the available training data consists of various 2D images taken from a 3D scene. The input images are randomly sampled on a hemisphere surrounding the scene and saved together with the information of the current camera position and angle. The goal is to generate new 2D images of the 3D scene from previously unseen angles and positions. *Neural radiance fields* (NeRFs) [Mil+21] have proven to be especially successful in this task. They accept 5D input coordinates – three of them describing the 3D spatial location $(x, y, z)$ and two of them describing the viewing direction $(\theta, \psi)$ – and output 4D information of the volumetric field – three for the color information $(r, g, b)$ and one for the density $\alpha$. In order to *render* a synthetic 2D image from this volumetric field, one needs to think of a camera ray travelling through the scene. The ray is discretized and the discrete coordinate points $(x_i, y_i, z_i)$ together with the viewing direction $(\theta, \psi)$ of the ray are passed to the NeRF. The NeRF produces for each 5D $(x_i, y_i, z_i, \theta, \psi)$ point a prediction of color $(r_i, g_i, b_i)$ and density $(\alpha_i)$. The information along one ray is then accumulated as color information $(R, G, B)$ corresponding to one pixel in the synthetic 2D image.

While NeRFs are currently most prominent in the field of computer vision, applications of neural fields are diverse and include inverse problems in *cryogenic electron microscopy* (Cryo-EM) [Zho+19] and surface reconstruction problems with explicit boundary curves [Pal+22].

**Adaptation to our Work.**   Our EmNeF approach is inspired by [Pal+22], where the authors consider the problem of surface reconstruction for explicitly given boundaries. Starting with Plateau's minimal surface problem, the authors embed the problem in the space of *k-currents* [Mor16, Chpt. 4] and approximate the problem of the embedded problem by training a neural field. Similarly, we aim to solve variational problems (5.1) by first embedding the problem using calibration-based lifting and then finding a non-linear approximation of the solution to the lifted problem

by stochastically optimizing the parameters of the neural fields. Neural fields convince through their expressive power. With this approach we hope to reduce the number of required parameters and to simplify the implementation.

### 5.2.3 Positional Embedding

It has been observed that neural networks learn low-frequency components in the solution faster than high frequency components. This behavior is referred to as *frequency bias* [Bas+20] or *spectral bias* [Rah+19]. As the distribution of the training data influences the spectral bias [Rah+19; Bas+20; Tan+20; ZRL21], manipulating said distribution can accelerate the training process. In [Rah+19] the authors demonstrate that components with high frequency are learnt faster if the input data lies in a low-dimensional, complexly shaped manifold of a high-dimensional input space. This is especially interesting with respect to neural fields; In comparison to other neural networks the input space of neural fields is often low-dimensional and rather *densely* distributed.

In order to accelerate the training process *positional embedding* strategies have been suggested; The idea is to define a function $\gamma : \mathbb{R}^n \to \mathbb{R}^N, n \ll N$, which maps low-dimensional input data, e.g., $x \in \mathbb{R}^2$, into a higher-dimensional space, e.g., $\gamma(x) \in \mathbb{R}^{1024}$ and which simultaneously makes the training data *more sparsely* distributed.

A comparison of different positional embedding strategies can be found in [ZRL21]. The authors argue that distance preservation between original and embedded space is important for generalization and that a high matrix rank of the embedded representation across positions causes better memorization of the training data. One of the discussed strategies is the *random Fourier features* (RFF) encoding which we use in the EmNeF approach.

**Random Fourier Features.** *Random Fourier features* have their origin in the *natural language processing* community [Xu+19] and were studied in the context of neural fields in [Tan+20; ZRL21]. The RFF encoding is defined as a map $\gamma : [0,1)^n \to \mathbb{R}^{2m}$ of the form

$$\gamma(x) := \begin{pmatrix} a_1 \cos(2\pi b_1^\top x) \\ a_1 \sin(2\pi b_1^\top x) \\ \vdots \\ a_m \cos(2\pi b_m^\top x) \\ a_m \sin(2\pi b_m^\top x) \end{pmatrix}, \tag{5.24}$$
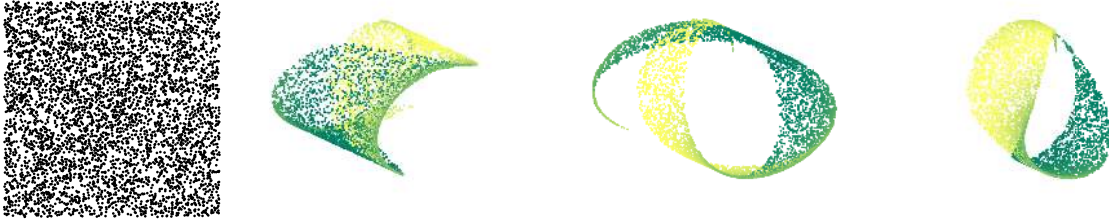
**Figure 5.2: RFF Encoding.** Positional encoding strategies are preprocessing steps used for coordinate-based neural networks. They increase the dimensionality of the input, which improves the network's capability of learning high-frequency components and leads to an acceleration of the training procedure. In the RFF encoding (5.24), randomly chosen spatial coordinates are projected onto higher-dimensional hyperspheres. Here, the sampled coordinates are in $[0, 1)^2$ (leftmost), and three possible projections for randomly chosen $b_1, b_2 \in \mathcal{N}(0, \sigma^2)$ as well as $a_1 = a_2 = 1$ are shown. The forth dimension of the projected coordinates is color-coded.

where the parameters $a_i \in \mathbb{R}$ are randomly chosen and the vectors $b_i \in \mathbb{R}^n$ are randomly sampled from the normal distribution $\mathcal{N}(0, \sigma^2)$. The RFF encoding can be considered as a preprocessing step; Before being passed to the trainable layers of the network, an input $x \in [0, 1)^n$ is replaced by its higher-dimensional representation $\gamma(x) \in \mathbb{R}^{2m}$ (see Fig. 5.2).

## 5.3 EmNeF

The idea of the EmNeF approach is to "learn" a parametric nonlinear approximation of the primal $v^* \in \mathscr{C}$ and dual $\varphi_x^* \in \mathscr{K}_x$, $\varphi_t^* \in \mathscr{K}_t$ solutions of the problem (5.6) by stochastically optimizing multiple neural fields.

We introduce three neural fields $F_{\Theta_v}^v : \Omega \times \mathbb{R} \to [0, 1]$, $F_{\Theta_x}^{\varphi_x} : \Omega \times \mathbb{R} \to \mathbb{R}^n$ and $F_{\Theta_t}^{\varphi_t} : \Omega \times \mathbb{R} \to \mathbb{R}$ which are parameterized by trainable weights $\Theta_v \in \mathbb{R}^{q_v}$, $\Theta_x \in \mathbb{R}^{q_x}$ and $\Theta_t \in \mathbb{R}^{q_t}$. We stochastically optimize these parameters for the problem

$$\inf_{\Theta_v \in \mathbb{R}^{q_v}} \sup_{\substack{\Theta_x \in \mathbb{R}^{q_x}, \\ \Theta_t \in \mathbb{R}^{q_t}}} \mathscr{F}\left(F_{\Theta_v}^v, F_{\Theta_x}^{\varphi_x}, F_{\Theta_t}^{\varphi_t}\right). \qquad (5.25)$$

As the primal and dual variables play contrary roles in minimizing/maximizing the primal-dual energy, the neural fields are later "trained against each other", which is reminiscent of the generator and discriminator in the GAN setting. Our approach is visualized in Fig. 5.3–5.4.

In the remainder of this section, we explain the architecture of the neural fields, the implementation of the constraint sets, the stochastic optimization of the parameters and the inference process.
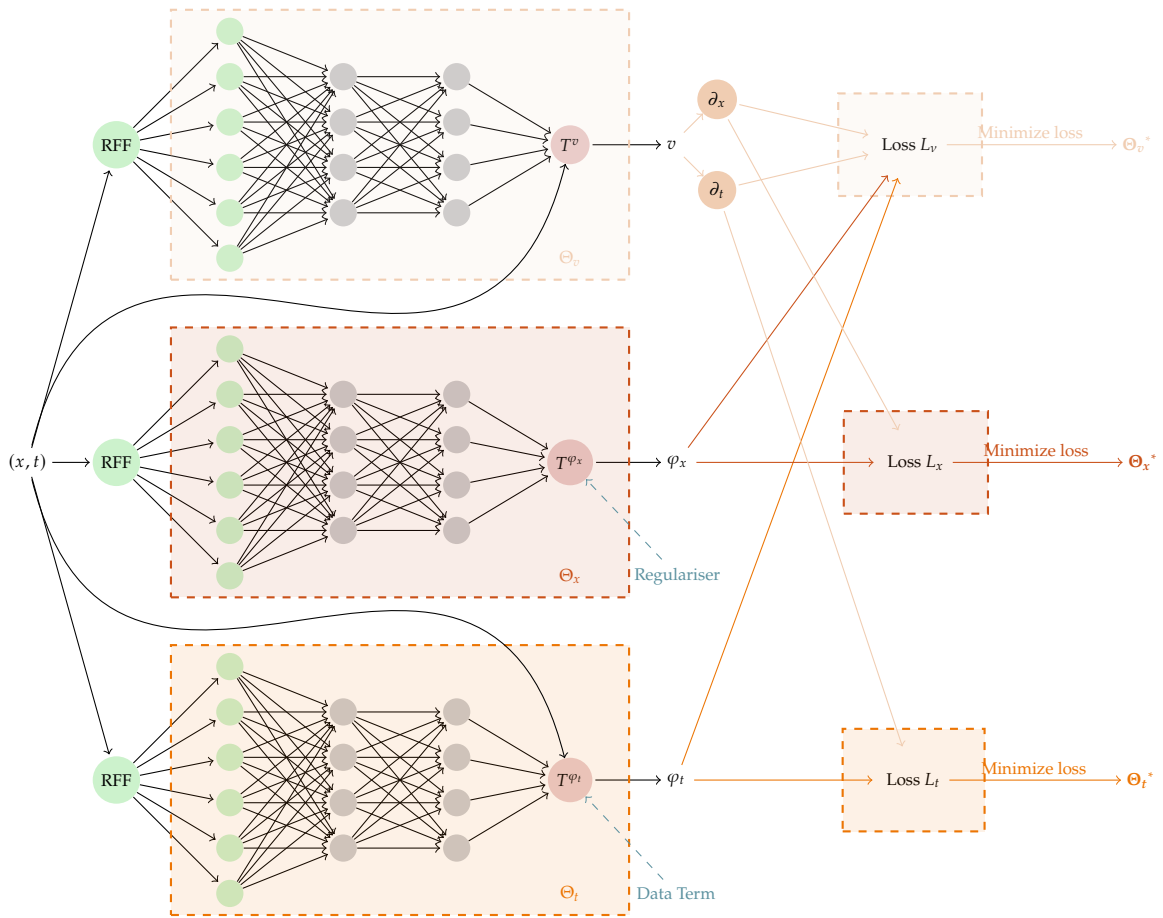
**Figure 5.3: EmNeF Architecture.** EmNeFs offer a unified strategy for solving problems of the form (5.1) building on the theory of calibration-based lifting. Instead of solving the lifted energy (5.7)-(5.10) using one of the discretization approaches discussed in Sec. 3.3 and the gradient-descent based PDHG algorithm as discussed in Sec. 3.4, we suggest solving the primal-dual problem with the help of neural fields. After the parameters have been optimized, the neural fields represent a non-linear parametric approximation of the solutions $v^*$, $\varphi_x^*$ and $\varphi_t^*$. The procedure of optimizing the parameters resembles the generative-adversarial networks (GAN) setting where multiple networks are trained against each other. The EmNeF approach is easy to implement.
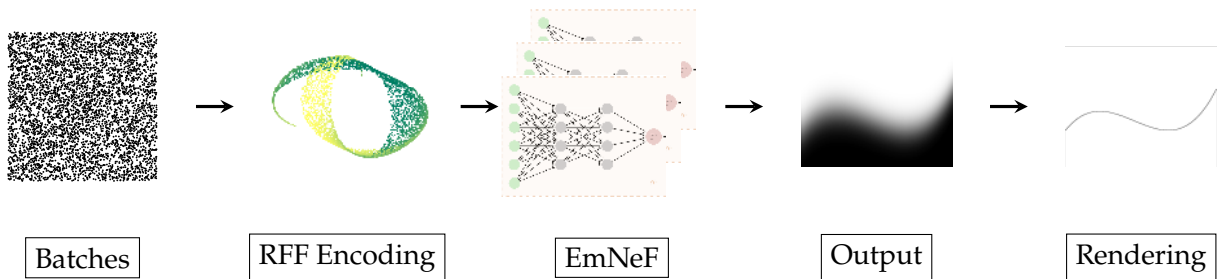


**Figure 5.4: EmNeF.** The EmNeF approach can be used for solving variational problems of the form (5.1). In this example we have $\Omega, \Gamma \subset \mathbb{R}^1$. During training, random coordinate points in $\Omega \times \Gamma$ are sampled and passed though an RFF encoding into the EmNeF. The parameters of the primal/dual neural fields are iteratively updated to minimize/maximize the energy (5.6). At the end of the training procedure, ideally, the three networks have learned an approximation of the optimal primal/dual variables. After the training is complete, a suitably fine grid can be chosen in order to infer the lifted solution $v^*$ with optimal resolution. Thresholding of $v^*$ results in an approximate minimizer $u^*$ of (5.1).

### 5.3.1 Architecture

All three neural fields $F_{\Theta_v}^v$, $F_{\Theta_x}^{\varphi_x}$ and $F_{\Theta_t}^{\varphi_t}$ have the same general architecture of one input, two hidden, and one output layer, see Fig. 5.5. Let us discuss these three main blocks:

**Input Layer.**   The input layer implements the random Fourier feature encoding defined in (5.24). The RFF parameters $a_i$, $b_i$, and $\sigma^2$ are typically manually chosen rather than trained. Consequently, this layer has no trainable parameters. We use it as a preprocessing step in order to improve the neural fields capability to learn high-frequency representation [Tan+20].

In our experiments, the RFF encoding increases the dimension of inputs $(x, t) \in \mathbb{R}^3$ from $n + 1 = 3$ to $2m = 2^{11}$. We set $a_i = 1$ and use $m$ random coefficients $b_i \in \mathbb{R}^{n+1}$ drawn from the multivariate normal distribution $\mathcal{N}(0, 2\pi\sigma^2 I)$, with variance $\sigma^2$. This means that the RFF encoding reduces for the notation $\boldsymbol{x} = (x, t)$ to $\gamma : \mathbb{R}^{n+1} \to \mathbb{R}^{2m}$,

$$\gamma(\boldsymbol{x}) := \left( \cos(b_1^\top \boldsymbol{x}), \sin(b_1^\top \boldsymbol{x}), ..., \cos(b_m^\top \boldsymbol{x}), \sin(b_m^\top \boldsymbol{x}) \right). \quad (5.26)$$

The choice of the variance $\sigma^2$ has a notable impact on the results and will be discussed further in Sec. 5.4.

**Hidden Layers.**   Each hidden layer has the same number of neurons – in our examples $2^5$ or $2^6$ – and we use ReLU activation functions for $F_{\Theta_v}^v$ and Tanh activation functions for $F_{\Theta_x}^{\varphi_x}$ and $F_{\Theta_t}^{\varphi_t}$ as shown in Fig. 5.5. For now, the choice of activation function is based on first experiments and observations; a more thorough theoretical investigations is desirable in the future.

**Last Layer.**   The last layer of the neural fields is especially important: By choosing suitable activation and transformation functions, the constraints imposed by $\mathscr{C}$, $\mathscr{K}_x$, and $\mathscr{K}_t$ can be enforced as hard constraints, as will be discussed in the next section.

### 5.3.2 Hard Constraints

In Sec. 5.2.2, we have discussed the implementation of constraints in the context of PINNs: Constraints can either be implemented as soft constraints by adding a Lagrangian multiplier to the loss function, or as hard constraints by customizing a network's output layer.
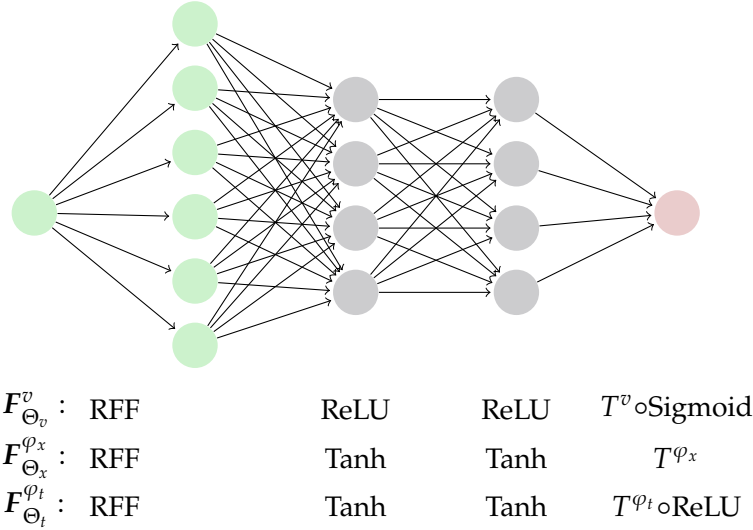
$$F^v_{\Theta_v} : \quad \text{RFF} \qquad\qquad \text{ReLU} \qquad \text{ReLU} \qquad T^v\circ\text{Sigmoid}$$
$$F^{\varphi_x}_{\Theta_x} : \quad \text{RFF} \qquad\qquad \text{Tanh} \qquad \text{Tanh} \qquad T^{\varphi_x}$$
$$F^{\varphi_t}_{\Theta_t} : \quad \text{RFF} \qquad\qquad \text{Tanh} \qquad \text{Tanh} \qquad T^{\varphi_t}\circ\text{ReLU}$$

**Figure 5.5: Network architecture.** The three neural fields $F^v_{\Theta_v}$, $F^{\varphi_x}_{\Theta_x}$, and $F^{\varphi_t}_{\Theta_t}$ share the same general architecture: Inputs to the neural fields are preprocessed using the RFF encoding in order to allow the neural fields to better learn high-frequency solutions. The RFF encoding increases the dimensionality of the inputs for example from 3 to $2^{11}$. It has no trainable parameters and we, therefore, consider it as part of the input layer. All three neural fields have two hidden layers with the same width (in our experiments $2^5$ or $2^6$), however, the activation functions are chosen differently as indicated in the figure. The output layers of the neural fields play a special role, as they are used to enforce the constraints imposed by $\mathscr{C}$, $\mathscr{K}_x$ and $\mathscr{K}_t$ as hard constraints. This is done by customizing the activation functions as indicated in the figure. The definition of the transformations $T^v$, $T^{\varphi_x}$ and $T^{\varphi_t}$ is given in (5.27), (5.34), and (5.36).

In this work, we implement the constraints $\mathscr{C}$, $\mathscr{K}_x$, and $\mathscr{K}_t$ as hard constraints. The reasons are twofold. First, soft constraints would require adding suitable Lagrangian multipliers, which would result in a more complicated primal-dual structure and require training/tuning additional variables. Second, it has been observed in the context of PINNs that hard constraints lead to superior results compared to soft constraints [Wu+23].

**Primal Constraint $\mathscr{C}$.** According to the constraint set $\mathscr{C}$ (5.8), the primal neural field $F^v_{\Theta_v}$ is required to assume values in $[0, 1]$ and to meet the boundary conditions.

We employ the sigmoid activation function (see Table 5.1), which guarantees values in $[0, 1]$, and define an additional pointwise transformation $T^v : [0, 1] \times \mathbb{R} \rightarrow [0, 1]$ in order to ensure that the boundary constraints are satisfied:

$$T^v(y, t) := \begin{cases} 1, & \text{if } t \leq \Gamma_{\min}, \\ 0, & \text{if } t \geq \Gamma_{\max}, \\ y, & \text{otherwise.} \end{cases} \tag{5.27}$$

Denoting the sigmoid function as $\sigma_S$, the last layer is completely described with

$$T^v \circ \sigma_S. \tag{5.28}$$

**Dual Constraint** $\mathcal{K}_x$**.** The dual constraint set $\mathcal{K}_x$ in (5.9) is determined by the choice of the regularizer. In our experiments, we use the total variation regularizer for which the integrand can be written as

$$\eta(p) = \lambda \|p\|_2, \tag{5.29}$$

where $\lambda > 0$ is a regularization parameter. For $\lambda = 1$, the (pointwise) Fenchel conjugate is calculated as

$$\eta^*(q) = \sup_{p \in \mathbb{R}^n} \langle p, q \rangle - \|p\|_2 = \begin{cases} 0, & \text{if } \|q\|_2 \leq 1, \\ \infty, & \text{else.} \end{cases} \tag{5.30}$$

In order to account for the regularization parameter, we use the conjugacy rule [RW09, p. 475]

$$\lambda f(p) \quad \overset{*}{\longleftrightarrow} \quad \lambda f^*(\lambda^{-1} v). \tag{5.31}$$

This means that for arbitrary $\lambda > 0$ the (pointwise) Fenchel conjugate of the total variation regularizer is given by

$$\eta^*(q) = \begin{cases} 0, & \text{if } \|q\|_2 \leq \lambda, \\ \infty, & \text{else.} \end{cases} \tag{5.32}$$

Therefore, we can rewrite the constraint set $\mathcal{K}_x$ in (5.9) as

$$\mathcal{K}_x = \{\varphi_x : \Omega \times \mathbb{R} \to \mathbb{R}^n \mid \|\varphi_x(x, t)\|_2 \leq \lambda, \ \forall (x, t)\}. \tag{5.33}$$

Consequently, we need to restrict the output of $F_{\Theta_x}^{\varphi_x}$ pointwise to a ball with radius $\lambda$. We define a projection $T^{\varphi_x} : \mathbb{R}^n \to \mathcal{B}_\lambda$ onto the $n$-dimensional scaled unit ball with radius $\lambda$:

$$T^{\varphi_x}(y) := \begin{cases} \lambda \dfrac{y}{\|y\|_2}, & \text{if } \frac{1}{\lambda} \|y\|_2 > 1, \\ y, & \text{otherwise.} \end{cases} \tag{5.34}$$

Note that this projection is not differentiable. This is theoretically problematic during the optimization process of the network's parameters; in practice it appears to work well, similarly to the ubiquitously used (and equally non-differentiable) ReLU nonlinearity. The transformation $T^{\varphi_x}$ completely describes the last layer.

**Dual Constraint** $\mathcal{K}_t$**.** The dual constraint set $\mathcal{K}_t$ (5.10) is largely determined by the integrand $\rho$ of the data term: The negative output of the dual neural field $F_{\Theta_t}^{\varphi_t}$ needs to be pointwise bounded from above by the data. We employ the ReLU activation function (see Table 5.1), which guarantees values in $[0, +\infty)$. Denoting

the output of the dual field right after application of the ReLU activation function as $\tilde{\boldsymbol{F}}_{\Theta_t}^{\varphi_t}$, it holds

$$\tilde{\boldsymbol{F}}_{\Theta_t}^{\varphi_t} \geq 0 \quad \Leftrightarrow \quad \tilde{\boldsymbol{F}}_{\Theta_t}^{\varphi_t} - \rho \geq -\rho \quad \Leftrightarrow \quad -(\tilde{\boldsymbol{F}}_{\Theta_t}^{\varphi_t} - \rho) \leq \rho. \quad (5.35)$$

Therefore, we define $T^{\varphi_t} : [-1, 1] \times \Omega \times \mathbb{R} \to \mathbb{R}$ pointwise as

$$T^{\varphi_t}(y, x, t) := y - \rho(x, t). \quad (5.36)$$

Denoting the ReLU activation function by $\sigma_R$, the last layer of the dual neural field $\boldsymbol{F}_{\Theta_t}^{\varphi_t}$ is completely described with

$$T^{\varphi_t} \circ \sigma_R. \quad (5.37)$$

**A Note on Implementation.**   As the constraints on $\boldsymbol{F}_{\Theta_x}^{\varphi_x}$ and $\boldsymbol{F}_{\Theta_t}^{\varphi_t}$ hold the complete information on the variational problem, implementing a new problem is as easy as rewriting the implementation of these constraints. Notably, only the integrand of the original, low-dimensional data term and the (pointwise) Fenchel conjugate of the integrand of the original, low-dimensional regularizer are needed and no further knowledge of the CBL embedding approach is required. This makes the EmNeF approach easy to implement, flexible, and more accessible than the sublabel-accurate discretization discussed in Sec. 3.3.

### 5.3.3 Training

While we use the word "training", the process can be thought of as stochastically optimizing the parameters of the neural fields. During training, we minimize the expected value of the integrand of $\mathcal{F}$ in (5.7), i.e.,

$$\mathbb{E}_{(x,t)} \left( \boldsymbol{F}_{\Theta_x}^{\varphi_x}(x, t) \nabla_x^h \boldsymbol{F}_{\Theta_v}^{v}(x, t) + \boldsymbol{F}_{\Theta_t}^{\varphi_t}(x, t) \nabla_t^h \boldsymbol{F}_{\Theta_v}^{v}(x, t) \right), \quad (5.38)$$

where $(x, t)$ are assumed to be uniformly distributed, and where we use the notation $\nabla_x^h \boldsymbol{F}_{\Theta_v}^{v}$, $\nabla_t^h \boldsymbol{F}_{\Theta_v}^{v}$ for the components of the distributional derivative $D\boldsymbol{F}_{\Theta_v}^{v}$.

**Batches.**   Instead of training the neural fields on a fixed, discrete grid $\Omega^h \times \Gamma^h$, we use in each training step $k$ randomly sampled coordinate points $X := \{(x_i, t_i) \mid (x_i, t_i) \in \Omega \times \Gamma_{\text{ext}}\}_{i=1}^{k}$ from a uniform distribution over the continuous domain $\Omega \times \Gamma_{\text{ext}}$ (see Fig. 5.6). Here, $\Gamma_{\text{ext}}$ denotes an interval which is slightly larger than $\Gamma$, i.e.,

$$\Gamma_{\text{ext}} := [\Gamma_{\min} - \Delta_\Gamma, \Gamma_{\max} + \Delta_\Gamma] \quad (5.39)$$

for some $\Delta_\Gamma > 0$. As the primal variable is required to be constant for $t \leq \Gamma_{\min}$ or $t \geq \Gamma_{\max}$, see (5.8), $\Omega \times \Gamma$ is the most interesting area. We sample from $\Omega \times \Gamma_{\text{ext}}$ in order to ensure that enough relevant points are sampled while also taking into account the constraint. In the terminology of neural networks, these random points $X$ are the *batches* the fields are trained on.

**Loss Functions.** Replacing the functions $v, \varphi_x, \varphi_t$ in (5.6)–(5.7) with the neural fields yields (again, informally, without considering any function spaces)

$$\inf_{\Theta_v \in \mathbb{R}^{q_v}} \sup_{\Theta_x \in \mathbb{R}^{q_x}, \Theta_t \in \mathbb{R}^{q_t}} \mathcal{F}\left(F_{\Theta_v}^v, F_{\Theta_x}^{\varphi_x}, F_{\Theta_t}^{\varphi_t}\right), \tag{5.40}$$

where

$$\mathcal{F}\left(F_{\Theta_v}^v, F_{\Theta_x}^{\varphi_x}, F_{\Theta_t}^{\varphi_t}\right) = \int_{\Omega \times \mathbb{R}} \left\langle \begin{pmatrix} F_{\Theta_x}^{\varphi_x} \\ F_{\Theta_t}^{\varphi_t} \end{pmatrix}, DF_{\Theta_v}^v \right\rangle. \tag{5.41}$$

In each training step, the parameters are updated successively by performing a gradient descent step on the parameters $\Theta_v$ and gradient ascent steps on the parameters $\Theta_x$ and $\Theta_t$ with respect to (5.41).

Given a batch $X := \{(x_i, t_i) \mid (x_i, t_i) \in \Omega \times \Gamma_{\text{ext}}\}_{i=1}^k$ as described above, the loss functions that the individual fields are trained on are as follows:

$$L_v(\Theta_v; X) = \frac{1}{k} \sum_{i=1}^k F_{\Theta_x}^{\varphi_x}(x_i, t_i)^\top \nabla_x^h F_{\Theta_v}^v(x_i, t_i)$$
$$+ F_{\Theta_t}^{\varphi_t}(x_i, t_i) \nabla_t^h F_{\Theta_v}^v(x_i, t_i), \tag{5.42}$$

$$L_x(\Theta_x; X) = \frac{1}{k} \sum_{i=1}^k - F_{\Theta_x}^{\varphi_x}(x_i, t_i)^\top \nabla_x^h F_{\Theta_v}^v(x_i, t_i), \tag{5.43}$$

$$L_t(\Theta_t; X) = \frac{1}{k} \sum_{i=1}^k - F_{\Theta_t}^{\varphi_t}(x_i, t_i) \nabla_t^h F_{\Theta_v}^v(x_i, t_i). \tag{5.44}$$

The gradients $\nabla_x^h F_{\Theta_v}^v$ and $\nabla_t^h F_{\Theta_v}^v$ can either be implemented with the help of finite differences or with automatic differentiation. While the latter approach prevails in literature, it has experimentally been observed in [LDR22] that the former can lead to better numerical results. Therefore we chose to implement the gradients $\nabla_x^h F_{\Theta_v}^v$ and $\nabla_t^h F_{\Theta_v}^v$ with the help of finite forward differences. While central differences have a better order of consistency, they have a high-frequency null space, which can lead to checkerboard artifacts in conjunction with total variation [Lel+13a].
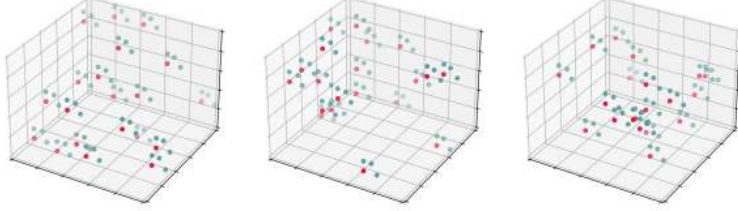
**Figure 5.6: Sampling.** In each training step, $k$ random coordinate points $X = \{(x_i, t_i) \in \Omega \times \Gamma\}_{i=1}^{k}$ are sampled (red). Together, these points form the batch, which the neural fields are trained on in this step. As we use finite differences for calculating the first-order terms in the loss functions, we additionally sample adjacent coordinates $(x_i + h_x e_1, t_i), (x_i + h_x e_2, t_i)$ and $(x_i, t_i + h_t)$ (blue), where $e_i$ are $i$-th unit vectors and $h_x$ is a vector that contains grid step sized for each $\Omega$-axis.

In order to compute the forward differences, we need to sample coordinates which are adjacent to the randomly sampled ones (see Fig. 5.6 for an example with $\Omega \subset \mathbb{R}^2$). Let $e_i$ denote the $i$-th unit vector. Let $h_x$ be a vector which contains the chosen grid step sizes with respect to the individual $\Omega$ axes and $h_t$ the grid step size with respect to the $\Gamma$ axis. Then the gradients are calculated as

$$\nabla_x^h F_{\Theta_v}^v(x_i, t_i) := \frac{1}{h_x} \begin{pmatrix} F_{\Theta_v}^v(x_i + h_x e_1, t_i) - F_{\Theta_v}^v(x_i, t_i) \\ \vdots \\ F_{\Theta_v}^v(x_i + h_x e_d, t_i) - F_{\Theta_v}^v(x_i, t_i) \end{pmatrix}, \quad (5.45)$$

$$\nabla_t^h F_{\Theta_v}^v(x_i, t_i) := \frac{1}{h_t} \left( F_{\Theta_v}^v(x_i, t_i + h_t) - F_{\Theta_v}^v(x_i, t_i) \right), \quad (5.46)$$

using finite forward differences both for the spatial dimensions and the range.

### 5.3.4 Inference

We are mainly interested in the output of the primal neural field $F_{\Theta_v}^v$, as the latter approximates the primal solution $v^*$. For the inference step, we chose a discrete grid $\Omega^h \times \Gamma^h$ with a particular fine resolution of the range

$$\Gamma^h = \left\{ \gamma_1^h, ..., \gamma_l^h \, \middle| \, \gamma_1^h = \Gamma_{\min}, \; \gamma_l^h = \Gamma_{\max}, \; \gamma_i^h < \gamma_{i+1}^h \right\} \quad (5.47)$$

for large $l$. Note that the runtime of the inference step scales linearly with the number of labels $l$ but is comparatively fast, see Fig. 5.7.

We evaluate $F_{\Theta_v}^v$ on said grid, choose the threshold $s = 0.5 \in [0, 1]$, and calculate $v^h : \Omega^h \times \Gamma^h \rightarrow \{0, 1\}$ as

$$v^h(x_i, t_i) = \begin{cases} 1, & \text{if } F_{\Theta_v}^v(x_i, t_i) > s, \\ 0, & \text{otherwise.} \end{cases} \quad (5.48)$$

**Figure 5.7: Runtime of Inference Step for a Grid of Size** $100 \times 100 \times l$**.** The $x$-axis denotes the number of labels $l$ and the $y$-axis the runtime in seconds. Neural field inference is generally very efficient, which allows to achieve precision by increasing the label count.
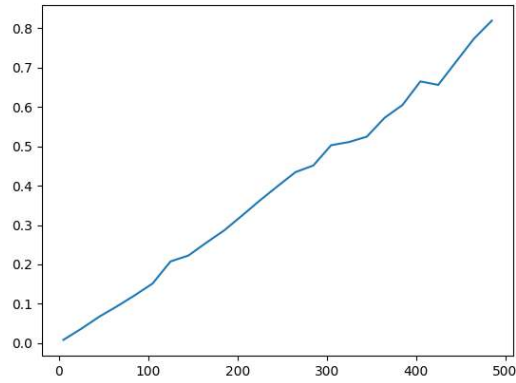
Furthermore, we compute $u^h : \Omega^h \to \mathbb{R}$ as

$$u^h(\boldsymbol{x}_j) = \gamma_1^h + \frac{\left(\gamma_l^h - \gamma_1^h\right)}{l} \sum_{i=1}^{l} v^h(\boldsymbol{x}_j, t_i). \qquad (5.49)$$

Together, (5.48)–(5.49) describe the projection of the neural field approximation of a solution to the embedded problem (5.6) onto the discrete version of the original solution space of (5.1). In the terminology of neural fields, this step can be viewed as the rendering process.

## 5.4 Numerical Results

In the following, we investigate a proof-of-concept implementation of the EmNeF method and test it on a convex denoising and non-convex stereo matching problem. For each problem we manually selected the variance parameter $\sigma$ associated to the RFF encoding, the number of nodes in the hidden layers, and the step sizes used for the SGD optimizers. In all experiments we increased the dimension of the inputs from 3 to $2^{11}$ with the RFF encoding and optimized the weights of the neural fields over 50.000 update steps using batches of size 2500.

The experiments were run on a 2 x 6-core Intel(R) Xeon(R) Gold 6128 CPU @ 3.40GHz system (24 logical cores), NVIDIA GeForce RTX 2080 Ti, Ubuntu 18.04.1 LTS, Python 3.8, and CUDA 10.1.

### 5.4.1 Convex Case: Denoising

First we considered the Rudin-Osher-Fatemi (ROF) denoising problem: For some input image $f$ we seek a solution $u \in \mathrm{BV}(\Omega; \mathbb{R})$

**(a)** Input image      **(b)** CVXPY solution u_cvxpy      **(c)** SA - PDHG solution u_prost

**(d)** EmNeF solution u_emnef      **(e)** Difference u_emnef − u_cvxpy      **(f)** Difference u_emnef − u_sapdhg
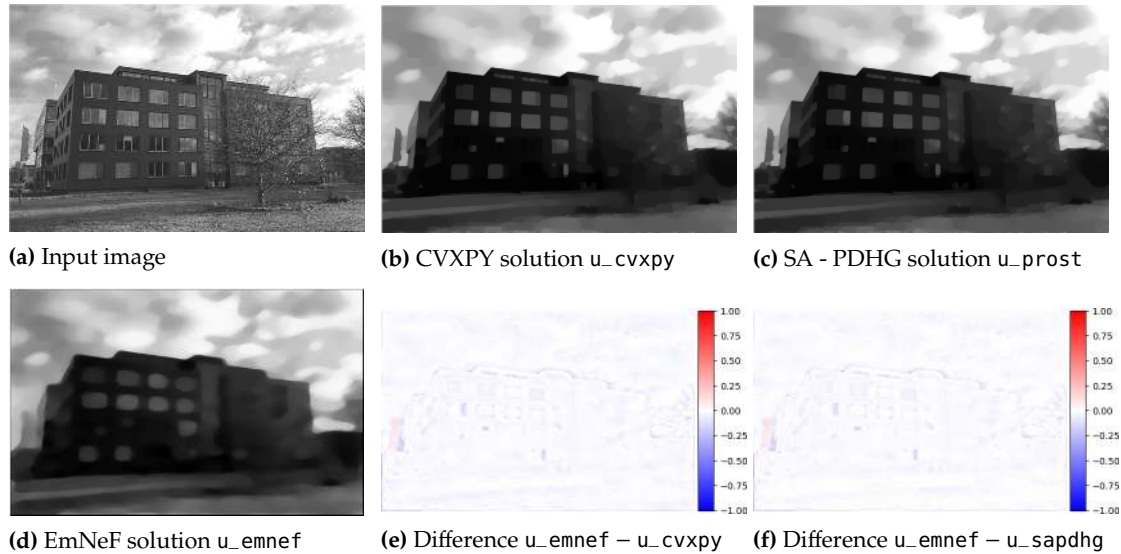
**Figure 5.8: Qualitative Comparison between EmNeF and Classical Convex Optimization Approaches on a Convex Problem.** We consider the ROF denoising problem (5.50) for the input image in **Fig. (a)**. While convex variational problems are not the main topic of interest for embedding techniques, this allows us to compare the EmNeF approach to classical optimization approaches both on the original and lifted energy. We compare the following three approaches: The original problem is solved using CVXPY in combination with the MOSEK backend, see **Fig. (b)**. The lifted problem is solved using sublabel-accurate discretization in combination with the PDHG algorithm and using the proposed EmNeF approach, see **Figs.(c)–(d)**. The EmNeF approach does only approximate the numerical CVXPY solution, which is further illustrated by the difference between the EmNeF and CVXPY result, see **Fig. (e)**. Differences are especially noticeable around the corners of the building and the windows.

which minimizes the energy

$$F(u) = \int_\Omega (u(x) - f(x))^2 \, \mathrm{d}x + \lambda TV(u). \qquad (5.50)$$

The problem is inherently convex, therefore there is no need for convex relaxation. However, as we can easily compute global minimizers of the original problem, applying the lifting approach allows us to compare the results achieved with the EmNeF approach to results achieved with a classic non-smooth convex solver – both on the original and the CBL energy. In our first experiment, we solved the ROF denoising problem (5.50) using three different methods:

▶ **CVXPY:** We first computed a numerical ground truth solution u_cvx of the original (convex) energy (5.50) using the CVXPY toolbox [Agr+18; DB16] with the MOSEK backend [ApS22].

▶ **SA - PDHG:** Second, as discussed in Chpt. 3, we embedded the problem using calibration-based lifting and solved the lifted energy with sublabel-accurate discretization and the PDHG algorithm. We used the prost [LM15] and sublabel_relax [LM16; Möl+15] toolboxes.

▶ **EmNeF:** Finally, we employed the proposed EmNeF approach as described in Chpt. 5.3.

The results are summarized in Fig. 5.8, Fig. 5.9, as well as in Table 5.2, and will be discussed after the following technical details.

**CVXPY Implementation Details.** Starting from the original problem (5.50), the image domain $\Omega = (a_1, b_1) \times (a_2, b_2) \subset \mathbb{R}^2$ was discretized using a regular grid

$$\Omega^h := \left\{ \left( x_j^1, x_k^2 \right) \,\middle|\, j = 1, ..., m_1, \ k = 1, ..., m_2 \right\}, \tag{5.51}$$

where $h_i := \frac{b_i - a_i}{m_i}$ and $x_j^i := a_i + (j - 0.5)h_i$. The total variation was calculated using finite forward differences and Neumann boundary conditions: Denoting the discretization of $u$ at $\left( x_j^i, x_k^i \right)$ by $u_{j,k}^h$, we define

$$\nabla^h u_{j,k}^h := \begin{pmatrix} \nabla_1^h u_{j,k}^h \\ \nabla_2^h u_{j,k}^h \end{pmatrix}, \tag{5.52}$$

where

$$\nabla_1^h u_{j,k}^h := \begin{cases} \frac{1}{h_1} \left( u_{j+1,k}^h - u_{j,k}^h \right), & \text{if } j \leq m_1 - 1, \\ 0, & \text{else}, \end{cases} \tag{5.53}$$

and

$$\nabla_2^h u_{j,k}^h := \begin{cases} \frac{1}{h_2} \left( u_{j,k+1}^h - u_{j,k}^h \right), & \text{if } k \leq m_2 - 1, \\ 0, & \text{else}. \end{cases} \tag{5.54}$$

Omitting the constant factors $\frac{1}{h_1 h_2}$, the discrete energy was thus given by

$$F^h(u^h) := \underbrace{\sum_{j=1}^{m_1} \sum_{k=1}^{m_2} \left( u_{j,k}^h - f_{j,k}^h \right)^2}_{D^h(u^h)} + \lambda \underbrace{\sum_{j=1}^{m_1} \sum_{k=1}^{m_2} \| \nabla^h u_{j,k}^h \|_2}_{\text{TV}^h(u^h)}. \tag{5.55}$$

We computed a minimizer of this discrete energy using CVXPY [Agr+18; DB16] in combination with the Mosek [ApS22] solver. For $\Omega = (0, 400) \times (0, 626)$ and $\lambda = 0.4$, Mosek achieved an objective function value of 1460.5110 within approximately 29 seconds using the default stopping criteria. The result is shown in Fig. 5.8b.

**SA - PDHG Implementation Details.** After lifting the energy (5.50) using the CBL approach, the domain $\Omega \times \Gamma$ was discretized

**Table 5.2: ROF Experiments.** We solved the ROF problem (5.50) for the input image Fig. 5.8a with three different methods: CVXPY (Fig. 5.8b), SA - PDHG (Fig. 5.8c) and the proposed EmNeF (Fig. 5.8d). The CVXPY method used a classic optimization approach on the original energy, the SA - PDHG method used a classic optimization approach on the lifted energy. The proposed EmNeF approach was based on the newly introduced learning framework on the lifted energy. As this convex problems is already efficiently classically solvable without embedding, the CVXPY approach performed best as expected. While the SA - PDHG approach came quite close to the objective function value achieved with CVXPY, the EmNeF approach only approximated the correct solution. While the EmNeF approach cannot (yet?) compete with the two established methods, it offers room for finetuning and future improvements. Especially for non-convex problems it is highly interesting, as it is relatively easy to implement and requires a comparatively small amount of labels.

| Approach | Labels $m_3$ | Runtime [min] | objective function value |
|---|---|---|---|
| CVXPY | – | 0:29 | 1460.5110 |
| SA - PDHG | 2 | 1:32 | 1468.5067 |
| SA - PDHG | 5 | 5:10 | 1468.8050 |
| EmNeF (mean) | 100 (inference) | 8:22 | 1911.3859 |
| No optimization (input image) | – | – | 4725.4132 |

using $\Omega^h$ from (5.51) and

$$\Gamma^h := \{\gamma_l \mid l = 1, ..., m_3\}, \tag{5.56}$$

where $h_3 := \frac{\Gamma_{\max} - \Gamma_{\min}}{m_3}$ and $\gamma_l := \Gamma_{\min} + (l-1)h_3$. The problem was then discretized using the sublabel-accurate discretization described in Sec. 3.3–3.3.3 and solved using the PDHG algorithm.

For the implementation, we relied on the `prost` [LM15] and `sublabel_relax` toolboxes [LM16; Möl+15]. The latter provides an implementation of the ROF problem with CBL lifting and sublabel-accurate discretization, also using finite forward differences and Neumann boundary conditions.

Using the input image in Fig. 5.8a, we solved the lifted ROF problem for $\Omega = (0, 400) \times (0, 626)$, $\Gamma = [0, 1]$ and $\lambda = 0.4$, once with $m_3 = 2$ and once with $m_3 = 5$ labels. Note that the choice of $m_3 = 2$ labels is in fact a reasonable choice in this scenario, as the original data term is convex and the sublabel-accurate discretization allows for a convex approximation of the data term in between the chosen labels. The additional choice of $m_3 = 5$ allows to get an impression of how the runtime scales with respect to the discretization of the label space $\Gamma$.

We set the maximum number of PDHG iterations to 50.000. The runtime was approximately 92 seconds for $m_3 = 2$ labels and 310 seconds for $m_3 = 5$ labels. We projected the lifted solutions onto the original solution space with the pointwise transformation (3.103). Plugging the projected solution into the discrete energy (5.55) gave a value of 1468.5067 for $m_3 = 2$ labels and of 1468.8050 for $m_3 = 5$ labels. The projected solution for $m_3 = 5$ is shown in Fig. 5.8c.

**EmNeF Implementation Details.** We employed the EmNeF approach as described above, i.e., with hard constraints, randomly

**Figure 5.9: Objective Function Value of EmNeF approach for ROF Problem.** As the EmNeF approach is stochastic, we solved the given ROF problem 20 times. Over the 20 runs we achieved an average objective function value of 1911.3859.
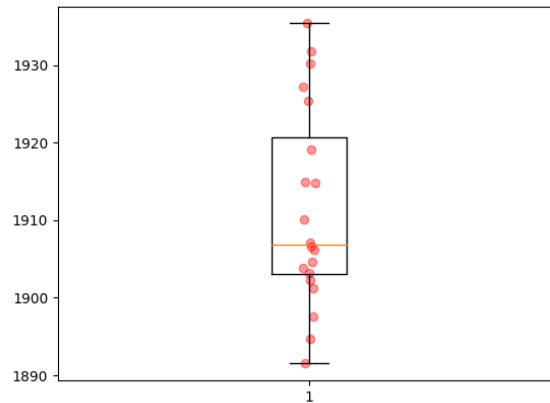
sampled training batches, RFF encoding and finite differences in the loss function. Again, we set $\Omega = (0, 400) \times (0, 626)$, $\Gamma = [0, 1]$ and $\lambda = 0.4$. We optimized the weights of the neural fields over 50.000 update steps using batches of size 2500, a primal learning rate of 0.3, dual learning rate of 0.9, as well as $\sigma = 4$.

As the approach is stochastic, we solved the problem 20 times and consider in the following the mean values over the 20 runs, see also Fig. 5.9. The training took on average $8 : 22$ minutes. For the inference step, we used the discrete grid $\Omega^h \times \Gamma^h$ with $\Omega^h$ from (5.51) and $\Gamma^h$ from (5.56) with $m_3 = 100$ labels. After rendering the solution, this resulted in u_emnef shown in Fig. 5.8d and plugging the rendered solution into (5.55) gave on average an objective function value of 1911.3859.

**Comparison of Results.** In case of the convex ROF problem, the classic CVXPY approach gave the best result while also requiring the least runtime. While the SA - PDHG approach came close to the objective function value, it was slower than the CVXPY approach. The EmNeF approach was the weakest approach in this scenario, as it had the longest runtime and reached the highest objective function value. However, the EmNeF approach is not restricted to convex problems and still young. While the implementation of the sublabel-accurate discretization is rather intricate, the implementation of the EmNeF approach is more simple and adapting the given code to a new problem is as easy as re-implementing the original pointwise data term. This makes the approach still interesting for the non-convex setting.

Comparing the results visually, the biggest differences are noticeable around the boundaries of the building and its windows. In the EmNeF result, the boundaries appear to be smoother, in the sense that corners are blurred while edges are preserved. We attribute these artifacts at least partially to the RFF encoding. Retraining
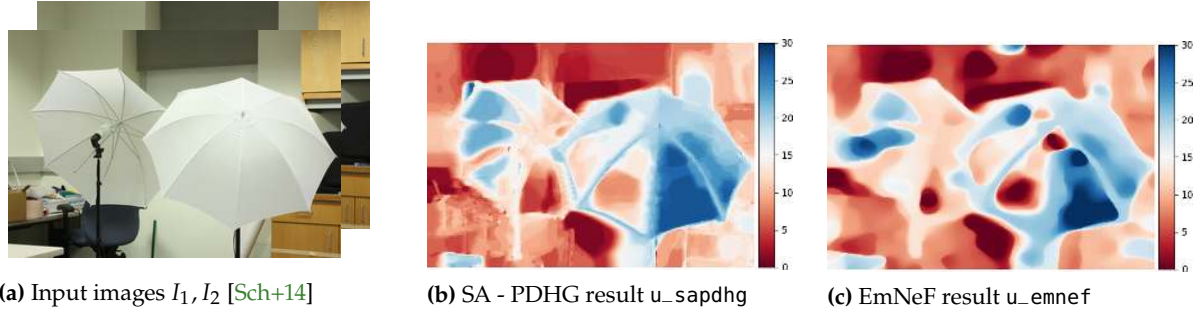
**(a)** Input images $I_1$, $I_2$ [Sch+14]   **(b)** SA - PDHG result `u_sapdhg`   **(c)** EmNeF result `u_emnef`

**Figure 5.10: Comparison of EmNeF for stereo matching to the lifting-based approach.** We considered the stereo matching problem (5.57)–(5.58) for the "Umbrella" [Sch+14] data set **Fig. (a)**. The variational problem was embedded using the CBL approach and solved using sublabel-accurate discretization and the PDHG algorithm **Fig. (b)** and once with the EmNeF approach **Fig. (c)**. Similar to the observations made in the ROF experiment, the EmNeF depth map is smoother compared to the depth map produced by SA - PDHG.

the model while using starting weights from the previous run and iteratively increasing the variance $\sigma$ of the RFF encoding does help to achieve sharper corners in the results. However, it is not advisable to choose a very large $\sigma$ from the start, as this might hinder the network from converging properly; we leave a detailed study to future work.

### 5.4.2 Stereo Matching

In contrast to the ROF denoising problem, the *stereo matching* problem is non-convex, which makes it more interesting for embedding methods such as our EmNeF approach. The aim in stereo matching is to construct a depth map of a 3D scene from multiple, given 2D images; see also p. 5.

Here, we assume that we are given two *rectified* 2D input images $I_1$ and $I_2$ (see Fig. 5.10a). Our aim is to solve the variational problem

$$\inf_{u \in BV(\Omega;\Gamma)} F(u), \quad F(u) := D(u) + \lambda TV(u), \tag{5.57}$$

where the data term is defined as

$$D(u) := \int_\Omega \min\{0.15; |I_1(x_1, x_2) - I_2(x_1, x_2 - u(x))|\} \, \mathrm{d}x. \tag{5.58}$$

We solved the problem (5.57) using two different methods:

▶ **SA - PDHG:** First, we embedded the given problem using calibration-based lifting and solved the lifted energy with sublabel-accurate discretization and the PDHG algorithm. We again used the `prost` [LM15] and `sublabel_relax` toolboxes [LM16; Möl+15].

▶ **EmNeF:** Second, we employed the proposed EmNeF approach as described in Chpt. 5.3.

**Table 5.3: Stereo Matching Experiments.** We solved the stereo matching problem (5.57) for the input images in Fig. 5.10a with two different methods: SA - PDHG (Fig. 5.10b) and the proposed EmNeF (Fig. 5.10c). The SA - PDHG method uses a classic PDHG optimization approach on the lifted energy using sublabel-accurate relaxation. The proposed EmNeF approach is based on the newly introduced learning framework on the lifted energy. The SA - PDHG approach reached a smaller objective function value while also requiring less runtime and was thus numerically superior. The EmNeF approach on the other hand is more accessible in the sense that it can be easily adapted to a new non-convex data term. While the SA - PDHG approach requires to implement projections onto the epigraph of the piecewise Fenchel conjugate of the data term, the EmNeF approach only requires to implement the original pointwise data term.

| Approach | $m_3$ | Max. Iterations | Runtime [min] | objective function value |
|---|---|---|---|---|
| SA - PDHG | 8 | 100.000 | 3:57 | 757.3621 |
| SA - PDHG | 8 | 200.000 | 7:59 | 757.4286 |
| SA - PDHG | 8 | 1.000.000 | 39:48 | 757.4701 |
| EmNeF | 100 (inference) | 50.000 | 9:00 | 932.3928 |
| $u \equiv 0$ | – | – | – | 2864.7905 |

More details on the implementation of the different approaches can be found in the following paragraphs. The results are summarized in Fig. 5.10 and Table 5.3, and are further discussed after the technical details.

**Discrete Energy.** Let us first discuss the discretization of the energy, which we used to compare the results achieved with the SA - PDHG and EmNeF approach. Similar to the ROF example, we defined $\Omega^h$ as in (5.51) and calculated the discrete total variation using finite forward differences and Neumann boundary conditions, see (5.52)–(5.55).

We assumed that the input images $I_1^h$ and $I_2^h$ are given on the cell-centered grid $\Omega^h$ (5.51). In order to evaluate $I_1^h$ on the transformed grid

$$\tilde{\Omega}^h := \left\{ \left( x_j^1, x_k^2 - cu^h\left(x_j^1, x_k^2\right) \right) \middle| \left(x_j^1, x_k^2\right) \in \Omega^h \right\}, \qquad (5.59)$$

we used a bilinear interpolation. Here, $c$ is a constant parameter, which declares the maximal expected, pixel-wise shift between $I_1^h$ and $I_2^h$. Denoting the result of the bilinear interpolation of $I_2^h$ on $\tilde{\Omega}^h$ by $\tilde{I}_2^h$, the discrete energy was thus given by

$$F^h(u^h) := D^h(u^h) + \lambda \, \mathrm{TV}^h(u^h), \qquad (5.60)$$

$$D^h(u^h) := \sum_{j=1}^{m_1} \sum_{k=1}^{m_2} \max\left\{ 0.15; \left| \left(I_1^h\right)_{j,k} - \left(\tilde{I}_2^h\right)_{j,k} \right| \right\}. \qquad (5.61)$$

In the following, we used this discrete energy in order to qualitatively compare the (projected) results of the SA - PDHG and EmNeF experiments.

**SA - PDHG Implementation Details.**    After lifting the energy (5.57) using the CBL approach, the domain $\Omega \times \Gamma$ was discretized using $\Omega^h$ from (5.51) and $\Gamma^h$ from (5.56). The problem was then discretized using the sublabel-accurate discretization described in Sec. 3.3–3.3.3 and solved using the PDHG algorithm.

In the repository [LM16], an implementation of a stereo matching problem with slightly different data term can be found. The data term was implemented as a cost volume, i.e., a matrix $C \in \mathbb{R}^{m_1, m_2, z}$ which stores a (pixel-wise) cost, where $z$ denotes the maximal expected shift (described in pixels):

$$c_{j,k,l} = \max \left\{ 0.15; \left| I_1 \left( x_j^1, x_k^2 \right) - I_2 \left( x_j^1, x_k^2 - lh_2 \right) \right| \right\}. \qquad (5.62)$$

The total variation was again implemented using finite forward differences and Neumann boundary conditions.

We ran our experiments for $\Omega = (0, 202) \times (0, 296)$, $\Gamma = [0, 1]$, $\lambda = 0.1$, $z = 30$, and $m_3 = 8$, and set the maximum number of PDHG iterations to $\left\{ 10^5, 2 \cdot 10^5, 10^6 \right\}$, which resulted in runtimes of approximately 237, 479, and 2388 seconds. We projected the lifted solutions onto the original solution space using the pointwise transformation (3.103) and plugged the projected solutions into the discrete energy (5.60), which gave an objective function value of approximately 757. All results are summarized in Tab. 5.3.

**EmNeF Implementation Details.**    We employed the EmNeF approach with hard constraints, randomly sampled training batches, RFF encoding and finite differences in the loss function. For $\Omega = (0, 202) \times (0, 296)$ and $\Gamma = [0, 1]$ we optimized the weights of the neural fields over 50.000 update steps using batches of size 2500, a primal learning rate of 0.15 and dual learning rate of 0.5. We set $\sigma = 3.0$, $\lambda = 0.1$ and assumed that the maximal shift is 30 pixels.

We trained the neural fields 20 times and in the following consider the mean values over the 20 runs, see also Fig. 5.11. On average, training took 540 seconds. For the inference step, we used the discrete grid $\Omega^h \times \Gamma^h$ with $\Omega^h$ from (5.51) and $\Gamma^h$ from (5.56) with $m_3 = 100$ labels. After inference, this resulted in the deformation u_emnef shown in Fig. 5.10c. Plugging the rendered solution into (5.60) gave an average objective function value of 932.3928.

**Comparison of Results.**    The SA - PDHG approach converged to a smaller objective function value while also requiring less runtime. Visually, the greatest difference between the solutions of the SA - PDHG and EmNeF approach is again the bluntness of corners and shapes in the solution of the EmNeF approach. We

**Figure 5.11: Objective Function Value of EmNeF Approach for Stereo Matching Problem.** As the EmNeF approach is stochastic, we solved the given stereo matching problem 20 times. Over the 20 runs, we achieved an average objective function value of 932.3928.
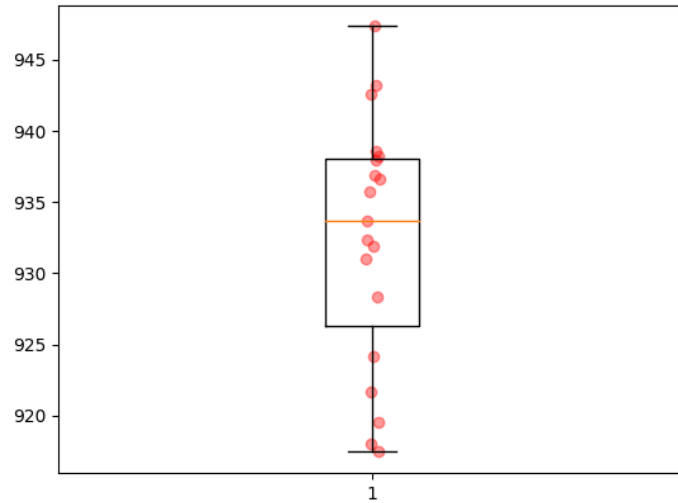
attribute this at least partially to the RFF encoding and specific choice of the variance parameter $\sigma^2$.

Although the EmNef approach in its current form is numerically inferior to the established SA - PDHG approach, the here presented results demonstrate the potential of the proposed method. The EmNeF approach is more accessible in the sense that it can be easily adapted to a new non-convex data term. While the SA - PDHG approach requires to implement projections onto the epigraph of the piecewise Fenchel conjugate of the data term, the EmNeF approach merely requires to implement the original pointwise data term. With more research dedicated to the influence of the RFF encoding, the convergence behavior as well as tweaking and finetuning of the method, the EmNeF approach could evolve into an easy-to-use strategy for solving non-convex variational problems.

## 5.5 Discussion and Outlook

With the EmNeF approach we have introduced a methodologically interesting approach for solving variational problems with possibly non-convex data term and TV regularizer of the form (5.1). By combining results from calibration-based lifting with modern learning techniques, we created a framework for diverse applications. While the results of the EmNeF approach are not (yet) on par with the ones achieved with established methods, we would like to point out certain advantages the EmNeF approach has to offer especially in the non-convex setting.

### 5.5.1 Accessibility of CBL Method

In the imaging community, the calibration-based lifting approach is celebrated for its practical usefulness as it allows to solve certain non-convex variational problems with convex optimization strategies. While the approach yields compelling results, it is rarely found in practical applications. We conjecture that this is at least partially caused by the relatively advanced mathematical theory and potential difficulties during discretization and optimization.

In Sec. 3.3, we have discussed different discretization schemes for an energy which has been embedded using the calibration-based lifting approach; A finite-difference discretization approach which is easy to implement but numerically expensive, and a finite-element discretization approach which leads to a sublabel-accurate discretization also employed in the SA - PDHG experiments that is numerically less expensive but more intricate to implement. The EmNeF approach proposed in this chapter is as easy to implement as the finite-difference discretization approach, yet it does not come with the high numerical costs:

We were able to achieve the above shown results with 135.361 (66.657) trainable parameters in the ROF (stereo matching) case. Putting these numbers into perspective, in the SA - PDHG examples we worked on a finite grid $\Omega^h \times \Gamma^h$ where $\Omega^h$ had $400 \cdot 626 = 250.400$ ($202 \cdot 296 = 59.792$) grid points in the ROF (stereo matching) case. This means that the results with the EmNeF approach were achieved using the equivalence of $1 + 0.5$ ($1 + 1.1$) parameters (*labels*) per grid point. In the experiments, we have demonstrated that the sublabel-accurate discretization requires only $m_3 = 2$ labels in the special case of an (originally) convex problem, see Tab. 5.2. This is, however, *not* the case for (originally) non-convex problems.

In conclusion, the EmNeF approach could make calibration-based lifting more accessible for practical applications. Compared to the sublabel-accurate discretization it does not only simplify the implementation, but it also offers the possibility of approximating the results using fewer parameters (*labels*).

### 5.5.2 Future Research

The EmNeF approach and neural fields in general are a rather new area of research and neither can nor should be viewed as a replacement for classic numerical optimization and discretization/approximation. The latter are built upon strong theoretical results developed over many decades and meet standards required in practice. As neural fields are still young, they lack some theoretical foundation and understanding – this, however, can both

136 5 <em>EmNeFs</em>: <em>Embedded Neural Fields</em>

be understood as current disadvantage and possibility for future improvements.

In the literature, one can find many theorems stating that infinitely deep or wide feed-forward neural networks are *universal function approximators* which can learn an implicit representation of any sufficiently regular function [HSW89; CMB00; SM17; Her+21]. In practice, however, neural networks are only of finite width and depth. The pressing question for the EmNeF approach is: How do the three neural fields need to be designed in terms of their width, depth and activation functions in order to have the expressive capacity of approximating functions in their respective solution spaces (5.8)–(5.10) well? Such an analysis of the approximation error is, e.g., done in [Bar94; Bar93].

We have already pointed out that even if the loss function is convex with respect to the network's output, it is typically non-convex with respect to the network's parameters. This is also the case for the EmNeF approach: Even though the CBL approach convexifies the variational problem, the loss functions of the three networks are non-convex with respect to the networks' parameters. Therefore, we still have to solve non-convex problems during training. Does the EmNeF approach (always) converge to a global minimizer and which global minimizer is preferred in a non-unique setting?

Research concerning neural fields is rapidly changing and growing. As applications of neural fields increase, theoretical results become ever more important and are a topic of ongoing research. In the future, we expect that missing theoretical results can be extended over time and help to improve the encouraging experimental results.

# Bibliography

[ABD03]   G. Alberti, G. Bouchitté, and G. Dal Maso. 'The calibration method for the Mumford-Shah functional and free-discontinuity problems'. In: *Calculus of Variations and Partial Differential Equations* 16.3 (2003), pp. 299–333. DOI: `https://doi.org/10.1007/s005260100152` (cited on pages 1, 9, 43).

[ABM14]   H. Attouch, G. Buttazzo, and G. Michaille. *Variational analysis in Sobolev and BV spaces: applications to PDEs and optimization*. SIAM, 2014 (cited on pages 1, 28).

[AFC22]   A. Alacaoglu, O. Fercoq, and V. Cevher. 'On the convergence of stochastic primal-dual hybrid gradient'. In: *SIAM Journal on Optimization* 32.2 (2022), pp. 1288–1318. DOI: `https://doi.org/10.1137/19M1296252` (cited on page 70).

[AFP00]   L. Ambrosio, N. Fusco, and D. Pallara. *Functions of bounded variation and free discontinuity problems*. Oxford mathematical monographs, 2000 (cited on pages 1, 4, 26–34, 42, 45, 46, 50, 51, 55, 80, 105).

[Agr+18]   A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd. 'A rewriting system for convex optimization problems'. In: *Journal of Control and Decision* 5.1 (2018), pp. 42–60 (cited on pages 108, 127, 128).

[AK06]   G. Aubert and P. Kornprobst. *Mathematical problems in image processing: partial differential equations and the calculus of variations*. Vol. 147. Springer Science & Business Media, 2006 (cited on page 1).

[Alt92]   H. W. Alt. 'Linear functional analysis'. In: *An application oriented introduction* (1992) (cited on page 80).

[Ami94]   Y. Amit. 'A nonlinear variational problem for image matching'. In: *SIAM Journal on Scientific Computing* 15.1 (1994), pp. 207–224. DOI: `https://doi.org/10.1137/0915014` (cited on page 3).

[AMP04]   L. Ambrosio, M. Miranda Jr, and D. Pallara. 'Special functions of bounded variation in doubling metric measure spaces'. In: *Calculus of variations: topics from the mathematical heritage of E. De Giorgi* 14 (2004), pp. 1–45 (cited on pages 44, 45, 51).

[Anz83]   G. Anzellotti. 'Pairings between measures and bounded functions and compensated compactness'. In: *Annali di Matematica pura ed applicata* 135 (1983), pp. 293–318 (cited on page 80).

[ApS22]   M. ApS. *MOSEK Optimizer API for Python 10.1.11*. 2022 (cited on pages 127, 128).

[Bar93]   A. R. Barron. 'Universal approximation bounds for superpositions of a sigmoidal function'. In: *IEEE Transactions on Information theory* 39.3 (1993), pp. 930–945 (cited on page 136).

[Bar94]    A. R. Barron. 'Approximation and estimation bounds for artificial neural networks'. In: *Machine learning* 14 (1994), pp. 115–133 (cited on page 136).

[Bas+20]   R. Basri, M. Galun, A. Geifman, D. Jacobs, Y. Kasten, and S. Kritchman. 'Frequency bias in neural networks for input of non-uniform density'. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 685–694 (cited on page 117).

[BCN02]    G. Bellettini, V. Caselles, and M. Novaga. 'The total variation flow in RN'. In: *Journal of Differential Equations* 184.2 (2002), pp. 475–525 (cited on pages 13, 14, 35, 74, 77).

[BH16]     K. Bredies and M. Holler. 'A pointwise characterization of the subdifferential of the total variation functional'. In: *arXiv preprint arXiv:1609.08918* (2016) (cited on pages 80, 81, 86).

[BL11]     K. Bredies and D. Lorenz. *Mathematische Bildverarbeitung*. Vol. 1. Springer, 2011 (cited on pages 1, 2).

[BL21]     D. Bednarski and J. Lellmann. 'Inverse Scale Space Iterations for Non-convex Variational Problems Using Functional Lifting'. In: *Scale Space and Variational Methods in Computer Vision*. Ed. by A. Elmoataz, J. Fadili, Y. Quéau, J. Rabin, and L. Simon. Cham: Springer International Publishing, 2021, pp. 229–241. DOI: https://doi.org/10.1007/978-3-030-75549-2_19 (cited on page 2).

[BL22]     D. Bednarski and J. Lellmann. 'Inverse Scale Space Iterations for Non-Convex Variational Problems: The Continuous and Discrete Case'. In: *Journal of Mathematical Imaging and Vision* (2022), pp. 1–16. DOI: https://doi.org/10.1007/s10851-022-01125-8 (cited on page 2).

[BL23]     D. Bednarski and J. Lellmann. 'EmNeF: Neural Fields for Embedded Variational Problems in Imaging'. In: *Scale Space and Variational Methods in Computer Vision*. Ed. by L. Calatroni, M. Donatelli, S. Morigi, M. Prato, and M. Santacesaria. Cham: Springer International Publishing, 2023, pp. 137–148. DOI: https://doi.org/10.1007/978-3-031-31975-4_11 (cited on page 2).

[BN06]     C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006 (cited on pages 1, 108).

[Bot12]    L. Bottou. 'Stochastic gradient descent tricks'. In: *Neural Networks: Tricks of the Trade: Second Edition* (2012), pp. 421–436. DOI: https://doi.org/10.1007/978-3-642-35289-8_25 (cited on page 110).

[Bur+06]   M. Burger, G. Gilboa, S. Osher, J. Xu, et al. 'Nonlinear inverse scale space methods'. In: *Communications in Mathematical Sciences* 4.1 (2006), pp. 179–212 (cited on page 75).

[Bur+15]   M. Burger, L. Eckardt, G. Gilboa, and M. Moeller. 'Spectral representations of one-homogeneous functionals'. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2015, pp. 16–27. DOI: https://doi.org/10.1007/978-3-319-18461-6_2 (cited on pages 73, 75).

[Bur+16]   M. Burger, G. Gilboa, M. Moeller, L. Eckardt, and D. Cremers. 'Spectral decompositions using one-homogeneous functionals'. In: *SIAM Journal on Imaging Sciences* 9.3 (2016), pp. 1374–1408 (cited on pages 4, 5, 9, 12–14, 39, 40, 73–76).

[BV04]   S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004 (cited on page 6).

[Cai+21]   S. Cai, Z. Wang, S. Wang, P. Perdikaris, and G. E. Karniadakis. 'Physics-informed neural networks for heat transfer problems'. In: *Journal of Heat Transfer* 143.6 (2021), p. 060801. DOI: https://doi.org/10.1115/1.4050542 (cited on pages 17, 114, 115).

[CB20]   L. Chizat and F. Bach. 'Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss'. In: *Conference on Learning Theory*. PMLR. 2020, pp. 1305–1338 (cited on page 111).

[CCP12]   A. Chambolle, D. Cremers, and T. Pock. 'A convex approach to minimal partitions'. In: *SIAM Journal on Imaging Sciences* 5.4 (2012), pp. 1113–1158 (cited on pages 8, 66, 67).

[CEN06]   T. F. Chan, S. Esedoglu, and M. Nikolova. 'Algorithms for finding global minimizers of image segmentation and denoising models'. In: *SIAM journal on applied mathematics* 66.5 (2006), pp. 1632–1648. DOI: https://doi.org/10.1137/040615286 (cited on page 8).

[Cha+18]   A. Chambolle, M. J. Ehrhardt, P. Richtárik, and C.-B. Schonlieb. 'Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications'. In: *SIAM Journal on Optimization* 28.4 (2018), pp. 2783–2808. DOI: https://doi.org/10.1137/17M1134834 (cited on page 70).

[Cha01]   A. Chambolle. 'Convex representation for lower semicontinuous envelopes of functionals in L1'. In: *Journal of Convex Analysis* 8.1 (2001), pp. 149–170 (cited on page 43).

[CMB00]   J. L. Castro, C. J. Mantas, and J. Benıtez. 'Neural networks with a continuous squashing function in the output are universal approximators'. In: *Neural Networks* 13.6 (2000), pp. 561–563 (cited on pages 107, 136).

[CP11]   A. Chambolle and T. Pock. 'A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging'. In: *Journal of Mathematical Imaging and Vision* 40.1 (May 2011), pp. 120–145. DOI: https://doi.org/10.1007/s10851-010-0251-1 (cited on pages 69, 70, 108).

[CSV00]   T. F. Chan, B. Y. Sandberg, and L. A. Vese. 'Active contours without edges for vector-valued images'. In: *Journal of Visual Communication and Image Representation* 11.2 (2000), pp. 130–141 (cited on page 8).

[Cuo+22]   S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli. 'Scientific machine learning through physics–informed neural networks: where we are and what's next'. In: *Journal of Scientific Computing* 92.3 (2022), p. 88. DOI: https://doi.org/10.1007/s10915-022-01939-z (cited on pages 17, 110).

[CV01]   T. F. Chan and L. A. Vese. 'Active Contours Without Edges'. In: *IEEE Trans. Image Proc.* 10.2 (2001), pp. 266–277. DOI: https://doi.org/10.1109/83.902291 (cited on page 15).

[DB16]     S. Diamond and S. Boyd. 'CVXPY: A Python-embedded modeling language for convex optimization'. In: *Journal of Machine Learning Research* 17.83 (2016), pp. 1–5 (cited on pages 108, 127, 128).

[Den+09]   J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 'Imagenet: A large-scale hierarchical image database'. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255 (cited on page 16).

[Eck89]    J. Eckstein. 'Splitting methods for monotone operators with applications to parallel optimization'. PhD thesis. Massachusetts Institute of Technology, 1989 (cited on page 69).

[EG15]     L. C. Evans and R. F. Gariepy. *Measure theory and fine properties of functions*. CRC press, 2015 (cited on page 28).

[ET99]     I. Ekeland and R. Temam. *Convex Analysis and Variational Problems*. Vol. 28. Siam, 1999 (cited on pages 1, 25, 81, 86).

[Eva22]    L. C. Evans. *Partial differential equations*. Vol. 19. American Mathematical Society, 2022 (cited on pages 10, 32).

[EZC10]    E. Esser, X. Zhang, and T. F. Chan. 'A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science'. In: *SIAM Journal on Imaging Sciences* 3.4 (2010), pp. 1015–1046 (cited on page 70).

[FB18]     C. Fougner and S. Boyd. 'Parameter selection and preconditioning for a graph form solver'. In: *Emerging Applications of Control and Systems Theory*. Springer, 2018, pp. 41–61 (cited on pages 70, 108).

[FR60]     W. H. Fleming and R. Rishel. 'An integral formula for total gradient variation'. In: *Archiv der Mathematik* 11.1 (1960), pp. 218–222. DOI: https://doi.org/10.1007/BF01236935 (cited on page 50).

[GDE21]    E. B. Gutiérrez, C. Delplancke, and M. J. Ehrhardt. 'Convergence properties of a randomized primal-dual algorithm with applications to parallel mri'. In: *Scale Space and Variational Methods in Computer Vision: 8th International Conference, SSVM 2021, Virtual Event, May 16–20, 2021, Proceedings*. Springer. 2021, pp. 254–266. DOI: https://doi.org/10.1007/978-3-030-75549-2_21 (cited on page 70).

[Gil13]    G. Gilboa. 'A spectral approach to total variation'. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2013, pp. 36–47. DOI: https://doi.org/10.1007/978-3-642-38267-3_4 (cited on pages 5, 12).

[Gil14]    G. Gilboa. 'A total variation spectral framework for scale and texture analysis'. In: *SIAM journal on Imaging Sciences* 7.4 (2014), pp. 1937–1961. DOI: https://doi.org/10.1137/130930704 (cited on page 5).

[GMB16]    G. Gilboa, M. Moeller, and M. Burger. 'Nonlinear spectral analysis via one-homogeneous functionals: Overview and future prospects'. In: *Journal of Mathematical Imaging and Vision* 56.2 (2016), pp. 300–319 (cited on pages 14, 75).

[Gol+13]   T. Goldstein, M. Li, X. Yuan, E. Esser, and R. Baraniuk. 'Adaptive primal-dual hybrid gradient methods for saddle-point problems'. In: *arXiv preprint arXiv:1305.0546* (2013) (cited on pages 70, 108).

[Goo+14]   I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. 'Generative Adversarial Nets'. In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger. Vol. 27. Curran Associates, Inc., 2014 (cited on page 112).

[Goo+20]   I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. 'Generative adversarial networks'. In: *Communications of the ACM* 63.11 (2020), pp. 139–144. DOI: https://doi.org/10.1145/3422622 (cited on page 112).

[Goo16]   I. Goodfellow. 'Nips 2016 tutorial: Generative adversarial networks'. In: *arXiv preprint arXiv:1701.00160* (2016). DOI: https://doi.org/10.48550/arXiv.1701.00160 (cited on pages 112, 113).

[Gui+23]   J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye. 'A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications'. In: *IEEE Transactions on Knowledge and Data Engineering* 35.4 (2023), pp. 3313–3332. DOI: 10.1109/TKDE.2021.3130191 (cited on pages 112, 113).

[Gun+18]   S. Gunasekar, J. Lee, D. Soudry, and N. Srebro. 'Characterizing implicit bias in terms of optimization geometry'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 1832–1841 (cited on page 111).

[Hal13]   B. C. Hall. *Quantum theory for mathematicians*. Springer, 2013 (cited on pages 36–39, 74).

[Her+21]   C. Hertrich, A. Basu, M. Di Summa, and M. Skutella. 'Towards lower bounds on the depth of ReLU neural networks'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 3336–3348 (cited on pages 107, 136).

[HG18]   E. Hait and G. Gilboa. 'Spectral total-variation local scale signatures for image manipulation and fusion'. In: *IEEE Transactions on Image Processing* 28.2 (2018), pp. 880–895. DOI: https://doi.org/10.1109/TIP.2018.2872630 (cited on pages 1, 5, 9, 14).

[HKY20]   G. Haskins, U. Kruger, and P. Yan. 'Deep learning in medical image registration: a survey'. In: *Machine Vision and Applications* 31 (2020), pp. 1–18. DOI: https://doi.org/10.1007/s00138-020-01060-x (cited on page 109).

[HL96]   J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms I: Fundamentals*. Vol. 305. Springer science & business media, 1996 (cited on pages 24, 47).

[HSW89]   K. Hornik, M. Stinchcombe, and H. White. 'Multilayer feedforward networks are universal approximators'. In: *Neural networks* 2.5 (1989), pp. 359–366 (cited on pages 107, 136).

[IG98a]   H. Ishikawa and D. Geiger. 'Occlusions, discontinuities, and epipolar lines in stereo'. In: *Computer Vision—ECCV'98: 5th European Conference on Computer Vision Freiburg, Germany, June, 2–6, 1998 Proceedings, Volume I 5*. Springer. 1998, pp. 232–248. DOI: https://doi.org/10.1007/BFb0055670 (cited on page 8).

[IG98b]      H. Ishikawa and D. Geiger. 'Segmentation by grouping junctions'. In: *CVPR*. Vol. 98. Citeseer. 1998, p. 125. DOI: https://doi.org/10.1109/CVPR.1998.698598 (cited on page 8).

[IG99]       H. Ishikawa and D. Geiger. 'Mapping image restoration to a graph problem.' In: *NSIP*. 1999, pp. 189–193 (cited on page 8).

[Ish03]      H. Ishikawa. 'Exact optimization for Markov random fields with convex priors'. In: *IEEE transactions on pattern analysis and machine intelligence* 25.10 (2003), pp. 1333–1336. DOI: https://doi.org/10.1109/TPAMI.2003.1233908 (cited on pages 8, 41).

[Kar+21]     G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. 'Physics-informed machine learning'. In: *Nature Reviews Physics* 3.6 (2021), pp. 422–440. DOI: doi:10.1038/s42254-021-00314-5 (cited on page 17).

[KB14]       D. P. Kingma and J. Ba. 'Adam: A method for stochastic optimization'. In: *arXiv preprint arXiv:1412.6980* (2014) (cited on page 110).

[KH+09]      A. Krizhevsky, G. Hinton, et al. 'Learning multiple layers of features from tiny images'. In: (2009) (cited on page 16).

[Koe+22]     L. Koestler, D. Grittner, M. Moeller, D. Cremers, and Z. Lähner. 'Intrinsic neural fields: Learning functions on manifolds'. In: *European Conference on Computer Vision*. Springer. 2022, pp. 622–639 (cited on page 106).

[Koe84]      J. J. Koenderink. 'The structure of images'. In: *Biological cybernetics* 50.5 (1984), pp. 363–370 (cited on page 9).

[Lau+16]     E. Laude, T. Möllenhoff, M. Möller, J. Lellmann, and D. Cremers. 'Sublabel-accurate convex relaxation of vectorial multilabel energies'. In: *European conference on computer vision*. Springer. 2016, pp. 614–627. DOI: https://doi.org/10.1007/978-3-319-46448-0_37 (cited on page 41).

[LBH15]      Y. LeCun, Y. Bengio, and G. Hinton. 'Deep learning'. In: *nature* 521.7553 (2015), pp. 436–444. DOI: https://doi.org/10.1038/nature14539 (cited on page 108).

[LDR22]      K. L. Lim, R. Dutta, and M. Rotaru. 'Physics informed neural network using finite difference method'. In: *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2022, pp. 1828–1833. DOI: 10.1109/SMC53654.2022.9945171 (cited on pages 115, 124).

[LeC98]      Y. LeCun. 'The MNIST database of handwritten digits'. In: *http://yann. lecun. com/exdb/mnist/* (1998) (cited on pages 16, 109).

[Lel+13a]    J. Lellmann, B. Lellmann, F. Widmann, and C. Schnörr. 'Discrete and Continuous Models for Partitioning Problems'. In: *Int. J. Comp. Vis* 104.3 (2013), pp. 241–269. DOI: https://doi.org/10.1007/s11263-013-0621-4 (cited on pages 8, 124).

[Lel+13b]    J. Lellmann, E. Strekalovskiy, S. Koetter, and D. Cremers. 'Total variation regularization for functions with values in a manifold'. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013, pp. 2944–2951. DOI: https://doi.org/10.1109/ICCV.2013.366 (cited on pages 9, 41).

[Lin+14]  T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Doll'a r, and C. L. Zitnick. 'Microsoft COCO: Common Objects in Context'. In: *CoRR* abs/1405.0312 (2014) (cited on page 16).

[LL16]  B. Loewenhauser and J. Lellmann. 'Functional lifting for variational problems with higher-order regularization'. In: *International Conference on Imaging, Vision and Learning based on Optimization and PDEs*. Springer. 2016, pp. 101–120. DOI: `https://doi.org/10.1007/978-3-319-91274-5_5` (cited on page 9).

[LL19]  A. S. Lundervold and A. Lundervold. 'An overview of deep learning in medical imaging focusing on MRI'. In: *Zeitschrift für Medizinische Physik* 29.2 (2019), pp. 102–127 (cited on page 16).

[LM15]  E. Laude and T. Möllenhoff. 'Prost'. In: *GitHub repository* (2015). DOI: `https://github.com/tum-vision/prost` (cited on pages 70, 97, 98, 101, 102, 108, 127, 129, 131).

[LM16]  E. Laude and T. Möllenhoff. 'sublabel$_r$*elax*'. In: *GitHub repository* (2016). DOI: `https://github.com/tum-vision/sublabel_relax` (cited on pages 97, 98, 101, 102, 108, 127, 129, 131, 133).

[LS11]  J. Lellmann and C. Schnörr. 'Continuous multiclass labeling approaches and algorithms'. In: *SIAM Journal on Imaging Sciences* 4.4 (2011), pp. 1049–1096. DOI: `https://doi.org/10.1137/100805844` (cited on pages 8, 66).

[Lu+21]  L. Lu, R. Pestourie, W. Yao, Z. Wang, F. Verdugo, and S. G. Johnson. 'Physics-informed neural networks with hard constraints for inverse design'. In: *SIAM Journal on Scientific Computing* 43.6 (2021), B1105–B1132 (cited on page 17).

[Mas79]  G. D. Maso. 'Integral representation on BV (Ω) of Γ-limits of variational integrals'. In: *manuscripta mathematica* 30.4 (1979), pp. 387–416. DOI: `https://doi.org/10.1007/BF01301259` (cited on pages 42, 47, 48, 56, 71).

[MC17]  T. Möllenhoff and D. Cremers. 'Sublabel-accurate discretization of nonconvex free-discontinuity problems'. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 1183–1191. DOI: `10.1109/ICCV.2017.134` (cited on pages 1, 9, 19, 42, 43, 58, 59, 62, 64, 68, 106).

[MC19a]  T. Möllenhoff and D. Cremers. 'Flat metric minimization with applications in generative modeling'. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 4626–4635 (cited on pages 19, 41, 114).

[MC19b]  T. Möllenhoff and D. Cremers. 'Lifting vectorial variational problems: A natural formulation based on geometric measure theory and discrete exterior calculus'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 11117–11126. DOI: `https://doi.org/10.1109/CVPR.2019.01137` (cited on pages 8, 41).

[Mic56]  E. Michael. 'Continuous Selections. I'. In: *Annals of Mathematics* 63.2 (1956), pp. 361–382. DOI: `https://doi.org/10.2307/1969615` (cited on pages 52, 54).

[Mil+21]   B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. 'NeRf: Representing scenes as neural radiance fields for view synthesis'. In: *Communications of the ACM* 65.1 (2021), pp. 99–106. DOI: https://doi.org/10.1145/3503250 (cited on pages 19, 106, 116).

[Mir64]   M. Miranda. 'Superfici cartesiane generalizzate ed insiemi di perimetro localmente finito sui prodotti cartesiani'. In: *Annali della Scuola Normale Superiore di Pisa-Scienze Fisiche e Matematiche* 18.4 (1964), pp. 515–542 (cited on page 44).

[MMS21]   R. Mulayoff, T. Michaeli, and D. Soudry. 'The implicit bias of minima stability: A view from function space'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 17749–17761 (cited on page 111).

[Möl+15]   T. Möllenhoff, E. Laude, M. Möller, J. Lellmann, and D. Cremers. 'Sublabel-Accurate Relaxation of Nonconvex Energies'. In: *CoRR* abs/1512.01383 (2015). DOI: https://doi.org/10.1109/CVPR.2016.428 (cited on pages 19, 41–43, 58, 64–68, 89, 96–98, 101, 102, 106, 108, 127, 129, 131).

[Möl20]   T. Möllenhoff. 'Efficient Lifting Methods for Variational Problems'. PhD thesis. Technische Universität München, 2020 (cited on page 8).

[Mor02]   M. G. Mora. 'The calibration method for free-discontinuity problems on vector-valued maps'. In: *Journal of Convex Analysis* 9.1 (2002), pp. 1–30 (cited on page 43).

[Mor16]   F. Morgan. *Geometric measure theory: a beginner's guide*. Academic press, 2016 (cited on pages 43, 114, 116).

[NW99]   J. Nocedal and S. J. Wright. *Numerical optimization*. Springer, 1999 (cited on pages 1, 6).

[Osh+05]   S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. 'An iterative regularization method for total variation-based image restoration'. In: *Multiscale Modeling & Simulation* 4.22 (2005), pp. 460–489 (cited on pages 12, 75, 76).

[Pal+22]   D. Palmer, D. Smirnov, S. Wang, A. Chern, and J. Solomon. 'DeepCurrents: Learning Implicit Representations of Shapes with Boundaries'. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 18665–18675. DOI: https://doi.org/10.1109/CVPR52688.2022.01811 (cited on pages 19, 116).

[PC11]   T. Pock and A. Chambolle. 'Diagonal preconditioning for first order primal-dual algorithms in convex optimization'. In: *2011 International Conference on Computer Vision*. IEEE. 2011, pp. 1762–1769 (cited on page 70).

[Per18]   A.-P. Perkkioe. 'Conjugates of integral functionals on continuous functions'. In: *Journal of Mathematical Analysis and Applications* 459.1 (2018), pp. 124–134. DOI: https://doi.org/10.1016/j.jmaa.2017.10.066 (cited on pages 52, 54).

[Poc+08]   T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers. 'A convex formulation of continuous multi-label problems'. In: (2008), pp. 792–805. DOI: https://doi.org/10.1007/978-3-540-88690-7_59 (cited on page 8).

[Poc+09]  T. Pock, D. Cremers, H. Bischof, and A. Chambolle. 'An algorithm for minimizing the Mumford-Shah functional'. In: *2009 IEEE 12th International Conference on Computer Vision*. IEEE. 2009, pp. 1133–1140 (cited on pages 19, 42, 58, 68, 70, 106, 108).

[Poc+10]  T. Pock, D. Cremers, H. Bischof, and A. Chambolle. 'Global solutions of variational models with convex regularization'. In: *SIAM Journal on Imaging Sciences* 3.4 (2010), pp. 1122–1145. DOI: https://doi.org/10.1137/090757617 (cited on pages 1, 8, 9, 19, 41–43, 50, 52, 58, 68, 71, 106).

[Rah+19]  N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville. 'On the spectral bias of neural networks'. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 5301–5310 (cited on page 117).

[Roc70]  R. T. Rockafellar. *Convex analysis*. Vol. 28. Princeton university press, 1970 (cited on pages 1, 23–25, 93, 103).

[ROF92]  L. Rudin, S. Osher, and E. Fatemi. 'Nonlinear total variation based noise removal algorithms'. In: *Physica D: Nonlinear Phenomena* 60.1-4 (Nov. 1992), pp. 259–268 (cited on pages 4, 76).

[RPK19]  M. Raissi, P. Perdikaris, and G. E. Karniadakis. 'Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations'. In: *Journal of Computational physics* 378 (2019), pp. 686–707. DOI: https://doi.org/10.1016/j.jcp.2018.10.045 (cited on pages 16, 114).

[Rud16]  S. Ruder. 'An overview of gradient descent optimization algorithms'. In: *arXiv preprint arXiv:1609.04747* (2016). DOI: https://doi.org/10.48550/arXiv.1609.04747 (cited on page 110).

[RW09]  R. T. Rockafellar and R. J. Wets. *Variational analysis*. Vol. 317. Springer Science & Business Media, 2009 (cited on pages 1, 23, 57, 92, 93, 122).

[SCC12]  E. Strekalovskiy, A. Chambolle, and D. Cremers. 'A convex representation for the vectorial Mumford-Shah functional'. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2012, pp. 1712–1719. DOI: https://doi.org/10.1109/CVPR.2012.6247866 (cited on page 8).

[Sch+14]  D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling. 'High-resolution stereo datasets with subpixel-accurate ground truth'. In: *German conference on pattern recognition*. Springer. 2014, pp. 31–42. DOI: https://doi.org/10.1007/978-3-319-11752-2_3 (cited on pages 3, 5, 97, 102, 131).

[Sch13]  B. Schweizer. *Partielle Differentialgleichungen: Eine anwendungsorientierte Einführung*. Springer-Verlag, 2013 (cited on page 7).

[SG01]  O. Scherzer and C. Groetsch. 'Inverse scale space theory for inverse problems'. In: *International Conference on Scale-Space Theories in Computer Vision*. Springer. 2001, pp. 317–325 (cited on page 9).

[SGC11]   E. Strekalovskiy, B. Goldluecke, and D. Cremers. 'Tight convex relaxations for vector-valued labeling problems'. In: *2011 International Conference on Computer Vision*. IEEE. 2011, pp. 2328–2335. DOI: https://doi.org/10.1109/ICCV.2011.6126514 (cited on page 8).

[SM17]   S. Sonoda and N. Murata. 'Neural network with unbounded activation functions is universal approximator'. In: *Applied and Computational Harmonic Analysis* 43.2 (2017), pp. 233–268 (cited on pages 107, 136).

[Sto48]   A. H. Stone. 'Paracompactness and product spaces'. In: (1948) (cited on page 54).

[Suz17]   K. Suzuki. 'Overview of deep learning in medical imaging'. In: *Radiological physics and technology* 10.3 (2017), pp. 257–273 (cited on page 16).

[Tan+20]   M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng. 'Fourier features let networks learn high frequency functions in low dimensional domains'. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 7537–7547 (cited on pages 117, 120).

[Val14]   T. Valkonen. 'A primal–dual hybrid gradient method for nonlinear operators with applications to MRI'. In: *Inverse Problems* 30.5 (2014), p. 055012 (cited on page 70).

[VBH16]   C. S. Von Bartheld, J. Bahney, and S. Herculano-Houzel. 'The search for true numbers of neurons and glial cells in the human brain: A review of 150 years of cell counting'. In: *Journal of Comparative Neurology* 524.18 (2016), pp. 3865–3895. DOI: https://doi.org/10.1002/cne.24040 (cited on page 108).

[VL19]   T. Vogt and J. Lellmann. 'Functional liftings of vectorial variational problems with Laplacian regularization'. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2019, pp. 559–571. DOI: https://doi.org/10.1007/978-3-030-22368-7_44 (cited on pages 9, 56).

[Vog+20]   T. Vogt, R. Haase, D. Bednarski, and J. Lellmann. 'On the Connection between Dynamical Optimal Transport and Functional Lifting'. In: *arXiv preprint arXiv:2007.02587* (2020) (cited on page 8).

[Vog19]   T. Vogt. 'Measure Valued Variational Models With Application In Image Processing'. PhD thesis. Universität zu Lübeck, 2019 (cited on pages 52, 53).

[Vog20]   T. Vogt. 'Measure-Valued Variational Models with Applications in Image Processing'. PhD thesis. University of Lübeck, 2020 (cited on pages 8, 9, 41, 52, 71).

[Wil04]   S. Willard. *General Topology*. Courier Corporation, 2004 (cited on page 54).

[Wu+23]   W. Wu, M. Daneker, M. A. Jolley, K. T. Turner, and L. Lu. 'Effective data sampling strategies and boundary condition constraints of physics-informed neural networks for identifying material properties in solid mechanics'. In: *Applied mathematics and mechanics* 44.7 (2023), pp. 1039–1068. DOI: https://doi.org/10.1007/s10483-023-2995-8 (cited on pages 115, 121).

[Xie+22]  Y. Xie, T. Takikawa, S. Saito, O. Litany, S. Yan, N. Khan, F. Tombari, J. Tompkin, V. Sitzmann, and S. Sridhar. 'Neural fields in visual computing and beyond'. In: *Computer Graphics Forum*. Vol. 41. 2. Wiley Online Library. 2022, pp. 641–676. DOI: https://doi.org/10.1111/cgf.14505 (cited on pages 17, 19, 106, 114, 116).

[Xu+19]  D. Xu, C. Ruan, E. Korpeoglu, S. Kumar, and K. Achan. 'Self-attention with functional time representation learning'. In: *Advances in neural information processing systems* 32 (2019) (cited on page 117).

[Ye+22]  Z. Ye, B. Haefner, Y. Quéau, T. Möllenhoff, and D. Cremers. 'Sublabel-accurate multilabeling meets product label spaces'. In: *Pattern Recognition: 43rd DAGM German Conference, DAGM GCPR 2021, Bonn, Germany, September 28–October 1, 2021, Proceedings*. Springer. 2022, pp. 3–17. DOI: https://doi.org/10.1007/978-3-030-92659-5_1 (cited on page 9).

[Ye+23]  Z. Ye, B. Haefner, Y. Quéau, T. Möllenhoff, and D. Cremers. 'A Cutting-Plane Method for Sublabel-Accurate Relaxation of Problems with Product Label Spaces'. In: *International Journal of Computer Vision* 131.1 (2023), pp. 346–362. DOI: https://doi.org/10.1007/s11263-022-01704-7 (cited on page 9).

[ZC08]  M. Zhu and T. Chan. 'An efficient primal-dual hybrid gradient algorithm for total variation image restoration'. In: *UCLA CAM Report* 34 (2008) (cited on page 70).

[Zeu+17]  L. Zeune, G. van Dalum, L. W. Terstappen, S. A. van Gils, and C. Brune. 'Multiscale segmentation via Bregman distances and nonlinear spectral analysis'. In: *SIAM journal on imaging sciences* 10.1 (2017), pp. 111–146. DOI: https://doi.org/10.1137/16M1074503 (cited on page 15).

[Zha+19]  Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu. 'Object detection with deep learning: A review'. In: *IEEE transactions on neural networks and learning systems* 30.11 (2019), pp. 3212–3232 (cited on page 16).

[Zho+19]  E. D. Zhong, T. Bepler, J. H. Davis, and B. Berger. 'Reconstructing continuous distributions of 3D protein structure from cryo-EM images'. In: *arXiv preprint arXiv:1909.05215* (2019) (cited on page 116).

[ZK12]  C. Zach and P. Kohli. 'A convex discrete-continuous approach for Markov random fields'. In: *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*. Springer. 2012, pp. 386–399. DOI: https://doi.org/10.1007/978-3-642-33783-3_28 (cited on page 8).

[ZRL21]  J. Zheng, S. Ramasinghe, and S. Lucey. 'Rethinking positional encoding'. In: *arXiv preprint arXiv:2107.02561* (2021) (cited on page 117).

# Danielle Bednarski

---

**Education**

| | |
|---|---|
| Since 10/2019 | **PhD Student**, *Institute of Mathematics and Image Computing*, University of Lübeck, Lübeck |
| | Advisor: *Prof. Dr. rer. nat. Jan Lellmann* |
| 2018-2019 | **Master's degree program: Mathematics in Medicine and Life Sciences**, *University of Lübeck*, Lübeck, with honours |
| 2014-2018 | **Bachelor's degree program: Mathematics in Medicine and Life Sciences**, *University of Lübeck*, Lübeck, very good |
| 2013 | **Allgemeine Hochschulreife**, *Maria-Ward-Gymnasium*, Nürnberg, with honours |

---

**Awards**

| | |
|---|---|
| 05/2021 | **Best Student Paper Award**, *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, Virtual Event |
| | "Inverse Scale Space Iterations for Convex and Non-Convex Variational Problems" |