Estimation of Motion Vector Fields of Complex Microstructures by Time Series of Volume Images

Tessa Nogatz

Vom Fachbereich Mathematik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades
Doktor der Naturwissenschaften
(Doctor rerum naturalium, Dr. rer. nat)
genehmigte Dissertation

D 386

Erstgutachterin	Prof. Dr. Claudia Redenbach
Zweitgutachter	Prof. Dr. François Hild
Datum der Disputation	05.12.2022

To Falco, my one and only.

Abstract

Mechanical tests form one of the pillars in development and assessment of modern materials. In a world that will be forced to handle its resources more carefully in the near future, development of materials that are favorable regarding for example weight or material consumption is inevitable. To guarantee that such materials can also be used in critical infrastructure, such as foamed materials in automotive industry or new types of concrete in civil engineering, mechanical properties like tensile or compressive strength have to be thoroughly described. One method to do so is by so called in situ tests, where the mechanical test is combined with an image acquisition technique such as Computed Tomography.

The resulting time series of volume images comprise the delicate and individual nature of each material. The objective of this thesis is to present and develop methods to unveil this behavior and make the motion accessible by algorithms. The estimation of motion has been tackled by many communities, and two of them have already made big effort to solve the problems we are facing. Digital Volume Correlation (DVC) on the one hand has been developed by material scientists and was applied in many different context in mechanical testing, but almost never produces displacement fields that allocate one vector per voxel. Medical Image Registration (MIR) on the other hand does produce voxel precise estimates, but is limited to very smooth motion estimates.

The unification of both families, DVC and MIR, under one roof, will therefore be illustrated in the first half of this thesis. Using the theory of inverse problems, we lay the mathematical foundations to explain why in our impression none of the families is sufficient to deal with all of the problems that come with motion estimation in in situ tests. We then proceed by presenting a third community in motion estimation, namely Optical flow, which is normally only applied in two dimensions. Nevertheless, within this community algorithms have been developed that meet many of our requirements. Strategies for large displacement exist as well as methods that resolve jumps, and on top the displacement is always calculated on pixel level. This thesis therefore proceeds by extending some of the most successful methods to 3D.

To ensure the competitiveness of our approach, the last part of this thesis deals with a detailed evaluation of proposed extensions. We focus on three types of materials, foam, fibre systems and concrete, and use simulated and real in situ tests to compare the Optical flow based methods to their competitors from DVC and MIR. By using synthetically generated and simulated displacement fields, we also assess the quality of the calculated displacement fields – a novelty in this area. We conclude this thesis by two specialized applications of our algorithm, which show how the voxel-precise displacement fields serve as useful information to engineers in investigating their materials.

Zusammenfassung

Mechanische Tests repräsentieren eine der wichtigsten Säulen in der Entwicklung und Bewertung moderner Materialien. Um neu entwickelte Materialien auch zuverlässig in kritischer Infrastruktur verwenden zu können, wie etwa der Automobilindustrie oder dem Baugewerbe, bedarf es einer gründlichen Charakterisierung insbesondere in Bezug auf Druck- und Zugfestigkeit. Ein Hilfsmittel hierfür ist durch die sogenannten In Situ Tests gegeben. In diesen Tests werden mechanische Prüfung und ein Bildgebungsverfahren wie Computertomographie kombiniert.

Das Resultat ist eine Zeitreihe von Volumenbildern, in der die individuelle, oft filigrane Reaktion des jeweiligen Material auf Belastungs verborgen ist. Diese Arbeit beschäftigt sich mit der Offenlegung dieses Verhaltens und der algorithmischen Auswertung der resultierenden Bewegungsfelder. Bewegungsschätzung ist ein klassisches Problem der Bildverarbeitung. Insbesondere in zwei Gebieten gibt es bereits umfangreiche Vorarbeiten in 3D. Einerseits sind hier Methoden zu nennen, die sich unter dem Namen Digitale Volumenkorrelation (DVC) zusammenfassen lassen. Entwickelt im Gebiet der Materialwissenschaften existieren bereits diverse Anwendungen auf Bildserien von mechanischen Tests. In der Regel sind diese Methoden aber nicht voxel-präzise, was letztendlich bedeutet, dass Verschiebungen, die nur wenige Voxel betreffen, nicht akkurat geschätzt werden können. Andererseits löst die medizinische Bildregistrierung (MIR) genau dieses Problem, indem sie jedem Voxel Bewegungsinformation zuordnet. Um eine Lösung zu berechnen werden allerdings oft starke Glattheitsannahmen getroffen, die, bezogen auf In Situ Tests, nicht erfüllbar sind.

Beide Ansätze zu vereinheitlichen wird ein Teil dieser Arbeit sein. Die Theorie der inversen Probleme stellt für uns einen ausgezeichneten Rahmen dar um zu argumentieren, warum beide Methoden nicht unseren Ansprüchen an Algorithmen zur Bewegungschätzung in In Situ Tests genügen. Wir greifen deswegen auf eine weitere Gruppe von Algorithem zurück, nämlich die des Optischen Flusses. Ursprünglich ausschließlich in 2D entwickelt, lassen sich trotzdem viele vorteilhafte Eigenschaften ausmachen: Strategien für große Verschiebungen, Abbilden von Sprüngen und Berechnung auf Pixelbasis sind hier nur kurz genannt und werden in der Arbeit ausführlich erläutert. Diese Arbeit beinhaltet deswegen eine Erweiterung einiger besonders geeigneter Algorithmen auf 3D.

Der letzte Teil der Arbeit beschäftigt sich mit einer ausführlichen Analyse und Auswertung inklusive eines Vergleiches mit Konkurrenzansätzen. Hierbei konzentrieren wir uns auf drei Materialtypen: Schäume, Fasersyteme und Beton. Wir verwenden synthetische und echte Bilder von In Situ Tests, um unsere Erweiterungen mit Mitbewerbern aus DVC und MIR zu vergleichen. Des weiteren wird auch die Qualität des berechneten Verschiebungsfeldes auf Basis simulierter Daten ausgewertet – ein Novum in diesem Bereich. Abschließend präsentieren wir zwei spezialisierte Anwendungen unserer Erweiterungen, in denen wir zeigen wie unsere Algorithmen IngenieurInnen in der Erforschung ihrer Materialien unterstützen können.

Acknowledgments

First of all I would like to express my deepest gratitude to my supervisors Prof. Dr. Claudia Redenbach at TU Kaiserslautern and Dr. Katja Schladitz at Fraunhofer ITWM for giving me the opportunity to do my PhD under their guidance. I have been blessed with an extraordinary well supervision and I am grateful for any advice and hint they gave me. Probably without knowledge they managed to encourage me when I needed it the most and I am very thankful that I had the oppurtunity to work with them.

Next, I would like to thank Prof. Dr. François Hild for consenting to be co-referee of my thesis.

This thesis would also not have been possible without all the colleagues in materials science, that provided me with their datasets. I would like to thank Jana Hubalkova and Harry Berek for Glass and MMC foam data. The experiment on the glass foam was supported by the German Research Foundation DFG within Collaborative Research Centre CRC 920 (project ID 169148856), subprojekt S01. In addition, I want to thank Claudia Voigt for providing me with the time series of refractorary concrete. Further, I would like to acknowledge Hannes Grimm-Strele, who provided me with the simulated displacement fields. The second LFT dataset set was acquired within project IGF 19532N 'Efficient Characterisation and Modelling of Anisotropic Failure of LFT for Crash Simulation' of the International Association for Technical Issues (iVTH) funded through the German Federation of Industrial Research Associations (AiF) in the programme for promoting the Industrial Collective Research (IGF) of the Federal ministry for economic Affairs and Energy (BMWi) on the basis of a decision of the German Bundestag, which is also gratefully acknowledged.

Furthermore, I would like to thank all colleagues and coworkers at ITWM and university for giving me such a nice welcome in Kaiserslautern. My office mate Xiaoyin Cheng, who was always in for a coffee or a chat and was always a great help when struggeling with ITWM software, Dascha Dobrowolskij for giving me an introduction to everything, and of course my dearest friend Petra Gospodnetic who always supported be in and out of office, let it be as rubber ducky or dog lender. Not to forget Olena Buchbinder, Martin Braun, Michael Godehardt and Franz Schreiber who were always happy to help with any issues. And of course Konstantin Hauch, Alexander Geng and Tin Barisin for never having to drink coffee alone.

Lastly, I would like to thank my family, and especially my husband Falco. Without you I would have never found the courage to go into research. Without you, nothing of what follows would have been written. I cannot express my gratitude for your unconditional love, your support and your ability to always make me smile, even when I absolutely do not feel to do so.

Contents

List of Figures

List of Tables

List of Symbols

$\mathbb{N}$	Natural numbers
$\mathbb{Z}$	Integers
$\mathbb{R}$	Real numbers
C_0^k	Space of k-times continuously differentiable functions with compact support
C^k	Space of k-times continuously differentiable functions
H^k	Sobolev spaces which are Hilbert spaces, i.e. $H^k = W^{k,2}$
$W^{k,p}$	Sobolev space of functions in L_p whose derivatives up to order k are also L_p
L_p	Space of p-times integrable functions
BV	Space of functions of bounded variation
$\mathscr{N}(T)$	Kernel of an operator T
$\mathscr{R}(T)$	Range of an operator T
$\mathscr{D}(T)$	Domain of an operator T
T	(compact) linear operator between Hilbert spaces
$T^\dagger$	Pseudoinverse of T
F	(nonlinear) operator between Banach spaces
F^*	Convex conjugate of an operator F
$\partial F(u)$	Subdifferential of F at U
prox_F	Proximal map with respect to F
$\mathcal{R}_\alpha$	Regularization operator
$\mathcal{R}$	Regularization functional
$\mathcal{D}$	Distance measure
$\delta_{\mathscr{A}}$	0-∞ characteristic function of a set $\mathscr{A}$
$\chi_{\mathscr{A}}$	0-1 characteristic function of a set $\mathscr{A}$
$\mathscr{E}$	Strain
J, E	Regularization functionals
I_0, I_1	Images
ϕ	Coordinate transformation displaying deformation
$\mathbf{u}$	Displacement vector field
(u, v, w)	Components of displacement vector field $\mathbf{u}$

Abbreviations

3DOF	3D Optical Flow
AAE	Average angular error
AEE	Average endpoint error
ALDVC	Augmented Lagrangian Digital Volume Correlation
DVC	Digital Volume Correlation
CT	Computed Tomography
Elas	Elastic image registration
FE	Finite Elements
IC-GN	Inverse compositional Gauss-Newton
LDDMM	Large deformation diffeomorphic metric mapping
LFT	Long fiber reinforced thermoplastics
lsc	lower semicontinuous
MIR	Medical image registration
MMC	Metal matrix composite
OF	Optical Flow
SBV	Special functions of bounded variation
TGV	Total generalized variation regularization
TV	Total variation regularization
TVMorph	Total variation regularization with morphological multiresolution analysis
RSME	Root mean-squared error
SSD	Sum of squared differences

Chapter 1

Introduction

Computed tomography (CT) is undoubtfully one of the game changers in medicine. With a mathematical theory 60 years older than the actual first device it was possible to image the inside of the human body in 3D. Unlike classical radiography, CT has no overlay of structures and offers a significant contrast of distinct tissue. Since the pioneering work of Radon [98], Cormack [35] and Hounsfield [59], many improvements in all fields of CT have been made. Detector, gantry, tube – none of the mechanical parts of a CT device has *not* been subject to extended research in the past 50 years. But also from an algorithmic point of view, a whole new research area was built. While the classical reconstruction technique *Filtered Back Projection* still serves its duty, iterative algorithms were able to reduce the radiation dose a patient is exposed to significantly.

But the march of victory is not limited to medical uses. From the very beginning, the application of CT to industrial applications was also considered. However, the demands for both applications, medical and industrial, soon began to diverge. Where in the beginning industrial CT would favor from the improvements regarding the resolution, soon medical imaging aimed for the reduction of radiation. Though this is very favorable for in-vivo applications, as radiation harms living organisms, this is almost never true for industrial applications. Even contrary: high radiation CT would provide insight in materials even with high absorption coefficients [18], such as steel-fiber reinforced concrete or metallic foams.

And yet another point marks a crucial difference: CT of living objects requires an as fast as possible acquisition as breathing and heart beat cannot be held. This matter was also tackled by the development of iterative reconstruction algorithms: As they require less data for sufficient reconstruction, not only less radiation was necessary but also less time in image acquisition. However, this again does not hold for materials: we do not expect heavy movement within a sample when a CT image is taken.

Due to these huge differences in requirements for human and industrial CT, soon also specialized devices for the latter emerged. Not bothering for acquisition time and radiation exposure, already in 1982 resolutions within the micrometer scale were described in the literature [39]. This is also the time when the wording *microtomography* evolved, marking the essential difference in small scale resolution. So from now on materials could be imaged with conservation of very small details. Until 2022 various materials have been investigated with the help of computed tomography. Already in 1999, Stock et al. [112] presented an extensive review on the different applications: Porosity in welds, fracture of bonding layers between ceramic plates, carbon fibre based composites, fatigue crack closure in metals, pathways for material transport in porous solids, deformation response in inorganic matrix

composites. Especially composite materials, as they are light-weight but still very durable and heat resistant, have always been an active research field. Metal matrix composites (MMC) for example were already used in the first Space Shuttle programs in the 1970s. Their exact micromechanical failure mechanisms were investigated much later with the help of microtomography [112]. MMCs are still an active research field and a good example how also the testing of materials evolved. Today, MMCs are often produced as foams, not as bulk material. Foam structures are of course much more complex than their solid counterpart. And again, CT plays an important role. With the help of so called in situ tests, exact failure mechanisms can be quantified. In in situ tests, a mechanical test is combined with non-destructive acquisition of 3D images of the specimen under several loading steps using CT. Most common and also initially, mechanical tests like compression, tensile or bending are applied, but also more exotic approaches like battery load can be found [128]. Though at first sight, this violates the assumption that nothing will move during image acquisition, the tests are often performed *quasi static*, i.e. a load is applied and held steady during acquisition. In situ investigations with the help of microtomography started around the early 1990 with the work of Stock et al. [113].

Quite soon after that, the urge for quantification of mechanical tests developed. Therefore, Bay et al. in 1999 [13] proposed a method to describe an in situ compression test by displacement vector fields. The investigated material was trabecular bone, which was compressed in several interrupted steps. Within each of the steps, a CT image was taken. The algorithm, going under the name *Digital Volume Correlation* following its two dimensional predecessor Digital Image Correlation, established a whole new paradigm in describing the outcome of an in situ test. The calculation of the displacement is entirely based on only a pair of images, say the image of the unloaded volume and the first compression step. As we will see later in this thesis, these calculations form an ill-posed problem and cannot be solved without additional constraints. The constraint of choice in the work of Bay et al. was to not calculate a voxelwise displacement, but to divide the image into subvolumes. The subvolumes were supposed to be much smaller than the original image, but much larger than a single voxel. The displacement is then calculated per subvolume. Though the representation of the subvolumes changed over the years, almost all methods going under the name DVC still proceed this way.

However, for the recent challenges in materials science, this procedure may not be sufficient anymore. Recent trends in material design tend towards more and more spatially thin structures, that still maintain their high performance as, again, for example MMC. Foamed structures only take a fracture of the material consumption compared to their bulk counterparts, but still perform remarkably well in macroscopic mechanical tests. Yet the failure mechanism of such a foamed material will heavily depend on the microscale structure – we expect failure of whole layers of the specimen, where the weakest struts fail first.

Unfortunately, when imaged, struts in these materials often only occupy few voxels. And even worse, they are often thinner than the minimal subvolume size of DVC. The exact interplay will be discussed later in this thesis. This discrepancy between the minimal resolution that is offered by DVC and the size of the smallest object of interest in in situ tests does not only limit to foam structures. Microcracks can usually be observed long before concrete macroscopically fails. In fibre-reinforced-plastics, failure is almost always preceded by fibre breakage and/or delamination. Again, when monitored in CT, fibres can only be resolved with few voxels diameter. Therefore, to understand the microscale mechanisms that foreshadow material failure, there is a need for algorithms that resolve displacement on

a voxel scale.

And yet another challenge comes with the estimation of motion in materials science: The observed maximal displacements can be very large. In tests foams for example are usually compressed until the sample can be nearly considered as bulk. This stage in a compression test is also of special interest, as the level of compression when this so called densification happens, belongs to a complete description of a foam's material properties. But for sure the displacement that describes this phenomenon will have its peak at several tenths of voxels.

In this thesis, a group of algorithms will be introduced, that not only feature sufficient accuracy, but also will be able to map the large displacements that occur during a materials test. These algorithms will be based on one of the oldest algorithms in motion estimation, namely Optical Flow, and their superior performance will be shown with well known means from the literature. The performance will also be demonstrated with several examples. Artificial ones for benchmarking, but also representatives for three kinds of materials: foams, fibre systems and concrete.

A General Framework for Motion Estimation

In motion estimation, we regard images as functions, that map from a bounded subset Ω of $\mathbb{R}^3$ to $\mathbb{R}$. For two images I_0 and I_1 of succeeding loading steps, we seek for a transformation $\phi\colon \mathbb{R}^3 \to \mathbb{R}^3$, such that

$$I_0(\mathbf{x}) = I_1(\phi(\mathbf{x})) = I_1 \circ \phi. \tag{1.1}$$

To reconstruct the motion ϕ, we need to fix a function space, not only for the images I_0 and I_1, but also for ϕ itself. The membership of the quantities in some space has however heavy influence on the methods we are allowed to use. Differentiability (weak or strong), smoothness, convexity, all these properties severely affect the outcome of the computation.

A crucial part in answering why motion estimation algorithms perform so differently is to unite them under a single roof. This roof will be inverse problems and their regularization. The classical theory of inverse problems deals with linear operator equations, that is for a linear operator T and given data y, a solution x to the (linear) operator equation

$$Tx = y \tag{1.2}$$

is sought. Here, T is usually a compact linear operator, and the right-hand side consists of some given data y. Finding an inverse $T^\dagger$ now often forms an *ill-posed* problem, so a problem, that violates at least one of the well-posedness conditions of Hadamard [40]:

- For all admissible data, a solution exists.
- For all admissible data, the solution is unique.
- The solution depends continuously on the data.

Obviously, in motion estimation we at least violate the second condition: We try to estimate three dimensional motion from scalar image data. Note that neither Hadamard's conditions are limited to linear compact operator equations, nor will we stay in this very narrow framework. However, these initial findings will be very helpful to understand *how* regularization works and *what* it is able to achieve.

Thesis Outline

This thesis proceeds as follows: In Chapter 2 we will describe the mathematical foundations that will be needed throughout this thesis. Starting with the classical theory of inverse problems in Hilbert spaces, it also includes the generalization to Banach spaces and the treatment of total variation regularization within these. Chapter 3 will give an overview on the current state of the art motion estimation in 3D, which can be generously split into Digital Volume Correlation and Medical Image Registration. In Chapter 4 we introduce another area, in which motion is estimated, namely (2D) Optical Flow, and directly formulate it in three dimensions. Chapter 5 deals with in situ testing, including a literature review on experiments on the investigated materials and their thorough introduction. Chapter 6 is dedicated to experiments. We will not only test how the algorithms perform compared to each other, but also how the different designs of Optical Flow models influence their performance in motion estimation. We conclude this thesis with an outlook to future work and closing remarks in Chapter 7.

Before proceeding we want to remark that parts of this thesis have been published elsewhere:

- Nogatz, T., Redenbach, C., Schladitz, K., 3D optical flow for large CT data of materials microstructures, *Strain*, 2022, 58(3), e12412. `https://doi.org/10.1111/str.12412`.

Chapter 2

Mathematical Preliminaries

This chapter covers all mathematical ingredients that will be needed throughout this thesis. We start with a short introduction into the classical theory of inverse problems to enlighten the difficulties that come with compact operator equations. The classical theory also gives a good intuition what a regularization can achieve. We continue by introducing variational methods for solving minimization problems. This theory offers methods, that allow to consider regularization not only in a Hilbert space setting, but also in the much more suitable Banach space setting. We then turn towards a special representative of regularization, namely bounded or total variation. We conclude the chapter with some specific strategies for solving the problems that arise in this thesis.

2.1 Inverse Problems

The majority of this section follows the monograph of Engl, Hanke and Neubauer [40]. However, Tikhonov regularization is introduced more conveniently in the more recent work of Hanke [52]. Regularization by projection is accurately treated in the monograph of Kirsch [66], which we will also follow in this topic.

2.1.1 Regularization - Definition and Basic Results

For a linear operator T, we define its domain by $\mathscr{D}(T)$, its range by $\mathscr{R}(T)$, and its Kernel by $\mathscr{N}(T)$.

Definition 2.1.1 ([40], 2.1.): Let $T\colon X \to Y$ be a bounded linear operator between Hilbert spaces X and Y. Consider the operator equation

$$Tx = y. \tag{2.1}$$

1. $x \in X$ is called *least-squares solution* of $Tx = y$ if

$$||Tx - y|| = \inf\{||Tz - y|| \mid z \in X\}.$$

2. $x \in X$ is called *best-approximate solution* of $Tx = y$ if x is a least-squares solution of $Tx = y$ and

$$||x|| = \inf\{||z|| \mid z \text{ is a least squares solution of } Tx = y\}$$

Note that in the above definition we did not require the operator equation $Tx = y$ to be solvable at all. However, if we cannot guarantee existence of an exact solution to

$$Tx = y, \tag{2.2}$$

we have a safety net of solutions that are minimal with respect to some norm. Note that nevertheless neither least-squares nor best-approximate solutions have to exist. Luckily, many inverse problems are formulated in Hilbert space settings in which – by restricting the right-hand side y – existence and uniqueness of solutions can be shown. For a linear and bounded operator $T: X \to Y$, where X and Y are Hilbert spaces, this is done by the Moore-Penrose generalized inverse.

Definition 2.1.2 ([40], 2.2.): Let

$$\tilde{T} := T|_{\mathscr{N}(T)^\perp}: \mathscr{N}(T)^\perp \to \mathscr{R}(T).$$

The *Moore-Penrose (generalized) inverse* $T^\dagger$ of a bounded linear operator $T: X \to Y$ is defined as the unique linear extension of $\tilde{T}^{-1}$ to

$$\mathscr{D}(T^\dagger) := \mathscr{R}(T) \oplus \mathscr{R}(T)^\perp$$

with

$$\mathscr{N}(T^\dagger) = \mathscr{R}(T)^\perp,$$

and $\oplus$ being the direct sum.

The Moore-Penrose inverse has a very special role; particularly, one can show that it is the (solution) operator, that maps y to the best-approximate solution of $Tx = y$.

Theorem 2.1.3 ([40], 2.5.): *Let $y \in \mathscr{D}(T^\dagger)$. Then, $Tx = y$ has a unique best-approximate solution, which is given by*

$$x^\dagger := T^\dagger y.$$

The set of all least-squares solutions is $x^\dagger + \mathscr{N}(T)$.

A similar characterization can also be stated for least-squares solutions.

Theorem 2.1.4 ([40], 2.6.): *Let $y \in \mathscr{D}(T^\dagger)$. Then $x \in X$ is a least-squares solution of $Tx = y$ if and only if the* normal equation

$$T^*Tx = T^*y \tag{2.3}$$

holds.

One can show that $y \in \mathscr{D}(T^\dagger)$ is necessary to state any kind of solution term in the sense of Definition 2.1.1. This also means, that in the context of infinite-dimensional operators a best-approximate solution does not necessarily exist – contrary to finite dimensional problems.

We should keep in mind that we develop a theory that must be able to tackle real-world problems. The right-hand side y in equation (2.1) will almost never be available without noise. We can only assume to have disturbed data y^δ with an (unknown) noise level δ, such that

$$||y - y^\delta|| \leq \delta. \tag{2.4}$$

This sheds a light on the fact, why Hadamard considered continuous dependence as one of the core points to mark a well-posed problem: Having perturbed data close to the noise free one, also the perturbed solution should be close to the real one. The following Lemma shows, that in compact operator equations (which many inverse problems count to), the above conclusion is not possible.

Theorem 2.1.5 ([5], Theorem 5.1): *Let X and Y be (infinite dimensional) Banach spaces, and let $T: X \to Y$ be compact and injective. Then, the inversion $T^\dagger : \mathscr{R}(T) \to X$ is unbounded.*

Hence, although $T^\dagger$ is also linear, it cannot be bounded and is therefore not continuous. (The monograph of Appell [5] gives this result in the context of general operator theory. For a result directly related to inverse problems, see [52].) For noisy input data y^δ we can therefore not guarantee that

$$T^\dagger y^\delta \to T^\dagger y \quad \text{for} \quad \delta \to 0, \tag{2.5}$$

and therefore especially not $x^\dagger \to x$.

The remedy to this is to drop the requirement of exact solutions and be content with an approximate, or *regularized solution*.

Definition 2.1.6 ([40], 3.1.): Let $T: X \to Y$ be a bounded linear operator between Hilbert spaces X and Y, $\alpha_0 \in (0, +\infty]$. For every $\alpha \in (0, \alpha_0)$, let

$$R_\alpha: Y \to X$$

be a continuous (not necessarily linear) operator. The family $\{\mathcal{R}_\alpha\}$ is called *a regularization* or *a regularization operator* for $T^\dagger$, if, for all $y \in D(T^\dagger)$, there exists a parameter choice rule $\alpha = \alpha(\delta, y^\delta)$, such that

$$\limsup_{\delta \to 0} \{\, ||\mathcal{R}_{\alpha(\delta, y^\delta)} y^\delta - T^\dagger y|| \quad | \quad y^\delta \in Y, \quad ||y - y^\delta|| \leq \delta \} = 0$$

holds. Here,

$$\alpha: \mathbb{R}^+ \times Y \to (0, \alpha_0)$$

is such that

$$\limsup_{\delta \to 0} \{\, \alpha(\delta, y^\delta) \,| \quad y^\delta \in Y, \quad ||y^\delta - y|| \leq \delta \} = 0.$$

For a specific $y \in \mathscr{D}(T^\dagger)$ satifying the above conditions, a pair $(\mathcal{R}_\alpha, \alpha)$ is called a *(convergent) regularization method* for solving $Tx = y$.

Regularization will be our main ingredient for all motion estimation algorithms presented in the following. Though introduced for Hilbert spaces, the theory can be extended to Banach spaces, although some results, that hold due to the natural structure of the Hilbert space have to be derived in Banach spaces "by hand." This resulting freedom will allow for a variety of different approaches in the choice of the regularization.

A regularization method in the sense of Definition 2.1.6 consists of two ingredients: the regularization operator $\mathcal{R}_{\alpha(\delta, y^\delta)}$ and the corresponding parameter choice rule $\alpha(\delta, y^\delta)$. Until now, the latter one is explicitly dependent on the noise level δ *and* on the perturbed data y^δ. It is also defined for a specific $y \in \mathscr{D}(T^\dagger)$, which marks another (hidden) dependence. However, dependence can also be established *only* based on the noise level. To distinguish this, we introduce the following convention.

Definition 2.1.7 ([40], 3.2.): Let α be a parameter choice rule according to Definition 2.1.6. If α does not depend on y^δ, but only on δ, then we call α an *a-priori parameter choice rule* and write $\alpha = \alpha(\delta)$. Otherwise, we call α an *a-posteriori parameter choice rule*.

Parameter choice rules have to be selected with care, as one can show that rules that dependent *only* on y^δ cannot produce a convergent regularization method.

Now that we know which ingredients are necessary in the treatment for inverse problems, we could ask how to construct them and how to design them, such that they converge, ideally fast. Operators, for example, that converge pointwise to the generalized inverse, i.e.

$$\mathcal{R}_\alpha \to T^\dagger \quad \text{pointwise on} \quad \mathscr{D}(T^\dagger) \quad \text{as} \quad \alpha \to 0, \tag{2.6}$$

form a regularization operator, and one can even find an a-priori parameter choice rule α, such that for every $y \in \mathscr{D}(T^\dagger)$ the regularization method $(\mathcal{R}_\alpha, \alpha)$ is convergent ([40], Proposition 3.4). For regularization methods, where the operator is linear and the parameter choice rule is a-priori, we can also fix the following characteristics:

Proposition 2.1.8 ([40], 3.7.): *Let $\{\mathcal{R}_\alpha\}$ be a linear regularization; for every $y \in \mathscr{D}(T^\dagger)$, let $\alpha: \mathbb{R}_+ \to \mathbb{R}_+$ be an a-priori parameter choice rule. Then $(\mathcal{R}_\alpha, \alpha)$ is a convergent regularization method if and only if*

$$\lim_{\delta \to 0} \alpha(\delta) = 0$$

and

$$\lim_{\delta \to 0} \delta ||\mathcal{R}_{\alpha(\delta)}|| = 0$$

hold.

2.1.2 Regularization by Projection

Regularization by projection is quite a simple and obvious approach to define a regularization in the sense of Definition 2.1.6. Even though the theory of inverse problems is built on operator equations that map between infinite dimensional spaces, many of them will eventually be discretized to fit to a real-world problem. One can now ask if not discretization alone poses a satisfactory regularization. In fact under certain conditions, this is the case, and the mathematical framework to embed is *regularization by projection*.

Definition 2.1.9 ([66], 3.1): Let X be a Banach space. Let $U \subset X$ be a closed subspace. A linear bounded operator $P: X \to X$ is called *projection operator* on U if

- $Px \in U \; \forall \; x \in X$,
- $Px = x \; \forall \; x \in U$.

Together with a projection, we can now define projection methods.

Definition 2.1.10 ([66], 3.4.): Let X and Y be Banach spaces and let $T: X \to Y$ be bounded and one-to-one. Furthermore, let $X_n \subset X$ and $Y_n \subset Y$ be finite dimensional subspaces of dimension n and $Q_n: Y \to Y_n$ be a projection operator. For a given $y \in Y$ the *projection method* for solving the equation $Tx = y$ is to solve the equation

$$Q_n T x_n = Q_n y \text{ for } x_n \in X_n. \tag{2.7}$$

For bases $\{\hat{x}_1, \ldots, \hat{x}_n\}$ and $\{\hat{y}_1, \ldots, \hat{y}_n\}$ of X_n and Y_n, respectively, one can express the ingredients of equation 2.7 via

$$Q_n y = \sum_{i=1}^{n} \beta_i \hat{y}_i \qquad \text{and} \qquad Q_n T \hat{x}_j = \sum_{i=1}^{n} A_{ij} \hat{y}_i, \quad j = 1, \ldots, n,$$

where β_i, A_{ij} are scalar coefficients. Clearly, one can also expand x_n in the same way

$$x_n = \sum_{j=1}^{n} \alpha_j \hat{x}_j.$$

Then, x_n is a solution of equation (2.7) if and only if $\alpha_1, \ldots, \alpha_n$ solve the system of linear equations

$$\sum_{j=1}^{n} A_{ij} \alpha_j = \beta_i, \qquad \text{or as matrix notation} \qquad A\alpha = \beta \tag{2.8}$$

Example 2.1.11:

a) Galerkin method: We slightly tighten the assumptions on X and Y namely to be (pre-) Hilbert spaces. Let Q_n be the *orthogonal* projection, i.e. Q_n satisfies

$$||Q_n y - y|| \leq ||w - y|| \quad \forall w \in Y$$

Because we are considering a pre-Hilbert setting, we can define an inner product, for which equation (2.7) is equivalent to

$$\langle T x_n, z_n \rangle = \langle y, z_n \rangle \quad \forall z_n \in Y_n.$$

The coefficients of the linear system 2.8 are given by

$$A_{ij} \coloneqq \langle T \hat{x}_j, \hat{y}_j \rangle \qquad \text{and} \qquad \beta_i = \langle y, \hat{y}_i \rangle.$$

b) Collocation method: We now come back to the more general Banach space setting and explicitly fix $Y = C[a, b]$ and $T \colon X \to C[a, b]$ bounded. Let $a = t_1 < \ldots < t_n = b$ be an ordered set of given points, the so called *collocation points*, and let $Y_n = \mathcal{S}_1(t_1, \ldots, t_n)$ be the space of linear splines. The projection operator Q_n is then given by an interpolation operator

$$Q_n y = \sum_{j=1}^{n} y(t_j) \hat{y}_j,$$

where the $\hat{y}_j$ are the basis functions of the space of linear splines

$$\hat{y}_j(t) = \begin{cases} \frac{t - t_{j-1}}{t_j - t_{j-1}}, & t \in [t_{j-1}, t_j], \text{ if } j \geq 2 \\ \frac{t_{j+1} - t}{t_{j+1} - t_j}, & t \in [t_j, t_{j+1}], \text{ if } j \leq n - 1 \\ 0, & t \notin [t_{j-1}, t_{j+1}]. \end{cases}$$

The projected operator equation (2.7) is then equivalent to

$$(T x_n)(t_i) = y(t_i) \quad \forall i = 1, \ldots, n.$$

The coefficients of the linear system (2.8) are given by

$$A_{ij} \coloneqq (T \hat{x}_j)(t_i) \qquad \text{and} \qquad \beta_i = y(t_i).$$

Note that a priori, convergence of x_n towards $x^\dagger$ cannot be guaranteed. An example can be found in [40]. Convergence can be achieved, if the following conditions are fulfilled.

Theorem 2.1.12 ([40], 3.20.): *Let $y \in \mathscr{D}(T^\dagger)$ and let x_n be the best-approximate solution of $Tx = y$ in X_n.*

i) x_n converges weakly to $x^\dagger$, i.e. $\langle x_n, w\rangle = \langle x^\dagger, w\rangle$ for all $w \in X$, if and only if $\{||x_n||\}$ is bounded.

ii) $x_n \to x^\dagger$ if and only if $\limsup_{n\to\infty}||x_n|| \leq ||x^\dagger||$.

Instead of considering strong convergence based on boundedness of the solution (which is not very useful, as we do not know the true solution), one can also express convergence conditions with the help of $T^\dagger$.

Proposition 2.1.13 ([40], 3.21.): *Let $y \in \mathscr{D}(T^\dagger)$ and let x_n be the best-approximate solution of $Tx = y$ in X_n. If*

$$\limsup_{n\to\infty}||(T_n^\dagger)^* x_n|| = \limsup_{n\to\infty}||(T_n^*)^\dagger x_n|| < \infty$$

holds, then $x_n \to x^\dagger$.

Once more we needed Hilbert spaces to pose a criterion for convergence, though the actual method can also be formulated in Banach spaces.

Until now we only have considered the convergence of projections given real input data y. We will now show that projections introduce a *hidden* regularization parameter and therefore serve as regularization just in the sense of Definition 2.1.6.

Our perturbed data will be such that

$$||Q_n(y - y^\delta)|| \leq \delta,$$

and by x_n^δ we denote the least-squares solution of

$$T_n x = y_n \quad \text{with} \quad T_n \coloneqq Q_n T,\ y_n \coloneqq Q_n y^\delta. \tag{2.9}$$

Theorem 2.1.14 ([40], 3.26.): *Let $y \in \mathscr{D}(T^\dagger)$ and let 2.9 hold. If $\delta/\mu_n \to 0$ as $\delta \to 0$ and $n \to \infty$, where μ_n is the smallest singular value of T_n, then*

$$x_n^\delta \to x^\dagger \text{ as } \delta \to 0,\ n \to \infty.$$

In the spirit of Proposition 2.1.8, we see that the pseudoinverse of the projected operator $T_n^\dagger$ serves as regularization operator. Recall one of the conditions for a convergent regularization method

$$\lim_{\delta\to 0} \delta||\mathcal{R}_{\alpha(\delta)}|| = 0.$$

Since $||T_n^\dagger|| = 1/\mu_n$, we have

$$\delta||\mathcal{R}_{\alpha(\delta)}|| = \delta||T_n^\dagger|| = \frac{\delta}{\mu_n},$$

which tends towards 0 for $\delta \to 0$ by the assumptions of Theorem 2.1.14. In equation (2.6), we saw that a parameter choice rule can be found if we have convergence of R_α towards $T^\dagger$ for $\alpha \to 0$. But this convergence is naturally given for $n \to \infty$, the role of the parameter choice rule is therefore taken by $1/n$.

Clearly, discretization is nothing else than a projection from infinite to finite (sub)spaces.

2.1.3 Tikhonov Regularization

Recall Definition 2.1.1. Here, a least-squares solution $x \in X$ was characterized by being minimal with respect to $||Tx - y||$, that is

$$||Tx - y|| = \inf\{||Tz - y|| \mid \forall z \in X\}.$$

For disturbed data $||y^\delta - y|| \leq \delta$ we already saw that we cannot guarantee convergence of the disturbed solution towards the real solution. However, considering solutions, such that Tx is closer to y^δ than the actual noise level δ as a starting point for a potential solution seems like an appealing idea. We therefore call the set

$$\mathcal{M}(y^\delta, \delta) = \{x \in X \mid ||Tx - y^\delta|| \leq \delta\}$$

the *set of admissible solution candidates* for some $\delta > 0$. The set will for sure contain the desired best-approximate solution, and in many cases even the true solution to $Tx = y$. Unfortunately, also solutions with undesirably large norms can be a member of this set.

Tikhonov therefore proposed to choose the element of minimal norm from $\mathcal{M}(y^\delta, \delta)$.

Proposition 2.1.15 ([52], 7.1.): *Let $T: X \to Y$ have dense range, $y \in \mathscr{R}(T)$, $y^\delta \in Y$ such that $||y^\delta - y|| \leq \delta$, and $\mathcal{M}(y^\delta, \delta)$ be the set of admissible solution candidates. Then, $\mathcal{M}(y^\delta, \delta)$ is nonempty, closed, and convex and there is exactly one element $x_* \in \mathcal{M}(y^\delta, \delta)$ of minimal norm. Moreover,*

$$||x_*|| \leq ||x||$$

holds for every solution x of $Tx = y$.

Let us now consider the case where available data is not overlaid by noise, i.e. we have $||y^\delta|| \leq \delta$. Then, the *Tikhonov approximation* $x_* \in \mathcal{M}(y^\delta, \delta)$ can be characterized by Gaussian normal equations.

Theorem 2.1.16 ([52], 7.2.): *Let $T: X \to Y$ have dense range, $y \in \mathscr{R}(T)$ and $y^\delta \in Y$ such that $||y^\delta - y|| \leq \delta$ with $||y^\delta|| > \delta$. Then, for the Tikhonov-approximation x_* holds*

$$||Tx_* - y^\delta|| = \delta \tag{2.10}$$

and

$$(T^*T + \alpha_* \mathbb{I})x_* = T^* y^\delta \tag{2.11}$$

for some $\alpha_* > 0$.

Proposition 2.1.17 ([52], 7.3.): *For every $\alpha > 0$ the operator $T^*T + \alpha_* \mathbb{I}$ is invertible with*

$$||(T^*T + \alpha_* \mathbb{I})^{-1}|| \leq 1/\alpha.$$

Therefore, for a given $\alpha_* > 0$ the solution x_* of (2.11) is uniquely determined. Moreover, the solution x^* of (2.11) also solves the unconstrained variational problem

$$\textit{minimize} \;\; \Phi_{\alpha_*}(x) \quad \textit{over} \quad x \in X, \tag{2.12}$$

where $\Phi_{\alpha_*}(x) = ||Tx - y^\delta|| + \alpha_* ||x||^2$.

It remains to show that the method of Tikhonov in fact forms a regularization.

Theorem 2.1.18 ([52], 7.4.): *Let T be injective with dense range in y, and let $\{\mathcal{R}_\alpha\}_{\alpha>0}$ be given by $\mathcal{R}_\alpha = (T^*T + \alpha_*\mathbb{I})^{-1}T^*$. If $y \in \mathscr{R}(T)$ and $\{y^\delta\}_{\delta>0} \subset Y$ is a family of approximations such that $||y^\delta - y|| \leq \delta$, and if $\alpha = \alpha(\delta)$ is chosen such that*

$$\alpha(\delta) \to 0 \qquad \textit{and} \qquad \delta^2/\alpha(\delta) \to 0 \qquad \textit{as} \qquad \delta \to 0,$$

then the approximations $\mathcal{R}_{\alpha(\delta)}y^\delta$ converge to $T^\dagger y$ as $\delta \to 0$.

2.1.4 Landweber Iteration

For finite systems of linear equations, the Richardson iteration uses a fixed point formulation to deduce an iterative solution method. We can derive a similar scheme in the infinite dimensional case. As starting point, we choose the normal equation (2.3) and formulate the following equivalent fixed point equation

$$x = x + T^*(y - Tx). \tag{2.13}$$

The fixed point operator $\mathbb{I} - T^*T$ is non-expansive if $||T||^2 < 2$. Note that it cannot be a contraction, we will however show that we can nevertheless reconstruct solutions in the regularization framework. The *Landweber iteration* is now given by the following rule

$$x_k^\delta = x_{k-1}^\delta + T^*(y^\delta - Tx_{k-1}^\delta), \quad k \in \mathbb{N}, \tag{2.14}$$

where the starting value x_0^δ is given by some initial guess x_*, which can be without loss of generality be assumed to be 0.

We now first establish the convergence of this iterative method for unperturbed data y, that is, the iteration reads

$$x_k = x_{k-1} + T^*(y - Tx_{k-1}), \quad k \in \mathbb{N}.$$

Theorem 2.1.19 ([40], 6.1.): *If $y \in \mathscr{D}(T^\dagger)$, then $x_k \to T^\dagger y$ as $k \to \infty$. If $y \notin \mathscr{D}(T^\dagger)$, then $||x_k|| \to \infty$ as $k \to \infty$.*

The following lemma gives a simple estimate on how strong the divergence of the Landweber iteration can be for perturbed data y^δ.

Lemma 2.1.20 ([40], 6.2.): *Let y, y^δ be a pair of right-hand side data with $||y^\delta - y|| \leq \delta$, and let $\{x_k\}$ and $\{x_l^\delta\}$ be the corresponding two iteration series by* (2.14)*. Then we have*

$$||x_k - x_k^\delta|| \leq \sqrt{k}\delta, \qquad k \geq 0.$$

What does that mean for the Landweber iteration? We can split the total error of the Landweber iteration the following way

$$\begin{aligned} ||T^\dagger y - x_k^\delta|| &= ||T^\dagger y - x_k + x_k - x_k^\delta|| \\ &\leq \underbrace{||T^\dagger y - x_k||}_{=:e_1} + \underbrace{||x_k - x_k^\delta||}_{=:e_2} \end{aligned} \tag{2.15}$$

Theorem 2.1.19 taught us that we can expect convergence for perturbed data when δ tends towards zero. If now the first Landweber iterations are computed, one can observe that the

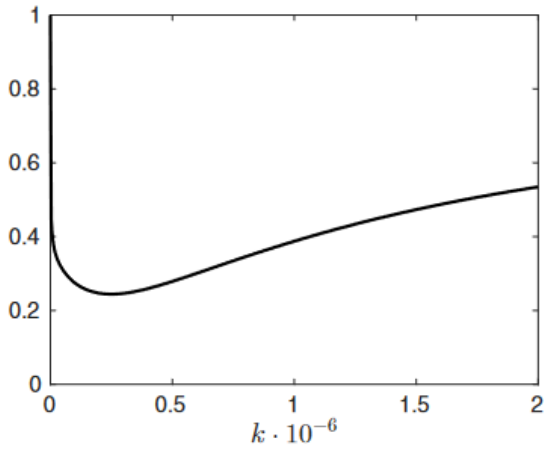


Figure 2.1: Phenomenon of semiconvergence. For the first iterations (denoted by k), the residual indicates convergence towards the true solution, before the influence of the data error takes over and causes divergence. Plot taken from [52].

residual is decreased – the procedure seems to converge to the true solution $T^\dagger y$. e_1 in (2.15) tends towards 0 and e_2 is negligible. But if we perform more and more iterations, the data error e_2 will grow and the method will produce worse approximations, see Figure 2.1.

This phenomenon, called *semiconvergence*, is often observed in iterative procedures to solve inverse problems. It is therefore of crucial importance to carefully observe the optimization process and stop accordingly.

2.1.5 Regularization for Nonlinear Operators

Recall, that in the case of Tikhonov regularization the relation between the variational formulation

$$\min_x ||Tx - y^\delta||^2 + \alpha||x||^2 \tag{2.16}$$

and

$$(T^*T + \alpha\mathbb{I})x_\alpha^\delta = T^*y^\delta \tag{2.17}$$

was by first order optimality. That this, (2.17) is a necessary condition for x to be a solution of (2.16). Because the variational formulation is also convex, the condition is also sufficient for x to solve (2.16) *globally*. This procedure can be immediately generalized to the nonlinear case. We (for now) consider a nonlinear operator $F\colon \mathscr{D}(F) \subset X \to Y$ between Hilbert spaces X and Y. Ill-posedness in this setting means, that the solution does not depend continuously on the data. For a proper treatment as in the previous sections, one has to assume at least some regularity of F (for example continuity and weakly (sequentially) closedness, see [40], p. 241).

Instead of minimum-norm or best-approximate solutions, we will consider x^*-minimum norm solutions $x^\dagger$. They are defined as closest solutions to some initial guess, i.e. $x^\dagger$ is a minimum-norm solution to x^*, if

$$F(x^\dagger) = y$$

and

$$||x^\dagger - x^*|| = \min\{||x - x^*|| \mid F(x) = y\}.$$

As in the linear case, such solutions do not have to exist. Hence, we assume in this section that a x^*-minimum norm solution $x^\dagger$ exists for data $y \in Y$. A condition of ill-posedness similar to Theorem 2.1.5, so a condition, when the inverse $F^\dagger$ is not continuous, is given by local injectivity around $x^\dagger$, provided that $x^\dagger$ is infinite dimensional.

For convergence results involving nonlinear equations, we refer to Engl [40]. We now want to focus on Tikhonov-like approaches to solve the nonlinear problem. That is, we formulate the variational equivalent to an x^*-minimal solution, i.e.

$$x_\alpha^\delta \in \operatorname*{arg\,min}_{x \in X} \left(\underbrace{||F(x) - y^\delta||^2 + \alpha||x - x^*||^2}_{=: J_\alpha(x)} \right). \tag{2.18}$$

Note that in general the above functional without any additional assumptions on F cannot be assumed to be convex. Moreover, if a solution exists, it might not be a global but only a local one. However, our regularization will only aim for global solutions. Frechet-differentiability of F can be transfered to the functional, so if F is Frechet-differentiable, so is J_α, and the same holds for F' and J'_α. We can therefore compute a derivative and verify, that a first order solution has to satisfy

$$F'(x_\alpha^\delta)^* \left(F(x_\alpha^\delta) - y^\delta \right) + \alpha(x_\alpha^\delta - x^*) = 0, \tag{2.19}$$

which, as we are in the nonlinear case, has now to be equipped with a sufficient condition for existence of a minimum. Existence of such a minimum is however already guaranteed by the continuity and the weak sequential closedness of F.

Theorem 2.1.21 ([106], 3.3.): *Let $F: X \to Y$ be continuous and weakly sequentially closed. Then, there exists a minimizer $x_\alpha^\delta \in X$ of the functional J_α defined by (2.18).*

Not only the Tikhonov-approach can be extended to nonlinear problems. The Landweber iteration for nonlinear problems reads

$$x_k^\delta = x_{k-1}^\delta + F'(x_{k-1}^\delta)^*(y^\delta - F(x_{k-1}^\delta)),\ k \in \mathbb{N}. \tag{2.20}$$

As opposed to Tikhonov regularization, there are no global convergence guarantees for the nonlinear Landweber iteration in general. One has to put very strong assumptions on F to achieve results, and a thorough description of this matter can be found in Engl [40].

One could also consider to use Newton type methods to solve the nonlinear problem. That is, in each step we perform a local linearization of F and solve the (then) linear equation

$$F'(x_k^\delta)(x_{k+1}^\delta - x_k^\delta) = y^\delta - F(x_k^\delta). \tag{2.21}$$

Though we now have to tackle a linear system, we still remain with an ill-posed one, as F', the derivative of a compact and continuous operator, is compact itself. Treatment via for example Tikhonov regularization gives

$$x_{k+1}^\delta = x_k^\delta + \left(F'(x_k^\delta)^* F'(x_k^\delta) + \alpha_k \mathbb{I} \right)^{-1} F'(x_k^\delta)^*(y^\delta - F(x_k^\delta)). \tag{2.22}$$

This method is known as the *Levenberg-Marquardt-Method*, which we will revisit in the next section in more detail in the linear case. Note that via linearization of the Tikhonov functional (2.16)

$$||y^\delta - F(x_k^\delta - F'(x_k^\delta))(z - x_k^\delta)||^2 + \alpha_k||z - x_k^\delta||$$

we arrive at the same iteration as in (2.22), when we minimize the quadratic functional for $z = x_{k+1}^\delta$.

2.2 Variational Methods

Having derived the foundation of inverse problems and considering our motion estimation equation (1.1), we immediately see that *linear* operator equations are not suitable to describe our problem. A workaround was already given in Section 2.1.5, where we transformed the nonlinear operator into a linear one by Taylor approximation. However, this is not the only nonlinearity that can occur. In fact for motion estimation it is even favorable to consider nonlinear regularization operators. They allow for solutions that exhibit jumps – a behavior that we will most certainly find in our motion fields when we consider in situ tests. The use of nonlinear functionals for regularization often comes with the switch from Hilbert to Banach spaces. Unfortunately we lose a very important property then: The existence of the generalized inverse $T^\dagger$, and convergence of a regularization by convergence of R_α to $T^\dagger$. We therefore need to introduce more sophisticated tools to prove convergence of regularization methods. The variational formulation will now be treated by calculus of variations, which we will shortly introduce before specifying its relevance for regularization in Banach spaces.

The content of the following section mainly follows three monographs. Subsection 2.2.1 follows the book of Aubert and Kornprobst [7]. Regularization in Banach spaces (Subsection 2.2.2) is treated as in the book of Scherzer [106]. A standard work in Optimal Control of Partial Differential Equations is the book of Tröltzsch [118], from which the parts presented here are also taken.

2.2.1 Direct Method

In Section 2.1.3 we showed that regularization can often be transformed to a minimization problem. This concept is not only very common in inverse problems (so in compact operator equations), but also in solving partial differential equations. Often, problems can be cast into functional equations of the form

$$F(u) = 0 \in \mathbb{R}^n,$$

where $F\colon V \to \mathbb{R}^n$, and u is the sought solution. If it is now possible to define a function $J\colon \mathbb{R}^n \to \mathbb{R}$, such that $\nabla J = F$, and such that J has a minimizer, we could equivalently search for a minimizer of J, i.e.

$$J(u) = \min_{v \in \mathbb{R}^n} J(v).$$

This duality between minimization and equation solving may be considered as one of the starting points of variational calculus. Take for example the Dirichlet problem for the Poisson equation in an open bounded set $\Omega \subset \mathbb{R}^n$

$$\begin{aligned} -\Delta u &= f \text{ in } \Omega \\ u &= g \text{ on } \partial\Omega. \end{aligned} \tag{2.23}$$

Riemann established the connection to the functional

$$J(v) = \int_\Omega \frac{1}{2}|\nabla v(x)|^2 - f(x)v(x)dx. \tag{2.24}$$

Specifically, by setting

$$K = \{v \mid v \in C^2(\bar{\Omega}), v(x) = g(x) \quad \text{for all} \quad x \in \partial\Omega\},$$

one can show that solving (2.23) and minimizing (2.24) over K is equivalent, if f and g are sufficiently smooth. Calculus of variations now generalizes this principle to functionals of the form

$$J(v) = \int_\Omega L(x, v(x), \nabla v(x))\, \mathrm{d}\mathbf{x}.$$

However, the Dirichlet-principle does not guarantee existence of a solution in any means. Moreover, it is not a trivial task at all: The minimization is performed in a subspace of infinite dimension.

Though implicitly assumed by Riemann in the specific case of the Poisson equation, it was to Hilbert in 1900 to formalize the proof of existence in his famous *direct method of calculus of variations*. We will shortly introduce the necessary preliminaries from functional analysis before presenting the method.

From now on U will denote a Banach space equipped with the norm $||\cdot||$, and its topological dual by U'. A norm on U' for some $l \in U'$ is given by

$$||l||_{U'} = \sup_{u \neq 0} \frac{||l(u)||}{||u||_U}.$$

We will endow U with two classical topologies.

Definition 2.2.1 ([7], Def. 2.1.1.):

1. The *strong topology*, denoted by $u_n \underset{U}{\longrightarrow} u$ is defined by $||u_n - u||_U \to 0$ for $n \to \infty$.
2. The *weak topology*, denoted by $u_n \underset{U}{\rightharpoonup} u$ is defined by $l(u_n) \to l(u)$ for $n \to \infty$ and for every $l \in U'$.

We can also define the dual U' with strong and weak topologies.

Definition 2.2.2 ([7], Def. 2.1.2):

1. The *strong topology*, denoted by $l_n \underset{U'}{\longrightarrow} l$ is defined by $||l_n - l||_{U'} \to 0$ or equivalently $\sup_{u \neq 0} \frac{||l_n(u) - l(u)||}{||u||_U} \to 0$ for $n \to \infty$.
2. The *weak topology*, denoted by $l_n \underset{U'}{\rightharpoonup} l$ is defined by $z(l_n) \to z(l)$ for $n \to \infty$ and for every $z \in (U')'$, the bidual of U.
3. The *weak-* topology*, denoted by $l_n \overset{*}{\underset{U'}{\rightharpoonup}} l$ is defined by $l_n(u) \to l(u)$ for $n \to \infty$ and for every $u \in U$.

The weak-* topology is interesting, as it allows to deduce a compactness results in Banach spaces that are only separable (contain a countable dense subset), but not reflexive. A Banach space is said to be reflexive if $(U')' = U$.

Theorem 2.2.3 ([7], Thm. 2.1.1.):

1. *Let U be a reflexive Banach space, let $K > 0$ and $u_n \in U$ a sequence such that $||u_n||_U \leq K$, then there exists $u \in U$ and a subsequence u_{n_j} of u_n such that $u_{n_j} \underset{U}{\rightharpoonup} u$ for $n \to \infty$.*

2. *Let U be a separable Banach space, let $K > 0$ and let $l_n \in U'$ such that $||l_n||_{U'} \leq K$, then there exist an $l \in U'$ and a subsequence l_{n_j} of l_n such that $l_{n_j} \xrightarrow[U']{*} l$ for $n \to \infty$.*

Another important property which our functionals will need to fulfill is *lower semi-continuity*.

Definition 2.2.4 ([7], Def 2.1.3): F is called *lower semi-continuous* (l.s.c.) for the weak topology if and only if for all sequences $u_n \rightharpoonup u_0$ we have

$$\lim_{u_n \rightharpoonup u_0} F(u_n) \geq F(u_0).$$

The same definition can be given with the strong topology.

We now consider the minimization of a function $J: U \to \mathbb{R}$ over a Banach space U

$$\inf_{v \in U} J(v).$$

The existence of a solution of such a problem can be proved via the classical direct method of calculus of variations. It consists of three steps.

1. One constructs a *minimizing sequence* $v_n \in U$, i.e. a sequence satisfying $\lim_{n\to\infty} J(v_n) = \inf_{v\in U} J(v)$.

2. If J is *coercive*, so if $\lim_{||v||\to\infty} J(v) = +\infty$, one can obtain a uniform bound $|v_n|_U \leq C$. If U is reflexive, we can, due to the previous theorem, deduce the existence of $v_0 \in U$ and of a weakly convergent subsequence v_{n_j}, i.e. such that $v_{n_j} \xrightarrow[U]{} v_0$.

3. If the inequality $\lim_{v_{n_j} \xrightarrow[U]{} v_0} J(v_{n_j}) \geq J(v_0)$ holds, we deduce that v_0 is a minimizer and that $J(v_0) = \inf_{v\in U} J(v)$.

Note that this method for existence of minimizers is quite a general one. In image processing however, we are often able to write J as a quadratic functional, that is

$$J(v) = \frac{1}{2}a(v, v) - F(v), \tag{2.25}$$

where a is a bilinear form on $U \times U$ and F a linear form on U. We can again show an equivalent characterization of solutions, namely that

$$J(u) = \min_{v\in U} J(v) \quad \Leftrightarrow \quad a(u, v) = F(v)\ \forall v \in U,$$

holds under certain conditions. It allows to apply the theorem of Lax and Milgram, that guarantees existence *and* uniqueness of a solution.

Theorem 2.2.5: *Let U be a Banach space, let $a: U \times U \to \mathbb{R}$ be a continuous U-elliptic bilinear form and let $F: U \to \mathbb{R}$ be linear and continuous. Then the variational equation*

$$a(u, v) = F(v) \quad \forall v \in U, \tag{2.26}$$

has a unique solution $u \in U$.

The attractiveness of this theorem comes from the fact, that by integration by parts all elliptic differential equations can be brought into the form of equation (2.26). Take for example again problem (2.23). Multiplying both sides by test functions which are 0 in $\partial\Omega$ and performing integration by parts gives

$$\int_\Omega \langle \nabla u(x), v(x) \rangle \, \mathrm{d}\mathbf{x} = \int_\Omega f(x) v(x) \, \mathrm{d}\mathbf{x}.$$

Existence and uniqueness of solutions of elliptic partial differential equations therefore nail down to fulfilling the conditions of Theorem 2.2.5. The test functions therefore have to be chosen, such that the bilinear form remains elliptic and continuous. Usually, this is achieved by using functions from Sobolev spaces. But the theorem does not only offer this theoretic result, it also immediately gives an ansatz for the numerical treatment of PDEs, namely the so called *Galerkin method*. Instead of choosing functions from an infinite dimensional Sobolov space, one could choose a finite dimensional subspace U_h and seek an approximation u_h to u such that

$$a(u_h, v) = f(v) \quad \forall v \in U_h. \tag{2.27}$$

Because U_h is now of finite dimension, there is a basis $\{\phi_1, \ldots, \phi_n\}$ and u_h has a representation

$$u_h = \sum_{i=1}^{n} u_i \phi_i,$$

and an approximate solution is given by the solution of the *linear* system of equations

$$A\mathbf{u}_h = b$$

with

$$A = [a(\phi_i, \phi_j)]_{ij} \in R^{n \times n}, \quad b = [f(\phi_j)]_j \in \mathbb{R}^n.$$

Remark 2.2.6: The careful reader will notice that we already introduced "a" Galerkin method in Section 2.1.2, which dealt with regularization by projection. It completely resembles the approach in the context of partial differential equations: first introduce a bilinear form, next find a suitable approximation to the solution and lastly solve a linear system. However, both methods suffer from the same problems: The finer the approximation is chosen, the closer we will be at a true solution, but at the same time the number of unknowns will grow and the linear system will be harder to solve.

2.2.2 Regularization in Banach Spaces

In the spirit of the Direct Method of calculus of variations, we are now able to formulate regularization in a Banach space setting. Recall once again that because our operators now live in a Banach space, we cannot fall back on the singular value decomposition anymore to prove the desired properties of a regularization. A consequence to this is the fact that regularization in Banach spaces almost always is formulated as minimization problem in the spirit of Tikhonov. Once more we will consider a (nonlinear) operator equation $F(u) = y$, but $F: U \to Y$ will map *from* a Banach space U *to* a Banach space Y. We will generalize the search for a solution u to

$$\min_v J(v), \quad \text{where} \quad J(v) = \mathcal{D}(v, y^\delta) + \alpha \mathcal{R}(v), \tag{2.28}$$

where $\mathcal{D}$ denotes a distance function that correlates $F(u)$ to the disturbed input data y^δ and $\mathcal{R}$ a regularization function to overcome the ill-posedness. To show that a general variational formulation like (2.28) regularizes an ill-posed problem, we dismantle the requested properties of a regularization into the following, and name the equivalent results in the Tikhonov case:

1. Existence: A functional as in equation (2.28) has to admit a solution for every $\alpha > 0$ and every $y \in Y$. This resembles the non-emptyness of $M(y^\delta, \delta)$ in Proposition 2.1.15.

2. Stability: For fixed $\alpha > 0$ and disturbed input data y^δ let u_α^δ denote a solution to (2.28). y^δ then has to depend continuously on u_α^δ. This resembles (2.10) in Theorem 2.1.16.

3. Convergence: For $\alpha \to 0$ and $v^\delta \to v$, we require the regularized solution u_α^δ to converge to the true solution u of $F(u) = y$. This resembles Theorem 2.1.18.

4. Convergence rates and stability estimates can also be of interest, but will not be treated here.

To show that a regularization in Banach spaces fulfills these properties, we have to assume the following.

- The Banach spaces U and Y are associated with topologies τ_U and τ_Y that are weaker than the norm topologies.
- $\mathcal{D}$ is given by $||\cdot||_U^p$ with $p \geq 1$.
- $||\cdot||_Y$ is sequentially lower semi-continuous with respect to τ_Y.
- $\mathcal{R}: U \to [0, \infty]$ is convex and sequentially lower semi-continuous with respect to τ_U.
- $\mathscr{D}(F) \cap \mathscr{D}(\mathcal{R}) \neq \emptyset$.
- For every $\alpha > 0$, $M > 0$ the level sets $\mathcal{M}_\alpha(M) := \{v \in U \mid J(v) \leq M\}$ are sequentially pre-compact with respect to τ_U.

Remark 2.2.7: In the Hilbert space setting some of the above assumptions are either naturally given or equivalent among themselves. For a thorough derivation see Remark 3.14 in [106].

The above assumptions are now sufficient to prove that J has a minimizer for $\alpha > 0$ and $v^\delta \in U$. Stability is given by the following theorem.

Theorem 2.2.8 ([106], 2.23.): *Let $\mathcal{D}, \mathcal{R}, \mathscr{D}, U, Y$ be as above. If (y_k) is a sequence converging to y^δ in Y with respect to the norm topology, then every sequence (u_k) with*

$$u_k \in \operatorname*{arg\,min}_{v \in \mathscr{D}} J(v)$$

has a subsequence that converges with respect to τ_U. The limit of every convergent subsequence $(u_{k'})$ of (u_k) is a minimizer u of $J(u)$ and $(\mathcal{R}(u_{k'}))$ converges to $\mathcal{R}(u)$.

To state convergence, we need to generalize our notion of solution. We call $u^\dagger \in \mathscr{D}$ an $\mathcal{R}$-minimizing solution, of $F(u) = y$, if $F(u^\dagger) = y$ and

$$\mathcal{R}(u^\dagger) = \min\{\mathcal{R}(v) \,|\, v \in \mathscr{D}(F), F(u) = y\}.$$

One can show that every solution is also an $\mathcal{R}$-minimizing solution. Convergence is now given by the following theorem.

Theorem 2.2.9 ([106], 3.26): *Let $\mathcal{D}, \mathcal{R}, \mathscr{D}, U, Y$ be as above. Assume that a solution to $F(u) = y$ exists (which then is also $\mathcal{R}$-minimizing) and that $\alpha\colon (0,\infty) \to (0,\infty)$ satisfies*

$$\alpha(\delta) \to 0 \quad \textit{and} \quad \frac{\delta^p}{\alpha(\delta)} \to 0 \textit{ as } \delta \to 0.$$

Moreover, assume that the sequence (δ_k) converges to 0, and that $y_k := y^{\delta,k}$ satisfies $||y - y_k|| \leq \delta_k$.
Set $\alpha_k := \alpha(\delta_k)$. Then every sequence (v_k) of elements minimizing J has a subsequence $(v_{k'})$ that converges with respect to τ_U. The limit $u^\dagger$ of every τ_u-convergent subsequence $(v_{k'})$ is an $\mathcal{R}$-minimizing solution of $F(u) = y$ and $\mathcal{R}(v_k) \to \mathcal{R}(u^\dagger)$. If, in addition, the $\mathcal{R}$-minimizing solution $u^\dagger$ is unique, then $v_k \to u^\dagger$ with respect to τ_U.

2.2.3 Optimal Control of Partial Differential Equations

There is a third mathematical area, that uses reformulation as unconstrained minimization problem, namely the area of *optimal control of PDEs*. An optimal control problem is a constrained minimization problem

$$\begin{aligned} &\min_{y,u} \; J(y,u) \\ &\text{s.t.} u \in U_{\text{ad}} \\ &\text{and } y = Su \end{aligned}$$

where the *state* y is governed by the *control* u, and u stems from a set of admissible functions U_{ad}. Though the minimization is performed with respect to both state *and* control, we are interested in a minimal state such that the control is minimal as well. For the sake of simplicity, we will derive the following results in terms of finite dimensional optimal control, that is, we consider the problem

$$\min_{y,u} \; J(y,u) \tag{2.29}$$

$$\text{s.t.} u \in U_{\text{ad}} \tag{2.30}$$

$$\text{and } Ay = Bu, \tag{2.31}$$

where A and B are finite dimensional operators, i.e. matrices. The extension to optimal control in infinite dimensions is rather straight forward and can, for example, be found in the monograph of Tröltzsch [118]. Existence of optimal control can for example be shown by setting $S = A^{-1}B$ and eliminating J in (2.29) by setting

$$J(y,u) = J(Su,u) =: f(u).$$

The *reduced problem*

$$\min f(u), \qquad u \in U_{\text{ad}}$$

then has a solution, if J is continuous, if U_{ad} is nonempty, bounded and closed, and if A is invertible. However, developing existence and construction of solutions based on S might be wasted time, as computing the inverse often is too costly. One will therefore often consider the *adjoint state* $\bar{p}$, that is a solution to the *adjoint equation*

$$A^T \bar{p} = \nabla_y J(\bar{y}, \bar{u}),$$

where $(\bar{y}, \bar{u})$ is an optimal pair of (2.29). The first step towards an unconstrained minimization problem is now to define a Lagrange-function $L(y, u, p)$, that also depends on the adjoint state p, as

$$L(y, u, p) \coloneqq J(y, u) - \langle Ay - Bu, p \rangle.$$

Note that at this point the formulation is only equivalent, if we still pose the constraint $u \in U_{\text{ad}}$. Usually, U_{ad} will consist of *box constraints*, that is

$$U_{\text{ad}} = \{u \in \mathbb{R}^m \mid u_a \leq u \leq u_b\}$$

or in the case of an infinite dimensional space U

$$U_{\text{ad}} = \{u \in U \mid u_a \leq u(x) \leq u_b\}.$$

In this case, we define

$$\begin{aligned} \mu_a &= \left(B^T \bar{p} + \nabla_u J(\bar{y}, \bar{u})\right)_+ \\ \mu_b &= \left(B^T \bar{p} + \nabla_u J(\bar{y}, \bar{u})\right)_- , \end{aligned}$$

which puts μ_a and μ_b componentwise to 0, if the right-hand side is positive or negative, respectively. This allows to formulate the celebrated Karush-Kuhn-Tucker (KKT) conditions, which state that for optimal control $\bar{u}$ and associated state $\bar{y}$, Lagrange-multipliers $\bar{b}, \mu_a, \mu_b$ exist and fulfill the following conditions

$$\begin{aligned} &\nabla_y L(\bar{y}, \bar{u}, \bar{p}, \mu_a, \mu_n) = 0 \\ &\nabla_u L(\bar{y}, \bar{u}, \bar{p}, \mu_a, \mu_n) = 0 \\ &\mu_a \geq 0, \quad \mu_b \geq 0 \\ &\langle u_a - \bar{u}, \mu_a \rangle = \langle \bar{u} - u_b, \mu_b \rangle = 0. \end{aligned}$$

Note that in the case of linear elliptic PDEs, the KKT-conditions are not only necessary but also sufficient conditions.

2.3 Convex Analysis

Let us now turn once more to the minimization of a functional F

$$F(u) = \inf_{v \in \Omega} F(v). \tag{2.32}$$

So far, we only dealt with very well-behaved functions, meaning that ∇F could always be computed to pose necessary conditions for a minimizer. Furthermore, in the case of compact operator equations, F was convex (so $\nabla F = 0$ was also a sufficient condition for a minimizer), or in the case of systems of PDEs, J led to a bilinear system that fulfilled the

conditions of Lax-Milgram to get existence, uniqueness and solution strategies. Implicitly, we therefore assumed a certain amount of regularity, namely that our problem is at least Gâteaux-differentiable. The next section will for one moment forget about the origin of F (or J), and will answer the question if we can lower the differentiability requirements as long as we guarantee convexity.

This section mainly follows [30], with some definitions and notions taken from the standard references [38] and [100].

2.3.1 Tools in Convex Analysis

The objectives of convex analysis is for sure convex functions, which can be introduced as in any basic calculus class.

Definition 2.3.1 ([38], I.2.1.): Let $\mathscr{A}$ be a convex subset of U, and F a mapping from $\mathscr{A}$ into $\overline{\mathbb{R}}$. F is said to be *convex*, if for every u and v in $\mathscr{A}$ we have

$$F(\lambda u + (1-\lambda)v) \leq \lambda F(u) + (1-\lambda)F(v), \tag{2.33}$$

for $\lambda \in [0,1]$, whenever the right-hand side is defined. It is said to be *strictly convex* if the strict inequality holds for $\lambda \in (0,1)$.

Note that this definition only holds for proper functions, i.e. functions that nowhere attain infinity. In the case of improper functions, we can define convexity with the help of the epigraph of a function.

Definition 2.3.2 ([38], I.2.2.): The *epigraph* of a function $F\colon V \to \overline{\mathbb{R}}$ is the set

$$\text{epi } F = \{(u,a) \in V \times \mathbb{R} | F(u) \leq a\}. \tag{2.34}$$

A function is convex if and only if its epigraph is convex. Usual operations are inherited, meaning that sums of convex functions are convex, and the pointwise supremum of a family of convex functions if also convex. Recall Definition 2.2.4, where we introduced the notion of a *lower semi-continuous* function. There are many equivalent ways to characterize lsc, some of them are

1. $\forall a \in \mathbb{R}\{u \in U | F(u) \leq a\}$ is closed
2. $\forall u_0 \in U,\ \lim_{u \to u_0} F(u) \leq F(u_0)$
3. $u \in U$, if $u_n \to u$, then $F(u) \leq \liminf_{n\to\infty} F(u_n)$

Example 2.3.3: A very easy and much used example of a convex function is the so called *indicator function*

$$\delta_{\mathscr{A}}(u) = \begin{cases} 0, & \text{if } u \in \mathscr{A} \\ +\infty, & \text{if } u \notin \mathscr{A}. \end{cases}$$

If $\mathscr{A}$ is convex, so is its indicator function.

One of the most important tools in convex analysis is the following.

Definition 2.3.4 ([38], I.4.1.): Let U, U^* be duals of each other connected by the dual pairing $< \cdot, \cdot >$. If $F: U \to \overline{\mathbb{R}}$, we call

$$F^*(y) = \sup_{u \in U} < u, y > -F(u),$$

conjugate function or *Legendre-Fenchel conjugate* of F.

The conjugate of the indicator function $\delta_{\mathscr{A}}$ is the so called *support function* of $\mathscr{A}$

$$\begin{aligned} \delta^*_{\mathscr{A}}(u^*) &= \sup_{u \in V} \{< u, u^* > -\delta_{\mathscr{A}}(u)\} \\ &= \sup_{u \in \mathscr{A}} < u, u^* > \end{aligned}$$

An important example of convex conjugates are those of the norms, that we will use throughout this manuscript.

Example 2.3.5: Let $F(u) = ||u||$. The convex conjugate $F^*(y)$ is then an indicator function of the unit ball with respect to the dual norm, i.e.

$$F^*(y) = \begin{cases} 0, & ||y||_* \leq 1 \\ +\infty & ||y||_* > 1. \end{cases} \tag{2.35}$$

In the special case of L_p spaces, we know that their duals are L_q spaces with $\frac{1}{p} + \frac{1}{q} = 1$ for $1 < p, q < +\infty$. Therefore, for $||\cdot|| = ||\cdot||_{L_2}$, we also have $||\cdot||_* = ||\cdot||_{L_2}$. From classical theory of function spaces we know that the case $p = 1$ has to be handled with slightly more care. Nevertheless, we can identify $||\cdot||_{L_\infty}$ as the dual of $||\cdot||_{L_1}$. However, L_1 is not a reflexive space, therefore the dual of L_∞ is *not* L_1. Thankfully, the dual of L_∞ is not of much interest for us.

Another important finding comes with the *biconjugate* F^{**}. By definition we can easily see that $F^{**} \leq F$ in general. But even more: for convex and lsc functions (our functions of desire), we have equality, i.e.

$$F^{**} = F.$$

The introduction to this section hinted that we can drop the restriction of differentiability, if we deal with convex functions. That is, because for convex functions we can define the *subdifferential*, which can be seen as generalization of a derivate.

Definition 2.3.6 ([38], I.5.1.): A function $F: U \to \overline{\mathbb{R}}$ is said to be *subdifferentiable* at a point $u \in U$ if there exists a $u^* \in U^*$ such that

$$F(v) \geq F(u) + < u^*, v - u > \quad \forall v \in U. \tag{2.36}$$

The element u^* is the called *subgradient* of F at u and the set of subgradients at u is called the *subdifferential* of F at the point u and is written as $\partial F(u)$.

Specifically, subdifferentiability generalizes the concept of Gâteaux-differentiability.

Definition 2.3.7 ([38], I.5.2.): Let F be a function of U onto $\overline{\mathbb{R}}$. We call the limit as $\lambda \to 0_+$, if it exists, of

$$\frac{F(u + \lambda v) - F(u)}{\lambda}$$

the *directional derivative of F at u in the direction of v* and denote it by $F'(u;v)$. If there exists $u^* \in U^*$ such that for all $v \in U$

$$F'(u;v) = \langle v, u^* \rangle,$$

we say that F is *Gâteaux-differentiable at u*, call u^* the *Gâteaux-differential at u of F*, and denote it by $F'(u)$.

The uniqueness of the Gâteaux-differential follows directly, it is characterized by

$$\forall v \in U, \quad \lim_{\lambda \to 0_+} \frac{F(u+\lambda v) - F(u)}{\lambda} = \langle v, F'(u) \rangle.$$

We will now show that for convex functions Gâteaux-differentiability and uniqueness of the subgradient coincide.

Proposition 2.3.8 ([38], I.5.3.): *Let F be a convex function of U into $\overline{\mathbb{R}}$. If F is Gâteaux-differentiable at $u \in U$, it is subdifferentiable at u and $\partial F(u) = \{F'(u)\}$. Conversely, if at a point $u \in U$, F is continuous and finite and has only one subgradient, then F is Gâteaux-differentiable at u and $\partial F(u) = \{F'(u)\}$.*

The close relationship between convex conjugate and subdifferential can be seen by the following proposition.

Proposition 2.3.9 (Legendre-Fenchel identity, [38], I.5.2): *For every function F of U into $\mathbb{R}$, we have that if $y \in \partial F(u)$, so is $u \in \partial F^*(y)$. Let furthermore F be convex, lsc and proper. Then*

$$y \in \partial F(u) \quad \Leftrightarrow \quad u \in \partial F^*(y) \quad \Leftrightarrow \quad F(u) + F^*(y) = \langle y, u \rangle. \tag{2.37}$$

In what follows, the *Proximal Map* or *Proximity Operator* will also play an important role. We consider the minimization problem

$$\min_{v \in U} \tilde{F}(v), \ \tilde{F}(v) := F(v) + \frac{1}{2\tau}||v-u||^2 \tag{2.38}$$

for some given $u \in U$. If F is convex, proper and lsc, so is $\tilde{F}$. Therefore, there is a unique $\tilde{u} \in U$, such that the minimum of

$$\tilde{F}(v) = F(v) + \frac{1}{2\tau}||v-u||^2$$

is attained at $\tilde{u}$. We define the map that maps u to the attained minimum $\tilde{u}$ by $\text{prox}_{\tau F}(u)$. One can show that this again defines a 1-Lipschitz monotone operator. In addition, it is itself the gradient of a convex function, and by basic subdifferential calculus we can show that

$$0 \in \partial F(\tilde{u}) + \frac{\tilde{u} - u}{\tau}. \tag{2.39}$$

Solving equation (2.39) for $\tilde{u}$ shows that $\tilde{u}$ is given exactly by the *resolvent* of the operator $\tau \partial F$ at u, namely $\tilde{u} = (I + \tau \partial F)^{-1} u$. We can also deduce Moreau's identity

$$u = (I + \tau \partial F)^{-1}(u) + \left(I + \frac{1}{\tau}\partial F^*\right)^{-1}\left(\frac{u}{\tau}\right) = \text{prox}_{\tau F}(x) + \text{prox}_{\frac{1}{\tau}F^*}\left(\frac{u}{\tau}\right),$$

which shows that as soon as we can compute one of the resolvents, we can use the identity to compute the other one. This will come in handy especially in imaging applications. Often, resolvents or proximal maps in these areas can be computed easily.

2.3.2 Minimization of Convex Functions

In the following, we will limit ourselves to spaces where bounded sequences admit convergent subsequences. In Section 2.2.1, we saw that this is an important condition to deduce the existence of a minimizer. As this is fulfilled for reflexive Banach spaces, V will be such and endowed with norm $||\cdot||$. We consider a function F, that maps from a non-empty *closed convex* subset of U, denoted by Ω onto $\mathbb{R}$. Further, F is assumed to be convex and lower semi-continuous. Our objective will be to find some $u \in \Omega$, such that

$$F(u) = \inf_{v \in \Omega} F(v). \tag{2.40}$$

The direct method of calculus of variations can now be framed into a proposition for existence of solutions.

Proposition 2.3.10 ([38], II.1.2.): *Let F be convex, l.s.c, and proper and assume that Ω in equation (2.40) is either bounded our that F is coercive for $u \in \Omega$. Then equation (2.40) has at least one solution. If F is strictly convex over Ω, this solution is unique.*

Note that the above problem can also be replaced by a minimization over the whole of U, namely by setting

$$\hat{F}(u) = \begin{cases} F(u), \text{ if } u \in \Omega \\ +\infty, \quad \text{if } u \notin \Omega. \end{cases} \tag{2.41}$$

The variety of solution strategies for convex minimization problems is manifold. Assume for the moment for example that F is differentiable. Then of course one would wonder if the classical gradient descent

$$u^{k+1} = u^k - \tau \nabla F(u^k)$$

with fixed stepsize τ forms a suitable iteration to obtain a minimum. The answer is no, as we know from classical literature that very strict conditions are necessary (on top of differentiability) to guarantee convergence, for example that ∇F is Lipschitz with some constant L and that $0 < \tau L < 2$. Less strict assumptions can be used if one solves the minimization via *implicit gradient descent*, i.e.

$$u^{k+1} = u^k - \tau \nabla F(u^{k+1}). \tag{2.42}$$

But how is a new iterate computed in this context? An iterate, that satisfies equation (2.42) also satisfies

$$\nabla F(u^{k+1}) + \frac{u^{k+1} - u^k}{\tau} = 0,$$

which is a critical point of the function

$$u \mapsto F(u) + \frac{||u - u^k||^2}{2\tau}. \tag{2.43}$$

For a convex and only lsc function this critical point is now exactly the proximal map of τF at u^k introduced in the previous section. Note that by defining the update as a critical point of (2.43), no smoothness on F is required. The representation

$$F_\tau(\bar{u}) \coloneqq \min_{u \in \Omega} F(u) + \frac{||u - \bar{u}||^2}{\tau} \tag{2.44}$$

can be also found under the name Moreau-Yosida regularization. It is easy to show that the gradient of F_τ given by

$$\nabla F_\tau(\bar{u}) = \frac{\bar{u} - \text{prox}_{\tau F}(\bar{u})}{\tau}, \tag{2.45}$$

does fulfill a Lipschitz condition with constant $1/\tau$ [31], and that equivalently

$$\text{prox}_{\tau F}(\bar{u}) = \bar{u} - \tau \nabla F_\tau(\bar{u}). \tag{2.46}$$

We can therefore replace the implicit gradient descent (2.42) by an explicit gradient descent of F_τ

$$u^{k+1} = \text{prox}_{\tau F}(u^k) = u^k - \tau \nabla F_\tau(u^k). \tag{2.47}$$

We will now turn towards a slight modification of equation (2.40). Especially in imaging applications it is often convenient to split the minimization problem into two parts

$$\min_{v \in U} F(K(v)) + g(v) \tag{2.48}$$

where

$$F \colon V \to (-\infty, +\infty], \quad g \colon U \to (-\infty, +\infty]$$

are convex lsc functions and $K \in \mathscr{L}(U, V)$ is a continuous linear operator mapping U into V. We already encountered this motif in regularization theory, where within Tikhonov regularization F and g can be identified with the corresponding norms, and K denotes the linear operator.

Recall that for convex and lsc functions it holds that [100]

$$F = F^{**} = \sup_{y \in Y} \langle u, y \rangle - F^*(y).$$

Plugging this into equation (2.48) gives

$$\min_{u \in V} F(K(u)) + g(u) = \min_{u \in V} \sup_{y \in Y} \langle y, Ku \rangle - F^*(y) + g(u). \tag{2.49}$$

The conditions to exchange $\min$ and $\sup$ are not very strict, it is for example already enough that $F(0) < \infty$ and g is continuous at 0. We can therefore write

$$\begin{aligned} \min_{v \in U} F(K(v)) + g(v) &= \min_{v \in U} \sup_{y \in Y} \langle y, Kv \rangle - F^*(y) + g(v) && (2.50) \\ &= \max_{y \in Y} \inf_{v \in U} \langle y, Kv \rangle - F^*(y) + g(v) && (2.51) \\ &= \max_{y \in Y} -F^*(y) - g^*(-K^* y), && (2.52) \end{aligned}$$

where the last formula is the *Fenchel-Rockefellar dual problem*. With the assumptions as above, we can show that it has at least one solution y^*. We now define the *Lagrangian* as

$$\mathcal{L}(v, y) \coloneqq \langle y, Kv \rangle - F^*(y) + g(v). \tag{2.53}$$

Then, if u is any solution of the *primal problem* (2.48), then (u, y^*) is a saddle point of the primal dual formulation (2.51). It holds

$$\mathcal{L}(u, y) \leq \mathcal{L}(u, y^*) \leq \mathcal{L}(v, y^*)$$

and we can also show that

$$0 \in \partial g(u) + K^* y^*,$$
$$0 \in \partial F^*(y^*) - Ku.$$

One can now formulate minimization algorithms that explicitly exploit the saddle point structure of (2.48). In the literature, they are usually found under the name *primal dual algorithms*, as they solve the primal and the dual part alternatingly. One of the first, very inviting approaches uses equation (2.51). It alternates between the solution of the primal variable u and the dual variable y by solving

$$\begin{aligned} u^{k+1} &= \text{prox}_{\tau g}(u^k - \tau K^* y^k) \\ y^{k+1} &= \text{prox}_{\sigma F^*}(y^k + \sigma K u^{k+1}), \end{aligned} \tag{2.54}$$

where we used the idea of equation (2.47) for each subproblem. Though this method has been applied successfully, convergence can only be shown for its over relaxed variant

$$\begin{aligned} u^{k+1} &= \text{prox}_{\tau g}(u^k - \tau K^* y^k) \\ y^{k+1} &= \text{prox}_{\sigma F^*}(y^k + \sigma K \bar{u}^k) \\ \bar{u}^{k+1} &= u^{k+1} + \theta(u^{k+1} - u^k). \end{aligned} \tag{2.55}$$

Especially in imaging applications, the solution of the proximal map is often very easily computed.

Example 2.3.11: In denoising for example, one tries to recover a clean image u from its noisy version f, where the relation is

$$u = f + \eta,$$

with η being an unknown noise. The problem is ill-posed, and a famous regularization is by Osher, Rudin and Fatemi [105]

$$\min_v \lambda ||\nabla v||_1 + \frac{1}{2}||v - f||_2^2. \tag{2.56}$$

In the notation of equation (2.48), we find $F = ||\cdot||_1$, $K = \nabla$ and $g = ||\cdot - f||_2$, which yields the saddle-point problem

$$\min_{v \in U} \sup_{y \in Y} \langle y, \nabla v \rangle > -\delta_{||\cdot||_2 \leq \lambda}(y) + \frac{1}{2}||v - f||_2^2, \tag{2.57}$$

where we used the findings from Example 2.3.5 to calculate the convex conjugate of the norm. Performing the minimization now requires to compute the proximal maps of $\delta_{||\cdot||_2 \leq \lambda}(y)$ and $\frac{1}{2}||v - f||_2^2$. The latter one can be easily derived by Moreau-Yosida regularization and reads

$$\hat{u} = \text{prox}_{\tau g}(\tilde{u}) \quad \Leftrightarrow \quad \hat{u}_{i,j} = \frac{\tilde{u}_{i,j} + \tau f}{1 + \tau}, \tag{2.58}$$

where $\tilde{u}$ is the argument of the proximal map as in algorithm (2.54) or (2.55). It remains to compute the proximal map of $\delta_{||\cdot||_2 \leq \lambda}(y)$. But proximal maps of indicators with respect to some norms are just the projection to balls with radius λ

$$y = \Pi_{||\cdot||_2 \leq \lambda}(y) \quad \Leftrightarrow \quad y_{i,j} = \frac{\tilde{y}_{i,j}}{\max\{1, \lambda^{-1}, |\tilde{y}_{i,j}|_2\}}, \tag{2.59}$$

where $\tilde{y}$ again denotes the argument as in algorithm (2.54) or (2.55).

2.4 Bounded Variation

Transferring regularization from Hilbert to Banach spaces will become necessary, as regularization in Hilbert spaces will almost always produce very smooth solutions. However, neither the images nor the motion that we expect can be considered as such. Images will have edges and in motion we will observe jumps. In this section we will therefore introduce the space of functions of bounded variation, which is not only very useful in image processing, but also turns out to be a suitable space for our motion estimation.

Bounded variation turned out to be beneficial in many applications. This is also reflected by the extensive literature. The main sources for this section are [30, 106, 29], and especially when we consider total generalized variation [22] and [23].

2.4.1 Basics from Functional Analysis

Functions of bounded variation are those functions, whose *total variation* TV is finite.

Definition 2.4.1 ([29]): Let $\Omega \subset \mathbb{R}^n$ be an open set and let $f \in L_1(\Omega)$. Define

$$\begin{aligned} TV(f) &= \int_\Omega |Df| \\ &= \sup \Big\{ \int_\Omega f \operatorname{div} \mathbf{g}\, dx \mid \\ &\qquad \mathbf{g} = (g_1, \ldots, g_n) \in C_0^1(\Omega; \mathbb{R}^n) \text{ and } |\mathbf{g}(\mathbf{x})| \leq 1 \text{ for } \mathbf{x} \in \Omega, \Big\} \end{aligned} \tag{2.60}$$

where $\operatorname{div} \mathbf{g} = \sum_{i=1}^n \frac{\partial g_i}{\partial x_i}$.

Example 2.4.2: For $f \in C^1(\Omega)$ the total variation is given by

$$\int_\Omega f \operatorname{div} \mathbf{g}\, dx = -\int_\Omega \sum_{i=1}^n \frac{\partial f_i}{\partial x_i} g_i \, dx$$

for *every* $g \in C_0^1(\Omega; \mathbb{R}^n)$ via integration by parts, and therefore

$$TV(f) = \int_\Omega |\nabla f| \, dx. \tag{2.61}$$

Unfortunately, such an identity cannot be derived in general. However, for functions that are finite with respect to TV, we can derive a similar identity.

Definition 2.4.3: A function $f \in L_1(\Omega)$ is said to have *bounded variation* in Ω if $TV(f) < \infty$. We define $\mathrm{BV}(\Omega)$ as the space of all functions in $L_1(\Omega)$ with bounded variation.

The classical theorem of Meyer and Serrin now states that if we relax the gradient in Example 2.4.2 to the distributional derivate, then this identity holds for functions in BV. The theorem in context of bounded variation can be found in [30]; a proof can be found in the original work of [85].

Theorem 2.4.4 ([30], Thm. 1.): *Let $\Omega \subset \mathbb{R}^N$ be an open set and let $u \in \mathrm{BV}(\Omega)$. Then there exists a sequence $(u_n)_{n\in\mathbb{N}}$ of functions in $C^\infty \cap W^{1,1}(\Omega)$ such that*

- $u_n \to u \quad$ *in* $\quad L_1(\Omega)$,
- $J(u_n) = \int_\Omega |\nabla u_n| dx \to J(u) = \int_\Omega |Du| \quad$ *as* $\quad n \to \infty$.

Remark 2.4.5: $W^{k,p}$ is the Sobolev space of functions whose derivatives up to k have finite L_p norm.

The theorem shows another remarkable fact: we can approximate functions in BV by smooth functions. However, the distributional derivative is quite a difficult quantity to calculate. At best, we can decompose it for BV functions into a weak gradient of a vector valued function in L_1, and a singular part $D^s u$

$$Du = \nabla u(x)dx + D^s u. \tag{2.62}$$

What does this mean for TV as an option for regularization? Let us in addition observe, that discontinuous functions *can* be of finite variation and therefore in BV. Take for example $\Omega = [-1, 1]$ and define a discontinuous function f via

$$f(t) = \begin{cases} -1, & -1 \le t < 0 \\ 1, & 0 < t \le 1 \\ 0, & \text{else.} \end{cases}$$

Applying the definition of total variation, we get

$$\int_\Omega f \operatorname{div} g \,\mathrm{dt} = \int_{-1}^{1} \frac{\partial g}{\partial t} \,\mathrm{dt} = -2g(0).$$

Calculating the supremum as in Definition 2.4.1 then gives

$$\int_{-1}^{1} |Df| = 2.$$

That means, if we find a way to approximate the total variation, we can expect discontinuous functions as an outcome of the regularization. We already saw that the approximation by smooth functions in Theorem 2.4.4 is one mean. But in fact, if we discretize the total variation, we can also get a weak convergence result, namely Gamma convergence.

Definition 2.4.6 ([106], 4.48.): Let (J_k) be a sequence of functionals on a metric space U. If there exists a functional J on U such that

1. for all sequences $(u_k) \to u$ one has $\liminf_k J_k(u_k) \ge J(u)$,
2. for each u there exists a sequence $(u_k) \to u$ such that $\limsup_k J_k(u_k) \le J(u)$,

then (J_k) is said to Γ-converge to the Γ-limit J, denoted by

$$J = \Gamma - \lim_k J_k.$$

We can for example approximate a three dimensional quantity by piecewise constant functions, i.e. for $\mathbf{u} = (u, v, w)$ is approximated by $\mathbf{u}_h = (u_h, v_h, w_h)$ via

$$\mathbf{u}_h(\mathbf{x}) = \sum_{i=1,j=1,k=1}^{N,M,L} \mathbf{u}_{ijk}\phi_{\mathbf{u}}(\mathbf{x}), \tag{2.63}$$

where $\phi_{\mathbf{u}}(\mathbf{x}) = \phi_i(x)\phi_j(y)\phi_k(z)$ and

$$\phi_i(x) = \begin{cases} 1, \text{if } x \in [x_i - \frac{h}{2}, x_i + \frac{h}{2}) \\ 0, \text{else,} \end{cases}$$

$$\phi_j(y) = \begin{cases} 1, \text{if } y \in [y_j - \frac{h}{2}, y_j + \frac{h}{2}) \\ 0, \text{else,} \end{cases}$$

$$\phi_k(z) = \begin{cases} 1, \text{if } z \in [z_k - \frac{h}{2}, z_k + \frac{h}{2}) \\ 0, \text{else,} \end{cases}$$

and $\mathbf{u}_{ijk} = (u_{ijk}, v_{ijk}, w_{ijk}) \in \mathbb{R}^3$ is a vector-valued coefficient.

A possible discretization for TV in three dimensions is to consider the sum of total variation on each vector component, that is

$$TV_h(u_h) = h \sum_{i,j,k} \sqrt{|u_{i+1,j,k} - u_{i,j,k}|^2 + |u_{i,j+1,k} - u_{i,j,k}|^2 + |u_{i,j,k+1} - u_{i,j,k}|^2}, \tag{2.64}$$

and

$$TV_h(\mathbf{u}) = TV_h(u) + TV_h(v) + TV_h(w). \tag{2.65}$$

Theorem 2.4.7 ([30], Prop. 3.1.): *Let for the sake of simplicity be $N = M = L$ and $h = 1/M$, $\Omega = (0,1)^3$. Set, for $\mathbf{u} \in L_1(\Omega)$, the discrete total variation as in (2.64) if there is a representation of $\mathbf{u}$ as in (2.63) and $+\infty$ else. Then, as $h \to 0$, TV_h Γ-converges to*

$$TV_h(\mathbf{u}; \Omega) = \begin{cases} \int_\Omega |D\mathbf{u}| & \text{if } \mathbf{u} \in \mathrm{BV}(\Omega) \cap L_1(\Omega), \\ +\infty & \text{if } \mathbf{u} \in \mathrm{BV}(\Omega) \backslash L_1(\Omega). \end{cases} \tag{2.66}$$

The last finding is of special importance, as the analytical representation via singular parts (2.62) leads to rather complicated representations via lower dimensional Hausdorff-measures, which are very difficult to compute in the discrete case. The theorem states that instead we can approximate a functional involving TV_h, and a solution to the *approximated* functional will eventually converge against the *true* functional, as $h \to 0$.

2.4.2 Application of Total Variation in Image Processing

The utilization of bounded variation (and also the idea of the last theorem to have a discretization that converges to the true functional) are ideas from the heart of image processing. They have been successfully applied to denoising, inpainting, restoration and segmentation. The utilization for motion estimation will be discussed later, but we want to take a closer look into segmentation with total variation. Mumford and Shah proposed to consider images in the space of special functions of bounded variation (SBV). The space consists of those functions of bounded variation, whose distributional derivatives can be decomposed

into a jump part and an absolutely continuous gradient, and it is therefore a subset of BV. An image $u_0 \colon \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ is therefore segmented by searching for an approximation $u \in$ SBV with jump set S_u, such that u is a minimizer of

$$J(v) \coloneqq \int_{\Omega \setminus S_v} |\nabla v|^2 dx + \nu \mathcal{H}^{d-1}(S_v) + \lambda \int_\Omega (v - u_0)\, dx. \tag{2.67}$$

$\mathcal{H}^{d-1}$ is the $(d-1)$-dimensional Hausdorff-measure. For $d = 2$ it is given by the length of the jump set S_u, which is the total length of the edges in u.

Ambrosio and Tortorelli [4] showed that the following sequence of elliptic functionals Γ-converges to equation 2.67

$$J_k(v, z) = \frac{1}{2} \int_\Omega \left(|\nabla v|^2 + |\nabla z|^2 \right) (1 - z)^{2k} + \nu^2 \int_\Omega \frac{k^2 z^2}{4} + \lambda \int_\Omega (v - u_0)^2\, dx. \tag{2.68}$$

Note that equation (2.68) replaced the Hausdorff-measure by smooth approximations and is therefore much easier to compute.

Due to the use of the Hausdorff-measure in the original formulation (2.67), only the length of the edges of u are penalized, but not their smoothness. A solution to the Mumford-Shah functional therefore can have junctions, corners and crack-tips – a very appealing property for segmentation.

The characterization of solutions to the Mumford-Shah functional as SBV functions happened much later than the initial work, and only with the help of the Ambrosio-Tortorelli limit.

We now return to a general formulation for 2D image processing tasks involving regularization. We once more consider the solution of $F(u) = y$ by regularization

$$\min_{v \in \mathrm{BV}} \frac{\lambda}{2} ||F(v) - y||^2 + TV(v). \tag{2.69}$$

As BV is a Banach space, existence and uniqueness of solutions have to be investigated with the means of regularization in Banach spaces. Therefore, following [29], we will consider BV functions of zero mean, i.e., the subspace

$$\mathrm{BV}_0(\Omega) = \{u \in \mathrm{BV}(\Omega) \mid \int_\Omega u\, dx = 0\}.$$

This choice is not a very harsh restriction – one can show that each solution u to (2.69) can be decomposed to

$$u = v + \frac{\langle y, T\mathbf{1} \rangle}{||T\mathbf{1}||^2} \mathbf{1},$$

where $v \in \mathrm{BV}_0$ and $\mathbf{1}$ is a constant function taking the value 1 everywhere. The minimization can therefore be shifted from BV to BV_0 by setting $y_0 = y - \frac{\langle y, T\mathbf{1} \rangle}{||T\mathbf{1}||^2} \mathbf{1}$.

Theorem 2.4.8 ([29], 3.9.): *There exists a solution to*

$$\min_{v \in \mathrm{BV}_0(\Omega)} J(v); \qquad J(v) = \frac{\lambda}{2} ||F(v) - y_0||^2 + TV(v) \tag{2.70}$$

in BV_0.

Note that the set of minimizers to (2.70) is convex, as it is composed of two convex terms. As soon as we pose some additional conditions, for example that the solution is of some minimal norm (which makes very much sense in the settings of inverse problems as we saw with the notion of best approximate solutions), the convexity guarantees uniqueness. In general, uniqueness can only be expected if T is injective.

Theorem 2.4.9 ([29], 3.11.): *Let F have a trivial nullspace and $\lambda > 0$. Then (2.70) has a unique minimizer.*

Now that we have found conditions for existence and uniqueness for minimizers, one might ask for conditions on the optimality of a possible solution. As BV functions may be non-differentiable, we have to formulate these conditions in terms of subgradients (cf Definition 2.3.6).

Clearly, u is a minimizer of a convex functional J if $0 \in \partial J(u)$, which is, due to the convexity of J, necessary *and* sufficient. The computation of the subdifferential of J now leads to the following optimality condition

$$\lambda F^*(Fu - y) + p = 0 \quad p \in \partial TV(u).$$

However this condition gives no recipe for computing an actual minimizer, as it is dependent on the subdifferential, and this quantity is usually hard to compute. One can for example consider its dual norm, the so called G-norm. For further investigations in this manner, see [84].

Recall Example 2.3.11

$$\min_v \lambda||\nabla v||_1 + \frac{1}{2}||v - f||_2^2. \tag{2.71}$$

Though we introduced the explicit notion of total variation regularization afterwards, this denoising method was already an example of such a method. We now want to introduce a different method to tackle TV-penalizers. We rewrite the problem as

$$\min_v J(v), \quad J(v) = \frac{1}{2}\int_\Omega \left(||v - f||_2^2 + \lambda\rho(|\nabla v|)\right) dx, \tag{2.72}$$

where ρ in this case is a function to tackle the non-differentiability of $|\nabla v|$, e.g. $\rho(s) = \sqrt{s^2 + \beta^2}$ for some small β. Because we used the representation of $TV(v)$ via its gradient, we will seek $u \in W^{1,1}$, which is not too much of a problem, thanks to Theorem 2.4.4. A necessary condition for u to be a minimizer is vanishing first variation, i.e.

$$L(u, v) = \int_\Omega \left((u - f)v + \lambda\rho'(|\nabla u|)\nabla u \cdot \nabla v\right) dx = 0 \quad \text{for all} \quad v \in W^{1,1}. \tag{2.73}$$

By integration by parts, we can also formulate u as weak solution of the Euler equation

$$\lambda \,\text{div}\, \left(\rho'(|\nabla u|)\nabla u\right) - (u - f) = 0, \tag{2.74}$$

with reflecting (homogeneous Neumann) boundary conditions

$$\frac{\partial u}{\partial n} = 0 \quad \text{on} \quad \partial\Omega.$$

The variational approach just described has a close relative called *diffusion filtering* [124]. As the name implies, a noisy image f is diffusion filtered by solving the diffusion equation for a filtered image $u(x, t)$, which reads

$$\partial_t u = \,\text{div}\, \left(\rho(|\nabla u|)\nabla u\right), \tag{2.75}$$

with initial state given by the original image $f(x)$

$$u(x,0) = f(x), \tag{2.76}$$

and boundary conditions as before. The resemblance to equation (2.74) becomes clear, if we write

$$\frac{u-f}{\lambda} = \text{div}\left(\rho'(|\nabla u|)\nabla u\right), \tag{2.77}$$

which is an implicit time discretization of the diffusion equation. No matter how we interpret this problem, we have to solve a nonlinear PDE to reconstruct the filtered image u. A possible approach is to approximate the variational formulation by a quadratic variant with the help of an auxiliary variable w by

$$\tilde{J}(u,w) = \frac{1}{2}\int_\Omega \left((u-f)^2 + \lambda(w|\nabla u|^2 + \psi(w))\right) dx, \tag{2.78}$$

where $\phi(w)$ is chosen such that the above functional will be convex with respect to w. It can therefore be neglected if the functional is already convex. A solution is now found by solving for w and u in an alternating way

$$\begin{aligned} w^k &= \arg\min_w \tilde{J}(u^k, w) \\ u^{k+1} &= \arg\min_u \tilde{J}(u, w^k). \end{aligned} \tag{2.79}$$

Due to convexity, we can immediately calculate w^k as

$$w^k = \rho(|\nabla u^k|).$$

The alternating equations (2.79) therefore boil down to *freezing* or to *lag* the nonlinear part of (2.72). In case of solving variational equations arising from partial differential equations, this method is known as the *Kačanov method*, in case of denoising with BV-functions it is called *lagged diffusivity*. A detailed convergence analysis is postponed to the next section.

Note that in the classical convergence analysis of the Kačanov method, ρ has to satisfy the following properties.

(ρ1) The diffusion function $\rho\colon [0,\infty) \to [0,\infty)$ is continuously differentiable;

(ρ2) The diffusion function ρ is decreasing, i.e. $\rho'(t) \leq 0$ for all $t \geq 0$;

(ρ3) There are positive constants m_ρ and M_ρ such that $m_\rho \leq \rho(t) \leq M_\rho$ for all $t \geq 0$;

(ρ4) There exists a positive constant c_μ such that $2\rho'(t^2)t^2 + \rho(t^2) \geq c_\rho$ for all $t \geq 0$; or equivalently that ϕ defined by

$$\phi(t) \coloneqq \int_0^t \rho(s^2)s\,ds, \quad t \geq 0$$

is strictly convex.

Clearly, the role of ρ is taken by the derivative of $\rho(t) = \sqrt{t^2+\epsilon^2}$. As

$$\rho'(t) = \frac{1}{2}\frac{1}{\sqrt{t^2+\epsilon^2}},$$

the above conditions are fulfilled.

2.4.3 Total Variation as Regularization

Let us now come back to the considerations if the general TV regularization (2.69) really forms a regularization. We still need to check if λ in fact serves as the classical regularization parameter, that we defined in the Hilbert space setting, and what we can say about convergence in case of noisy data. Note that it is quite difficult to define a "real" regularization operator $\mathcal{R}_\alpha$ (or in this specific case $\mathcal{R}_\lambda$), that can be checked to fulfill the conditions of Definition 2.1.6. For convergence for exact data y there exists the following weak convergence result.

Theorem 2.4.10 ([29], 5.4.): *Let $y \in \mathscr{R}(T)$, and u^λ be a solution of J for a specific λ. Then every subsequence of (u^λ) has a weak-* convergent subsequence. Every weak-* accumulation point is a solution to $Tu = y$ with minimal total variation.*

For noisy data y^δ we find the following.

Theorem 2.4.11 ([29], 5.5.): *Let $y \in \mathscr{R}(T)$ and let y^δ be such that $||y - y^\delta|| = \delta$. Moreover, let the noise be bounded from above. Let λ be such that $\lambda(\delta) \to \infty$ as $\delta \to 0$ and $\lambda(\delta)\delta^2 \to 0$. Then every subsequence of (u^λ) has a weak-* convergent subsequence. Every weak-* accumulation point is a solution to $Tu = y$ with minimal total variation.*

Having now justified that total variation can in fact be used as regularization, one could now ask if besides allowing or even promoting jumps, there is another mathematical justification to use total variation.

2.4.4 Total Generalized Variation

Recall that we eventually want to compute the deformation a body underwent during a materials test. It therefore seems natural to check what continuum mechanics has to offer to *simulate* the outcome of such an experiment. In fact, already in 1980 Temam [117] introduced the space of *bounded deformation* to model *plastic* deformation, which occurs after an initial elastic reversible stage. To relate bounded deformation to bounded variation, we need the following notions from tensor calculus. The vector spaces of *k-tensors* and *symmetric k-tensors* are defined by

$$\mathcal{T}^k(\mathbb{R}^d) = \left\{ \xi : \underbrace{\mathbb{R}^d \times \ldots \times \mathbb{R}^d}_{k\text{-times}} \to \mathbb{R} \,|\, \xi \text{ } k\text{-linear} \right\}$$

and

$$\mathrm{Sym}^k(\mathbb{R}^d) = \left\{ \xi : \underbrace{\mathbb{R}^d \times \ldots \times \mathbb{R}^d}_{k\text{-times}} \to \mathbb{R} \,|\, \xi \text{ } k\text{-linear and symmetric} \right\},$$

respectively. Just as in the classical theory, we can identify the space of Radon measures with the dual of continuous functions vanishing on the boundary. For (symmetric) tensor *fields*, i.e. mappings from a bounded set Ω to either $\mathcal{T}^k(\mathbb{R}^d)$ or $\mathrm{Sym}^k(\mathbb{R}^d)$, we define

$$\mathscr{M}(\Omega, X) = C_0(\Omega, X)^*, \qquad X \in \{\mathcal{T}^k(\mathbb{R}^d), \mathrm{Sym}^k(\mathbb{R}^d)\}$$

with Radon norm

$$||u||_{\mathscr{M}} = \sup\{\langle u, \phi \rangle \mid \phi \in C_0(\Omega, X),\ ||\phi||_\infty \leq 1\}.$$

Definition 2.4.12: Let $l \in \mathbb{N}, l \geq 0$. The space $\mathrm{BD}(\Omega, \mathrm{Sym}^l(\mathbb{R}^d))$ of *symmetric tensor fields of bounded deformation* is defined as the Banach space

$$\mathrm{BD}(\Omega, \mathrm{Sym}^l(\mathbb{R}^d)) = \{u \in L_1(\Omega, \mathrm{Sym}^l(\mathbb{R}^d)) \mid \mathscr{E}u \in \mathscr{M}(\Omega, \mathrm{Sym}^{l+1}(\mathbb{R}^d))\},$$

with norm $||\mathbf{u}||_{\mathrm{BD}} = ||\mathbf{u}||_1 + ||\mathscr{E}\mathbf{u}||_{\mathscr{M}}$ and $\mathscr{E}$ being the weak symmetrized derivative.

As with total variation, we can express the Radon norm $||\mathscr{E}\mathbf{u}||_{\mathscr{M}}$ by its dual representation

$$||\mathscr{E}\mathbf{u}||_{\mathscr{M}} = \sup \left\{ \int_\Omega \mathbf{u} \cdot \operatorname{div} \phi \, \mathrm{d}\mathbf{x} \mid \phi \in \mathscr{C}_c^1(\Omega, \mathrm{Sym}^{l+1}(\mathbb{R}^d)), ||\phi|| \leq 1 \right\}.$$

For $l = 0$ and $l = 1$ we now arrive at the desired spaces [22], as

$$\mathrm{BD}(\Omega, \mathrm{Sym}^0(\mathbb{R}^d)) = \mathrm{BV}(\Omega) \quad \text{and} \quad \mathrm{BD}(\Omega, \mathrm{Sym}^1(\mathbb{R}^d)) = \mathrm{BD}(\Omega),$$

where $\mathrm{BD}(\Omega)$ is the space of bounded deformation as Temam introduced it together with Strang to describe perfectly plastic deformations [117]. Just as for TV, we can write $\mathscr{E} = (D\mathbf{u} + D^T\mathbf{u})/2$ with D being the distributional gradient as an Definition 2.4.1. The quantity $\mathscr{E}$ in this special case is also called *strain tensor*. Temam developed the theory for a suitable space for plastic deformation very similarly to image processing with bounded variation, but several years earlier: He showed, that the Sobolev space $W^{1,1}$ only contains functions that are too smooth to produce solutions in the plastic case. The "natural candidate" BV is also not a correct candidate. Plasticity can be characterized by finite strain energy, but as finite strain does not dominate the total variation (so the components of the derivative), we lose coerciveness and cannot apply the direct method of calculus of variations anymore. He therefore proposed BD and many publications followed manifesting the correctness of this space. However, the careful reader noted that BD and BV do not coincide, even worse: One can derive examples of functions that have bounded deformation but fail to have bounded variation. It is nevertheless not too strict to consider total variation as regularizer in motion estimation, as piecewise rigid displacements *have* bounded variation [3].

2.5 Solution Strategies

2.5.1 Kačanov Method

The following section is based on the quite recent work of Heid et al. [54], who introduced a modified Kačanov method that works under less strict conditions. We will however show that we fulfill the strict conditions of the original method, but we profit from their nice introduction either way.

In their general setting V will denote a reflexive Banach space equipped with a norm $||\cdot||_V$ and $K \subset V$ a closed, convex set. Note that there is a small but significant difference to our Banach space U from the previous section, namely the reflexivity. One of the main objects of interest will be a (nonlinear) Gâteaux-continuously differentiable functional $H\colon K \to \mathbb{R}$ with strongly monotone Gâteaux derivative, i.e. H' satisfies

$$\langle H'(u) - H'(v), u - v \rangle \geq \nu ||u - v||_V^2 \quad \text{for all} \quad u, v \in K,$$

for a constant $\nu > 0$. Such functions admit a unique minimizer in K, as the following proposition shows.

Proposition 2.5.1 ([54], 2.1.): *Suppose that $H\colon K \to \mathbb{R}$ is a (Gâteaux-)continuously differentiable functional with the strong monotonicity condition as above. Then there exists a unique minimizer $u^* \in K$, i.e. $H(u^*) \leq H(v)$ for all $v \in K$. Furthermore, $u^* \in K$ is the unique solution of the weak inequality*

$$\langle H'(u^*), v - u^* \rangle \geq 0 \quad \textit{for all} \quad v \in K.$$

Remark 2.5.2: Y is assumed to be reflexive to guarantee that every bounded sequence has a convergent subsequence. As we deal with functions in BV, we cannot guarantee reflexivity. However, we will use Kačanov's method in motion estimation in a similar manner as in diffusion filtering (2.72), which allows us to consider an approximate problem in $W^{1,1}$.

K will be furthermore characterized by the following assumption.

(K) The set $X := \{u - v \,|u, v \in K\}$ is a linear closed subspace of V, and $x + v \in K$ for all $x \in X$ and $v \in K$.

This allows for an even stronger characterization (but still a weak formulation) of the minimizer, namely

$$\langle H'(u^*), v \rangle = 0 \quad \text{for all} \quad v \in X. \tag{2.80}$$

In general, the Kačanov method deals with mappings $a\colon K \times V \times X \to \mathbb{R}$ and right-hand sides $b\colon K \times X \to \mathbb{R}$, which satisfy the following properties

(A1) For any given $u \in K$, we suppose that $a(u; \cdot, \cdot)$ is a bilinear form on $V \times X$, and $b(u, \cdot) \in X^*$.

(A2) There exist positive constants $\alpha, \beta > 0$ such that, for any $u \in K$, the form $a(u; \cdot, \cdot)$ is uniformly bounded on $V \times X$ and coercive on $X \times X$ in the sense that

$$a(u; v, w) \leq \beta ||v||_V ||w||_V \quad \text{for all} \quad v \in V \text{ and for all } w \in X,$$

and

$$a(u; v, v) \geq \alpha ||v||_V^2 \quad \text{for all} \quad v \in X,$$

respectively; in particular, if the set K satisfies property (K), then it follows that

$$A(u; v - w, v - w) \geq \alpha ||v - w||_Y^2 \quad \text{for all} \quad v, w \in K.$$

(A3) There are Gâteaux continuously differentiable functionals $G\colon K \to \mathbb{R}$ and $B\colon K \to \mathbb{R}$ such that, for all $u \in K$ it holds $G'(u)|_X = a(u; u, \cdot)$ and $B'(u)|_X = b(u)$ in X^*.

(A4) The (continuously differentiable) functional $H\colon K \to \mathbb{R}$ given by $G(u) - B(u)$, $u \in K$ satisfies the strong monotonicity condition.

The characterization of the minimizer (2.80) for the functional H, if K satisfies (K) reads

$$0 = \langle H'(u), v \rangle = \langle G'(u^*) - B'(u^*), v \rangle = a(u^*; u^*, v) - \langle b(u^*), v \rangle \text{ for all } v \in X. \tag{2.81}$$

For a given $u \in K$ we now define the *linear* operator $A(u)\colon V \to X^*$, $v \mapsto A(u)v$ by

$$\langle A(u)v, w \rangle = a(u; v, w) \quad \text{for all} \quad w \in X. \tag{2.82}$$

Using A, the weak formulation (2.81) then reads

$$A(u^*)u^* = b(u^*) \quad \text{in } X^*.$$

By (A2), we can deduce that $a(u;\cdot,\cdot)$ is a bounded and coercive bilinear form on the closed subspace $X \times X$. That allows to apply the Lax-Milgram theorem: For any $u \in K$ and linear maps $\ell \in X^*$, we find a unique $w_{u,\ell}$ such that $A(u)w_{u,\ell} = \ell$ in X^*. Therefore, A is invertible on X for every $u \in K$. That allows us to define an iterative procedure by setting

$$u^{n+1} = u^n - \rho^n, \quad n \geq 0, \tag{2.83}$$

where ρ is defined as the solution to the *linear* system

$$A(u^n)\rho^n = F(u^n) \tag{2.84}$$

with

$$F(u) \coloneqq H'(u) = A(u)u - b(u) \in X^*. \tag{2.85}$$

We can now state convergence of the Kačanov method.

Theorem 2.5.3 ([54], 2.5.): *Assume (A1)-(A4) and in addition let the following* key *inequality hold*

$$G(u) - G(v) \geq \frac{1}{2}\left(a(u;u,u) - a(u;v,w)\right) \quad \textit{for all} \quad u,v \in K. \tag{2.86}$$

1. *For $n = 0, 1, \ldots$ the quadratic variational problem*

$$u^{k+1} = \min_u \frac{1}{2}a(u^k;u,u) - b(u) \tag{2.87}$$

has a unique solution u^{k+1} and for $u^{k+1} \in K$

$$a(u^k;u^{k+1},v-u^{k+1}) \geq b(v-u^{k+1})$$

holds.

2. *If in addition $H'(u)$ satisfies the strong monotonicity condition, the Kačanov method converges, that is*

$$u_k \to u \quad \textit{in } V \quad \textit{as} \quad k \to \infty. \tag{2.88}$$

2.5.2 Levenberg-Marquardt Algorithm

As hinted in the previous sections, we now come to the derivation of the method of Levenberg and Marquardt for curve fitting problems. We follow the derivation of [53]. Originally, it is designed to solve least squares curve fitting algorithms, but it also arises in the context of regularization of nonlinear ill-posed problems.

In the classical context, one seeks a solution to the nonlinear curve fitting problem

$$\min_x \frac{1}{2}||F(x)||_2^2 =: \Phi(x), \tag{2.89}$$

where $F\colon \mathscr{D}(F) \subset \mathbb{R}^n \to \mathbb{R}^m$ with $m \geq n$. For sufficiently smooth F (which we will assume for now), we know that

$$\nabla\Phi(\hat{x}) = 0 \qquad \text{and} \qquad H_\Phi(\hat{x}) \text{ positive definite}$$

are the necessary and sufficient conditions for Φ having a local minimum in $\hat{x}$. Candidates for minimizing points, so stationary points, can be computed by the well-known Newton method

$$x^{k+1} = x^k - (\nabla\Phi(x^k))^{-1}\Phi(x^k). \tag{2.90}$$

Note that the classical method was developed for $m = n = 1$, but the extension to arbitrary $m = n$ is straight forward. In our case, so $m \neq n$, the inverse has to be replaced with a generalized or Pseudo-inverse, as $\nabla\Phi(x^k)$ is not necessarily quadratic.

Computing the iteration (2.90) is however a very tedious task, as it involves second order derivatives of F. Am remedy is provided by an additional linearization step, i.e.

$$x^{k+1} = \min_x \Phi(x), \quad \Phi(x) \approx \frac{1}{2}||F(x^k) + \nabla F(x^k)\left(x - x^k\right)||_2^2.$$

x^{k+1} can now be computed via a Newton step with respect to F, that is

$$x^{k+1} = x^k - (\nabla F(x^k))^{-1}F(x^k),$$

or, if one prefers to solve the linear system

$$\nabla F(x^k)\left(x^{k+1} - x^k\right) = -F(x^k).$$

This method is usually known under the name Gauss-Newton-method. Convergence is unfortunately only guaranteed for starting points close to the actual solution which is mainly due to the possible ill condition of $\nabla\Phi(x^k)$. If $\nabla\Phi(x^k)$ is ill-conditioned the difference $x^{k+1} - x^k$ can become arbitrary large. The idea of the *Levenberg-Marquardt-method* is to limit this difference, i.e. to minimize within a ball-shaped trust region of radius ρ_k

$$R_k = \{x \in \mathbb{R}^2 \mid ||x - x^k||_2 \leq \rho_k\}.$$

The corresponding minimization problem then reads

$$\min \frac{1}{2}||F(x^k) + \nabla F(x^k)\left(x - x^k\right)||_2^2 \tag{2.91}$$

$$\text{s.t. } ||x - x^k||_2 \leq \rho_k. \tag{2.92}$$

This *constrained* curve fitting problem is now minimized on a compact subset $R_k \subset \mathbb{R}^n$ – it therefore has at least one solution. With the notation

$$A_k = \nabla F(x^k), \qquad h = x - x^k, \qquad b^k = -F(x^k)$$

we arrive at a *linear* constrained minimization problem

$$\min \frac{1}{2}||A_k h - b^k||_2^2 =: \Psi(h) \tag{2.93}$$

$$\text{s.t. } ||h||_2 \leq \rho_k. \tag{2.94}$$

After solving for h, the next iterate is computed via $x^{k+1} = x^k + h$.

Let us now consider the special case of an iterate h^k satisfying $||h^k|| = \rho_k$ exactly. We can then deduce that there exists some $\lambda_k > 0$ such that

$$\nabla\Psi(h^k) = A_k^* A_k h^k - A^* b^k = -\lambda_k h^k$$

or

$$\left(A_k^* A_k - \lambda_k \mathbb{I}\right) h^k = A^* b^k,$$

which is exactly the Tikhonov representation of the nonlinear inverse problem (2.22).

Chapter 3

Motion Estimation in Volume Images

After laying the mathematical foundations, we can now turn towards the exact description of algorithms that estimate motion in volume images. We can identify two communities, that estimate motion in volume images, but develop their algorithms rather independently from each other: Materials scientists with their methods going under the name *Digital Volume Correlation* (DVC), and medical image analysts, that call their methods *Medical Image Registration* (MIR). Due to the vast number of algorithms in these two areas, our choice of algorithms presented here is far from complete. We limited ourselves to either cornerstones of their kind or to those of special interest as they seemed particularly suitable.

However, we refer to the following reviews and monographs for a more complete overview in the respective areas. For DVC, the work of Bay et al. [12] and Buljac et al. [27] presents a great starting point when reading into the topic. For MIR the monograph [86], and [96] with special focus on the method we present here should be mentioned.

For all motion estimation algorithms, our setting will be the following. We consider images I_0, I_1 as maps from an open and bounded domain $\Omega \subset \mathbb{R}^3$ to the real numbers, so $I_0, I_1 \colon \Omega \subset \mathbb{R}^3 \to \mathbb{R}$. I_0 and I_1 resemble the same specimen at different, but successive loading stages, so for example unloaded and loaded, compressed at 2% and compressed at 4%. The images I_0 and I_1 are related by a transform $\phi \colon \mathbb{R}^3 \to \mathbb{R}^3$ that transfers the images such that

$$I_0(\mathbf{x}) = I_1(\phi(\mathbf{x})) = I_1 \circ \phi. \tag{3.1}$$

Ideally, the transformation ϕ mimics the motion that the real-world-sample underwent. An obvious choice is therefore to express ϕ in terms of displacement, so by

$$\phi(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x}).$$

But one could also think of interpreting ϕ as a path that transfers every particle in I_0 to a (preferably unique) position in I_1. The former will be the basis for Digital Volume Correlation, the latter for Medical Image Registration.

In spirit of regularization and variational formulation, we usually relate I_0 and I_1 by a distance function $\mathcal{D}$, for example

$$\mathcal{D}(I_0, I_1) = ||I_0(\mathbf{x}) - I_1(\phi(\mathbf{x}))||^2_{L_2},$$

which is also called the *weak formulation*, and will pose a regularization $\mathcal{R}(\mathbf{u})$ on the displacement $\mathbf{u}$, for example

$$\mathcal{R}(\mathbf{u}) = ||\mathbf{u}||_{H^1}.$$

3.1 Digital Volume Correlation

3.1.1 Local DVC

The starting point of DVC is a displacement based weak formulation first proposed by Bay et al. [13]. The algorithm starts by separating both images into subvolumes of K^3 voxels. Each subvolume is uniquely identified by its center $\mathbf{x}$. Then, for each subvolume individually, the displacement $\mathbf{u}$ is estimated at center $\mathbf{x}$ by solving

$$\min_{\mathbf{u}} \sum_{i=1}^{K^3} \left(I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x})) - I_0(\mathbf{x} + m_i)\right)^2, \tag{3.2}$$

where m_i denotes the offset to all locations within the subvolume. We now want to extend (3.2) to the whole image in order to fit the formulation to our general motion estimation approach. Obviously, this approach is regularization by projection. Assume that our image consists of $H \times W \times L$ or a total of N_{total} voxels. Our reconstructed displacement field will therefore also be of dimension $H \times W \times L \times 3$, but most certainly not of N_{total} distinct values. The number of distinct values coincides with the number N_{disp} of "supervoxels", with $N_{\text{total}} >> N_{\text{disp}}$ – the whole subvolume in (3.2) will be annotated with the displacement $\mathbf{u}$ at $\mathbf{x}$. An approximation $\mathbf{u}_h$ to $\mathbf{u}$ can be now be formulated by piecewise constant functions covering the supervoxels. By defining $\mathbf{x}_d = (x_d, y_d, z_d)$ as the center point of a subvolume, the displacement is approximated by

$$\mathbf{u}_h(\mathbf{x}) = \sum_{d}^{N_{\text{disp}}} \mathbf{u}_d \phi_{\mathbf{u}_d}(\mathbf{x}), \tag{3.3}$$

where $\phi_{\mathbf{u}_d}(\mathbf{x}) = \phi_d^x(x)\phi_d^y(y)\phi_d^z(z)$ and

$$\phi_d^x(x) = \begin{cases} 1, \text{if } x \in [x_d - \frac{K}{2}, x_d + \frac{K}{2}) \\ 0, \text{else,} \end{cases}$$

$$\phi_d^y(y) = \begin{cases} 1, \text{if } y \in [y_d - \frac{K}{2}, y_d + \frac{K}{2}) \\ 0, \text{else,} \end{cases}$$

$$\phi_d^z(z) = \begin{cases} 1, \text{if } z \in [z_d - \frac{K}{2}, z_d + \frac{K}{2}) \\ 0, \text{else,} \end{cases}$$

and $\mathbf{u}_d \in \mathbb{R}^3$ is a vector-valued coefficient. Note that in the above representation the superscripts only indicate which indicator is chosen. The minimization with respect to $\mathbf{u}_h$ is therefore equivalent to minimizing with respect to the coefficients $\mathbf{u}_d$. Taking into account all subvolumes in parallel, the minimization reads

$$\min_{\mathbf{u}} \sum_{d=1}^{N_{\text{disp}}} \sum_{i=1}^{K^3} \left(I_1(\mathbf{x}_d + m_i + \mathbf{u}_h(\mathbf{x}_d)) - I_0(\mathbf{x}_d + m_i)\right)^2, \tag{3.4}$$

and because our center points of the subvolumes are unique

$$\min_{\mathbf{u}} \sum_{d=1}^{N_{\text{disp}}} \sum_{i=1}^{K^3} \left(I_1(\mathbf{x}_d + m_i + \mathbf{u}_d \phi_{\mathbf{u}}(\mathbf{x}_d)) - I_0(\mathbf{x}_d + m_i)\right)^2. \tag{3.5}$$

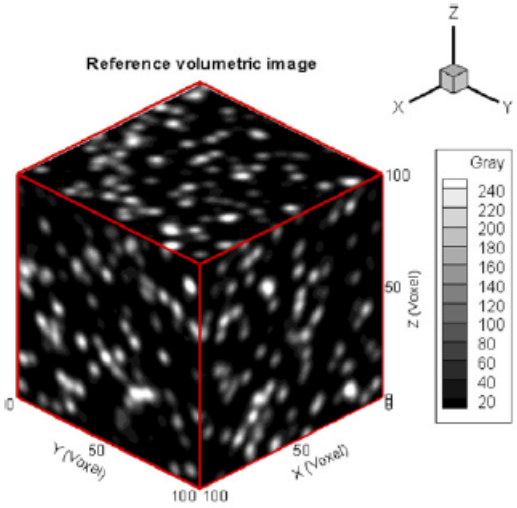


Figure 3.1: Speckle pattern. Image taken from [95].

The original work proceeds by using the Levenberg-Marquardt method to iteratively generate a solution. That is, $\mathbf{u}^{k+1}$ is computed by $\mathbf{u}^{k+1} = \mathbf{u}^k + p_k$, and p_k is the solution to the linear system

$$(J_k^T J_k + \lambda_k \mathbb{I}) p_k = -J_k \left(I_1(\mathbf{u}^k) - I_0 \right),$$

where J_k is the Jacobian of I_1 evaluated at $\mathbf{u}^k$, and where we omitted the dependence on $\mathbf{x}$ and m_i, and λ_k a sequence of nonnegative constants.

In Section 2.1.5 we learned that Levenberg-Marquardt itself is a regularization method with parameter λ_k. Therefore, theoretically it is possible to push N_{disp} to N_{total}, so to shrink the supervoxel, such that $K^3 = 1$. By formulating only the system matrix for the Gauss-Newton-Method (so by setting $\lambda_k = 0 \; \forall \, k$), we get a very visual impression on how the problem's ill-posedness manifests in a real world example: If we push the subset size to one displacement vector per voxel, the system matrix has only diagonal entries – namely the corresponding squared derivatives. If any two neighboring points in the image have the same gray values, the system matrix is singular and thus not invertible anymore. Tikhonov regularization (and thus Levenberg-Marquardt) now adds a small pertubation of size λ_k to each diagonal element to guarantee invertibility. Note that the condition number still might be bad, and thus the result will not be a good one. Regularization by projection, so choosing subvolumes much bigger than the actual voxel size, will now cause the system matrix to be overdetermined, instead of underdetermined, as before. The Tikhonov approach (adding $\lambda_k \mathbb{I}$) is therefore not necessary anymore, at least if the often required speckle pattern is present in the image, see Figure 3.1. Such a pattern guarantees that no subvolume is completely empty, which, again, guarantees invertibility.

Bay himself and his group also proposed the first improvements to this method [111]. Instead of only computing displacement, so three translational degrees of freedom per subvolume, they propose to use six degrees of freedom, so to also add those related to rotations. This changes equation (3.2) to

$$\min_{\mathbf{u},R} \sum_{i=1}^{K} \left(I_1(\mathbf{x} + n_i + \mathbf{u}(\mathbf{x})) - I_0(\mathbf{x} + m_i)\right)^2. \tag{3.6}$$

Note that now different offset vectors n_i and m_i have to be used. That is due to the fact that rotation has to be applied by a matrix, therefore the relation $n_i = Rm_i$ for R being a rotation matrix, i.e. $R^T = R^{-1}$, holds. This variant transitions into the original simply by setting $R = \mathbb{I}$. However, in our applications we will usually assume that no rotation occurs or that rigid transformation has been applied as a preprocessing step. Therefore we will not go much more into detail regarding these kinds of transformations.

The much more interesting improvement, also in the same work by Smith et al. [111], was to introduce a different metric to measure correlation. Where the original version (3.2) can be considered as a sum-of-squared-difference (SSD) metric, one can also use a Normalized-Cross-Correlation (NCC) metric, i.e. by setting

$$E_0 = \sum_{i=1}^{K} I_0(\mathbf{x} + m_i)^2,$$

$$E_1 = \sum_{i=1}^{K} I_1(\mathbf{x} + n_i + \mathbf{u})^2,$$

one can define the NCC coefficient via

$$C_n(\mathbf{u}) = 1 - \frac{\sum_{i=1}^{K} I_0(\mathbf{x} + m_i) I_1(\mathbf{x} + n_i + \mathbf{u})}{\sqrt{E_0 E_1}}. \tag{3.7}$$

In 2011 Gates et al. [43] proposed to use 12 degrees of freedom, so to estimate the displacement and the first order derivatives. To cope with the higher computational cost this extension caused, they limited the maximal possible displacement and proposed a parallel interpolation scheme. Pan et al. [94] proposed to use zero-mean NCC shortly after.

In the last decade, the focus was put on improvement of the optimization. First, a distinction between *additive* and *compositional* approaches was made. Because of the large number of unknowns in DVC, the optimization of the above problem is almost always done via iterative schemes. *Compositional* in this context means, that after each iteration of the solution algorithm all terms involving the displacement to be optimized are updated with the result from the iteration. This specifically involves all terms of the form $I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x}))$, so images evaluated at a displaced position. More detail on how and where to incorporate compositional methods can be found in Baker et al. [8]. In this context, the method of Bay et al. [13] can be classified as additive. The inverse compositional Gauss-Newton (IC-GN) scheme of Pan et al. [94] then led to significant improvement in accuracy and speed. Further improvements on IC-GN, especially aiming for fast and memory efficient computations on high resolution images, were made by Pan et al. [93]. Note that algorithms that only use Gauss-Newton methods can be considered to be regularized *only* by projection. Their theoretical limit therefore lies in subvolume sizes of several voxels.

Another problem with approaching motion estimation via local DVC immediately arises. As all displacement vectors are estimated independently from each other, they may be highly uncorrelated. In practice, one would expect that neighboring vectors more or less act the same, at least outside fracture or other discontinuous regions. Otherwise, the mechanical interpretation might be difficult. To tackle this problem, a different approach to DVC has been introduced, which will be described in the next section.

3.1.2 Global DVC

At first glance, the formulation of global DVC is not too different from its local variant. However, one usually starts with a minimization over the domain Ω

$$\min_{\mathbf{u}} \int_{\Omega} \left(I_1(\mathbf{x}+\mathbf{u}(\mathbf{x})) - I_0(\mathbf{x})\right)^2 \, \mathrm{d}\mathbf{x}, \tag{3.8}$$

and expresses the displacement field $\mathbf{u}$ in terms of basis functions

$$\mathbf{u}(\mathbf{x}) = \sum_n u_n \psi_n(\mathbf{x}), \tag{3.9}$$

where u_n are scalar values. Note that this coincides with our formulation in equation (3.4). The central difference now lies in the different basis functions, namely by choosing them as Finite Elements. A very obvious option are cuboidal elements, as they resemble the underlying voxel grid very well. Roux et al. [104] did so, by choosing 8-node cubic elements with piecewise trilinear functions. For a precise formulation of these functions, we refer to Zienkiewicz, Chapter 6.3 [129].

The minimization procedure is performed very similarly to the local case. As naturally done by Levenberg-Marquardt, the nonlinearity is eliminated by a Taylor-approximation

$$I_1(\mathbf{x}+\mathbf{u}) \approx I_1(\mathbf{x}) + \mathbf{u}(x) \cdot \nabla I_0(\mathbf{x}).$$

Instead of solving directly for the displacement $\mathbf{u}$, a solution for an increment is computed, that is, one solves for $\mathrm{d}\mathbf{u}$ and then updates

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \mathrm{d}\mathbf{u}.$$

The increment can be computed by solving a system of linear equations

$$\mathbf{M}\,\mathrm{d}\mathbf{u} = \mathbf{b}.$$

Clearly, this forms a Newton-scheme, $\mathbf{M}$ is therefore given by the Hessian, so componentwise by

$$M_{ij} = \sum_{\mathbf{x}} \left(\nabla I_0(\mathbf{x}) \cdot \psi_i(\mathbf{x})\right) \left(\nabla I_0(\mathbf{x}) \cdot \psi_j(\mathbf{x})\right).$$

The elements of the right-hand side vector are given by

$$b_i = \sum_{\mathbf{x}} \left(\nabla I_0(\mathbf{x}) \cdot \psi_i(\mathbf{x})\right) \left(I_0(\mathbf{x}) - I_1(\mathbf{x}+\mathbf{u}^n)\right)$$

Obviously, because subvolumes are now finite elements, we have a coupling between respective displacement vectors. However, the resulting vector field is very smooth. This may or may not be an advantage. When the underlying motion is expected to be very smooth, it is most certainly not erroneous to only allow for smooth displacement. Even for discontinuous displacement, global DVC offers helpful tools. In concrete for example, the segmentation of cracks based on gray-value levels usually performs very poorly. We will later see that the inability to match cracks properly in motion estimation can be a very useful tool for detecting cracks.

Anyhow, if one is interested in exact material behavior, the smoothness of the computed displacement field causes problems. One possibility to overcome this problem is to estimate

crack surfaces and to use them as masks, which indicate where displacement shall not be computed. Rethore et al [99] propose to do so by solving for displacement and crack surface in parallel. In a first step, the global DVC is performed as described before. The resulting residual

$$\rho_u(\mathbf{x}) = |I_1(\mathbf{x} + \mathbf{u}(\mathbf{x})) - I_0(\mathbf{x})| \tag{3.10}$$

will exhibit a large error at voxels which display the crack, due to the smoothness assumptions in global DVC. The residual can therefore be thresholded and used as an initial mask to indicate where to expect a crack and where not. Based on this mask, global DVC is recomputed, but this time only outside the crack. Inside the crack, a different set of functions is then used. This procedure is repeated until convergence.

Again, local DVC suffers from an ill-conditioned system matrix when pushing the size of the subvolumes (now in the shape of finite elements) towards voxel level. Leclerc et al. [70] proposed to regularize by the finite element stiffness matrix $\mathbf{K}$, so to set

$$\mathcal{R}(\mathbf{u}) = ||\mathbf{K}\mathbf{u}||_{L_2}. \tag{3.11}$$

3.1.3 Augmented Lagrangian DVC

Recently, Yang at al. [126] proposed to combine both local and global DVC by an augmented Lagrangian approach. In their work, which is abbreviated by ALDVC, two subproblems are solved in an alternating manner: First, a local, IC-GN approach is used to calculate accurate subvolume displacements $\mathbf{u}$. Then a global FE based DVC step calculates an auxiliary displacement field $\hat{\mathbf{u}}$, that suffices the properties, that an outcome of global DVC has *and* is as close as possible to the local DVC displacement $\mathbf{u}$.The optimization therefore is performed as follows. Let the image domain Ω be separated into subvolumes such that $\bigcup_i \Omega_i = \Omega$. Then for each subvolume the classical DVC SSD assumption (3.8) is augmented by terms relating $\mathbf{u}_i$ to the auxiliary variable $\hat{\mathbf{u}}$, and a kinematic compatibility constraint, namely that $\mathbf{F}_i = \nabla\hat{\mathbf{u}}\left(\mathbf{x}_i^0\right)$ and $\mathbf{u}_i = \hat{\mathbf{u}}(\mathbf{x}_i^0)$, where $\mathbf{x}_i^0$ denotes the center point of the subvolume Ω_i. Here, the $\mathbf{F}_i$ will serve as auxiliary variables and are subject to the optimization as well $\hat{\mathbf{u}}_i$ and $\mathbf{u}_i$. Therefore, the following functional, denoted by $\mathcal{L}_0$, is computed such that it is minimal with respect to $\mathbf{u}_i$

$$\begin{aligned}\mathcal{L}_0 \coloneqq \int_{\Omega_i} &\left(I_1(\mathbf{x} + \mathbf{u}_i(\mathbf{x}) + \mathbf{F}_i\left(\mathbf{x} - \mathbf{x}_i^0\right)) - I_0(\mathbf{x})\right)^2 + \frac{\mu}{2}\left|\hat{\mathbf{u}}_i - \mathbf{u}_i\right|^2 \\ &+ \lambda_i\left(\hat{\mathbf{u}}_i - \mathbf{u}_i\right) + \frac{\alpha}{2}\left|\nabla\hat{\mathbf{u}}_i - \mathbf{F}_i\right|^2 + \beta_i : \left(\nabla\hat{\mathbf{u}}_i - \mathbf{F}_i\right)\,\mathrm{d}\mathbf{x}.\end{aligned} \tag{3.12}$$

Now the locally computed unrelated subvolumes are correlated globally by the sum over all subvolumes:

$$\begin{aligned}\mathcal{L} \coloneqq \sum_i \int_{\Omega_i} &\left(I_1(\mathbf{x} + \mathbf{u}_i(\mathbf{x}) + \mathbf{F}_i\left(\mathbf{x} - \mathbf{x}_i^0\right)) - I_0(\mathbf{x})\right)^2 \\ &+ \frac{\mu}{2}\left|\hat{\mathbf{u}}_i - \mathbf{u}_i\right|^2 + \lambda \cdot \left(\hat{\mathbf{u}}_i - \mathbf{u}_i\right) + \frac{\alpha}{2}\left|\nabla\hat{\mathbf{u}}_i - \mathbf{F}_i\right|^2 \\ &+ \beta_i : \left(\nabla\hat{\mathbf{u}}_i - \mathbf{F}_i\right)\,\mathrm{d}\mathbf{x}.\end{aligned} \tag{3.13}$$

Before proposing the alternating algorithm, the authors perform a simplification step that has essential influence on the nature of the computed displacement fields. By setting

$\mathbf{W}_i \coloneqq \beta_i/\alpha$ and $\mathbf{v}_i \coloneqq \lambda_i/\mu$ they derive

$$\begin{aligned}\mathcal{L} = &\sum_i \int_{\Omega_i} \left(I_1(\mathbf{x} + \mathbf{u}_i(\mathbf{x}) + \mathbf{F}_i\left(\mathbf{x} - \mathbf{x}_i^0\right)) - I_0(\mathbf{x})\right)^2 \\ &+ \frac{\alpha}{2}\left|\nabla\hat{\mathbf{u}}_i - \mathbf{F}_i + \mathbf{W}_i\right|^2 + \frac{\mu}{2}\left|\hat{\mathbf{u}}_i - \mathbf{u}_i - \mathbf{v}_i\right|^2 \mathrm{d}\mathbf{x}.\end{aligned} \tag{3.14}$$

This simplification makes heavy use of a structure that comes only with Hilbert spaces, namely that a norm in this space is induced by a scalar product. This simplification therefore only works if we assume $\mathbf{u} \in L_2$. This manifests a strong restriction and in the later evaluation we will see that it causes the computation of very smooth solutions.

A full iteration step is now performed by solving two independent subproblems. Given $\{\mathbf{F}_i^k\}$, $\{\mathbf{u}_i^k\}$, $\{\hat{\mathbf{u}}_i^k\}$, $\{\mathbf{W}_i^k\}$ and $\{\mathbf{v}_i^k\}$, where $\{X_i\} = \{X_i | i \in I\}$ and I being an index set to number all components, solve

- Subproblem 1 (local DVC): Hold $\{\hat{\mathbf{u}}_i^k\}$, $\{\mathbf{W}_i^k\}$ and $\{\mathbf{v}_i^k\}$ fixed and minimize (as we do in the local setting) independently for each i (where i is an index for the component, *not* for the iteration)

$$\{\mathbf{F}_i^{k+1}\}, \{\mathbf{u}_i^{k+1}\} = \underset{\{\mathbf{F}_i\},\{\mathbf{u}_i\}}{\arg\min}\, \mathcal{L}(\{\mathbf{F}_i\}, \{\mathbf{u}_i\}, \{\hat{\mathbf{u}}_i^k\}, \{\mathbf{W}_i^k\}, \{\mathbf{v}_i^k\}).$$

- Subproblem 2 (global DVC): Hold $\{\mathbf{F}_i^{k+1}\}$, $\{\mathbf{u}_i^{k+1}\}$, $\{\mathbf{W}_i^k\}$ and $\{\mathbf{v}_i^k\}$ and solve

$$\{\hat{\mathbf{u}}_i^{k+1}\} = \underset{\{\hat{\mathbf{u}}_i\}}{\arg\min}\, \mathcal{L}(\{\mathbf{F}_i^{k+1}\}, \{\mathbf{u}_i^{k+1}\}, \{\hat{\mathbf{u}}_i\}, \{\mathbf{W}_i^k\}, \{\mathbf{v}_i^k\}).$$

- Subproblem 3 (Lagrange Multiplier update): Update $\{\mathbf{W}_i^k\}$ and $\{\mathbf{v}_i^k\}$ for each i individually by

$$\begin{aligned}\mathbf{W}_i^{k+1} &= \mathbf{W}_i^k + \left(\nabla\hat{\mathbf{u}}_i^{k+1} - \mathbf{F}_i^{k+1}\right) \\ \mathbf{v}_i^{k+1} &= \mathbf{v}^k + \left(\hat{\mathbf{u}}^{k+1} - \mathbf{u}^{k+1}\right).\end{aligned}$$

A stopping criterion can be given either by a maximal number of iterations or by stopping when $|\hat{\mathbf{u}}^{k+1} - \mathbf{u}^{k+1}|$ falls below some user-given threshold.

3.1.4 Convergence Analysis and Benchmarking

The convergence analysis of DVC methods leaves room for improvement. Usually, it is only carried out in terms of benchmarking. That means the authors test if for some artificial example their algorithm produces reasonable results. Consequently, the operability of DVC is rather proven empirically than mathematically. This is already visible in the very first approach of Bay et al. [13]. One can show that the Levenberg-Marquardt method only converges for Lipschitz-differentiable functions. Images, especially those used in materials science, are however for sure not Lipschitz, as they can have arbitrary jumps. The same holds for displacement, and as we saw in Section 2.4, functions *with* jumps seem to be of particular interest in describing deformation. By computing Taylor-approximations to circumvent the highly non-convex and nonlinear evaluation of $\mathbf{u}$ inside the image I_1, one for sure introduces an error towards the true solution. Consequently, it is even more important

to carefully assess the evaluation and benchmarking the respective authors chose. A similar approach is carried out to evaluate the performance of ALDVC.

Benchmarking in DVC is executed in various ways. Bay et al. [13] used repeated scans of the same specimens without any loading and calculated displacement between them. For sure, the outcome should be a zero displacement field, or at most only corrupted by noise. The deviation to this was used to quantify the approximation error. Liu et al. [71] introduced artificially loaded images, to estimate deviation from a non-zero displacement field. However, they used only one *global* displacement vector, so the estimation of displacement results in a heavily overdetermined system. A very thorough investigation was performed by Pan et al. [95], where on top of translation as in Liu, in plane translation (so setting one of the components to 0) and uniaxial tension (varying the value of one component piecewise) was inspected.

Global DVC is usually evaluated in its performance when noisy images are considered. In [69], the original image was correlated with the same image but with artifically added noise. Again, the deviation from zero displacement was considered. In this case, one can even derive closed form solutions.

Let us spend some last words on the terminology frequently used in DVC. Almost all algorithms claim to be "subvoxel accurate" [95] and that they are precise within the range of 0.002 voxels [13]. This must be understood as a purely technical result. If the displacement, that the present sample underwent, can be resolved by the coarse displacement that DVC offers, then these accuracy results hold true. If we recall the notion of supervoxels from the beginning of this chapter, this means that the number of distinct values of the original displacement must not be larger than the number of supervoxels (times three). However, as we will see later, it is nearly impossible to resolve low scale displacement in magnitudes of few voxels with this approach.

3.2 Medical Image Registration

Medical Image Registration (MIR) has been around for very long and therefore the fixation of an initial work is quite hard. Already in 1996, Elsen et al. [125] decided that it is time for a proper review. Since then, approximately every 10 years the list of algorithms is updated. Note that also several books [86, 51, 45] have been published in this area. For a recent review, we propose to combine the review of classical methods by Oliveira et al. [92] with the more recent review by Fu et al. [41], focusing on application of Deep Learning in that area.

In this thesis, we will therefore present only selected approaches that seemed suitable for our applications, and refer to the above mentioned reviews for a general overview in that topic.

MIR fits exactly in our framework of motion estimation by regularization. It usually consists of functionals, that can be split into two parts: The so called distance measure denoted by $\mathcal{D}$ and the regularizer denoted by $\mathcal{R}$. Instead of searching for a displacement u, a *transformation* ϕ is computed. The transformation shall be such that $I_0 \circ \phi = I_1$. As in DVC, ϕ will be then the solution to a minimization problem

$$\min_{\phi} \mathcal{D}(I_0, I_1, \phi) + \lambda \mathcal{R}(\phi), \tag{3.15}$$

where λ is a positive real constant.

Remark 3.2.1: Though the regularizer here is defined to be dependent on ϕ, most of the algorithms will assume a direct dependence on the motion model. Detailed explanations will be postponed to the explicit algorithm description.

Many MIR algorithms require the transformation ϕ to be diffeomorphic, that means the transformation is differentiable and has a differentiable inverse. This allows a sound mathematical treatment, but will suffer from the same problems as global DVC, namely a very smooth description of motion.

3.2.1 Elastic Image Registration

Elastic image registration (EIR) was first proposed by Broit in 1981 [24]. Most of its variants choose the infinitesimal equivalent of the sum of squared differences

$$\mathcal{D}^{\text{SSD}}(I_0, I_1, \phi) \coloneqq \int \left(I_1(\phi(\mathbf{x})) - I_0(\mathbf{x})\right)^2 \, \mathrm{d}\mathbf{x} \tag{3.16}$$

as distance measure, but some also considered its mass preserving modification

$$\mathcal{D}^{\text{MP}}(I_0, I_1, \phi) \coloneqq \int \left(I_1(\phi(\mathbf{x}))\text{det}\nabla\phi(x) - I_0(\mathbf{x})\right)^2 \, \mathrm{d}\mathbf{x}.$$

As the name indicates, *elastic* image registration is inspired by the mechanical model of elasticity. Recall from the introduction that elasticity assumes that with release of forces, the body returns to its reference configuration. The behavior under application of force is usually described by a displacement based formulation, so that

$$\phi(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x}), \tag{3.17}$$

where $\mathbf{x}\colon \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$.

Then a regularizer can be formulated in terms of the displacement $\mathbf{u}$, that is, the *linearized elastic potential* is proposed as

$$\mathcal{R}^{\text{elas}}(\mathbf{u}) = \int_\Omega \frac{\mu}{4} \left(\nabla\mathbf{u} + \nabla^T\mathbf{u}\right)^2 + \frac{\lambda}{2} \left(\operatorname{div}\mathbf{u}\right)^2 \, \mathrm{d}\mathbf{x}. \tag{3.18}$$

This regularizer has a very nice physical interpretation as it is based on the deformation of an elastic body. Its Euler-Lagrange equation becomes the so called *Navier-Lamé-equation*, which reads

$$f = \mu\Delta\mathbf{u} + (\lambda + \mu)\nabla \operatorname{div}\mathbf{u}.$$

Note that in this derivation, the linearization of the strain tensor E is essential. The *Green-St Venant strain tensor* E is defined by

$$E = \frac{1}{2}\left(C - \mathbb{I}\right), \qquad C = \nabla\phi^T\nabla\phi, \tag{3.19}$$

with ϕ being the deformation a body undergoes. If ϕ only consists of small deformations, so that $\phi(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x})$ with small $\mathbf{u}$, and that we deal with a *St Venant-Kirchhoff material*, E can be written as

$$E(\mathbf{u}) = \frac{1}{2}\left(\nabla\mathbf{u}^T + \nabla\mathbf{u} + \nabla\mathbf{u}^T\nabla\mathbf{u}\right),$$

which is up to constants the regularizer that finds its way into (3.18). For a detailed derivation of the mechanical aspects, we refer to Ciarlet, and for a derivation with respect to motion estimation to the monograph of Modersitzki [86].

Due to the derivation of the equations via linearization, we cannot deviate from the assumptions of small strains. In image registration however, we do not want to limit us to this case. In continuum mechanics, the framework to deal with bodies that are subjected to large deformations is called *hyperelasticity*. A material is called hyperelastic, if its stress tensor can be expressed via the derivative of the *stored energy function*. If the strain tensor E is small, such a function for example reads

$$W(C) = \frac{\lambda}{2}\left(\text{trace}(E)\right)^2 + \mu\,\text{trace}(E^2) + o(||E||^2),$$

where $C = \mathbb{I} + 2E$. A stored energy function W should satisfy the following properties (which we here consider in dependence of some deformation ϕ instead of a tensor C)

- $W(\phi) \to \infty \quad$ for $\quad \det \nabla\phi \to 0$,
- $W(\phi) \geq c_1\{||\nabla\phi||^p + ||\text{cof}\nabla\phi||^q + \det(\nabla\phi)^r\} + c_2$,

where $c_1 = 0$, $c_2 \in \mathbb{R}$ and $p, q, r > 1$. Burger et al. [28] proposed to use the second growth condition as regularizer in the form

$$\mathcal{R}_{\text{hyper}}(\phi) = \int \alpha_1 \text{length}(\phi) + \alpha_2 \text{surface}(\phi) + \alpha_3 \text{volume}(\phi)\,\mathbf{dx}, \tag{3.20}$$

where

$$\begin{aligned}
\text{length}(\phi) &\coloneqq ||\nabla\phi - \mathbb{I}||^2_{\text{Fro}} \\
\text{surface}(\phi) &\coloneqq (||\text{cof}\nabla\phi||_{\text{Fro}} - 3)^2 \\
\text{volume}(\phi) &\coloneqq \frac{(\det\nabla\phi - 1)^2}{\det\nabla\phi},
\end{aligned}$$

where cofactor and determinant are given by

$$\text{cof}\nabla\phi = \begin{pmatrix}
\partial_2\phi_2\partial_3\phi_3 - \partial_3\phi_2\partial_2\phi_3 & \partial_3\phi_2\partial_1\phi_3 - \partial_1\phi_2\partial_3\phi_3 & \partial_1\phi_2\partial_2\phi_3 - \partial_2\phi_2\partial_1\phi_3 \\
\partial_3\phi_1\partial_2\phi_3 - \partial_2\phi_1\partial_3\phi_3 & \partial_1\phi_1\partial_3\phi_3 - \partial_3\phi_1\partial_1\phi_3 & \partial_2\phi_1\partial_1\phi_3 - \partial_1\phi_1\partial_2\phi_3 \\
\partial_2\phi_1\partial_3\phi_2 - \partial_2\phi_1\partial_2\phi_2 & \partial_3\phi_1\partial_1\phi_2 - \partial_1\phi_1\partial_3\phi_2 & \partial_1\phi_1\partial_2\phi_2 - \partial_2\phi_2\partial_2\phi_2
\end{pmatrix},$$

$$\begin{aligned}
\det\nabla\phi = {} & \partial_1\phi_1\partial_2\phi_2\partial_3\phi_3 + \partial_2\phi_1\partial_3\phi_2\partial_1\phi_3 + \partial_3\phi_1\partial_1\phi_2\partial_2\phi_3 \\
& - \partial_1\phi_3\partial_2\phi_2\partial_3\phi_1 - \partial_2\phi_3\partial_3\phi_2\partial_1\phi_1 - \partial_3\phi_3\partial_1\phi_2\partial_2\phi_1,
\end{aligned}$$

and the Frobenius norm $||A||_{\text{Fro}} = \sqrt{\sum A^2_{i,j}}$. The above penalizers on length, surface and volume changes have direct mechanical interpretations. Remember that (hyper)elastic image registration borrows ideas from mathematical elasticity. There, one is frequently concerned with the question how a deformation ϕ alters (infinitesimal) volume, surface and length elements. Often, the relation can be established with basic tools from linear algebra, volume elements for example are transformed via $\det\nabla\phi$ (which we all know from calculus classes, as it is the core of change of variables in multiple dimensions). Similar results hold for the cofactor matrix as stated above – it describes how surface elements are

transformed by a deformation ϕ. Unfortunately, this mechanically justified penalizer is not convex anymore and so is the resulting minimization. A convexification is possible, but does not penalize surface shrinkage anymore, which is not desirable in the case of (hyper)elastic materials.

To pose an existence result for the hyperelastic minimization, Burger et al. [28] require the deformation to be in the Sobolev space $W^{1,2}$. In [86], the solutions for the elastic image registration problem even have to be C^2 (it holds that $C^2 \subset W^{1,2}$). Both spaces are also Hilbert spaces. Again, as in the case of inverse problems, we remark that this produces very smooth solutions. In elasticity, and also in MIR, this choice is perfectly justifiable, as no tearing or fracture is apparent. This fact can be assumed for imaging of the human body as well. If we monitor movement of the lung during breathing, or heart beats, we do not expect any damage. However, in the design of material tests, one almost always wishes to monitor plastic deformation, which initiates material failure. Cracks and fracture for sure will introduce new surface area, and therefore non-smooth solutions. Hence, we once more declare this approach to registration to be only of limited usefulness in our setting.

Optimization is performed fairly similarly to Section 2.2.1. The equations are cast into a bilinear and a linear form, and after discretization the resulting linear systems are solved by a linear solver of choice.

Recently, Aggrawal et al. [2] proposed a framework to introduce at least some level of discontinuity into (hyper)elastic image registration. In the case of cracked specimens smooth transformations can be assumed at least outside the crack. That means if one could reconstruct a phase field function for a crack, it can be used to weigh the registration functionals. Hence, let $\Gamma \subset \Omega$ be the crack, and let

$$p\colon \Omega \to [0,1],$$

be its phase field function. p should ideally be an indicator, which only attains values 0 outside and 1 on the crack, but often smoothly varies in $[0,1]$. Then, for $\mathcal{D}^{\text{SSD}}$ as in (3.16) and $\mathcal{R}_{\text{hyper}}$ as in (3.20), the indicator is incorporated as follows

$$J(\phi, p) \coloneqq \int_\Omega \alpha(p)(I_1(\phi(\mathbf{x})) - I_0(\mathbf{x})) + \beta(p)\mathcal{R}_{\text{hyper}}(\phi(\mathbf{x})) + \gamma e(p)\,\mathrm{d}\mathbf{x}, \tag{3.21}$$

where $e(p) = p^2$, $\alpha(p) = (1-p)^2$ and $\beta(p) = \epsilon + \lambda(1-p)^n$, $\epsilon, \lambda > 0$, $n \in \mathbb{N}$. As the independence of p in $J(\phi, p)$ indicated, optimization is now performed for ϕ *and* p in an alternating manner, i.e.

$$\phi^{k+1} = \arg\min_{\phi} J(\phi, p^k)$$
$$p^{k+1} = \arg\min_{p} J(\phi^{k+1}, p).$$

The penalty e on the crack indicator originates from the energy function of a strongly brittle material. Following Marigo et al. [78], such materials are assumed to fulfill the following properties.

1. The material is governed by a damage parameter which is strictly increasing from 0 (undamaged) to 1 (completely damaged).

2. The state of each volume element is characterized by a triplet $(\varepsilon, p, \mathbf{g})$. ε denotes the strain, p the damage parameter, and $\mathbf{g}$ being the gradient of the damage vector

($\mathbf{g} = \nabla p$). Note that the damage parameter will coincide with the crack indicator. In the optimization process in the image registration, the strict condition of p being an indicator is dropped and replaced by a characterization as damage parameter as stated here.

3. The bulk energy density of the material is given by a state function depending on $(\varepsilon, p, \mathbf{g})$. It can be assumed to consist of three parts: a stored elastic energy, dependent on (ε, p), a local and a non-local part of the dissipated energy. The non local part is given by $\frac{1}{2} w_1 \ell \mathbf{g} \cdot \mathbf{g}$. The local part is a positive increasing function depending on p.

Inspired by the Mumford-Shah functional, Marigo et al. now propose energy functions that suffice the Ambrosio-Tortorelli limit (equation 2.68), which we already met in the Mumford-Shah functional for image segmentation (equation 2.67). That is,

$$e(p) = p^2 + \ell |\nabla p|^2. \tag{3.22}$$

However, the authors of [2] decided to drop the second part of the energy, i.e. set ℓ to 0, to "keep the numbers of tunable parameters small." Analysing the results, we can observe the following. As it is a variant closely related to Mumford-Shah, and because it poses a weight based on the *image* and *not* on the displacement, the crack indicator does not only indicate cracks but also edges. In the challenges we want to tackle this is fatal: Foams in CT will merely consist of high contrast sharp edges, and therefore this method of crack reconstruction labels the *whole* sample as crack. The same holds also for concrete, which based on the content, is rather similar to the application of the original authors of [2]: They analysed brain images, containing tears due to sample preparation. One can observe cracks, pores and different material phases, just as in concrete, and the algorithm fails to extract the crack only, and not other materials boundaries.

3.2.2 Large Displacement Diffeomorphic Metric Mapping

To overcome the problem that large deformations cannot be mapped by elastic regularization yet another solution was proposed. Instead of modeling the motion via displacements, i.e. $\phi(\mathbf{x}) = \mathbf{x} + \mathbf{u}(\mathbf{x})$, Christensen et al. [33] proposed to use a transformation map that is now described by ordinary differential equations. The transformation map ϕ will be the result of an integration by time of smooth, time-dependent velocity fields $\mathbf{v}$: $[0,1] \times \Omega \to \mathbb{R}^d$

$$\dot{\phi}(\tau, \mathbf{x}) = \mathbf{v}(\tau, \phi(\tau, \mathbf{x})), \qquad \phi(0, \mathbf{x}) = \mathbf{x}, \tag{3.23}$$

where we denoted the temporal derivative by $\dot{\phi}$. From a continuum mechanics point of view, this can be considered as moving from solid to fluid mechanics. Note that both theories are based on Euler's first law of motion, therefore Christensen used exactly the same regularizer as in (3.18), but this time not posed on displacement, but on velocity. The overall registration is then only complemented by an additional Euler-step to calculate ϕ from $\mathbf{v}$.

Nowadays, this method is counted as a variant of *Large Displacement Diffeomorphic Metric Mapping* (LDDMM). All algorithms belonging to that framework have exactly the novelty of Christensen et al. in common, namely that they describe the transformation by time-dependent velocities. As this approach almost always produces diffeomorphic mappings, the designation is self-explanatory. LDDMM has since then become one of the most used frameworks in MIR, with numerous improvements and applications.

In 2005 for example (in fact this is the first work using the term LDDMM), Beg et al. [14] moved away from the justification in continuum mechanics and proposed a method, that is very attractive from an optimization point of view. Instead of choosing the regularizer as elastic potential, the minimization problem now reads

$$\hat{\mathbf{v}} = \underset{\mathbf{v}:\dot{\phi}_t=\mathbf{v}_t(\phi_t)}{\arg\min} \left(\int_0^1 ||\mathbf{v}_t||_V^2 + \frac{1}{\lambda^2} ||I_0 \cdot \phi - I_1||_{L_2}^2 \, dt \right). \tag{3.24}$$

They were able to show that an optimizer of this formulation generates a geodesic path of shortest length on the manifold of diffeomorphisms which connect the images.

For further advancements and applications, a very detailed review can be found in Polzin [96]. In this work, we will focus on the special representative, namely on the work of Mang et al. [76]. They propose to use, again, the SSD-distance as similarity measure and either the H1 seminorm ($\mathcal{R}(\mathbf{v}) = \nabla\mathbf{v}$) or the vectorial Laplace operator ($\mathcal{R}(\mathbf{v}) = \Delta\mathbf{v}$) as regularizer. The key difference is now the formulation as optimal control problem. The image is considered as a time dependent state variable, so $I\colon \bar{\Omega} \times [0,1] \to \mathbb{R}, (\mathbf{x}, t) \mapsto I(\mathbf{x}, t)$. $I(\cdot, 1)$ shall therefore correspond to the *final state*, i.e. $I(\mathbf{x}, 1) = I_1(\phi(\mathbf{x}))$. The velocity $\mathbf{v}$ serves as control variable. When considering motion estimation in images, there is a very natural constraint that can be posed. Consider the case of only small movements $\Delta\mathbf{x}$ between two arbitrary time steps $I_t \coloneqq I(\cdot, t)$ and $I_{t+\Delta t} \coloneqq I(\cdot, t + \Delta t)$. Our very first assumption on the preservation of intensity reads

$$I_t(\mathbf{x}) = I_{t+\Delta t}(\mathbf{x} + \Delta\mathbf{x}).$$

A first order Taylor approximation with respect to $\mathbf{x}$ *and* t of the left-hand side gives

$$I_{t+\Delta t}(\mathbf{x} + \Delta\mathbf{x}) = I_t(\mathbf{x}) + \nabla I \cdot \Delta\mathbf{x} + \partial_t I \Delta t,$$

which therefore, after dividing by Δt and considering the infinitesimal limit results in

$$\partial_t I + \nabla I \cdot \mathbf{v} = 0.$$

Remark 3.2.2: We will revisit the above derivation later in this thesis, as it also depicts the formulation of the very first Optical Flow algorithms.

Collecting all the ingredients so far and adding a condition on the incompressibility on $\mathbf{v}$ leads us to the following constrained minimization problem

$$\begin{aligned} \min_{\mathbf{v}} \coloneqq & \frac{1}{2} \int_\Omega \left(I_0 - I_1 \circ \phi\right)^2 \, \mathrm{d}\mathbf{x} + \int_0^1 \mathcal{R}(\mathbf{v}) \, dt \\ & \text{subject to } \mathscr{C}[I, \mathbf{v}] = 0, \end{aligned} \tag{3.25}$$

with

$$\mathscr{C}(I, \mathbf{v}) \coloneqq \begin{cases} \partial_t I + \nabla I \cdot \mathbf{v} & \text{in } \Omega \times (0, 1] \\ I - I_1 & \text{in } \Omega \times \{0\} \\ \gamma(\nabla \cdot \mathbf{v}) & \text{in } \Omega. \end{cases}$$

The constraint $I - I_1 = 0$ in $\Omega \times \{0\}$ might not seem intuitive at first sight. It means, that at $t = 0$ the difference of $I_1(\phi)$ and I_0, which is nothing else than using the "classical" distance of I_0 and I_1 as boundary condition for the PDE constraint.

The PDE constraints are hyperbolic, therefore the KKT-conditions from Section 2.2.3 are only necessary. For sufficient conditions, one has to consider the second variation and choose a corresponding solver. We refer to the original work of Mang et al. [76] for a detailed derivation of the numerical solution.

The same authors put much effort in improving their method with regards to computation time and memory usage. LDDMM almost never takes "the easy way out", so performing a linearization step to gain necessary *and* sufficient optimality conditions, as one derives a convex minimization, for which first order variation yields the desired optimality. Therefore, the algorithm CLAIRE, as baptized by the authors, manifests the most suitable representative of this group for us. Note however, that it suffers from the same problems as (hyper)elastic registration. In fact, because (hyper)elastic registration produces also diffeomorphic transformations, it can even be considered to be a variant of LDDMM. Once more we calculate smooth solutions that seem to be a problem when fracture occurs.

Chapter 4

Optical Flow for Volume Images

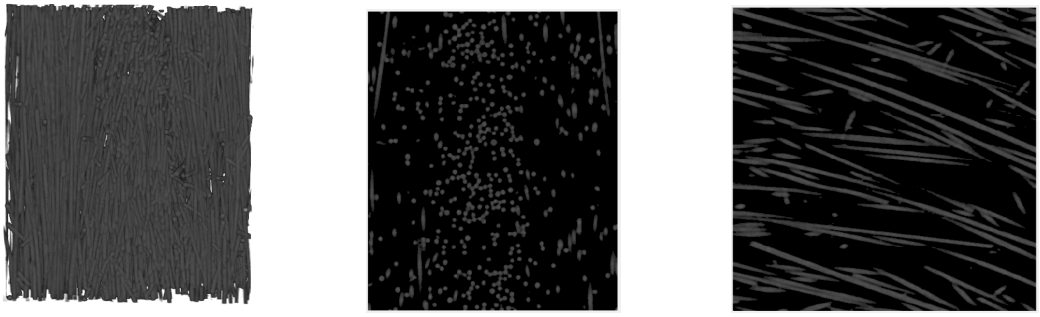
In the previous chapter we have seen a variety of algorithms that were developed for the special purpose to estimate motion in images that stem from CT acquisition schemes (or other tomographic approaches in MIR). One could therefore ask why we do not stick with the existing solutions but borrow algorithms from a different area of motion estimation. The answer to this question was already hinted within some of the algorithm descriptions: Neither DVC nor MIR fulfill our requirements sufficiently.

Macroscopically, the evolution of a material during the test can be described and monitored rather easily: Elastic materials for instance elongate when tension is applied, or shrink when compressed. For materials like foams, however, though they seem to behave elastically at first glance, a closer look into the microstructure gives better insight into the mechanisms of failure. For instance, metal-matrix composite (MMC) foams deform in compression tests first elastically before full layers collapse. The failure of the foam is preceded by breaking struts. A change on a small scale therefore influences the stability of the whole component. A similar observation can be made for concrete: fault zones are generally preceded by a large number of microfractures. CT devices that generate images with voxel sizes in the micrometer range (μCT) are means to resolve this microstructural behavior. Nevertheless, the challenge to quantify this deformation on the micrometer scale remains.

Unfortunately, this task is very complex. Even when the imaged components are manufactured in sizes of a few millimeters to centimeters, the resulting CT data sets may have a total number of up to $2\,048^3$ voxels in 16-bit gray value range. Therefore, an algorithm is required that is not only efficient in time but also in its use of memory.

Another challenge is unveiled immediately, too: the quality of classical algorithms for estimating deformation fields in two or three dimensions increases if the image displays many different shapes with distinct gray values. In CT data of materials one hardly ever observes more than a few distinct intensity levels.

In addition, estimation in medical images often benefits from the fact that the objects which undergo the motion are fully contained in the image. If, for example, a CT of the head is aligned to another one, the outer boundary is already a very powerful feature to support the registration process. Furthermore, these features can be used to generate landmarks to support the process: Either a doctor sets the landmarks intuitively manually [101] or an algorithm constructs landmarks automatically based on a clearly detectable feature [97]. In contrast, constructing landmarks is highly challenging, if not impossible, for materials samples. Figure 4.1, which is an excerpt of a sample of long-fiber reinforced thermoplastics, shows a typical example lacking both, an outward bound of the imaged structure and unique



(a) Volume (b) xz-slice (c) yz-slice

Figure 4.1: Volume and slice view of the fiber system in a long glass fiber reinforced composite (LFT). Sample Sabic. Imaging Fraunhofer ITWM. Image size $500 \times 600 \times 500$ at voxel edge length 5μm.

structural features as candidates for landmarks.

In fact, even with in situ testing it is not guaranteed that the scanned areas before and after the material test coincide. A meaningful test for example might demand a certain thickness of the material, so that the material bears a minimal required load. If a foam or a fiber system is eventually manufactured in components with a thickness of several centimeters, performing tests on few layers that cover only micrometers may not be meaningful. Therefore, to guarantee the microstructure to be properly resolved, only parts of the sample are scanned. If the material is only slightly elastic in this case, the boundaries in both scans do not coincide anymore.

In summary, an algorithm for estimating displacement has to fulfill the following requirements: First, our method must be a local one in the sense that local behavior such as breaking struts or delamination of fibers is resolved accurately. Next, our algorithm needs to work robustly, even if the volume images do not contain very rich features. And last, the algorithm must yield a valid displacement field, even if the fields of view of the images do not coincide. Evaluating the algorithms presented in the previous chapter with respect to these requirements unveils why they are not sufficient: DVC lacks the ability to resolve local behavior, not only because of the restrictions regarding the size of the subvolumes, but also because some methods indirectly pose strong smoothness assumptions which cannot be fulfilled by the motion we expect from in situ tests. This second point also counts for MIR, but even more severely: As these algorithms require the motion to be diffeomorphic we have no chance to map any kind of crack or fracture properly.

The remedy we propose is to extend algorithms from 2D. Estimating motion between digital images is a problem scientists deal with nearly as long as they deal with processing the images itself. A quick search on “Optical Flow” on Google Scholar gives 5 Million results, nearly as much as probing “Deep Learning” (5.1 million results), and much more than MIR (2.7 million) or DVC (70 000).

However, Optical Flow is mainly considered a plain 2D algorithm. Although some extensions to volume images exist ([56, 89]), “3D” in Optical Flow usually has a completely different meaning, namely the incorporation of time as third dimension (so to describe the motion over more than two images), or by deriving three dimensional information from two dimensional data (scene flow).

In the following chapter, we will give an introduction to its (two dimensional) roots and spend some words about the recent trends. We will then present the three dimensional extension of one of the most celebrated algorithms (Large Displacement Optical Flow), and its famous successor and variants of them (primal dual methods).

4.1 The Roots

In this contribution, we want to set the roots of Optical Flow to the celebrated algorithms of Lucas-Kanade [73] and Horn-Schunck [58], which both date back to the year 1981. Once more we assume intensity constancy, namely that for images $I_0, I_1 \colon \Omega \subset \mathbb{R}^3 \to \mathbb{R}$ we have

$$I_0(\mathbf{x}) = I_1(\mathbf{x} + \mathbf{u}(\mathbf{x})) \tag{4.1}$$

with a suitable displacement $\mathbf{u} \colon \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$. As in LDDMM, if the displacements are small, a first order Taylor approximation yields the infinitesimal equivalent of equation (4.1)

$$\nabla I \cdot \mathbf{u} + \partial_t I = 0. \tag{4.2}$$

Again, no matter which variant we consider, we try to derive a three dimensional (in our case, in the original formulation of course only two dimensional) displacement vector field from scalar image data. The problem is therefore ill-posed. Interestingly, the early works of Optical Flow can be divided rather similarly to DVC, namely into a local and a global approach [26]. The local approach is given by the Lucas-Kanade algorithm, which assumes that the displacement vector field is piecewise constant, that is, in a neighborhood of size ρ, $\mathbf{u}$ is reconstructed via a weighted least squares fit by minimizing

$$J^{\mathrm{LK}}(\mathbf{u}) = K_\rho * \left((\nabla I \cdot \mathbf{u} + \partial_t I)^2 \right),$$

where $K_\rho *$ is the convolution of the right-hand side with a Gaussian kernel of size ρ. Note the small but significant difference to DVC: Though we consider the displacement in a neighborhood, we still perform the minimization for *each* voxel with respect to its neighborhood, and not *once* for the whole neighborhood.

The functional to be minimized is convex, therefore first order variation is sufficient for finding a minimizer and the resulting linear system for each component of the displacement field (u, v, w) reads

$$\begin{pmatrix} K_\rho * (I_x)^2 & K_\rho * (I_x I_y) & K_\rho * (I_x I_z) \\ K_\rho * (I_x I_y) & K_\rho * (I_y)^2 & K_\rho * (I_y I_z) \\ K_\rho * (I_x I_z) & K_\rho * (I_y I_z) & K_\rho * (I_z)^2 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} -K_\rho * (I_x I_t) \\ -K_\rho * (I_y I_t) \\ -K_\rho * (I_z I_t) \end{pmatrix},$$

where (I_x, I_y, I_z) are the components of ∇I and $\mathbf{u} = (u, v, w)$.

In the *global* approach, as in DVC, no neighborhoods are considered anymore and the problem is tackled based on the whole image. Contrary to DVC, this implies a *dense*, so voxel scale flow field computed for *all* image positions. Horn and Schunck proposed to minimize a *regularized* energy functional of the form

$$J^{\mathrm{HS}}(\mathbf{u}) = \int_\Omega (\nabla I \cdot \mathbf{u} + \partial_t I)^2 + \alpha \left(|\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2 \right) \mathrm{d}\mathbf{x},$$

for $\alpha > 0$ and $|\cdot|$ the Euclidean norm. The functional is convex and the Euler-Lagrange-equations read

$$\begin{aligned} 0 &= \Delta u - \frac{1}{\alpha}(I_x^2 u + I_x I_y v + I_x I_z w + I_x I_t) \\ 0 &= \Delta v - \frac{1}{\alpha}(I_x I_y u + I_y^2 v + I_y I_z w + I_y I_t) \\ 0 &= \Delta w - \frac{1}{\alpha}(I_x I_z u + I_y I_z v + I_z^2 w + I_z I_t). \end{aligned}$$

To overcome the restrictions to small displacements (which is required to compute a linearization as in (4.2)), Optical Flow is almost always computed within a coarse-to-fine scheme. That means, the image is filtered with for example a Gaussian or a Laplacian filter, and then downscaled. This procedure is repeated, until the expected displacement in the resulting coarse images *can* be assumed to be small and linearization can be performed with only a small error. Then, the displacement is computed on this coarse scale. Images and displacement are upsampled, and used as starting value for the Optical Flow computation on the next finer scale. Another procedure to overcome the limitations to small displacements is to employ a warping strategy. That is, an additional outer iteration is introduced. In this iteration $I_1(\mathbf{x}+\mathbf{u})$ is reinitialized by using the displacement that was calculated by the previous iteration, to *warp* $I_1(\mathbf{x}+\mathbf{u})$ towards I_0. The resulting image is used as "new" I_1 in the next iteration. We will later see that this procedure drastically improves the performance when large displacement is apparent, but we also want to note that it comes with an equally drastic increase of computation time.

However, both the local and the global approach produce continuous solutions, once more due to the choice of the smooth penalizer on $\nabla\mathbf{u}$. From the vast number of contributions dealing with Optical Flow, one can see that much effort was made to produce *discontinuous* solutions, that allow for a more realistic motion model. One of the first authors to introduce bounded variation into motion estimation was Cohen in 1993 [34]. Instead of using the L_2 norm of $\nabla\mathbf{u}$, he proposed to use the L_1-norm. However, to obtain solutions the author used variational methods and required $\mathbf{u}$ to be in the Sobolev-space H^1. This choice has to be handled with care, as it is again Hilbert and – ad hoc – again too smooth for our demands. Under certain circumstances one can nevertheless use smooth approximations of a similar form. The next section will cover such an algorithm and its 3D extension. The work is one of the milestones of TV Optical Flow, namely the work of Brox et al. [25].

4.2 Large Displacement Optical Flow

The starting point of Brox et al. [25] is to not use the classical L_2 distance measure alone, but to use a concave function $\Psi(s) = \sqrt{s^2+\varepsilon^2}$, $\varepsilon = 0.0001$, to reduce the proneness towards outliers

$$J_{\text{Data}}(\mathbf{u}) = \int_\Omega \Psi\left(|I_0(\mathbf{x}) - I_1(\mathbf{x}+\mathbf{u})|\right)\,\mathrm{d}\mathbf{x}, \tag{4.3}$$

where $|\cdot|$ is the Euclidean norm.

The problem is regularized by total variation, but this time (again) Ψ is applied. Immediately, we see another merit of such a function, as it constitutes a differentiable approximation to TV

$$J_{\text{Smooth}}(\mathbf{u}) = \int_\Omega \Psi\left(|\nabla u|+|\nabla v|+|\nabla w|\right)\,\mathrm{d}\mathbf{x}. \tag{4.4}$$

The novelty of [25] is a third penalty on differences in gradients of both images. The original purpose of this so called gradient constancy assumption was to enhance robustness with respect to changes in illumination. We noticed that in our applications this term puts special emphasis on features, mainly on prominent edges. It connects the gradients of the original image and the deformed image by

$$J_{\text{Grad}}(\mathbf{u}) = \int_\Omega \Psi\left(|\nabla I_0(\mathbf{x}) - \nabla I_1(\mathbf{x}+\mathbf{u})|\right) \, \mathrm{d}\mathbf{x}. \tag{4.5}$$

The improvement caused by this additional term can be seen in Section 6.2.1. The variational problem now consists of finding a solution $\mathbf{u} = (u, v, w)$, such that

$$J_{\text{total}}(\mathbf{u}) = J_{\text{Data}}(\mathbf{u}) + \lambda J_{\text{Grad}}(\mathbf{u}) + \mu J_{\text{Smooth}}(\mathbf{u}) \tag{4.6}$$

is minimal. λ and μ are regularization parameters and are chosen problem dependent, but larger than 0.

Calculating the first variation $J'(u)$, which is given by

$$J'_{\text{total}}(u) = \frac{\partial}{\partial \tau} J_{\text{total}}(\mathbf{u} + \tau \mathbf{v})|_{\tau=0},$$

yields componentwise

$$\begin{aligned} 0 = {} & \Psi'(I_d^2) I_d I_x \\ & + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xx} I_{xd} + I_{xy} I_{yd} + I_{xz} I_{zd}) \\ & + \mu \text{div}\left(\Psi'\left(|\nabla u| + |\nabla v| + |\nabla w|\right) \nabla u\right) \end{aligned} \tag{4.7}$$

$$\begin{aligned} 0 = {} & \Psi'(I_d^2) I_d I_y \\ & + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xy} I_{xd} + I_{yy} I_{yd} + I_{yz} I_{zd}) \\ & + \mu \text{div}\left(\Psi'\left(|\nabla u| + |\nabla v| + |\nabla w|\right) \nabla v\right) \end{aligned} \tag{4.8}$$

$$\begin{aligned} 0 = {} & \Psi'(I_d^2) I_d I_z \\ & + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xz} I_{xd} + I_{yz} I_{yd} + I_{zz} I_{zd}) \\ & + \mu \text{div}\left(\Psi'\left(|\nabla u| + |\nabla v| + |\nabla w|\right) \nabla w\right) \end{aligned} \tag{4.9}$$

where

$$\begin{aligned} I_x &= \partial_x I_1(\mathbf{x}+\mathbf{u}) & I_{xx} &= \partial_{xx} I_1(\mathbf{x}+\mathbf{u}) \\ I_y &= \partial_y I_1(\mathbf{x}+\mathbf{u}) & I_{xy} &= \partial_{xy} I_1(\mathbf{x}+\mathbf{u}) \\ I_z &= \partial_z I_1(\mathbf{x}+\mathbf{u}) & I_{xz} &= \partial_{xz} I_1(\mathbf{x}+\mathbf{u}) \\ I_d &= I_0(x) - I_1(\mathbf{x}+\mathbf{u}) & I_{yy} &= \partial_{yy} I_1(\mathbf{x}+\mathbf{u}) \\ I_{xd} &= \partial_x I_d & I_{yz} &= \partial_{yz} I_1(\mathbf{x}+\mathbf{u}) \\ I_{yd} &= \partial_y I_d & I_{zz} &= \partial_{zz} I_1(\mathbf{x}+\mathbf{u}) \\ I_{zd} &= \partial_z I_d. \end{aligned} \tag{4.10}$$

Recall now the Kačanov method from Section 2.5.1 applied to total variation denoising. We saw that applying the method results in freezing the nonlinear parts. In the case of denoising, these were only given in the regularizer. If we now consider equations (4.7)-(4.9), we see that almost all terms regarding $\mathbf{u}$ are non-linear. Therefore, two nested Kačanov methods are applied. The first one freezes the stand-alone partial derivatives. That is, all terms involving I_x, I_y, I_z or their second order terms, are equipped with an iteration index k, whereas all other terms are equipped with $k+1$. As an example, the resulting equation for u reads

$$\begin{aligned}0 &= \Psi'((I_d^{k+1})^2) I_d^{k+1} I_x^k \\ &+ \lambda \Psi'((I_{xd}^{k+1})^2 + (I_{yd}^{k+1})^2 + (I_{zd}^{k+1})^2) \\ &\qquad (I_{xx}^k I_{xd}^{k+1} + I_{xy}^k I_{yd}^{k+1} + I_{xz}^k I_{zd}^{k+1}) \\ &+ \mu \operatorname{div}\left(\Psi'\left(|\nabla u^{k+1}|+|\nabla v^{k+1}|+|\nabla w^{k+1}|\right)\nabla u^{k+1}\right).\end{aligned}$$

The next step also involves a Taylor approximation, but of quite a different nature compared to equation (4.2). Instead of linearizing with respect to $\mathbf{u}$, we will linearize only with respect to small increments (du, dv, dw) such that $u^{k+1} = u^k + du^k$, $v^{k+1} = v^k + dv^k$ and $w^{k+1} = w^k + dw^k$ holds. In contrast to linearizations as in (4.2), this is a less strong restriction, as we can assume that *increments* are in fact small. Furthermore, as we apply Kačanov's method, $\mathbf{u}$ will be computed via fixed point iterations (see (2.83)). Separation of $\mathbf{u}$ into increment and update is therefore a very natural procedure. Taylor approximations of the increments now read

$$\begin{aligned}I_d^{k+1} &\approx I_d^k + I_x^k du^k + I_y^k dv^k + I_z^k dw^k \\ I_{xd}^{k+1} &\approx I_{xd}^k + I_{xx}^k du^k + I_{xy}^k dv^k + I_{xz}^k dw^k, \\ I_{yd}^{k+1} &\approx I_{yd}^k + I_{xy}^k du^k + I_{yy}^k dv^k + I_{yz}^k dw^k \\ I_{zd}^{k+1} &\approx I_{zd}^k + I_{xz}^k du^k + I_{yz}^k dv^k + I_{zz}^k dw^k\end{aligned} \tag{4.11}$$

where all terms involving I are now evaluated for the current known displacement (u^k, v^k, w^k). Again, for the sake of simplicity, we only show the result for u. We denote

$$(\Psi')_{D1}^k \coloneqq \Psi'\left(\left(I_d^k + I_x^k du^k + I_y^k dv^k + I_z^k dw^k\right)^2\right),$$

$$\begin{aligned}(\Psi')_{D2}^k \coloneqq \Psi'\Big(&(I_{xd}^k + I_{xx}^k du^k + I_{xy}^k dv^k + I_{xz}^k dw^k)^2 \\ &+(I_{yd}^k + I_{xy}^k du^k + I_{yy}^k dv^k + I_{yz}^k dw^k)^2 \\ &+(I_{zd}^k + I_{xz}^k du^k + I_{yz}^k dv^k + I_{zz}^k dw^k)^2\Big),\end{aligned}$$

and

$$(\Psi')_S^k \coloneqq \Psi'\left(|\nabla(u^k + du^k)|+|\nabla(v^k + dv^k)|+|\nabla(w^k + dw^k)|\right),$$

and therefore linearization yields the following optimality condition for u

$$
\begin{aligned}
0 = (\Psi')^k_{D1} I^k_x (I^k_d + I^k_x du^k + I^k_y dv^k + I^k_z dw^k) \\
+\lambda (\Psi')^k_{D2} \Big(I^k_{xx} (I^k_{xd} + I^k_{xx} du^k + I^k_{xy} dv^k + I^k_{xz} dw^k) \\
+ I^k_{xy} (I^k_{yd} + I^k_{xy} du^k + I^k_{yy} dv^k + I^k_{yz} dw^k) \\
+ I^k_{xz} (I^k_{zd} + I^k_{xz} du^k + I^k_{yz} dv^k + I^k_{zz} dw^k) \Big) \\
+\mu \mathrm{div} \Big((\Psi')^k_S \nabla (u^k + du^k) \Big).
\end{aligned}
$$

The remaining nonlinearity is now removed with a similar step: All terms dependent on u, v or w that occur inside Ψ' are equipped with an additional index l, the others with the index $l+1$. The nested fixed point iteration for u therefore finally reads

$$
\begin{aligned}
0 = (\Psi')^{k,l}_{D1} I^k_x (I^k_d + I^k_x du^{k,l+1} + I^k_y dv^{k,l+1} + I^k_z dw^{k,l+1}) \\
+\lambda (\Psi')^{k,l}_{D2} \Big(I^k_{xx} (I^k_{xd} + I^k_{xx} du^{k,l+1} + I^k_{xy} dv^{k,l+1} + I^k_{xz} dw^{k,l+1}) \\
+ I^k_{xy} (I^k_{yd} + I^k_{xy} du^{k,l+1} + I^k_{yy} dv^{k,l+1} + I^k_{yz} dw^{k,l+1}) \\
+ I^k_{xz} (I^k_{zd} + I^k_{xz} du^{k,l+1} + I^k_{yz} dv^{k,l+1} + I^k_{zz} dw^{k,l+1}) \Big) \\
+\mu \mathrm{div} \Big((\Psi')^{k,l}_S \nabla (u^k + du^{k,l+1}) \Big).
\end{aligned}
$$

At this point, all nonlinearities are removed except the one within the divergence in the very last line. These will be removed by extending Brox diffusivity to 3D, that is, we compute at a voxel position (ℓ, m, n)

$$
\begin{aligned}
&\mathrm{div} \left(\Psi_S \, \nabla (u^k + du^{k,l+1}) \right) \\
&\quad \approx \Psi^S_{\ell+1/2,m,n} \left(u^k_{\ell+1,m,n} + du^{k,l+1}_{\ell+1,m,n} - u^k_{\ell,m,n} - du^{k,l+1}_{\ell,m,n} \right) \\
&\quad - \Psi^S_{\ell-1/2,m,n} \left(u^k_{\ell,m,n} + du^{k,l+1}_{\ell,m,n} - u^k_{\ell-1,m,n} - du^{k,l+1}_{\ell-1,m,n} \right) \\
&\quad + \Psi^S_{\ell,m+1/2,n} \left(u^k_{\ell,m+1,n} + du^{k,l+1}_{\ell,m+1,n} - u^k_{\ell,m,n} - du^{k,l+1}_{\ell,m,n} \right) \\
&\quad - \Psi^S_{\ell,m-1/2,n} \left(u^k_{\ell,m,n} + du^{k,l+1}_{\ell,m,n} - u^k_{\ell,m-1,n} - du^{k,l+1}_{\ell,m-1,n} \right) \\
&\quad + \Psi^S_{\ell,m,n+1/2} \left(u^k_{\ell,m,n+1} + du^{k,l+1}_{\ell,m,n+1} - u^k_{\ell,m,n} - du^{k,l+1}_{\ell,m,n} \right) \\
&\quad - \Psi^S_{\ell,m,n-1/2} \left(u^k_{\ell,m,n} + du^{k,l+1}_{\ell,m,n} - u^k_{\ell,m,n-1} - du^{k,l+1}_{\ell,m,n-1} \right).
\end{aligned}
$$

Here, we exemplarily denote

$$\begin{aligned}\Psi^S_{\ell+1/2,m,n} =& \Bigg(\left((u+du)^{k,l}_{\ell+1,m,n} - (u+du)^{k,l}_{\ell,m,n}\right)^2 \\ &+ \left(\frac{1}{2}\left(\frac{(u+du)^{k,l}_{\ell+1,m+1,n} - (u+du)^{k,l}_{\ell+1,m-1,n}}{2}\right.\right. \\ &\qquad \left.\left. + \frac{(u+du)^{k,l}_{\ell,m+1,n} - (u+du)^{k,l}_{\ell,m-1,n}}{2}\right)\right)^2 \\ &+ \left(\frac{1}{2}\left(\frac{(u+du)^{k,l}_{\ell+1,m,n+1} - (u+du)^{k,l}_{\ell+1,m,n-1}}{2}\right.\right. \\ &\qquad \left.\left. + \frac{(u+du)^{k,l}_{\ell,m,n+1} - (u+du)^{k,l}_{\ell,m,n-1}}{2}\right)\right)^2\Bigg)^{1/2},\end{aligned}$$

where we slightly abused our notation to indicate that the quantities $(u+du)^{k,l}$ and similar ones are ready to compute at this stage of iteration. Note finally that the terms $\Psi^S_{\ell-1/2,m,n}$, $\Psi^S_{\ell,m+1/2,n}$, $\Psi^S_{\ell,m-1/2,n}$, $\Psi^S_{\ell,m,n+1/2}$ and $\Psi^S_{\ell,m,n-1/2}$ are computed similarly. For all other derivatives we use central difference schemes.

In addition, we slightly change the computation of gradients compared to the original contribution. Following [123], we use a blended version of the derivative. That is, we compute all terms involving image gradients via

$$\nabla I = (1-\beta)\nabla I_1(x+\mathbf{u}) + \beta \nabla I_0(x). \tag{4.12}$$

The choice of β is problem dependent, we decided to fix it to a value of $\beta = 0.5$. The final inner iteration is now achieved by collecting all terms $du^{k,\ell}$, $dv^{k,\ell}$ and $dw^{k,\ell}$ on one side. This results in a 3-by-3-matrix equation for each $\mathbf{u}$, which we solve by successive overrelaxation (SOR) [53].

4.3 TV-L_1 Optical Flow

Primal dual methods usually do not consider the gradient constancy assumptions as above, but aim for solving a *linearized* variational Optical Flow equation. That is, as in the previous section, the L_1 distance measure is considered. Because

$$\int_\Omega |I_0(\mathbf{x}) - I_1(\mathbf{x}+\mathbf{u})|\,\mathrm{d}\mathbf{x}$$

is neither linear nor convex, a first order Taylor approximation is performed, that is

$$I_1(\mathbf{x}+\mathbf{u}) = I_1(\mathbf{x}+\mathbf{u}_0) + \nabla I_1^T(\mathbf{x}+\mathbf{u}_0)(\mathbf{u}-\mathbf{u}_0),$$

and the minimization is performed for the functional

$$\mathcal{J}_{\text{PD}} = \int_\Omega \lambda |I_0(\mathbf{x}) - I_1(\mathbf{x}+\mathbf{u}_0) - (\nabla I_1)^T\mathbf{u} + (\nabla I_1)^T\mathbf{u}_0| + |\nabla \mathbf{u}|\,\mathrm{d}\mathbf{x}. \tag{4.13}$$

To bring the above equation into a form where the primal dual algorithms from Section 2.3.2 are easily applicable, we separate the motion estimation based on the image data from the

reconstruction of a BV solution, that is, we introduce an auxiliary variable $\mathbf{u}$ and solve

$$\min_{\mathbf{u},\mathbf{v}} \mathcal{J}_\theta(\mathbf{u},\mathbf{v}) = \int_\Omega |\nabla \mathbf{u}| + \frac{1}{2\theta}||\mathbf{u}-\mathbf{v}||_2^2 + \lambda|\rho(\mathbf{v})|\,\mathrm{d}\mathbf{x}, \tag{4.14}$$

where $\rho(\mathbf{u}) = I_0(\mathbf{x}) - I_1(\mathbf{x}+\mathbf{u}_0) - (\nabla I_1)^T\mathbf{u} + (\nabla I_1)^T\mathbf{u}_0$ for some initial but known $\mathbf{u}_0$. Equation 4.14 is then solved by alternating between the solution concerning the $\mathbf{u}$ and the $\mathbf{v}$ parts

$$\mathbf{u}^{k+1} = \min_{\mathbf{u}} \int_\Omega |\nabla \mathbf{u}| + \frac{1}{2\theta}||\mathbf{u}-\mathbf{v}^k||_2^2\,\mathrm{d}\mathbf{x} \tag{4.15}$$

$$\mathbf{v}^{k+1} = \min_{\mathbf{v}} \int_\Omega ||\mathbf{u}^{k+1}-\mathbf{v}||_2^2 + \lambda|\rho(\mathbf{v})|\,\mathrm{d}\mathbf{x}. \tag{4.16}$$

Subproblem (4.15) is exactly one of those problems, where posing the saddle point formulation and applying the primal dual method (2.55) yields a fast and efficient algorithm, whose terms are much easier to evaluate. Before applying the framework of Section 2.3.2, we need to find a finite dimensional equivalent that approximates our problem sufficiently. The discrete total variation of an approximation $\mathbf{u}_h$ to $\mathbf{u}$ as in (3.3) [30] is given by

$$TV_h(\mathbf{u}_h) = TV_h(u_h) + TV_h(v_h) + TV_h(w_h), \tag{4.17}$$

where

$$TV_h(u_h) = h^2 \sum_{i,j,k} \frac{1}{h}\sqrt{|u_{i+1,j,k}-u_{i,j,k}|^2 + |u_{i,j+1,k}-u_{i,j,k}|^2 + |u_{i,j,k+1}-u_{i,j,k}|^2}, \tag{4.18}$$

$TV_h(v_h)$ and $TV_h(w_h)$ are defined analogously. Note that this amounts to discretizing the gradient by forward differences. $\mathbf{u}_h$ is now a multidimensional array with real entries, that is $\mathbf{u}_h \in U \coloneqq \mathbb{R}^{N\times M\times L\times 3}$. The gradient ∇ is a linear map to $V = U\times U\times U$. Let $\langle\cdot,\cdot\rangle_U$ and $\langle\cdot,\cdot\rangle_V$ denote the standard Euclidean scalar products. The adjoint ∇^* of the gradient is then denoted by $-\operatorname{div}$ and for any $\mathbf{u}\in U$ and $\mathbf{p}\in V$ defined by

$$\langle \nabla\mathbf{u}, \mathbf{p}\rangle_V = \langle \mathbf{u}, \nabla^*\mathbf{p}\rangle_U = -\langle \mathbf{u}, \operatorname{div}\mathbf{p}\rangle_U. \tag{4.19}$$

Easy computations give

$$(\operatorname{div}\mathbf{p})_{i,j,k} = p^x_{i,j,k} - p^x_{i-1,j,k} + p^y_{i,j,k} - p^y_{i,j-1,k} + p^z_{i,j,k} - p^z_{i,j,k-1},$$

where $\mathbf{p} = (p^x, p^y, p^z)$. The discrete equivalent of subproblem (4.15) is then given by

$$\min_{\mathbf{u}\in U} \lambda||\nabla\mathbf{u}||_{2,1} + \frac{1}{2}||\mathbf{u}-\mathbf{v}^k||_2^2, \tag{4.20}$$

with $||\mathbf{p}||_{2,1} = \sum_{i,j,k}\sqrt{(p^x_{i,j,k})^2 + (p^y_{i,j,k})^2 + (p^z_{i,j,k})^2}$. We can now identify

$$F(\cdot) = ||\cdot||_{2,1}, \qquad K = \nabla, \qquad \text{and} \qquad g(\mathbf{u}) = \frac{1}{2\theta}||\mathbf{u}-\mathbf{v}||_2^2. \tag{4.21}$$

The primal dual iteration in this setting is

$$\begin{aligned}
\mathbf{p}^{k+1} &= \operatorname{prox}_{\sigma F^*}(\mathbf{p}^k + \sigma K\bar{\mathbf{u}}^k)\\
\mathbf{u}^{k+1} &= \operatorname{prox}_{\tau g}(\mathbf{u}^k - \tau K^*\mathbf{p}^k)\\
\bar{\mathbf{u}}^{k+1} &= \mathbf{u}^{k+1} + \theta(\mathbf{u}^{k+1}-\mathbf{u}^k).
\end{aligned}$$

The conjugate function F^* of F is the indicator function δ_P, where

$$P = \{\mathbf{p} \in Y \mid ||\mathbf{p}||_\infty \leq 1\}$$

with the discrete maximum norm $||\mathbf{p}||_\infty = \max_{i,j,k} \sqrt{(p^x_{i,j,k}) + (p^y_{i,j,k})^2 + (p^z_{i,j,k})^2}$. The proximal operator $\text{prox}_{\sigma F^*}$ was given by the resolvent $(I + \sigma \partial F^*)^{-1}$, which in this case is the pointwise projection onto L_2 balls. $\mathbf{p}$ is therefore given by

$$\mathbf{p}^{k+1} = \frac{\mathbf{p}^k + \sigma \nabla \bar{\mathbf{u}}^k}{\max(1, |\mathbf{p}^k + \sigma K \bar{\mathbf{u}}^k|)}. \tag{4.22}$$

$\text{prox}_{\tau g}$ is also given by the resolvent, which, as it is a pointwise quadratic problem, is trivially given by

$$\mathbf{u}^{k+1} = \frac{\mathbf{u}^k - \tau \operatorname{div} \mathbf{p}^k + \mathbf{v}}{1 + \tau\lambda}. \tag{4.23}$$

To solve subproblem (4.16), we first remark that there is no dependence on neighboring voxels in $\mathbf{v}$ as we do not consider any gradients. We can therefore solve the equation voxelwise. Next, we make a case distinction with respect to $\rho(\mathbf{v})$. By considering the three cases $\rho(\mathbf{v}) < 0$, $\rho(\mathbf{v}) > 0$ and $\rho(\mathbf{v}) = 0$, we can derive from first order optimality that

$$\mathbf{v}^{k+1} = \mathbf{u}^{k+1} + \begin{cases} \lambda\theta\nabla I_1 & \text{if } \rho(\mathbf{u}^{k+1}) < -\lambda\theta|\nabla I_1|^2 \\ -\lambda\theta\nabla I_1 & \text{if } \rho(\mathbf{u}^{k+1}) > \lambda\theta|\nabla I_1|^2 \\ -\rho(\mathbf{u}^{k+1})\nabla I_1/|\nabla I_1|^2 & \text{if } |\rho(\mathbf{u}^{k+1})| \leq \lambda\theta|\nabla I_1|^2. \end{cases} \tag{4.24}$$

4.4 Total Generalized Variation Optical Flow

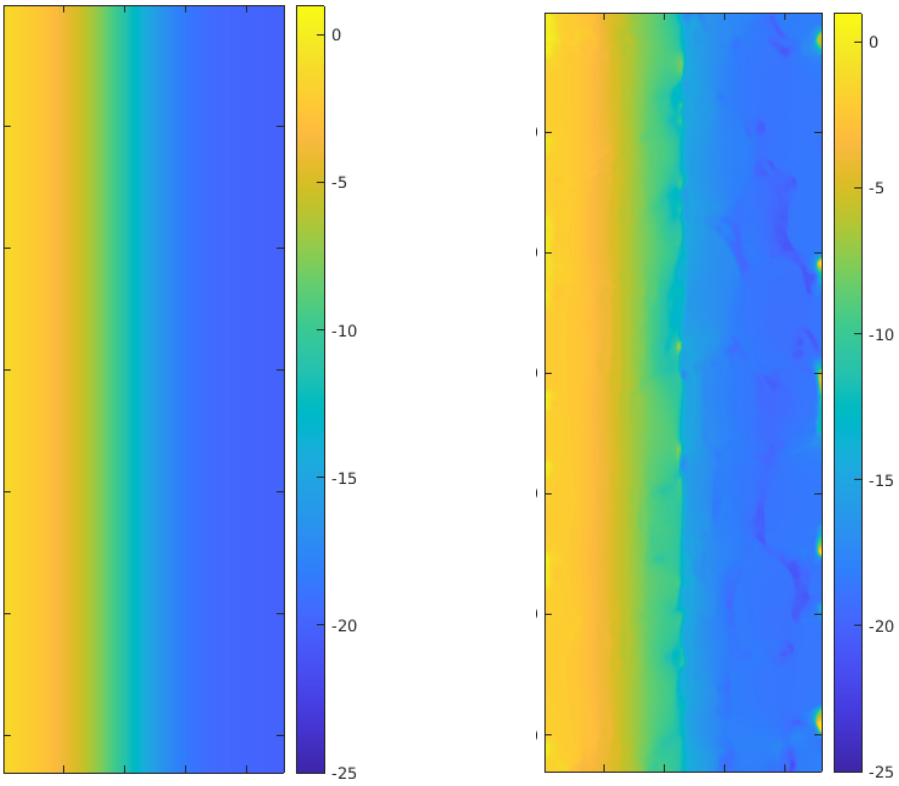
There are various analyses on the performance of Total Variation regularization. A very thorough investigation was done by Jalalzai et al. [62], who considered the denoising of 1D functions. In denoising of two dimensional images (where TV regularization actually has its roots) one flaw is very apparent: smooth variations are often approximated via piecewise constant functions. This phenomenon is also called staircasing due to the very obvious analogy. This effect can also be observed in three dimensions, and for motion estimation. In Figure 4.2, we see an artificial displacement field and its reconstructed version by TV-L_1 Optical Flow. The stairs are very evident, whereas the original varies smoothly.

We have already seen a generalization of total variation that can also tackle this problem. Total generalized variation introduced by Bredies et al. [23] and shortly recapitulated in this thesis in Section 2.4.4, can be used as regularizer via defining

$$TGV^k_\alpha(u) = \sup \Big\{ \int_\Omega u \operatorname{div}^k v dx \mid v \in C^k_c(\Omega, \text{Sym}^k(\mathbb{R}^d)), \\ ||\operatorname{div}^l v||_\infty \leq \alpha_l, l = 0, \cdots, k-1, \Big\} \tag{4.25}$$

and its vectorial equivalent $TGV^k_\alpha(\mathbf{u}) = TGV^k_\alpha(u) + TGV^k_\alpha(v) + TGV^k_\alpha(w)$. A very important order of this regularizer is $k = 2$, which reads

$$TGV^2_\alpha(u) = \min_w \left(\alpha_1 \int_\Omega |\nabla u - w| \, \mathrm{d}\mathbf{x} + \alpha_0 \int_\Omega |\nabla w| \, \mathrm{d}\mathbf{x} \right). \tag{4.26}$$



(a) Ground Truth (b) TV-L_1

Figure 4.2: Comparison of TV-L_1 flow to a ground truth. Staircasing artifacts are visible in the center.

Compared to TV-L_1 Optical flow, only subproblem (4.15) changes. The saddle-point formulation for $k = 2$, adapted to our notation in 3D, reads

$$\min_{\mathbf{u},\mathbf{w}} \max_{\mathbf{p},\mathbf{q}} \langle \nabla \mathbf{u} - \mathbf{w}, \mathbf{p} \rangle + \langle \mathscr{E}\mathbf{w}, \mathbf{q} \rangle + \frac{1}{2}||\mathbf{u} - \mathbf{v}^k||_2 - \delta_{||\cdot||_{2,\infty} \leq \alpha_1}(\mathbf{p}) - \delta_{||\cdot||_{2,\infty} \leq \alpha_0}(\mathbf{q}).$$

Here, $\mathscr{E} = (D\mathbf{u} + D^T\mathbf{u})/2$ is the strain tensor (cf Section 2.4.4).

To follow the notations from Section 2.3.2, we find

$$F^*(\mathbf{p}, \mathbf{q}) = \delta_{||\cdot||_{2,\infty} \leq \alpha_1}(\mathbf{p}) + \delta_{||\cdot||_{2,\infty} \leq \alpha_0}(\mathbf{q})$$
$$K = \begin{pmatrix} \nabla & -\mathbb{I} \\ 0 & \mathscr{E} \end{pmatrix}, \quad G(\mathbf{u}) = ||\mathbf{u} - \mathbf{v}^k||_2,$$

and the dual operators to ∇ and $\mathscr{E}$ are suitable divergence terms.

Most algorithms however to not compute iterations based on the *full* operator K, but perform separate updates for $\mathbf{u}, \mathbf{w}$ and their dual variables. Again, as F^* only consists of

indicators of norms, we denote by P_{α_1} and P_{α_0} their respective projections. Then

$$\begin{aligned}
\mathbf{p}^{k+1} &= P_{\alpha_1}(\mathbf{p}^k + \sigma(\nabla \bar{\mathbf{u}}^k - \bar{\mathbf{w}}^k)) \\
\mathbf{q}^{k+1} &= P_{\alpha_0}(\mathbf{q}^k + \sigma \mathscr{E} \bar{\mathbf{w}}^k) \\
\mathbf{u}^{k+1} &= \operatorname{prox}_{\tau G}(\mathbf{u}^k + \tau \operatorname{div} \mathbf{p}^k) \\
\mathbf{w}^{k+1} &= \mathbf{w}^k + \tau(\mathbf{p}^{k+1} + \operatorname{div} \mathbf{q}^k) \\
\bar{\mathbf{u}}^{k+1} &= \mathbf{u}^{k+1} + \theta(\mathbf{u}^{k+1} - \mathbf{u}^k)) \\
\bar{\mathbf{w}}^{k+1} &= \mathbf{w}^{k+1} + \theta(\mathbf{w}^{k+1} - \mathbf{w}^k)).
\end{aligned}$$

The exact derivation of the procedure can be found in [23] or [31].

4.5 Wavelet Optical Flow

The voxel scale resolution of displacement is not always the desired quantification of motion. Take for example [32], where the lack of performance of DVC in failed areas, so at cracks, was used to segment the crack itself. However, performing a full cycle of motion estimation, no matter if DVC, MIR or Optical Flow, just to evaluate the failure of the method does not seem efficient. It would therefore be desirable to develop a method that reliably estimates motion on low scales in the coarse-to-fine framework. Due to the downscaling by Gaussian filters however, one is very likely to smoothen out cracks in an early stage, especially at the beginning of the image pyramid. We therefore propose to replace the coarse-to-fine scheme by Gaussian image pyramids in Optical Flow by wavelets. In fact, wavelets based on lifting steps seem to be extraordinarily reasonable in this case.

The short introduction into formulating wavelet transforms by lifting follows the work by its inventor Wim Sweldens [114].

4.5.1 Lifted Wavelets

Wavelets, as the name indicates, are families of wave-like functions. They are usually used to split data into different frequency components (so scales), which enables a distributed investigation of these scales. Most appealing about wavelets in this context is the fact, that they – opposite to Fourier transformations for example – are local not only in their spatial domain, but also approximately local in their frequency domain. Here, *approximately* means up to small variance. Wavelets are formed via dyadic translates of one so called *mother wavelet*, that is, all members of the family $\phi_{j,m}$ are computed via

$$\phi_{j,m} = 2^{-j/2}\phi(2^{-j} - m).$$

When one carefully chooses ϕ, one can construct an orthonormal basis of L_2 with the family $\phi_{j,m}$. A very old but well-known example are the *Haar-wavelets*, whose mother wavelet is given by

$$\phi(x) = \begin{cases} 1, & 0 \le x < \frac{1}{2}, \\ -1, & \frac{1}{2} \le x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.27}$$

One consequence of $\phi_{j,m}$ being an orthonormal basis of L_2 is, that every function in $f \in L_2$ can be approximated by a finite linear combination of $\phi_{j,m}$ up to an arbitrarily small precision. We have

$$f = \sum_{j,m} \gamma_{j,m} \phi_{j,m}, \tag{4.28}$$

where $\gamma_{j,m}$ are the *wavelet coefficients*.

This finding is even a sufficient condition, so when one wants to prove that an orthonormal family $\phi_{j,m}$ forms an orthonormal basis, one proves that every function can be approximated this way [36]. In such a proof, one successively approximates f over coarser and coarser intervals. The difference between two resolution steps can then be written as a linear combination of the $\phi_{j,m}$'s. This immediately hints, why decomposition by wavelets can be seen as a multiresolution analysis. One deduces nested spaces $(V_j)_{j\in\mathbb{Z}}$, which satisfy the following properties

1. $\cdots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \cdots$,
2. $\bigcap_{j\in\mathbb{Z}} V_j = \{0\}, \overline{\bigcup_{j\in\mathbb{Z}} V_j} = L_2(\mathbb{R})$,
3. $f \in V_j \leftrightarrow f(2^j \cdot) \in V_0$,
4. $f \in V_0 \rightarrow f(\cdot - n) \in V_0$ for all $n \in \mathbb{Z}$.

Multiresolution in this context means that all V_j are scaled versions of each other. In the Haar case one for example has

$$\text{proj}_{V_{j-1}} = \text{proj}_{V_j} + \sum_{k\in\mathbb{Z}} \langle f, \phi_{j,k} \rangle \phi_{j,k}.$$

We now collect the favorable properties that we just learned wavelets offer.

1. Wavelets form an orthonormal basis (even more general, a Riesz-basis).
2. One can explicitly calculate the coefficients of the basis expansion.
3. Wavelets are approximately local in space and frequency.
4. Wavelets can be used for multiresolution analysis.

Within the "Wavelet-Boom" between the late 1980s and early 2000s, many communities have proposed improvements and concepts in this area. A rather late one is the concept of *Lifted Wavelets*, that transfers the favorable properties of wavelets into a pure space domain decomposition. The Haar-wavelet in equation (4.27) for example also has a pure space formulation: Instead of continuous f take its discretized version $f[k]$ sampled at a finite number of $k \in \mathbb{Z}$ points. (That is of course not a restriction in any real-world application.) The approximation of f in the next coarser space f_{-1} is then

$$f_{-1}[k] = \frac{f[2k] - f[2k+1]}{2},$$

and the wavelet coefficient is given by $\gamma_{-1,k} = f[2k+1] - f[2k]$.

The idea of such lifted wavelets is, that correlation of structures is already apparent in the space domain and a transfer to the frequency domain might therefore not be necessary.

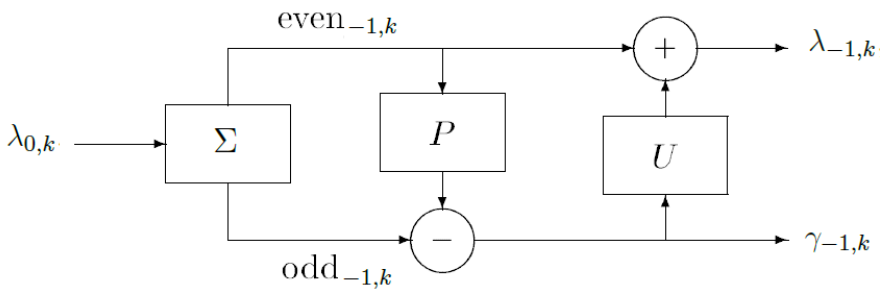


Figure 4.3: Schematic lifting scheme. The scheme is traversed from the left.

The scheme is best described by considering an abstract example. Once more consider a signal f, which we know at a fixed sampling distance, e.g. $d = 1$. This "initial" or "original" scale will be denoted by $\lambda_{0,k} = f(k)$, $k \in \mathbb{Z}$. An approximation in the next coarser space can now be to just consider the even samples, that it

$$\lambda_{-1,k} \coloneqq \lambda_{0,2k} \quad \text{for} \quad k \in \mathbb{Z}. \tag{4.29}$$

Ideally, the loss of information in such a downscaling step should be small. That means, that the difference between $\lambda_{-1,k}$ and $\lambda_{0,k}$ is ideally kept low. The coefficients, that will encode the difference, will be denoted by $\{\gamma_{-1,k}\}$. In fact, these are, as in the classical approach to wavelets, the *wavelet coefficients*.

Now, a very obvious choice is to just put the odd coefficients as wavelet coefficients, that is $\gamma_{-1,k} = \lambda_{0,2k+1}$, $k \in \mathbb{Z}$. This is also called the *lazy wavelet*. However, this will not fulfill our requirement to produce small wavelet coefficients, even worse: the signal will be roughly the same as the approximation. We need a more advanced scheme to do so.

We for now keep with the separation of even and odd samples. But now we *predict* the odd samples $\{\lambda_{0,2k+1}\}$ based on $\{\lambda_{-1,k}\}$ by computing the average of two (even) neighbors. The wavelet coefficient then reads

$$\gamma_{-1,k} \coloneqq \lambda_{0,2k+1} - \frac{1}{2}\left(\lambda_{-1,k} + \lambda_{-1,k+1}\right).$$

If the signal we encode contains correlated structures (which it does, as it will be an image), these coefficients are small. However, proceeding with an iteration like that will cause aliasing of the approximation at the coarsest level. This can be prevented by requiring that the coefficients $\lambda_{j,k}$ that can be found on all levels are roughly the same, i.e.

$$\sum_k \lambda_{-1,k} = \frac{1}{2}\sum_k \lambda_{0,k}. \tag{4.30}$$

This can be achieved by performing a similar step as before – but this time we use the neighboring *wavelet* coefficients and *lift* the $\lambda_{-1,k}$:

$$\lambda_{-1,k} = \lambda_{-1,k} + \frac{1}{4}\left(\gamma_{-1,k-1} + \gamma_{-1,k}\right).$$

Let us now formalize this procedure. The *lifting scheme* consists of the following steps.

1. Split the signal into subbands by invertible transformation Σ. For the sake of simplicity, we use 2 subbands. We have

$$(\text{even}_{-1,k}, \text{odd}_{-1,k}) = \Sigma(\lambda_{0,k}).$$

2. Predict the wavelet coefficient (also called *detail signal*) by $\text{odd}_{-1,k}$. We have

$$\gamma_{-1,k} = \text{odd}_{-1,k} - P(\text{even}_{-1,k}),$$

   where P is an (invertible) prediction operator.

3. Update the signal approximation $\lambda_{-1,k}$ by the previously predicted wavelet coefficients

$$\lambda_{-1,k} = \text{even}_{-1,k} + U(\gamma_{-1,k}),$$

   with update operator U.

The scheme is also depicted in Figure 4.3. The signal enters the scheme from the left. $\lambda_{0,k}$ is split by Σ, indicated by two departing lines from the corresponding box. $\text{even}_{-1,k}$ then traverses P and is joined with $\text{odd}_{-1,k}$. Here the result is already $\gamma_{-1,k}$. The detail signal passes through the lower line to be eventually transformed by U and assembled with $\text{even}_{-1,k}$, which then results in the final, downscaled signal $\lambda_{-1,k}$.

4.5.2 Min/Max Lifting on the Quincunx Grid

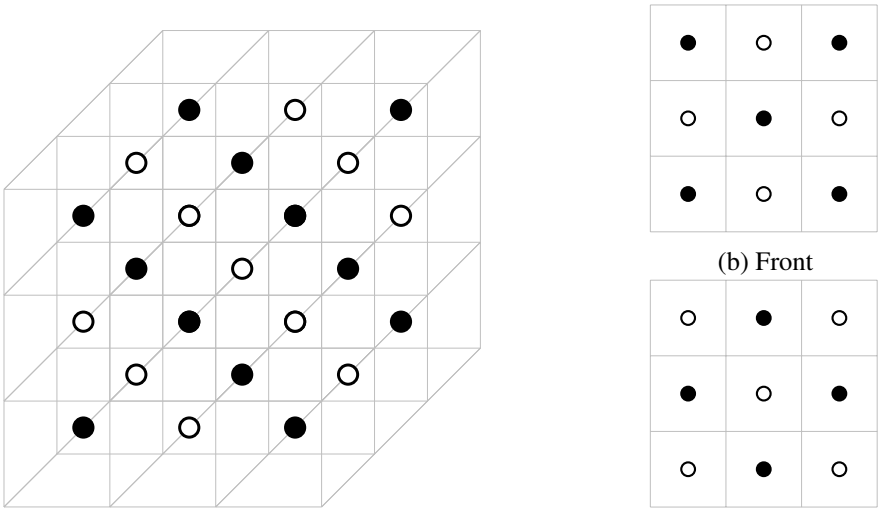
Until now we did not pay much attention to the generalization of the splitting operator Σ to higher dimensions. However, images are regarded as two or even three dimensional signals. Fortunately, there exists a quite straight forward extension to two and three dimensions, namely by the quincunx scheme, see Figure 4.4 (b) for 2D and Figure 4.4 (a) for three dimensions. With each pixel or voxel we now associate a black or a white dot. "Splitting into even and odd" will then mean to split into black and white points. Classically, the quincunx is the arrangement of black points in Figure 4.4 (b). By associating pixels or voxels with their points, we can again establish a new grid, in which the nodes are identified with these points. We will denote this grid by *image grid* or *Cartesian grid*. Further, we now denote by S the set of all original points, by Q the set of black points, and by $R = S \backslash Q$ the set of white points. We now define an adjacency relation to identify neighbors s' of s, i.e. $s \sim s'$ if $||s - s'||_1 = 1$. $\sim$ is a symmetric relation on $S \times S$. Note that only either s or s' can be a member of Q if $s \sim s'$, but not both at the same time. In a first step, we therefore have $\lambda_{-1,k}(q) = \lambda_{0,k}(q)$ and $\gamma_{-1,k}(r) = \lambda_{0,k}(r)$ for $q \in Q$ and $r \in R$. We now define specific prediction and update operators by

$$P(\lambda)(r) = \max_{q:q\sim r} \lambda(q) \tag{4.31}$$

and

$$U(\gamma)(q) = \max\{0, \max_{r:r\sim q} \gamma(r)\}. \tag{4.32}$$

This scheme is known as Max-Lifting [55], but the maximum can also be exchanged by a minimum. The use of Max- or Min-Lifting can be directly translated into the morphological operators dilation or erosion, respectively. Schemes that use this kind of prediction and update operators are therefore also called Morphological Wavelets.



(a) 3D Quincunx

(b) Front

(c) Back

Figure 4.4: Separation of an image grid into quincunx. (a) shows the separation in 3D. Subfigures (b) and (c) show representation by two layers, where (b) coincides with a two dimensional scheme. White and black voxels are split into two signals.

If we closely observe the calculations (4.31) and (4.32), we see that the calculations can and should be performed inplace. As a consequence, when starting calculations on a Cartesian (image) grid *in two dimensions*, we do not end up with such a grid again, but by a 45 degree tilted version (or with diagonal instead of horizontal and vertical neighboring relations), see Figure 4.4 (b). If we now want to perform a prediction and an update step on the tilted version, we can use the exact same neighboring relations (horizontal and vertical neighbors) to apply the wavelet transform. This is even mathematically justified: The dilation matrix, that transforms a quincunx sublattice into a Cartesian lattice in two dimensions is a similarity matrix. Unfortunately, such a scheme which

- allows for a 2-channel design (split into even and odd, black and white),
- generates a Cartesian lattice again after d iterations,
- *and* is a similarity transform,

does not exist in any dimension larger than $d = 2$. However, the third point is quite a crucial one: It guarantees that we can "reuse" neighboring relations we stated once. The consequences for 3D are exactly that: We cannot find a scheme, that produces a Cartesian grid after 3 iterations *and* can be split into two distinct channels. We need to borrow one of the steps from a different wavelet transform. An intuition on why we fail to find a scheme that is as similar to in the 2D case, can be given if we consider the Voronoi cells of the resulting 3D quincunx grid.

Voronoi cells can be used to state an equivalent notion of neighborhood between points: Two points are neighbors if their Voronoi cells share exactly one face (one edge in 2D). Let us now consider the stepwise outcome of a wavelet transform on a quincunx grid. The first step, so splitting into black and white balls works fine in 2D and in 3D. We remain with two distinct sets from which we can predict and update. Performing prediction and update

then yields an approximation on the positions of the black points. In 2D, we now just rotate the black points by 45 degree, and once more we can use the previous neighboring relation, split the remaining points into two distinct sets, perform prediction and update steps on those and arrive eventually on a Cartesian grid with $\frac{1}{2}$ of the number of pixels in each coordinate direction, so an image whose size is $\frac{1}{4}$ of the original.

In 3D, the first step works similarly. But now considering the remaining points, we immediately unveil the issue in 3D. The Voronoi cells of the 3D quincunx grid have the shape of a rhombic dodecahedron, a twelve-faced polygon with rhombic faces, which means that every point has twelve neighbors. A similarity transform however requires the same amount of neighbors as in the Cartesian lattice, which are six. Even worse, the amount of twelve neighbors prevents us from dividing the remaining points into two distinct subsets, from which we can predict and update. We therefore borrow the idea of 5/3-LeGall filters [68], which will not be introduced in detail here.

The resulting Voronoi cell is now a cuboid, with two equal sides and the remaining one twice as long. The cell is not aligned with Cartesian coordinates. Clearly we loose isotropy at this stage. It may therefore be worth checking if the cuboidal cell should be aligned problem dependent to achieve better performance in whatever the wavelets are used for. This study will however not be part of this thesis.

4.5.3 Efficient Implementation of Morphological Wavelets

In the previous section we saw that a generalization of splitting a one dimensional signal into even and odd parts is to split a multidimensional signal based on the quincunx grid. This can also be seen when we transform the multidimensional index to a linear one, i.e. a signal sampled at $n \times m$ or $n \times m \times l$ points, is transformed to a one dimensional signal by mapping

$$(i, j) \mapsto i + j \cdot n, \tag{4.33}$$

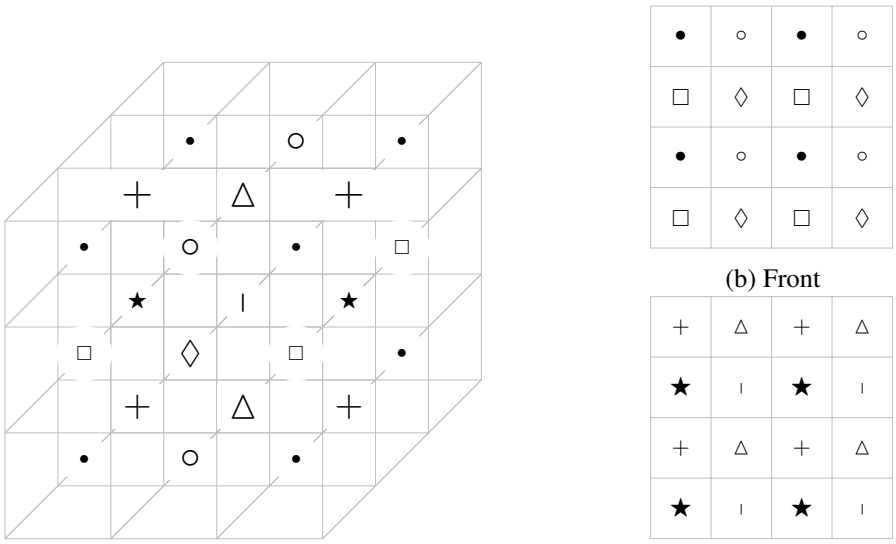
or

$$(i, j, k) \mapsto i + (j + k \cdot n) \cdot m, \tag{4.34}$$

respectively. Then, the quincunx grid corresponds to the even and odd points of the transformed signals. However, transformation to 1D or modulo-checking for being even or odd are rather costly operations. We therefore extend the efficient implementation approach of [37] to three dimensions.

For implementation, we do not split the lattice into two subsets, but eight, as in Figure 4.5. The symbols $\bullet, \lozenge, \triangle, \bigstar, \circ, \square, +, \imath$ are used to demonstrate the belonging of each voxel. Instead of modulo operations, this split can be efficiently performed by taking every second point in each dimension and only varying the starting points (either first or second point) in each dimension. We will call the application of the Morphological Wavelet Approximation to a Cartesian image grid *Horizontal/Vertical Lifting* or *hv*-Lifting. We start by splitting the image into $(\bullet, \lozenge, \triangle, \bigstar)$ and $(\circ, \square, +, \imath)$. In the prediction step neighboring relations will be used as in Figure 4.6. Here, the center character always depicts the point that is predicted and the lines indicate which neighbors are used for prediction as in (4.31). The neighbors for the update step are found in Figure 4.7.

We will call the transition from the dodecahedral grid to the cuboidal grid *Diagonal Lifting 1* or *d1-Lifting*. The points are now partitioned into $(\bullet, \bigstar)$ and $(\lozenge, \triangle)$. Predict and update neighboring relations can be found in Figure 4.8 and Figure 4.9, respectively.

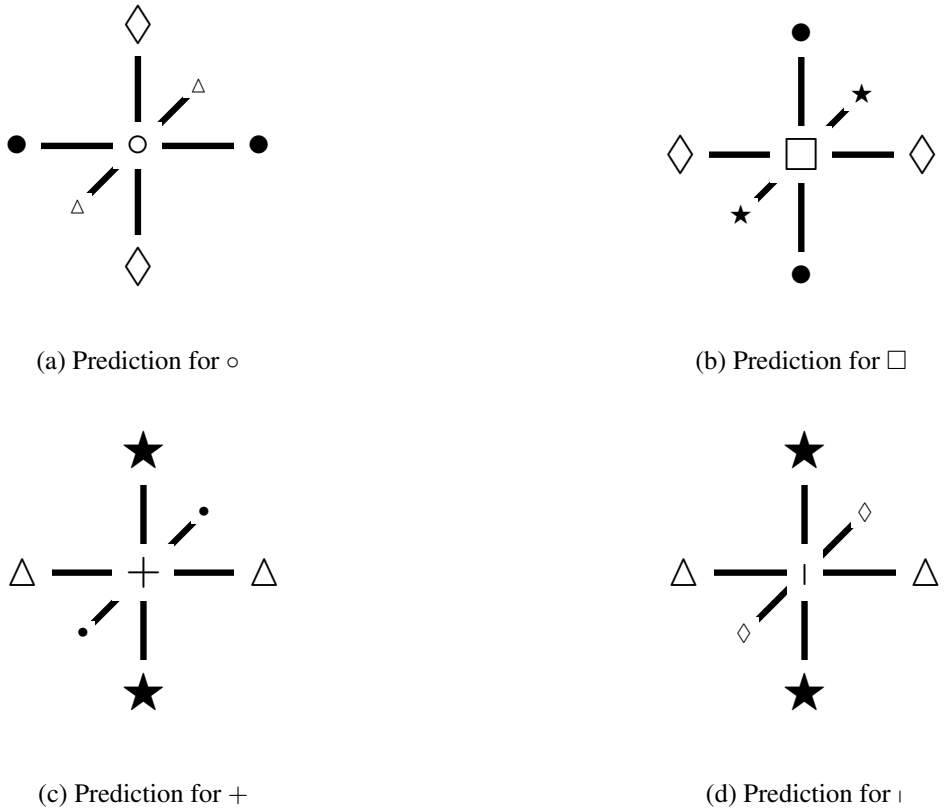


(a) 3D Quincunx with symbols

(b) Front

(c) Back

Figure 4.5: Separation of an image grid into quincunx. (a) shows the separation in 3D. Subfigures (b) and (c) show representation as two layers, where (b) coincides with a two dimensional scheme.



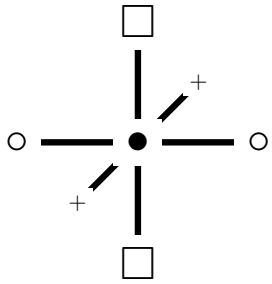
(a) Prediction for ○

(b) Prediction for □

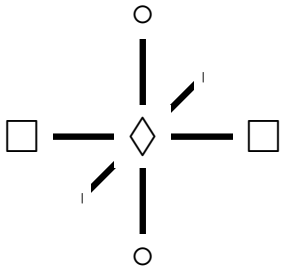
(c) Prediction for +

(d) Prediction for ı

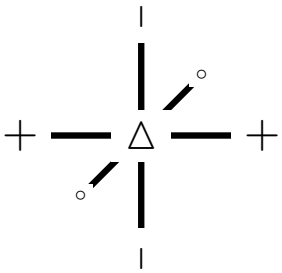
Figure 4.6: Horizontal/Vertical Lifting prediction step. $(\circ, \square, +, \imath)$ are predicted by their neighbors as indicated in the Subfigures (a), (b), (c) and (d), respectively.



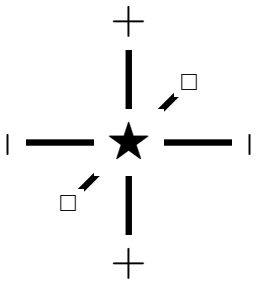
(a) Prediction for •



(b) Prediction for ◊

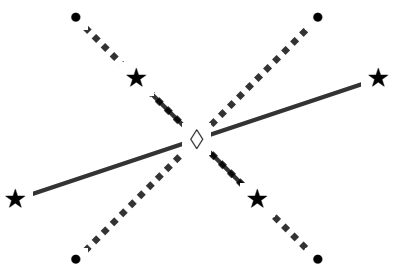


(c) Prediction for △

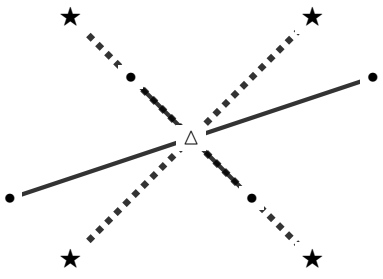


(d) Prediction for ★

Figure 4.7: Horizontal/Vertical Lifting update step. (•, ◊, △, ★) are updated by their neighbors as indicated in the Subfigures (a), (b), (c) and (d), respectively.

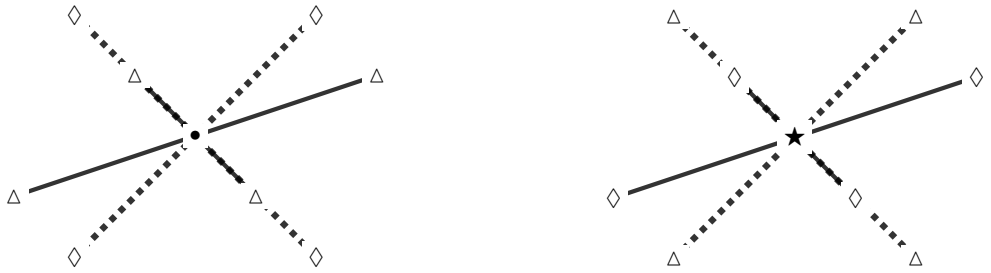


(a) Prediction for ◊



(b) Prediction for △

Figure 4.8: Diagonal Lifting I prediction step. (◊, △) are predicted by their neighbors as indicated in the Subfigures (a) and (b), respectively.



(a) Update for •

(b) Update for ★

Figure 4.9: Diagonal Lifting I prediction step. $(\bullet, \bigstar)$ are predicted by their neighbors as indicated in the Subfigures (a) and (b), respectively.

(a) Prediction for ★

(b) Update for •

Figure 4.10: Diagonal Lifting II prediction and update step. ★ are predicted by their neighbors as indicated in the Subfigure (a) and • is updated by neighbors as in (b), respectively.

Application of the wavelet transform to the cuboidal grid is denoted by *Diagonal Lifting 2* or *d2-Lifting*. As only two sets are left the split is clear, the neighboring relations are however depicted in Figure 4.10.

4.5.4 Derivatives on the Lifted Grids

In the previous subsection we learned that only in every third step we end up with a Cartesian grid in the wavelet transform. The other two steps produce grids that consist of cells in shapes of dodecahedra and tilted cuboids, respectively. However, if we want to perform motion estimation via Optical Flow on all scales, we need to estimate derivatives also on these special grids. We will use the divergence theorem to do so. In what follows, we strongly follow the notations of [115] for unstructured grids, which clearly have our structured ones as special case. A cell of a grid will be denoted by P, with Ω_P its volume and S_P its bounding surface, respectively. Then, for continuously differentiable Φ we have

$$\int_{\Omega_p} \nabla\Phi \, \mathrm{d}\mathbf{x} = \int_{S_P} \Phi\vec{n} \, \mathrm{d}s.$$

Our bounding surface S_P consists of F planar faces S_f, and each face has a constant unit normal, therefore

$$\int_{\Omega_p} \nabla\Phi \,\mathrm{d}\mathbf{x} = \sum_{f=1}^{F} \left(n_f \int_{S_f} \Phi \,\mathrm{d}s \right). \tag{4.35}$$

Now apply the midpoint rule: It states that the mean value of a quantity over a cell P (or face f) is equal to its value at centroid $\mathbf{P}$ of the cell (or c_f of the face) plus a second order correction. We get

$$\begin{aligned} \frac{1}{|\Omega_P|} \int_{\Omega_P} \nabla\Phi \,\mathrm{d}\mathbf{x} &= \nabla\phi(\mathbf{P}) + O(h^2) \\ \frac{1}{|S_f|} \int_{|S_f|} \Phi \,\mathrm{d}s &= \Phi(c_f) + O(h^2), \end{aligned}$$

where h is the characteristic grid spacing. This yields

$$\begin{aligned} \int_{\Omega} \nabla\Phi \,\mathrm{d}\mathbf{x} &= \nabla\phi(\mathbf{P})|\Omega_P| + O(h^4) \\ \int_{S_f} \Phi \,\mathrm{d}s &= \Phi(c_f)|S_f| + O(h^3). \end{aligned}$$

Plugging this into (4.35), we get

$$\nabla\Phi(\mathbf{P}) = \frac{1}{|\Omega|} \sum_{f=1}^{F} \Phi(c_f)|S_f| n_f + O(h). \tag{4.36}$$

Normally, one would now need to bother about evaluation of Φ at cell and face centers. But keep in mind that we calculate derivatives on voxel grid, so Φ is just constant over the cell and the face.

Having now derived an ingredient to compute gradients on arbitrary grids, we can apply the TV Optical Flow variant of Section 4.3 to the quincunx grids. We just replace all differences by their corresponding components in the gradient, computed at every grid position as in equation (4.36).

4.6 Extensions to Volume Images

We want to conclude this chapter by a small review on algorithms, that can be counted as extensions of Optical Flow to 3D from the literature. The first one to mention here is the work by Hermann et al. [57], which uses a similar procedure as in Section 4.3, but with a different similarity measure. They propose to compare the census transforms of both images. The census transform of an image assigns a binary vector to each image position, which encodes features at that position. A similarity measure based on this transform then only measures the activated bits in the vector. In equation (4.14), this only influences ρ, therefore only the optimization of (4.16) changes. The binary vector, denoted by C at each voxel position is generated by considering the set of all neighboring voxels $N_{\mathbf{x}}$ of $\mathbf{x}$ and setting

$$C(x) = \{\chi(I(\mathbf{x}) \geq I(\mathbf{y}))\}_{\mathbf{y}\in N_{\mathbf{x}}},$$

where χ returns 1 if the condition is fulfilled and 0 otherwise. ρ is then given by the Hamming distance of the census transforms C_0 and C_1 of both images, i.e.

$$\rho(\mathbf{x}, \mathbf{u}) = \Gamma(C_0(\mathbf{x}) \oplus C_1(\mathbf{x} + \mathbf{u})),$$

where $\oplus$ concatenates both bitstrings and Γ counts the positive bits. The solution of (4.16) requires the computation of gradients of ρ. Unfortunately the authors did not describe how they proceeded here. In 2020, Manandhar et al. [75] proposed to use the exact same setup to estimate motion in 3D fluorescence images. However, when performing the optimization, they use a continuous approximation to the census transform. This approximation can then easily be incorporated into classical optimization procedures.

The same authors describe a very similar extension in [56]. They now use the algorithm of Subsection 4.2. Again, they replace the L_1 similarity measure by a census based approach, but keep the distance measure based on the image gradients. Interestingly, the use of census transforms was also investigated in [49] by one of the authors of [25]. There, the census transform does not replace the data term but is directly related to the gradient constancy by its continuous limit. Moreover, the census transform is shown to perform better than the classical gradient constancy assumption only in case of strong illumination changes. It can therefore be considered as a *more robust* version. The price for this increased robustness however is the loss of information. The census cost transform is a binary measure, and therefore loses parts of already sparse information of our data. These two reasons, the loss of information and the fact that CT data usually does not suffer from strong illumination changes caused us to consider the 3D extension of the original work, consisting of the constancy assumption, the gradient constancy assumption and the smoothness constraint.

Another extension that can roughly be counted as Optical Flow extension are two recent works by Nie et al. [89, 88]. Both methods take up the findings of Temam, which we presented in Section 2.4, i.e that the space of functions of bounded variation should be replaced by bounded deformation to produce mechanically admissible solutions in case of plastic deformations. Note however that they apply their algorithm to estimate motion in images of lungs during free breathing, where the benefits of considering nonsmooth displacements may not pay off the loss of invertibility of the transformation.

In their first contribution, Nie et al. [89] propose to use the L_2-norm similarity measure together with a L_1-penalty on the strain, that is, they minimize

$$E(\mathbf{u}) = ||I_0(\mathbf{x}) - I_1(\mathbf{x} + \mathbf{u})||_2 + \lambda |\mathscr{E}(\mathbf{u})|.$$

They follow a standard procedure: They compute Euler-Lagrange-equations and then derive a gradient descent scheme. However, such approaches often only produce valid results in terms of minimal residuals, if the image is transported within the optimization process. This means that instead of computing a residual based on the displacement that is computed during the whole optimization process, one uses the outcome of the last warping stage. The resulting displacement fields are often rather sparse and therefore cannot be a reliable mean for material testing. This is also visible in the contribution of Nie et al. [89], where the quiver plots of the displacement fields seem to consist of only few large estimates, whereas the remaining vectors seem to have small length. But we already saw that total variation especially helps to estimate piecewise rigid vector fields, if optimized correctly.

The subsequent contribution [88], generalizes bounded deformation to bounded generalized deformation, in the same manner as Bredies et al. did for total variation [23]. The

optimization this time is applied with the means of convex optimization (Section 2.3.2). However, the careful reader will note that we stated, based on the work of Bredies [23], that bounded deformation is already covered by $k = 2$ when considering arbitrary orders of total generalized variation. This shows also in the optimization process: The authors of [88] say that the only difference between their work and the seminal work of Bredies et al. lies in the computation of the projection operators P_{α_1} and P_{α_0}, as they are not computed componentwise but for whole vector fields. This basically nails down to divide by a norm involving all components compared to dividing only by some of the components.

Unfortunately, there is no implementation available to the public of the TGV Optical Flow implementation of Chambolle et al. [31]. This algorithm has been proposed by the same group, that proposed TGV as a regularizer in the general setting. Our impression however is that a stabilization as proposed by Nie et al., so dividing by the norm, is necessary anyway, when TGV is used as a regularizer. It is therefore in fact difficult to assess how much bounded generalized deformation really deviates from total generalized variation.

Chapter 5

In Situ Testing

Microtomography, so the imaging of materials with computed tomography at scales of micrometers, has been around almost as long as medical computed tomography. But still in 1999, Stock [112] did not give X-ray tomography much chance to overtake classical 2D radiography as imaging modality of choice in non-destructive material tests due to "the high cost of instrumentation." However, 15 years later, Maire et al. [74] manifested the increasing success in using 3D images of materials for analyses by a thorough review on the findings and advantages that come with microtomography. Numerous research facilities all over the world have formed and overcome the financial issues by offering their setups for industry corporations.

It is therefore no surprise that today's main interest lies in the investigation of highly advanced materials. In a world that becomes more and more demanding, materials that deal with resources sustainably have to be developed and analyzed. In situ tests play an important role, as they allow to monitor behavior under load of such new materials and without this knowledge no one can justify the use in buildings, cars or planes.

Nevertheless, there are limits in incorporating mechanical tests into a CT device. Let us shortly recall the process of generating images by CT. The basis of CT are the findings by Röntgen, who discovered x-radiation, whose attenuation of X-rays is given by the Beer-Lambert law

$$I/I_0 = \exp(-\mu d), \tag{5.1}$$

where I_0 is the initial intensity at the X-ray source, I the intensity measured at a detector after traversing through the object, μ the attenuation coefficient and d the material thickness. Infinitesimally, the loss of intensity dI along a short path ds is given by

$$dI = -\int \mu(s) ds. \tag{5.2}$$

By taking the limit, we arrive at the linear differential equation $I' = -\mu I$. Integration yields

$$-\log \frac{I_0}{I} = \int \mu(s) ds. \tag{5.3}$$

In CT, we now rotate source and detector of the X-rays around the object and collect slice-wise all line integrals of μ of the object. The famous Radon transform is now exactly performing this step. For a specific line parametrized by $r > 0$ and a two dimensional unit vector $\phi \in \mathbb{S}^1$, it computes the line integral over that line

$$(R\mu)(r, \phi) \coloneqq \int_{\mathbb{R}} \mu(r\phi + s\phi^{\perp}) \, \mathrm{d}s = -log \frac{I_0}{I}.$$

Note that the quantity on the very right can be measured. Speaking of measurement, the CT acquisition process now consists of parameterizing ϕ and r by discrete values. In fact, one can prove that if we do this by infinitely many values, we can reconstruct the original object from μ perfectly. In a more realistic setting, the rule of thumb on how many projections N are needed to reconstruct an object of effective diameter d at a resolution r is $N > \pi d/r$, which results in the classical number of projections of around 1000. Note that the above framework does not allow for motion. The object under investigation has to stand still during the whole acquisition process. This is for sure a problem in medical imaging, but also for performing in situ tests. As we are intrinsically interested *in* the motion from such a test, using an imaging modality not allowing for motion at all does not seem plausible.

There are two directions in which a remedy can be found. One is the very recent trend to dynamic reconstruction operators [50, 44]. These have hardly been introduced and are a massive open research topic. The second one is to perform quasi static or interrupted testing. That is, mechanical engineers estimate the important stages of loading of a material by small ex situ experiments, for example elastic regime, plastic regime and failure. During the CT scan, the load is then applied until a certain level is reached and kept constant during the acquisition.

It is therefore essential, that the experiment can be interrupted: This is for example the case in compression, bending and tensile tests. Note also that the interrupted state has to be kept constant for acquisition times around an hour.

In situ tests are documented for a large variety of materials. We will now focus on three special kinds, namely materials exhibiting foam like structures, fiber systems, and concrete. The current state-of-the-art in microtomography and in situ testing of these materials will be presented next.

One example of the subsequent use of in situ tests is by Betts et al. [17]. The authors performed in situ tests on single struts of an open-cell metal foam and afterwards used their results to perform finite element simulations on full foams. Marter et al. [79] performed a study, in which by the Virtual Fields Method [46] material parameters were estimated. The parameters were compared with the real values, and with competitive methods to measure them, such as crosshead displacement and optical measurements. The estimates by DVC turned out to be very reliable, where crosshead measurements underestimated the Youngs modulus (which can be interpreted as the materials stiffness) significantly.

5.1 Foams

Foams are a lightweight alternative to their solid counterpart. But the reduced demand in resource is not the only advantage that can be found when considering cellular materials. Metal foams for example experience growing popularity in the automotive industry because of their favorable stiffness and energy absorption properties [65].

In this contribution, we only consider compressive experiments of foams. This conforms with their general application: Compared to specimens from bulk material, which behave more or less the same no matter if compression or tension is applied, foams fail to bear high tension. Under compression, one can observe that after an initial regime of linear elastic behavior, they exhibit a *plateau stress*. This means that beyond the elastic limit, foams fail at nearly constant stress. Failure in foams is manifold and dependent on the underlying bulk material. Struts in ceramic and glass foams fracture very suddenly under load, elastomeric foams buckle, and metal foams yield [6]. In the plateau region, the transition from linear to

plastic behavior is usually initiated by the failure of a single strut. This causes a weakened cell within the foam, which corrupts also the stability of neighboring cells. These are then very likely to collapse next, and this propagates to a full level. This proceeds until all cells densify to a near bulk material.

Note that though foams undergo initial linear elastic behavior, they may exhibit plasticity already at this stage [65].

The first work that describes in situ CT tests on cellular structures is in fact the initial work on DVC by Bay et al. [13]. Point of interest of this work and a succeeding one [111] was to assess the heterogenity of trabecular bone, which is in fact a porous, foam like structure, and correlate it to the performance during an in situ test. In 2007, Roux et al. [104] investigated a polypropylene solid foam. Polypropylene can be considered as foam in foam, as it consists of porous beads, where each bead itself again exhibits a foam-like structure. The analysis of such complex materials highly benefits from a *local* strain analysis. Therefore, the objective of Roux et al. was not only to compute local strain at the scale of the porous beads but also *inside* the beaded structures. The method of choice was a global DVC to allow for a straightforward comparison to Finite Element simulations afterwards. However, the data posed challenges exceeding the limits of DVC. The thickness of the struts of the foam was only few voxels. It is therefore very difficult to map the local behavior, even with the authors' choices of rather small subvolumes with 8 or 16 voxels in each direction. This was also manifested by the fact that the authors could not achieve convergence with their approach at a higher compression rate, though visually many cells seemed to be intact and only a small band of densification may have caused trouble in our impression. Nevertheless, the authors were able to detect the desired shear bands at the lower compression stage.

Jirousek in 2011 [63] also investigated trabecular bone, but this time by NCC local DVC. Bouterf et al. [20] examined nail pull tests in plasterboard. With the help of in situ synchrotron images, one could see that plasterboard tends to fail very suddenly and after a short period of compaction. Bouterf et al. also investigated indendation tests in foamed gypsum [21]. The high uncertainty of displacement and strain motivated the authors to develop a reduced but more reliable kinematic basis for the displacement.

Yet many following contributions performing in situ tests lack of an algorithmic exploration of the motion a sample underwent during the test.

Adrien et al. [1] investigated compression tests of syntactic foams with Synchrotron X-Ray tomography. Syntactic foams consist of hollow glas spheres embedded into a polymer matrix and are used for thermal insulation of pipelines. Only by the opportunity to investigate the interior of the specimens, it was possible to observe that a soft matrix spreads load equally among the spheres and eventually the weakest, in this case the largest, break first. In 2011, Berek et al. [15] investigated Metal Matrix Composite Foams in in situ tests. As it was a fairly new material at that time, the aim of the study was to show agreement with literature: MMC foams first deform elastically before they enter a plastic regime. Failure is eventually manifested in densification. Note that we will use exactly this time series later in our evaluation chapter. Ballaschk et al. [10] used MMC cruciform structures and investigated the highly non-uniform behavior of these. Wang et al. [120] were able to show different behavior of different regimes in pomelo peel compression experiments. Hubalkova et al. [60] performed compression tests on glass and ceramic foams. These foams tend to fracture very suddenly, which makes their assessment by motion estimation a very difficult task. Note again that though the findings in the publication by Hubalkova et al. were not supported by DVC or other algorithms, we catch up on this matter in the next chapter.

5.2 Fibers

Fibers or fibrous structures usually appear in two variants. The first one is by fiber reinforced composites (FRC), that is a material consists of a matrix, usually polymer, ceramic or metal, which is reinforced by fibers. Especially for polymer matrix such materials are very attractive as they combine a lightweight material with the strength of stiff fibers [108]. In addition, the production process (often by injection molding) allows for a very flexible design of shapes. Yet a very natural question in this case is how fibers influence the stability and failure mechanics of specimens. For FRC, the failure can be devided into four stages [102]:

1. Damage at fiber ends
2. Fiber failure
3. Debonding of fibers from matrix
4. Damage growth in the matrix.

The first three stages can usually only be observed in sychrotron imaging, which allows for voxelsizes below 1μm. Several studies aimed to understand exactly these failure mechanisms, for example in short glass fiber reinforced thermoplastics [102, 103], in E-glass fiber reinforced epoxy resin [80] and carbon fiber reinforced epoxy laminate [109], and with special regard to influence of preexisting cracks in the same material in [110]. Synchroton facilities and measurements are however far from being easily accessible to standard research. Therefore, the fourth point, damage growth in the matrix, ideally correlated to the underlying fiber structure was investigated by Hufenbach et al. [61] by a laboratory CT with resolutions of approximately 25μm. They also performed compression experiments, similar studies were performed by Wang et al. [122], also in Micro CT. Wang et al. focused on kink bands, which usually can be found shortly before fibers fail in compression tests.

All of the contributions mentioned until this point have a very severe point in common: None of them investigates the motion between the different loading steps. Quantification of strain and stress is only done globally, and local material damage is investigated visually. One of the rare works to change this is by Borstnar et al. [19]. The authors use DVC on carbon fiber reinforced plastics to quantify delamination. Again, as delamination was the damage mechanism of interest, synchrotron images are needed. The method of investigation was a local DVC algorithm with a rather large subvolume size of 150^3 voxels. Therefore, the resulting displacement fields and the computed local strain are unfortunately very rough, though the image quality in our impression would allow for more detailed description.

Mazars et al. [82] carried out in situ tests on woven ceramic matrix composites at varying temperature. Not only did they use a global DVC to derive a qualitative damage scenario, but they also located cracks within the material – again with the help of DVC. Mendoza et al. [83] also considered woven composites and used full-field measurements computed by DVC to assess the influence of weaving anomalies. Recently, the second category of fiber systems, namely entangled ones without matrix, were investigated by Johansson et al. [64] by in situ tests and motion estimation by DVC. The underlying materials, single ply and 3-ply paperboard, were investigated at a resolution of 4μm within a tensile test. The local DVC at a subvolume size of 20^3 voxels was able to map the auxetic behavior of the paperboard, and gave a rough impression on formation on strain maxima at crack formation.

Lastly, we want to point towards a review covering plenty of different aspects of in situ tests of fiber systems, namely the work of Garcea et al. [42].

5.3 Concrete

Roughly speaking, concrete belongs to a very similar category as fibers in Section 5.2. It is a composite, meaning it consists of two or more materials, mixed together such that it forms a fully dense and strongly bonded material. As in all composite materials, different mixes lead to different material properties. In the work of Maryamh et al. [81], fiber reinforced concrete was investigated. The authors found dependencies between production parameters, the fibre positions and directions, and the mechanical performance. In 2003, Wang et al. [121] investigated the performance of different pavement varieties under load. But concrete is not only used in pavement, but also as common material in buildings. Both areas rely on the stability and durability of concrete. Therefore, thorough mechanical testing is – once more – inevitable.

Mechanically, concrete exhibits only a small elastic regime. Already at relatively low loading stages, microcracks, that developed during the manufacturing process, start to evolve. Formation of new microcracks follows, usually at porous areas, or at large voids. Failure of concrete is eventually induced, when the microcracks develop into macrocracks [67].

Various in situ studies have been performed to quantify this behavior. Trtik et al. [119] investigated tensile load of concrete with synchrotron based microtomography and observed bridging and branching on the microscale. Landis et al. [67] used in situ compression tests to improve numerical simulations of fracture processes. Yang et al. [127] computed material parameters of concrete with the help of DVC. The algorithm was also used to visualize the crack evolution. The motion, that concrete undergoes during drying in the manufacturing process was investigated by Wan et al. [122], also with the help of DVC. Chateau et al. [32] used DVC indirectly to segment cracks. Local DVC usually does not perform very well at jumps due to the implicit smoothness assumptions (see Section 3.1.1). Computing the residual will show the largest error exactly at the crack surface, and thresholding of the residual then yields the segmented crack.

5.4 Samples in this Contribution

The choice of limiting to foams, fibers and concrete is justified by the samples that we will investigate in this contribution. In the following section we will shortly describe the specific underlying materials and the corresponding tests, as well as their outcome in the volume images.

5.4.1 Synthetically Deformed Ceramic Foam

The first data set is given by synthetically deformed CT images of a ceramic foam. The original image is of size $630 \times 630 \times 230$ voxels at a voxel edge length of $70.88\,\mu$m. The

chosen displacement field follows [72], and is defined by

$$\begin{aligned} u(x,y,z) &= \begin{cases} z \cdot 0.005 \cdot K & \text{if } z < D/2, \\ (z - D/2) \cdot 0.005 \cdot K & \text{if } z > D/2 \end{cases} \\ v(x,y,z) &= 0 \\ w(x,y,z) &= -0.2 - \frac{K - 0.2}{\sqrt{0.5 \cdot e^{-0.04 \cdot (z - D/2)}}}, \end{aligned} \tag{5.4}$$

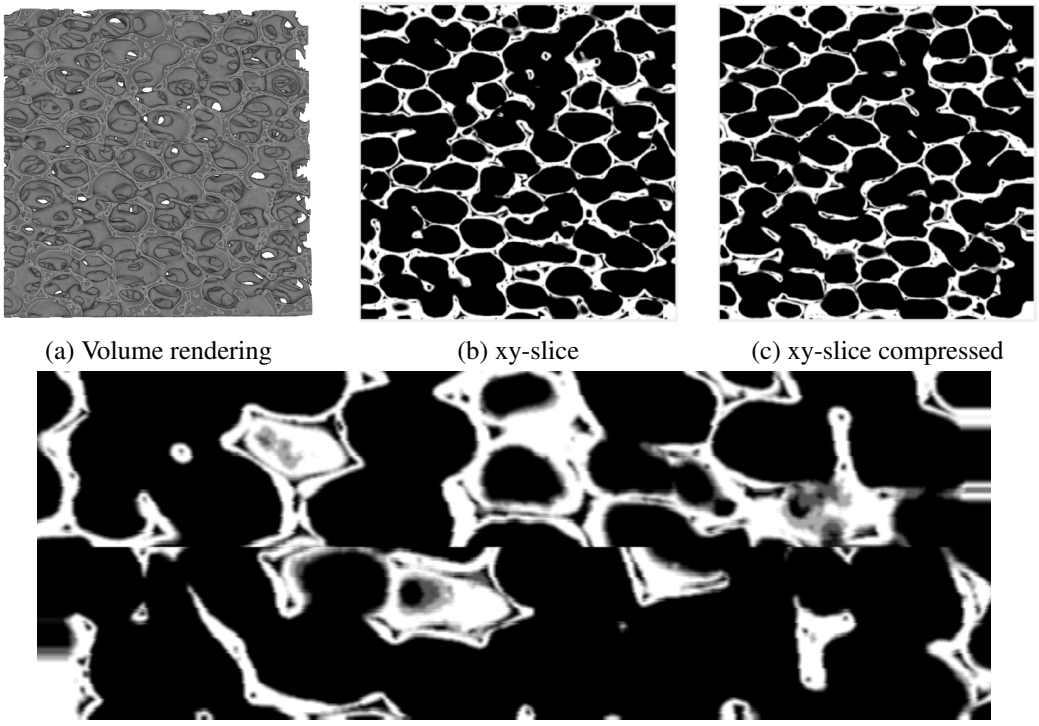
where K is the maximal displacement and $D = 230$ is the image size in z-direction. The displacement field exhibits two difficulties: The deformation in z-direction is nonlinear and mimics a compression. The deformation in x-direction is discontinuous mimicking struts breaking due to load. Both effects can be seen in Figure 5.1. The displacement field will be used in two ways. First, for fixed values $K = 20$ and $D = 230$, the performance of the Optical Flow variant in Subsection 4.2 will be compared against its competitors from materials science and MIR. Second, we will present a study on how different variants of Optical Flow perform regarding maximal displacement. To do so, the image of the foam is downscaled by factor 2 in each direction, and 80 deformed images are computed by starting with $K = 1$ and increasing it by 0.5 until $K = 40$. Though we may expect larger absolute displacement in other in situ tests, the displacement compared to the image size is still considerably big. This study will therefore aid in choosing appropriate algorithms and understanding what to expect from motion estimation algorithms.

Due to the known ground truth displacement field, we cannot only evaluate the quality of the residual but also the quality of the computed displacement vector fields. Exact definitions of error measures to do so will be given in the next chapter.

5.4.2 Simulated Displacement of LFT

The next data set that we will investigate consists of long fiber reinforced thermoplastics (LFT), for which a tensile test has been simulated. LFTs can be produced inexpensively and provide favorable properties with respect to stiffness and impact strength [90], therefore they are a popular replacement for metals in automotive industry. In order to fully exploit their potential it is crucial to characterize their material behavior under load reliably. In situ testing is the method of choice to accomplish this task as the LFT's macroscopic behavior is known to be driven by the microstructure, in particular the fiber component. In situ testing enables better understanding of the relationship of microstructural geometric and macroscopic physical properties by direct observation of microstructural changes during loading.

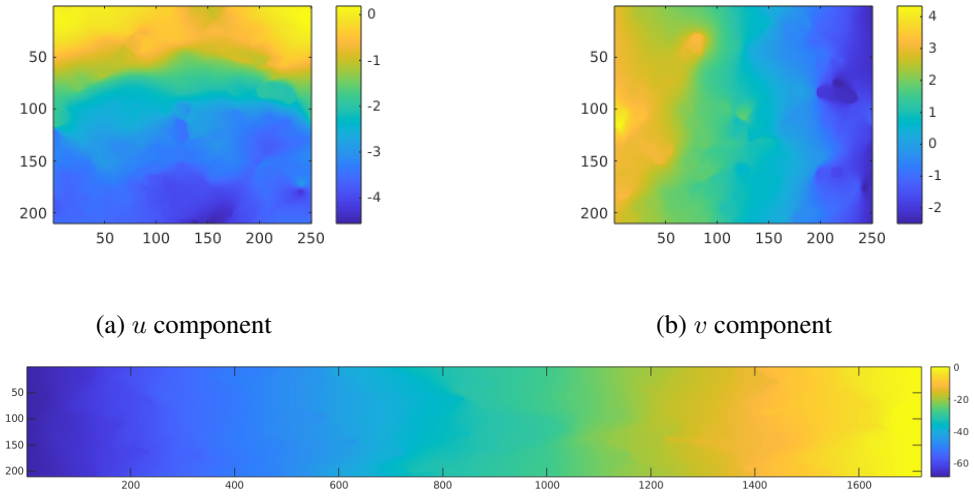
The simulation of the tensile test accounts for both matrix and fiber phase of the material. It is based on an efficient homogenization technique [107] and incorporated into the software FeelMath. Though introduced for general purpose simulations, the method has been successfully applied to LFTs [48, 47]. The simulated fields display a maximal displacement of 70 voxels, we therefore incorporate a scheme fairly similar to the one for the synthetically deformed foam. We scale the simulated fields linearly by a scalar t between 0.1 and 1.0 in discrete steps of 0.05. The resulting 19 displacement calculations are once more used to benchmark the influence of larger deformations. Slices of the displacement at scaling parameter $t = 1$ can be found in Figure 5.2.



(a) Volume rendering (b) xy-slice (c) xy-slice compressed

(d) xz-slice with discontinuity

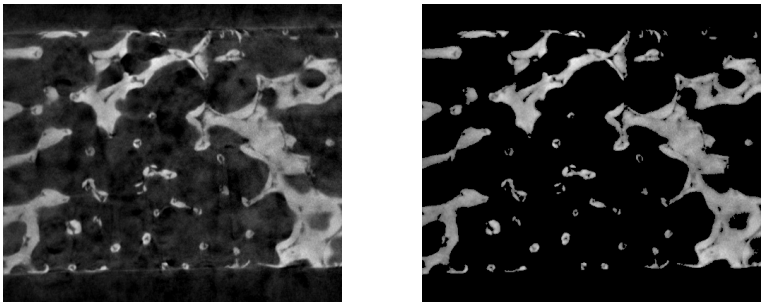
Figure 5.1: Volume and slice view of the synthetically compressed ceramic foam. Note the discontinuity in the middle layer in (d). Sample Foseco. Imaging Fraunhofer IZFP. Image size $630 \times 630 \times 230$ at voxel edge length 70.88μm.



(a) u component (b) v component

(c) w component

Figure 5.2: Simulated displacement fields for LFT. Images show a slice of u, v and w component respectively.



(a) Initial (b) Masked

Figure 5.3: MMC foam: Result of the background noise removal

5.4.3 Tensile Tests of LFT

Another example of an LFT will be used to assess the performance regarding large data sets. Until now no sample exceeded 1000 voxels in any dimension, but as mentioned it is essential that the presented motion estimation algorithms can also be applied to large volume images.

We use a μCT image of the central part of a tensile test specimen of size $1162\times769\times1481$ voxels at a voxel edge length of $5\,\mu$m. A volume rendering and a slice view illustrating the spatially sparse microstructure can be found in Figure 4.1.

5.4.4 Compression Tests on MMC Foams

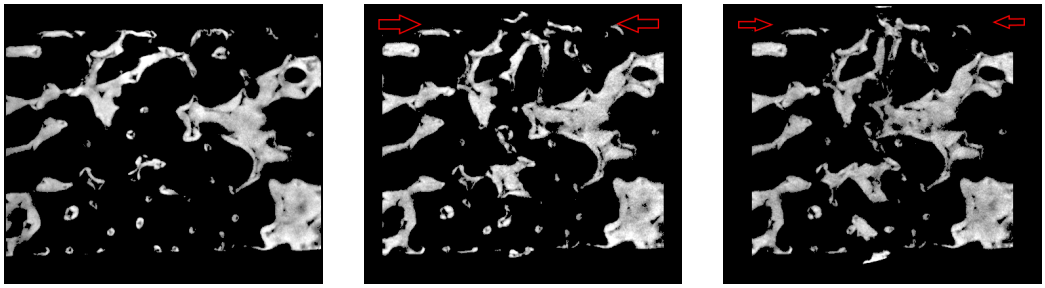
For the second data set we choose CT images of an MMC foam acquired during a compression test. The foam consists of highly alloyed stainless Cr-Mn-Ni TRIP-steel reinforced by magnesia partially stabilized zirconia (Mg-PSZ)-particles [16]. MMC foams are used as lightweight materials. Yet they need to be mechanically sufficiently stable. Their load-bearing capacity can be investigated in in situ tests. In the experiment described in [16], the foam was compressed by $2\,\%$, $10\,\%$, and $16\,\%$. The image size is $420 \times 420 \times 480$ voxels at a voxel edge length of $30\,\mu$m.

Figure 5.3(a) shows a slight corruption in the background caused by the high X-ray absorption of metals. We therefore pre-process the images: The whole image is binarized by Otsu's method, and the resulting binary image is used as a mask for the original image. Figure 5.3(b) shows the result. Note that these images are brightened to make the background corruption visible.

Metal foams feature rather large struts and do not fall in the category of foams whose failure is preceded by breaking struts. They are nevertheless a very suitable example to test the power of our algorithm. In fact, one can observe three stages of deformation during compression: linear elasticity, plastic collapse, and densification. Hopefully, our algorithm is able to map all three kinds of motion. The stages of deformation can be seen in Figure 5.4.

5.4.5 Compression Tests on Glass Foams

The third foam that we use to evaluate our method consists of recycled float glass. The CT images are acquired during an in situ compression test. In detail, we investigate the volumes at strain levels 1% and 3.8%. More information about the testing procedure can be



(a) Unloaded

(b) 10% compression, linear elastic and early plastic deformation

(c) 16% compression, plastic deformation and early densification

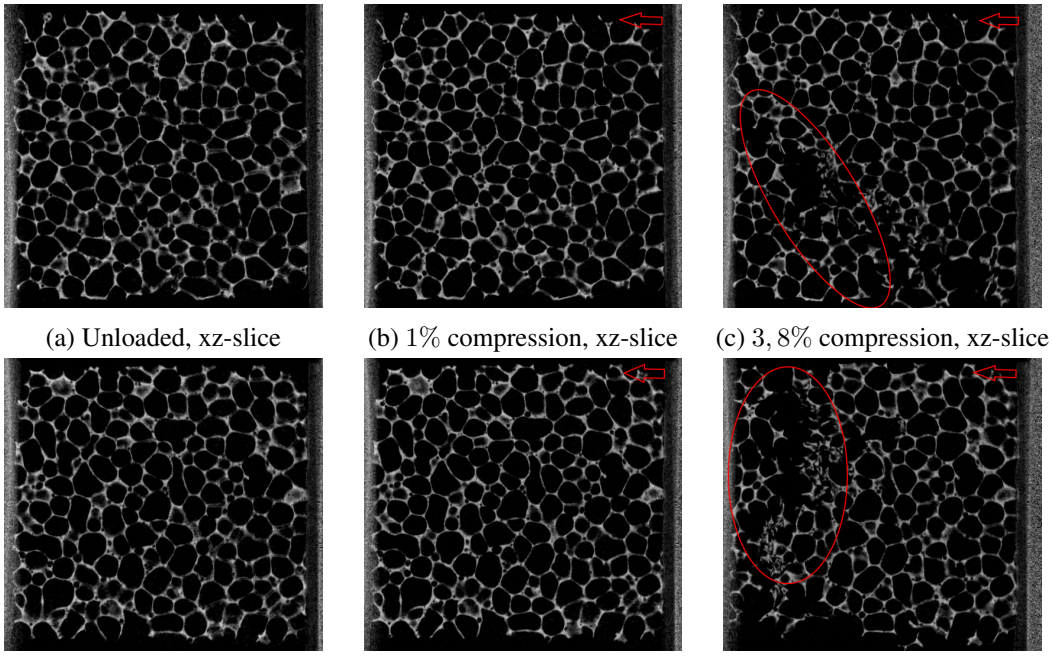
Figure 5.4: yz-slices showing the collision behavior of the MMC foam. Red arrows indicate compression direction. Sample, in situ testing, and imaging: TU Bergakademie Freiberg. Image size $420 \times 420 \times 480$ at voxel edge length 30μm.

found in [60]. The image size is $450 \times 450 \times 400$ voxels at a voxel edge length of $40\,\mu$m. Figure 5.5 shows the same slice at the different loading steps. While images (a) to (c) show the evolution of a crack orthogonal to the compression direction, we can also observe diagonal cracks in images (d) to (f).

5.4.6 Compression Tests on Refractory Concrete

The last time series discussed in this thesis consists of refractory concrete under load. Refractory concrete is a special type of concrete that is designed to withstand high temperatures. It usually consist of a hydraulic cement phase and is combined with heat resistant, refractory aggregates or fillers [87]. Their heat resistance allows for usage in domestic flues and chimneys, but they are also common in fire training areas.

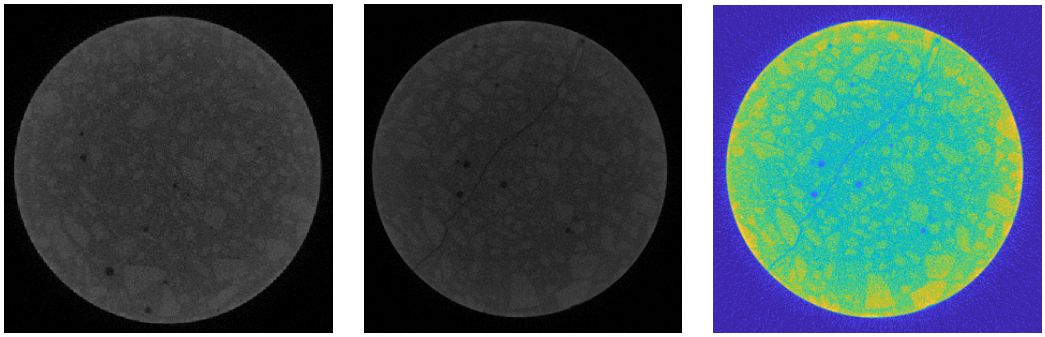
In this study, cylindrical samples have been investigated during compression tests. Three loading stages have been applied on the circular cross sections, where already at the first stage a fine crack can be observed. Figure 5.6 shows the sample, where 5.6 (a) is the unloaded stage, and 5.6 (b) and 5.6 (c) shows the loaded stage in gray and color transformed respectively, to make the crack more visible.



(a) Unloaded, xz-slice (b) 1% compression, xz-slice (c) $3,8\%$ compression, xz-slice

(d) Unloaded, yz-slice (e) 1% compression, yz-slice (f) $3,8\%$ compression, yz-slice

Figure 5.5: Slices of the compressed white glass foam. Observe that the first loading stage in (b) and (e) hardly influences the structural integrity. The second loading stage however introduces a diagonal crack visible in the lower left corner in (c) and a crack orthogonal to pressure direction visible in (f). The samples are compressed in negative z-direction, which is indicated by red arrows. Sample, in situ testing and imaging by TU Bergakademie Freiberg.



(a) Unloaded, xz-slice (b) Loaded, gray (c) Loaded, color

Figure 5.6: Slices of concrete sample.

Chapter 6

Evaluation of Motion Estimation Algorithms

It's too easy in computer vision to develop an algorithm that does something plausible *on a few images rather than something* correct

— RICHARD SZELISKI[1]

The following chapter deals with a thorough evaluation of our methods. By defining three error measures, we not only measure the goodness based on the warped images, but also on reconstructed displacement fields. In contrast to Optical Flow, there exist no commonly accepted benchmarks for DVC. We therefore generate two ground truth displacement fields for evaluation: One is a rather synthetic, hardly realistic field, that nevertheless exhibits the difficulties we expect in materials tests: jumps, smoothly varying parts and strongly differing components. The second ground truth displacement consists of simulated displacement – it is therefore of high relevance. Parts of the evaluation have already been published by the author of this thesis in [91]. The displacement studies in Section 6.2.2 and Section 6.2.3 as well as the applications in Section 6.3 have not been part of this contribution.

6.1 Error Measurements

Assessing the error, that comes with any motion estimation algorithm is an important point in evaluating the performance of the methods. Luckily, we can once more have a recourse on methods provided by the Optical Flow community. The amount of methods that have been developed in 2D made a structured and commonly accepted evaluation unavoidable. The 2D vision community widely accepts to evaluate Optical Flow based on three measures [9]. The first one, also manifested as meaningful by the DVC community [69], is the *root mean-squared error* (RMSE). Before stating the error formally, we want to fix the notion of *residual*. Recall that our algorithms are supposed to reconstruct displacement vector fields $\mathbf{u}$ such that $I_0(\mathbf{x}) = I_1(\mathbf{x}+\mathbf{u})$. It is therefore very obvious to assess the performance of an algorithm based on this exact relation. Hence, one can consider the *absolute value of*

[1] Richard Szeliski is one of the co-developers of the famous "Middlebury" benchmark in Optical Flow [9]. The citation is taken from his monograph on Computer Vision [116].

the initial residual

$$r_0 = |I_0(\mathbf{x}) - I_1(\mathbf{x})|,$$

and the *absolute value of residual* after warping,

$$r = |I_0(\mathbf{x}) - I_1(\mathbf{x} + \mathbf{u})|,$$

where $I_1(\mathbf{x} + \mathbf{u})$ was computed by interpolation at deformed coordinates. We will consider normalized images, that means that both residuals have a maximal value of 1. Note that we will often slightly carelessly call these quantities only *initial residual* and *residual*.

The RMSE is now given by averaged sum of the squared residual, i.e. if N is the number of voxels, then

$$\text{RMSE} \coloneqq \left(\frac{1}{N} \sum_x \left(I_1(\mathbf{x} + \mathbf{u}) - I_0(\mathbf{x}) \right)^2 \right)^{\frac{1}{2}}. \tag{6.1}$$

In our setting, this will be the method of choice for all data sets, where we do not have access to a ground-truth displacement field. This type of error can always be computed, and in general an algorithm is considered to perform well, if the RMSE regarding the residual is much lower than the RMSE regarding the initial residual. Anyhow, a low residual only partially allows for conclusions concerning the quality of the displacement vector fields. This is due to the nature of our ill-posed problem: Again, we estimate a three dimensional quantity (displacement) based on a one dimensional relation (intensity difference), so we cannot expect a unique solution.

Consequently one needs to find a measure to evaluate displacement as well. A crucial part in the following two sections thus will be the comparison of the computed displacement field to ground truths, which have been artificially generated. The Optical Flow community mainly uses two means to do so. The first one is the average angular error (AAE)

$$\text{AAE} \coloneqq \frac{1}{N} \sum_{\mathbf{x}} \arccos \Big(\frac{1 + \mathbf{u} \cdot \mathbf{u}_{GT}}{\sqrt{1 + u^2 + v^2 + w^2} \sqrt{1 + u_{GT}^2 + v_{GT}^2 + w_{GT}^2}} \Big), \tag{6.2}$$

where $\mathbf{u} = (u, v, w)$ and $\mathbf{u}_{GT} = (u_{GT}, v_{GT}, w_{GT})$ are the components of the computed and the ground truth displacement field, respectively. The second one is the average endpoint error (AEE), which can be understood as a residual in the displacement field, i.e.

$$\text{AEE} \coloneqq \frac{1}{N} \sum_{\mathbf{x}} \Big(\sqrt{(u - u_{GT})^2 + (v - v_{GT})^2 + (w - w_{GT})^2} \Big). \tag{6.3}$$

AAE measures the angular deviation between computed and ground truth displacements, but penalizes in regions of non-zero motion more than in areas of zero motion. This seems to be arguable, the second measure AEE was introduced. Note that however none of the measures seems to be common neither in DVC nor in MIR. But keeping in mind that we compute motion between in situ tests, the quality of the constructed displacement fields may even be of higher importance than low residuals. A thorough evaluation of the relevance of both vector field measures may assist the findings in this contribution, but will be postponed to future work.

Method	Abbr.	Source Code	Ref.
Optical Flow	3DOF	https://gitlab.itwm.fraunhofer.de/kuschnerus/3dof	[91]
local DVC	ICGN	https://github.com/FranckLab/ALDVC	[126]
global DVC	FEM	https://github.com/jyang526843/FE_Global_DVC	[126]
ALDVC	ALDVC	https://github.com/FranckLab/ALDVC	[126]
Elastic	Elas	https://github.com/C4IR/FAIR.m	[86]
LDDMM	CLAIRE	https://github.com/andreasmang/claire	[77]

Table 6.1: Testbed for performance comparison on synthetically deformed foam including links and references.

6.2 Performance Evaluation of Motion Estimation Algorithms

The objective of the first evaluation will be a performance comparison with other state-of-the-art methods. As representative in Optical Flow based methods we chose the algorithm presented in Section 4.2, denoted by 3DOF, and compared it to elastic (Section 3.2.1), and LDDMM (Section 3.2.2) image registration methods, and local (Section 3.1.1), global (Section 3.1.2) and combined (Section 3.1.3) DVC methods. Following the names of their implementations, the algorithms for comparison are denoted by ICGN, FEM, ALDVC, Elas and CLAIRE, respectively.

Our final choice of methods was not exclusively influenced by the fact whether they seem suitable from a mathematical viewpoint. Even further – methods that were available open-source or are capable of dealing with large data sets were preferred. That led to a testbed of six methods, whose links to source code and reference can be found in Table 6.1. The library FAIR [86] would also allow to use Hyperelastic Image Registration. Unfortunately, there is no matrix-free version of the implementation, and already a downscaling by factor 2 in each coordinate direction exceeds the memory of our largest available cluster computer with 1400 GB RAM. Furthermore, the evaluation on the artificial example already shows why the smoothness assumptions in MIR are not suitable for materials science. Our impression is that the algorithms in CLAIRE [76] have a better chance in at least partly mapping jumps. In addition, these algorithms are, as already mentioned, very costly. In the evaluation on the real-world examples therefore only CLAIRE is used for comparison. The parameters for all methods can be found in Tables 6.2 (3DOF), 6.4 (CLAIRE) and 6.3 (DVC). The parameters have been extracted by a small prestudy. Coarse ranges were extracted from the literature on $2D$ methods, and fine tuning was then performed on 3D datasets. Note that due to the size of the images, applying an exhaustive parameter search is hard, if not impossible.

6.2.1 Synthetically Deformed Foam - Single Computation

The RMSE values for this data set already demonstrate the superior performance of our algorithm. Figure 6.1 shows a slice view of the residuals of the data sets at a critical position. The view exhibits the main difficulty of the data set, namely the discontinuity. All algorithms struggle to compute a perfect residual here, 3DOF showing the lowest error.

Table 6.6 contains RMSE, AAE and AEE for the synthetically deformed foam with maximal displacement $K = 20$. Again, our method 3DOF clearly yields the lowest values. The values in the range achieved here by 3DOF are considered to prove good performance already in two dimensions. The outstanding performance can also be observed visually. Figure 6.2 shows slices of the displacement field in all coordinate directions of our method,

Data set	3DOF							
	λ	μ	M	N	W	L	d	σ
Synth. Def. Foam	1	20	15	15	3	10	0.9	$\frac{1}{\sqrt{2d}}$
MMC 10%	10	10	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
MMC 16%	10	10	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
Glass Foam 1%	20	5	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
Glass Foam 3.8%	20	5	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
LFT	50	50	10	5	3	15	0.98	$\frac{1}{\sqrt{2d}}$

Table 6.2: Parameter choice for our algorithms. λ and μ regularization parameters, M number of outer iterations, N number of inner iterations, W number of warps, L number of levels, d downscaling factor, σ smoothing factor.

Data set	ICGN	FEM		ALDVC	
	S_V	S_{FEM}	α	S_{FEM}	S_V
Synth. Def. Foam	40	10	2	10	40
MMC 10%	40	20	2	10	40
MMC 16%	40	10	2	10	40
Glass Foam 1%	40	10	2	10	40
Glass Foam 3.8%	40	10	2	10	40
LFT	40	120	2	40	120

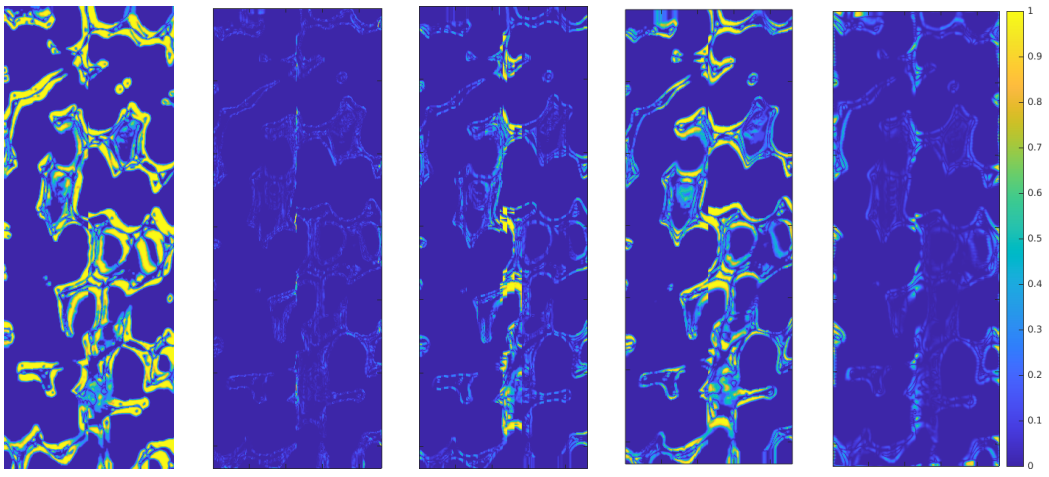
Table 6.3: Parameter choice for DVC based methods. S_V subvolume size for ICGN based local DVC, S_{FEM} subset size for global FEM based DVC, α regularization parameter. Note that for ALDVC the used parameters are the recommended minimal choice of parameters by the original authors, except for LFT, where the sizes had to be increased due to the sample size.

Data set	CLAIRE	
	N	n_t
Synth. Def. Foam	40	16
MMC 10%	20	8
MMC 16%	20	8
Glass Foam 1%	20	8
Glass Foam 3.8%	20	8
LFT	20	8

Table 6.4: Parameter choice for CLAIRE. N maximal number of iterations, n_t time stepping.

Data set	3DOF	ICGN	FEM	ALDVC	CLAIRE
MMC 10%	**0.0110**	0.0225	0.0202	0.0224	0.1245
MMC 16%	**0.0101**	0.0173	0.0163	0.0172	0.1467
Glass Foam 1%	**0.0124**	0.0164	0.0147	0.0146	0.0627
Glass Foam 3.8%	**0.0182**	0.0237	0.0234	0.0224	0.0767
LFT	**0.0073**	0.0277	0.0283	0.0277	0.2930

Table 6.5: RMSE (equation 6.1) for our application cases. Best performing method highlighted in bold.



(a) Initial (b) 3DOF (c) ALDVC (d) Elas (e) CLAIRE

Figure 6.1: xz-slices of residuals of the synthetically deformed ceramic foam. Yellow color in the residual indicates the largest possible error, dark blue indicates a perfect match between both data sets. The scale for all the data sets can be found on the very right of the figure. The images are normalized, the maximum possible error is therefore 1.

Measure	3DOF	ICGN	FEM	ALDVC	Elas	CLAIRE
RMSE	**0.0876**	0.1265	0.1343	0.1069	0.115	0.1732
AAE	**2.29°**	2.34°	2.31°	2.31°	11.5°	15.1°
AEE	**1.60**	2.57	1.99	2.05	5.42	24.14

Table 6.6: Error measures of all methods for the synthetically deformed foam.

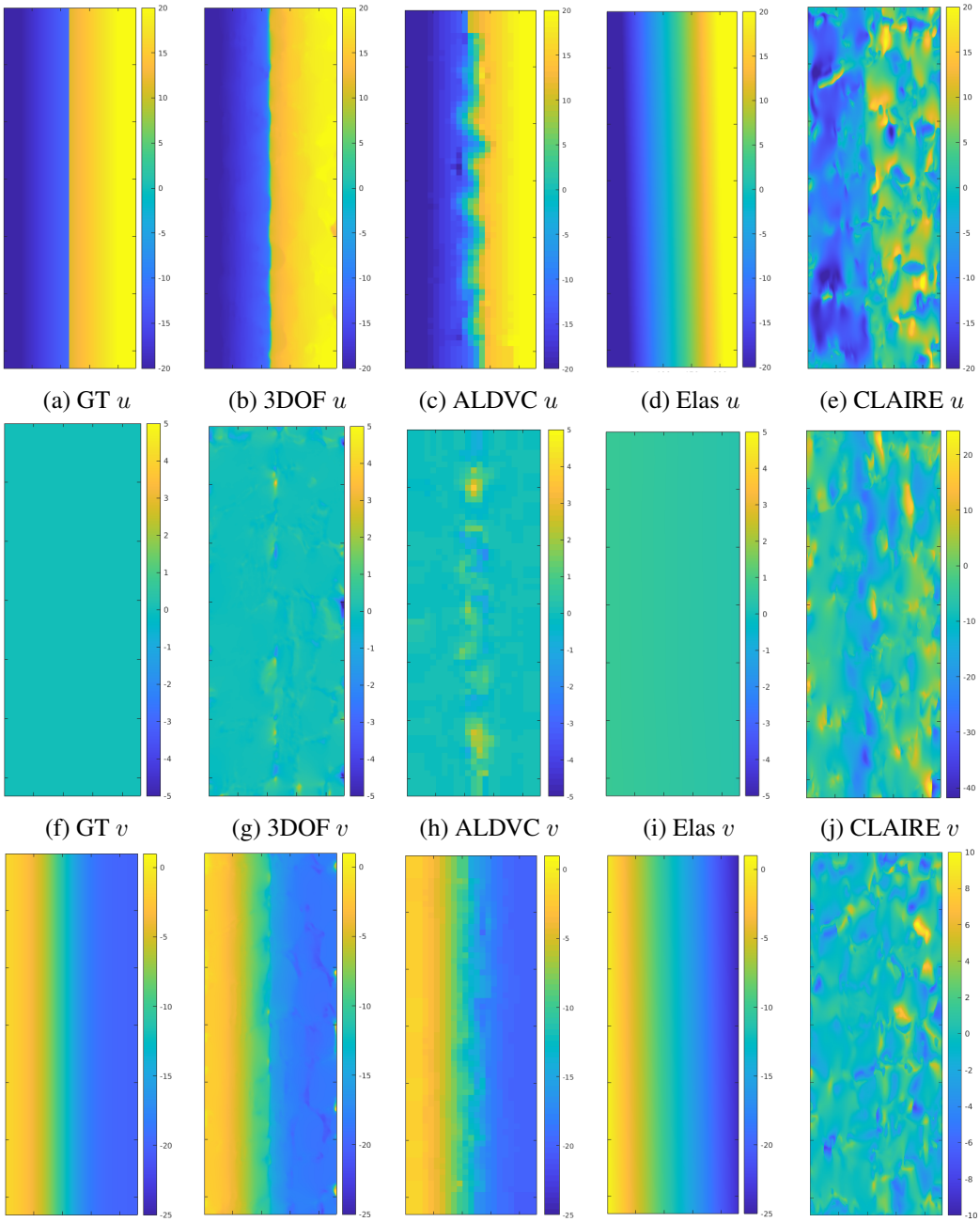
ALDVC as best performing DVC based method, Elas and CLAIRE. Especially the discontinuity in u is computed best by our algorithm. Note that for the displacement fields in u and v direction, the color axis for CLAIRE had to be modified due to large outliers. Figure 6.2 also quite remarkably shows what we hinted in the theoretical investigation of the motion estimation algorithms: ALDVC (Section 3.1.3) and elastic image registration (Section 3.2) seek for the displacements in spaces that produce functions which are too smooth to capture our artifical discontinuity in the u-component. But exactly these smoothness assumptions of course favor the reconstruction of the w-component, as the displacement here is smooth. The 3DOF introduces an edge in the displacement field that is not in the ground truth field. However, let us remark that it is often very unlikely to find such discontinuities only in one of the components, while the other ones remain smooth. In the case of fracture, this error may therefore be not very severe.

At points of discontinuity, most methods including ours still struggle. This can be seen in the difference images of Figure 6.3. In Figure 6.3 (a), the initial difference image is shown. Here, white pixels represent components, that overlay in both images. Magenta color corresponds to the solid component that is only apparent in the original, undeformed foam, whereas the green color corresponds to the solid component of the synthetically deformed foam. A favorable result would therefore be a difference image, that consists only of white color. Our algorithm nearly achieves this, see Figure 6.3 (b). Note that in this case the magenta color displays the initial foam deformed by the computed displacement field. The same holds for Figures 6.3 (c) and 6.3 (d), which show the difference image resulting from ALDVC and CLAIRE. Compared to this result, our algorithm performs much better, however the zoom shows the inability of the algorithm to eventually *break* the strut.

We also used this synthetic example for a small study on how the parameter λ influences the quality of the estimation of the displacement vector field. In Figure 6.4, we can see that a careful choice of λ results in a minimum in the RMSE and, most important, yields a drastic improvement over setting $\lambda = 0$, so not including the difference in gradients at all. Note that nonetheless the choice of the parameters μ and λ is highly problem dependent. To our knowledge, there are no rules for calculating the optimal parameters, they can only be found by testing, for example, on downscaled versions of the images to reduce computation time.

6.2.2 Synthetically Deformed Foam - Maximal Displacement Study

The synthetically deformed foam and the available ground truth can also be used for a performance study regarding increasing displacement. We compared four different implementations of Optical Flow: The large displacement Optical Flow from Section 4.2, again denoted by 3DOF, an approach which solves the total variation by primal dual algorithms from Section 4.3, denoted by TV, the morphological primal dual scheme from Section 4.5, denoted by TVMorph and a primal dual implementation of the TGV regularizer, denoted

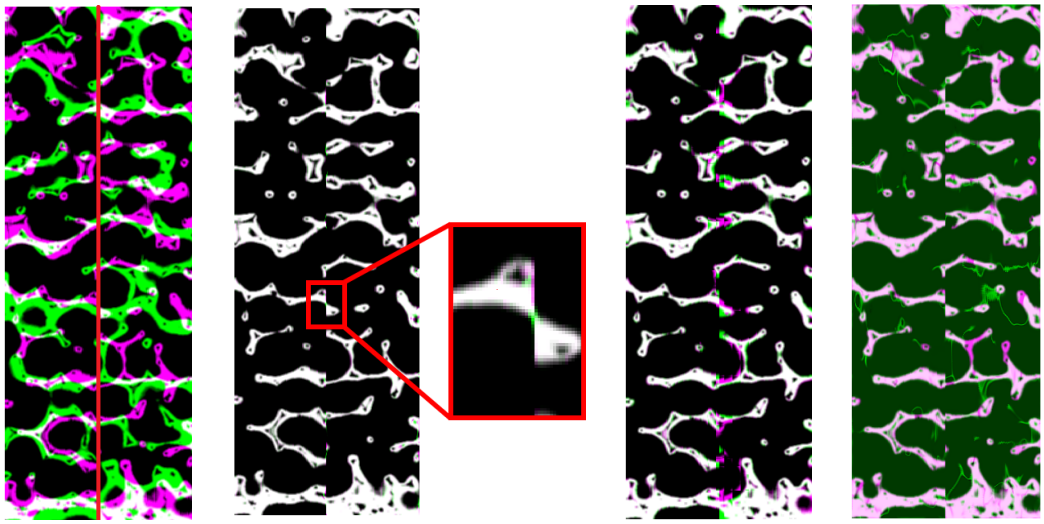


(a) GT u (b) 3DOF u (c) ALDVC u (d) Elas u (e) CLAIRE u

(f) GT v (g) 3DOF v (h) ALDVC v (i) Elas v (j) CLAIRE v

(k) GT w (l) 3DOF w (m) ALDVC w (n) Elas w (o) CLAIRE w

Figure 6.2: Slices of the ground truth (GT) and computed displacement fields for the synthetically deformed ceramic foam. Yellow color indicates movement along the horizontal axis, blue refers to the opposite direction.



(a) No warping (b) 3DOF with zoom (c) ALDVC (d) CLAIRE

Figure 6.3: Difference images of the synthetically deformed foam. White: pixels in the solid component in original and deformed image. Magenta: pixels only in the solid component of the initial foam in (a) and deformed by the computed displacement fields in (b)–(d). Green: pixels only in the solid component of the synthetically deformed foam. In (a) we also marked the discontinuity by a red line. To not distort the results, we forgo this marker in the computed images. The zoom in (b) shows that the displacement field does not fully capture discontinuous behavior. 3DOF did not break the strut.

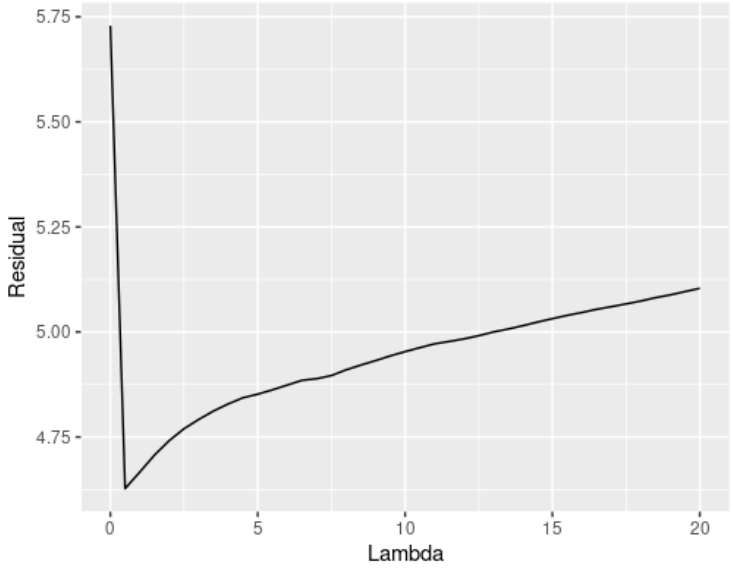


Figure 6.4: Influence of the parameter λ. The residual plot shows the improvement over $\lambda = 0$ and also that an optimal value can be found at $\lambda = 1$.

Parameter	θ	λ	(α_0, α_1)	σ	τ
TV	0.2	25	–	1	0.25
TVMorph	0.2	25	–	1	0.25
TGV	0.2	–	$(0.1, 10)$	0.1	0.25

Table 6.7: Parameters for additional Optical Flow based methods.

by TGV (Section 4.4), and lastly, as best competitor also ALDVC. The parameters for the additional TV methods can be found in Table 6.7. Figure 6.6 now shows the performance with the best performing set of parameters. RMSE-wise, TV performs best, with 3DOF and TGV achieving similar results. The angular measures identify 3DOF best, with only marginal differences to other Optical Flow based algorithms as the displacements turn large. As already expected, ALDVC cannot compete with the voxel-based methods residualwise, as it struggles to compute the jump in the u-component properly. Interestingly, TVMorph seems to stabilize the AAE for large displacements.

The angular measures identify 3DOF best for small and medium displacements, with only marginal differences to other Optical Flow based algorithms. Interestingly, TVMorph and ALDVC seem to stabilize the AAE for large displacements. Especially ALDVC still produces reliable results for large displacements. An explanation for this can be given by the usage of an initial guess. Based on Fourier methods [11], a preceding computation is performed to fulfill the requirement of the Gauss-Newton-method, namely to start the iteration close to the actual optimum. This strategy is highly related to the idea of the coarse-to-fine strategy in Optical Flow: Some *easier* method is used to generate a starting point for the current iteration, either by efficient approximate methods like in ALDVC or on a coarser scale as in 3DOF. However, for the plots in Figures 6.6 we used roughly the same amount of multiresolution steps for all Optical Flow Methods (10 with downscaling factor 0.9 for the classical, 9 for the morphological scheme). The reason why 3DOF then performs worse than ALDVC nevertheless is that due to large displacements, the approximations on coarse scales become bad initial guesses. Recall that the coarse-to-fine scheme in Optical Flow is incorporated to ensure the applicability of linearization by Taylor approximation. This means, that on the coarsest scale the maximal displacement should be around one voxel. One can therefore pose the following rule of thumb when deciding for the choice of parameters d and L

$$Kd^{L-1} = 1, \tag{6.4}$$

where K is the maximal expected displacement on the finest scale, d is the downscaling factor, and L is the number of levels. The heatmap in Figure 6.5 displays this relation visually. If one now follows this rule and alters the parameters to fit for large displacements, we again observe a better performance compared to ALDVC, even if preinitialization is performed for large displacement.

Another major difference in the choice of parameters in the Optical Flow based algorithms can be found in the number of warps used to derive satisfactory results. Where 3DOF only needs 2 warping steps, we need at least 10 for TV and TGV to produce an error that is sufficiently low. This becomes even more visible in the accompanying studies in Figure 6.7. Here we see that increasing the number of warps linearly also increases the maximal possible displacement in 3DOF linearly. However doing the same with TV does have less and less influence on the performance. That means that the treatment of the

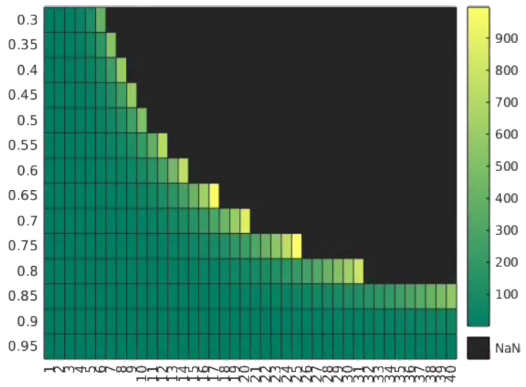


Figure 6.5: Heat map indicating how parameters for the image pyramid should be chosen. X-axis: Number of levels in the pyramid. Y-axis: Downscaling factor. Color: Maximal displacement value in voxels that can be captured by the whole coarse-to-fine scheme. NaN values indicate parameter combinations that exceed image limits, as this scheme condenses the image edge length below 1.

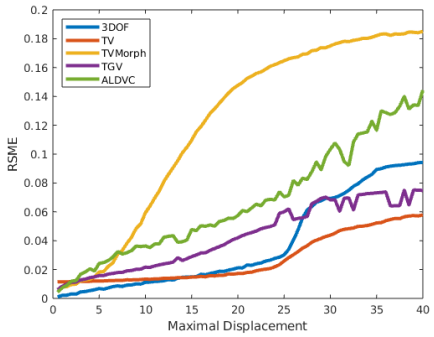
nonlinearities in 3DOF in fact is beneficial compared to just performing a first-order Taylor linearization as in TV or TGV.

Yet another fact is interesting when observing the error plots: Assessment of which method performs best seems to be highly dependent on the error measure. If we looked for example only at RMSE (Figure 6.6), one would conclude that TV performs best. Even further – in 3DOF it would be highly questionable if increasing the number of warps from 2 to 5 has any beneficial effect at all, see Figure 6.7 (a). But if we consider the error measures on the displacement fields (Figures 6.7 (c) and 6.7 (e)) we see that more warps allow for larger maximal displacement. Vice versa, if one expects to observe large displacements in the in situ test, we highly recommend to increase the number of warps, even if this also causes an increase of computational time.

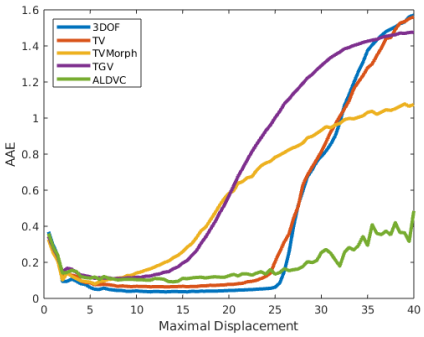
6.2.3 LFT - Maximal Displacement Study

The second study on the performance of the algorithms for increasing displacement was executed on simulated displacement fields of LFT. Though the fields seem to be smoother than what we will construct in the foam samples without ground truth, it still offers many of the difficulties that come with motion estimation in materials tests: We can find plateau regions or jumps, but also large smoothly varying areas in the component images.

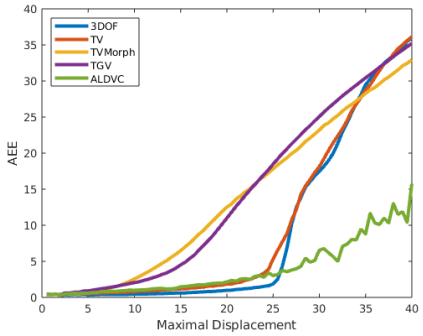
Figure 6.8 shows RMSE, AAE and AEE for the test series. The RMSE of all methods is remarkably low – it starts by a factor 10 lower than what we found in the previous section. The discrepancy between 3DOF and the methods that solve based on primal dual approaches is once more explainable due to the use of Kačanov's method. Using 5 warps in the TV based approach seems just not to be enough to compete with the 2 warps of the Kačanov method. Let us investigate this finding a little bit further. Observing the AAE for ALDVC, we find a minimum at maximal displacement of approximately 20, which corresponds to the scaling parameter $t = 0.3$. All other methods seem to grow linearly for increasing t. We therefore extracted the residuals for all methods at parameter $t = 0.3$, and also recomputed the displacement fields for TV and TVMorph with 15 instead of 5 warps. We see that both



(a) RMSE



(b) AAE

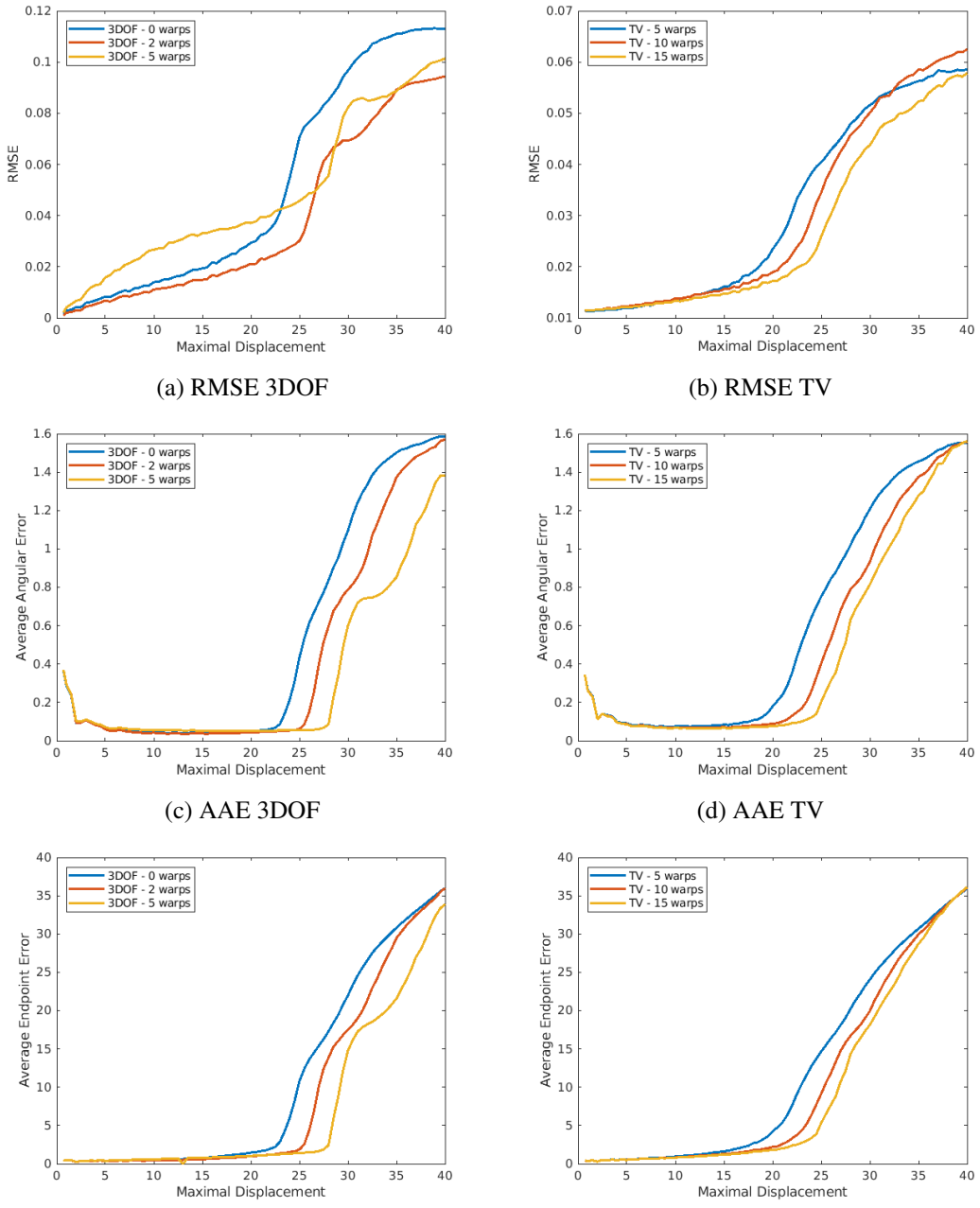


(c) AEE

Figure 6.6: RMSE, AAE and AEE measures for maximal displacement study on synthetically deformed foam.

Data set	Time (in hours)	RMSE	AAE	AEE
3DOF	4.5	0.0003	0.003	0.04
TV – 5 warps	3.9	0.006	0.16	4.20
TV – 15 warps	13.3	0.0026	0.02	0.62
TGV	18.5	0.0087	0.48	7.27
TVMorph – 5 warps	1.9	0.0089	0.21	5.80
TVMorph – 15 warps	5.5	0.0058	0.095	3.59
ALDVC	2.4	0.0016	0.12	23.88

Table 6.8: Calculation time and error measures for scale parameter $t = 0.3$ (maximal displacement ≈ 20).



(a) RMSE 3DOF

(b) RMSE TV

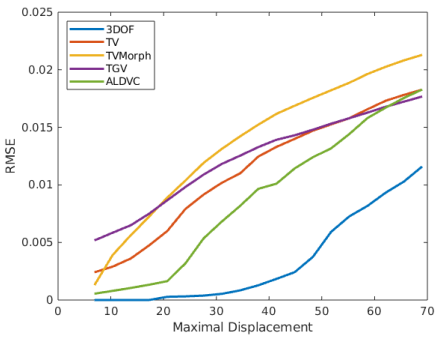
(c) AAE 3DOF

(d) AAE TV

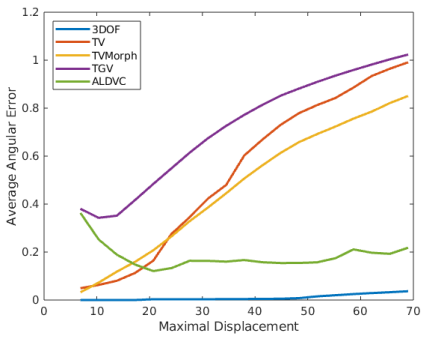
(e) AEE 3DOF

(f) RMSE TV

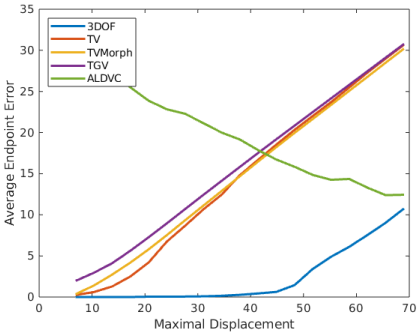
Figure 6.7: Error measures for 3DOF and TV with different numbers of warps.



(a) RMSE

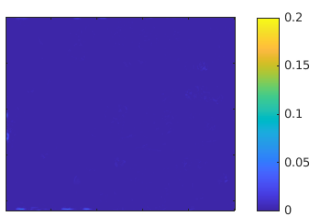


(b) AAE

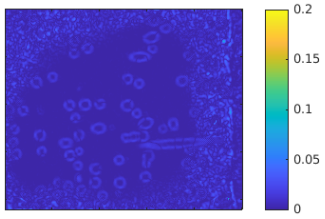


(c) AEE

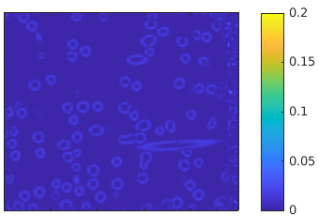
Figure 6.8: RMSE, AAE and AEE measures for simulated displacement fields of LFT. 3DOF performs significantly better than all other methods.



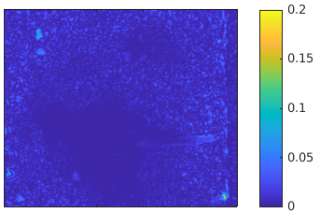
(a) 3DOF



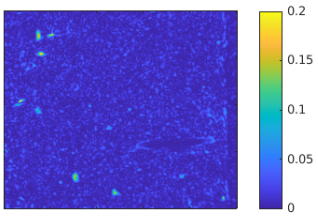
(b) TV – 5 warps



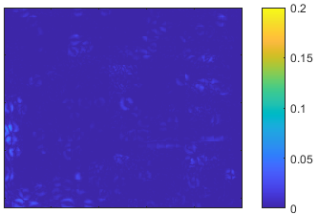
(c) TV – 15 warps



(d) TVMorph – 15 warps



(e) TGV



(f) ALDVC

Figure 6.9: Residuals of simulated displacement for scale parameter $t = 0.3$ (maximal displacement approximately 20 voxel).

methods still do not perform as remarkably as 3DOF, which exhibits a residual that is nearly flawless (see Figure 6.9), but yet all the methods produce surprisingly accurate residuals compared to what we saw in the previous section. Let us proceed with an inspection on the displacement fields in Figure 6.10, in particular the w-component. We see that, except 3DOF, ALDVC and TV with 15 warps, all other methods only roughly display the correct behavior. The error mainly occurs on the left-hand-side of the depicted slice. These are in fact the areas of large displacement, and – consulting also the initial residual of this slice in Figure 6.11 – we see that large displacement cannot be reconstructed very well in areas of sparse spatial content with a low number of warps in TV. This makes sense: Taking a second glance at the terms involving the optimization for 3DOF and TV we see that many terms involving I_d remain in the iterations of 3DOF, where almost all terms are replaced by gradient approximations in TV. Once more the "freezing" of Kačanov seems to be beneficial for motion estimation in materials science. Note that also the visually appealing performance of ALDVC can be explained (see Figure 6.10 (g)). Fibers do admit the required speckle pattern for which DVC was designed in the first place. However, once more due to the choice of subvolumes instead of voxel bases, we lose detail in the calculated fields. The consequence is a significantly worse performance than 3DOF, but a very competitive one compared to other methods.

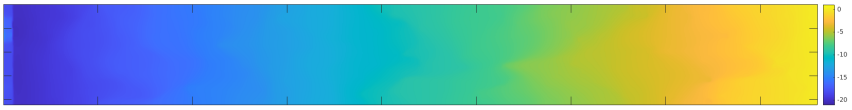
The low error measures for all of the methods can be explained by the absence of large jumps in the simulated displacement fields. The finite element approach in simulation (once more) assumes implicit smoothness, namely by assuming the displacement is an element of a Hilbert space. It will be left to future work to compare simulated displacements with computed ones, where the computation will be performed based on the actual tensile test on these data sets. Note however that bringing these displacement fields into accordance will entail an investigation in different function spaces and how well solutions can be approximated in these.

A last aspect that has been checked with these samples is the computation time of all methods, which can be found in the first column of Table 6.8. All Optical Flow based methods have been executed in the exactly same setup, ALDVC on a less powerful machine. For sure, TVMorph with 5 warps and ALDVC produce results in significantly less time – ALDVC computes less degrees of freedom by definition, where TVMorph reduces computational time algorithmically due to its keener downsampling scheme. What we also want to emphasize is that when using the same resources in time and memory, 3DOF achieves much better results than TV with 5 warps. To accomplish comparable performance of TV to 3DOF, we have to triple the computation time, which still seems feasible in this example. But recalling the dimension of $210 \times 250 \times 1720$ voxels, this can be considered only a moderate image size, and looking at the results, one could easily justify the need for even more warps. We however want to remark that all methods leave room for improvement: 3DOF currently uses successive overrelaxation, a method that is difficult to parallelize, and all methods based on primal-dual approaches would heavily benefit from the use of a GPU instead of (as in our case) a CPU.

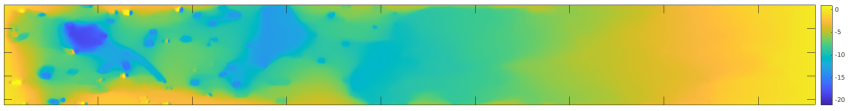
6.2.4 LFT - In Situ Test

Having dealt with synthetically generated pairs of images in the previous section, we now want to turn towards a real tensile test on LFT. Note that this data set and the one from Section 6.2.3 do *not* coincide and stem from completely different experimental setups.

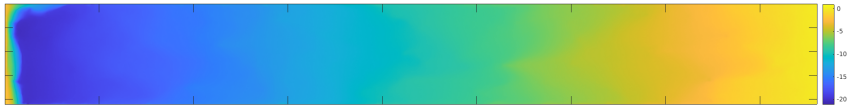
(a) Ground truth



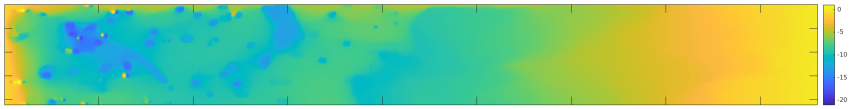
(b) 3DOF



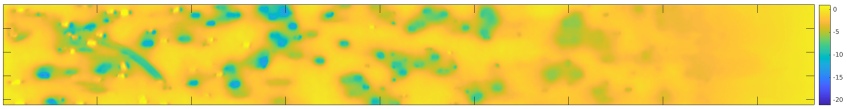
(c) TV – 5 warps



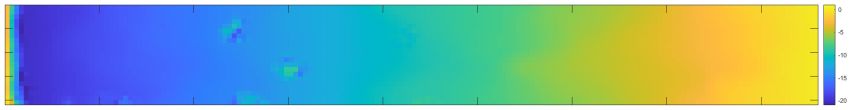
(d) TV – 15 warps



(e) TVMorph



(f) TGV



(g) ALDVC

Figure 6.10: w-component of simulated displacement for scale parameter $t = 0.3$ (maximal displacement approximately 20 voxel).

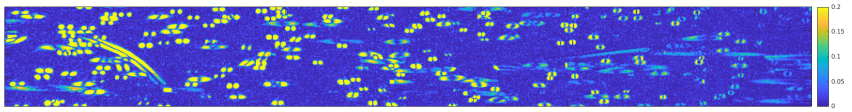
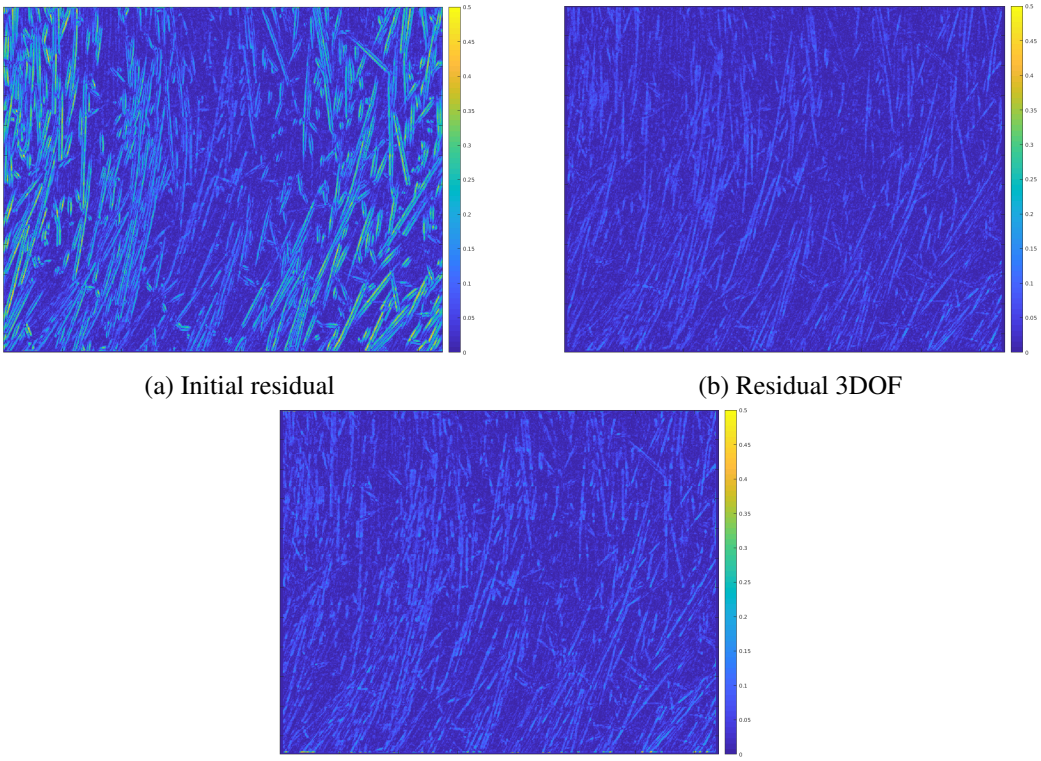


Figure 6.11: Initial residual extracted at the same position as w-components in Figure 6.10.



(a) Initial residual

(b) Residual 3DOF

(c) Residual ALDVC

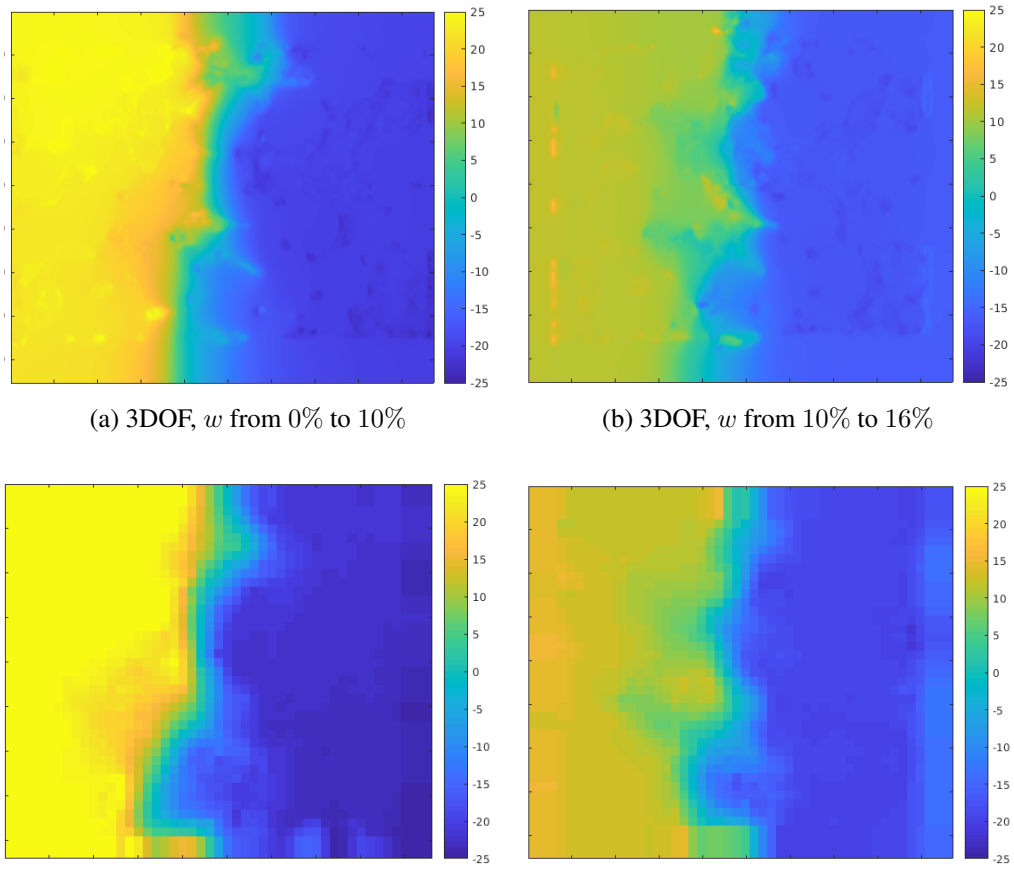
Figure 6.12: LFT: slice view of the residuals

Figure 6.12 shows the residuals for the LFT data set. The algorithm managed to match the individual fibers properly, and to significantly reduce the overall RMSE, when computed with the initial residual. Note that the initial RMSE is already much lower than for all data sets we discussed earlier. This can also be seen in Figure 6.12(a), where the maximal absolute difference value can be found at ≈ 0.2, where for other data sets it was close to 1, see for example Figure 6.1. This means that the expected maximal displacement is also much lower than for the foam samples we already investigated. Nevertheless, the residual of our method in Figure 6.12(b) shows a slight improvement and a lower residual than ALDVC in Figure 6.12(c). Therefore, our proposed method does not only compute large displacement of foams in a very exact way, but also smaller displacement of fibers.

6.2.5 MMC Foam

Table 6.5 again shows that we outperform the state-of-the-art methods with respect to the RMSE. Note that from now on, we will only focus on the analysis of our results and renounce further comparisons to CLAIRE and local and FE-based DVC, but show comparisons to ALDVC, as it performed best among the comparison methods.

Our interest also lies in the displacement field. As the data set was generated during a compression test, the computed displacement should reflect the behavior during the test. Figure 6.13 shows, that our algorithm again describes the increasing compression best. We can see a very detailed edge where material densifies in the middle. ALDVC resolves this behavior rather roughly. The displacement field between 10% and 16% presents a plausible



(a) 3DOF, w from 0% to 10%

(b) 3DOF, w from 10% to 16%

(c) ALDVC, w from 0% to 10%

(d) ALDVC, w from 10% to 16%

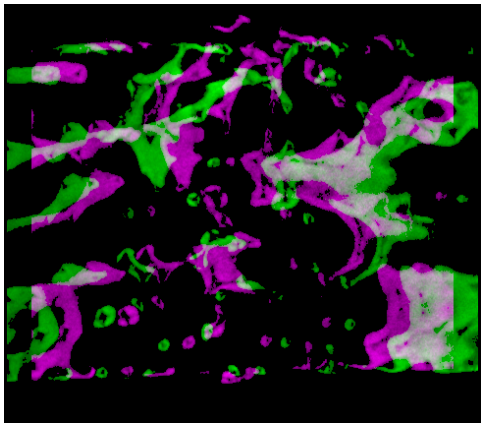
Figure 6.13: xz-slices of displacement field of the MMC foam. Again, yellow color indicates movement along the horizontal axis, blue refers to the opposite direction. The computed displacement field reflects the expected behavior during the experiment.

discontinuity along the failure. Note that we only focus on w, the displacement field in z-direction, as it is along compression direction and therefore the most interesting one.

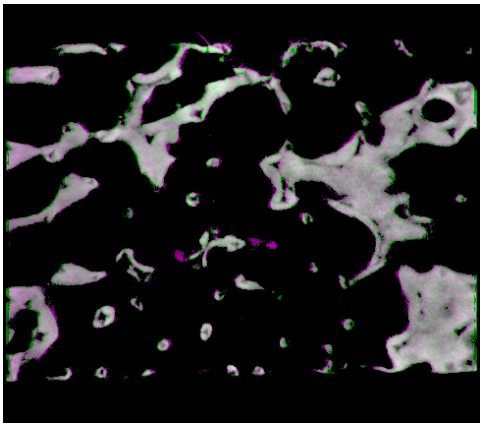
However, the difference images in Figure 6.14, composed in the same way as in Section 6.2.1, again show the difficulty of brittle and plastic behavior. We can clearly see that the result on those parts yet needs to be improved in future work.

6.2.6 White Glass Foam

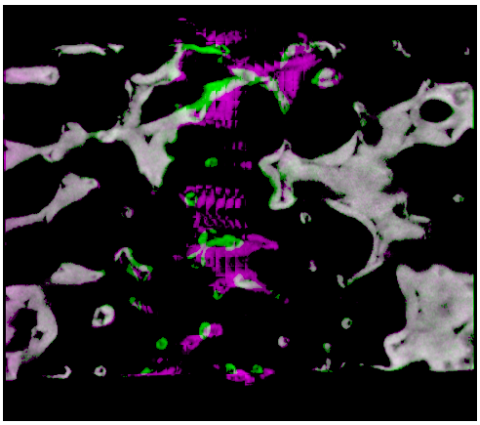
We proceed with a data set that is the main representative of data sets of special interest for us. Its spatial content is very sparse, as it consists of very thin struts, and is fractured rather immediately between 1% and 3.8% of loading. Figure 6.15 shows the displacement of the glass foam from 1% to 3.8%. We did not include the displacement field of the first loading stage here, as it does not contain any large damage behavior. The qualitative result of the displacement field perfectly maps the behavior described in [60]. This example also shows remarkably the difference between DVC-based methods like ALDVC and our approach: We compute a true voxel-based displacement field that is not gained by interpolation as in DVC. The accurate outcomes of this approach are apparent over all directions, but are par-



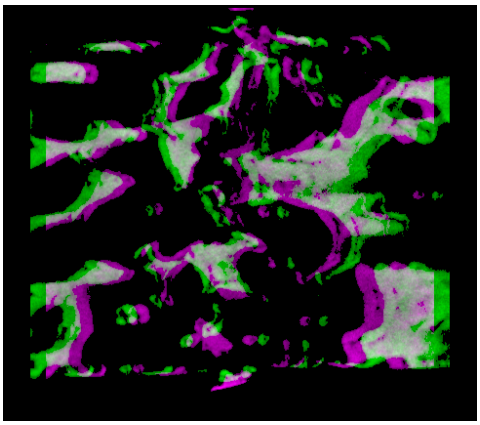
(a) Initial, 0% to 10%



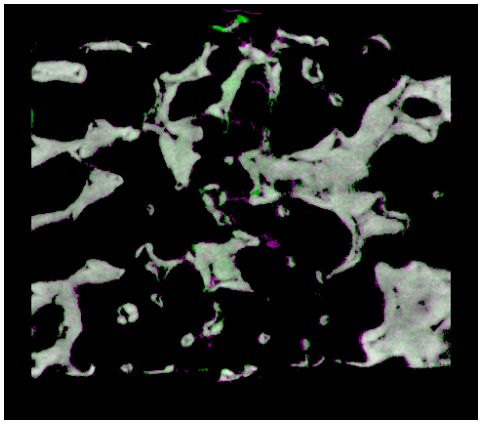
(b) 3DOF, 0% to 10%



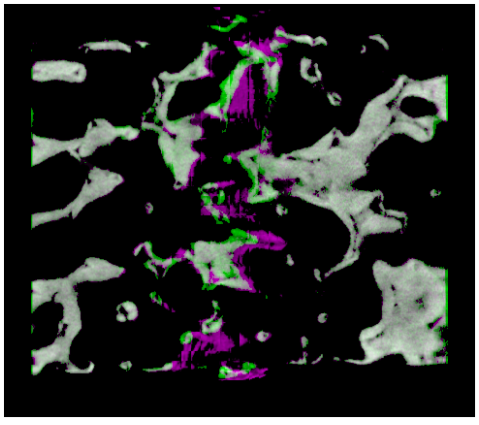
(c) ALDVC, 0% to 10%



(d) Initial, 10% to 16%

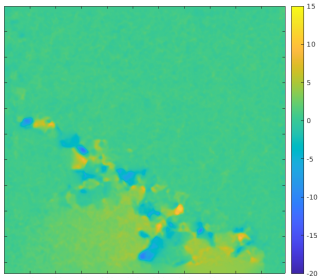


(e) 3DOF, 10% to 16%

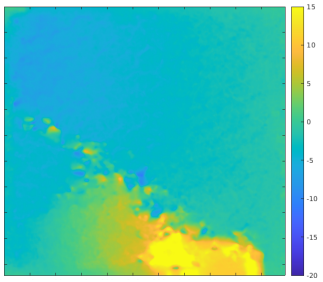


(f) ALDVC, 10% to 16%

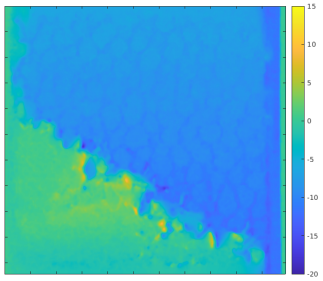
Figure 6.14: xz-slices of the difference image of MMC foam. White: pixels in the solid component in original and deformed image. Green: pixels only in the solid component of the initial foam in (a) and (d) and deformed by the computed displacement fields in (b), (c), (e) and (f). Magenta: pixels only in the solid component of the deformed foam. Especially brittly fractured parts are difficult to match, as we can see in magenta in (b), (c), (e) and (f).



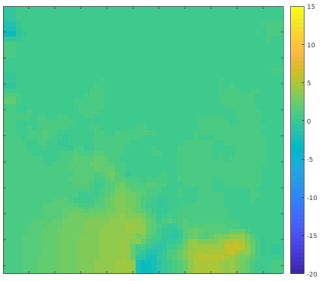
(a) 3DOF, xz-slice u, 1% to 3.8%



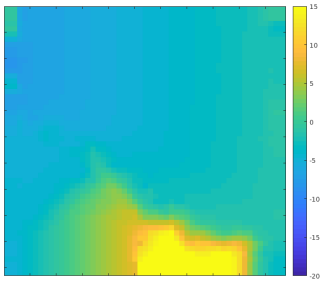
(b) 3DOF, xz-slice v, 1% to 3.8%



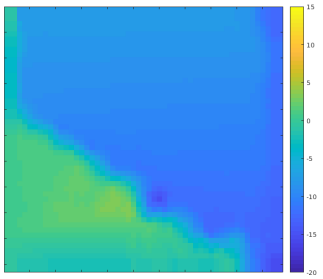
(c) 3DOF, xz-slice w, 1% to 3.8%



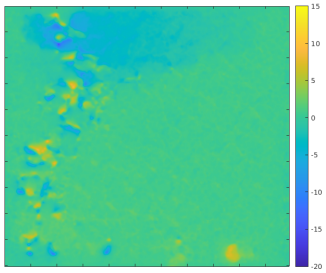
(d) ALDVC, xz-slice u, 1% to 3.8%



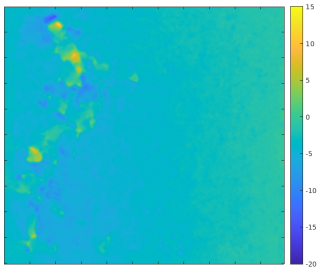
(e) ALDVC, xz-slice v, 1% to 3.8%



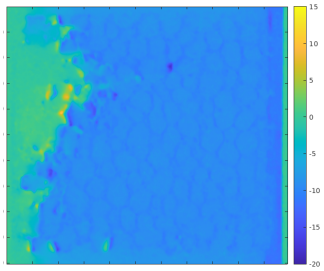
(f) ALDVC, xz-slice w, 1% to 3.8%



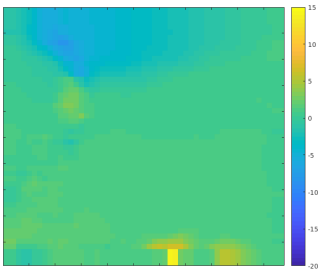
(g) 3DOF, yz-slice u, 1% to 3.8%



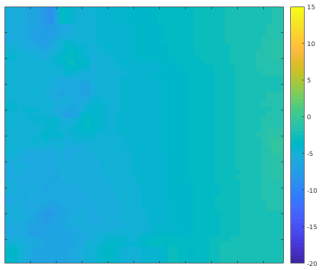
(h) 3DOF, yz-slice v, 1% to 3.8%



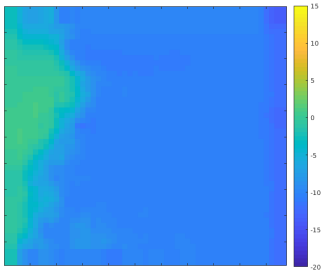
(i) 3DOF, yz-slice w, 1% to 3.8%



(j) ALDVC, yz-slice u, 1% to 3.8%

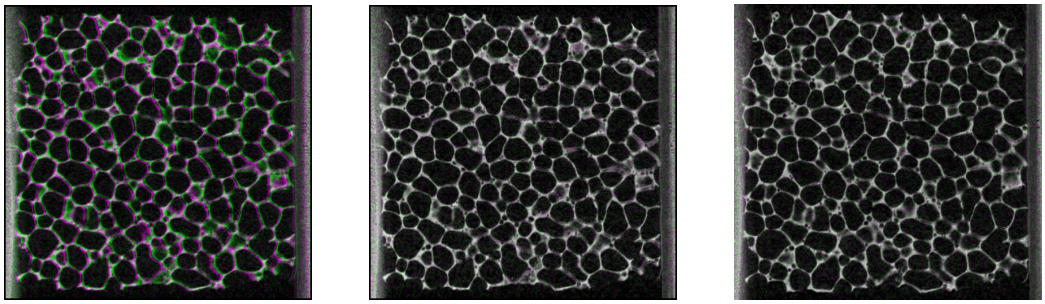


(k) ALDVC, yz-slice v, 1% to 3.8%



(l) ALDVC, yz-slice w, 1% to 3.8%

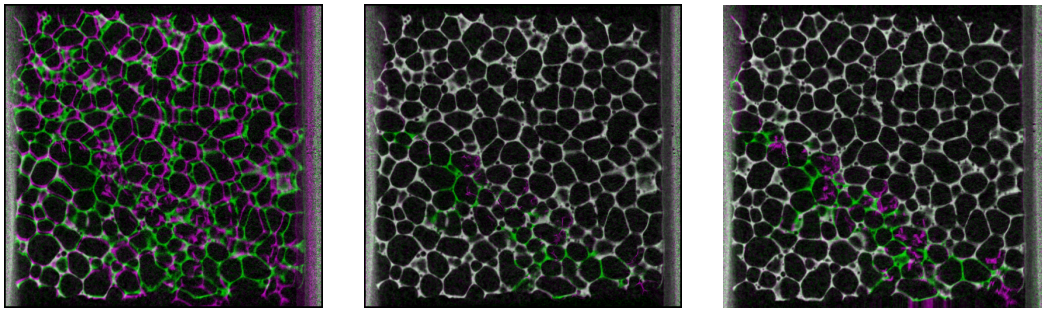
Figure 6.15: Slices of the displacement field of the glass foam. The discontinuity along the previously observed fault zones can be well observed in the computed displacement fields in z-direction in (c) and (i). Both the averaging nature of the DVC based displacement field and the fine details of 3DOF can be observed very well.



(a) Initial, 0% to 1%

(b) Computed 3DOF, 0% to 1%

(c) Computed ALDVC, 0% to 1%



(d) Initial, 1% to 3.8%

(e) Computed 3DOF, 1% to 3.8%

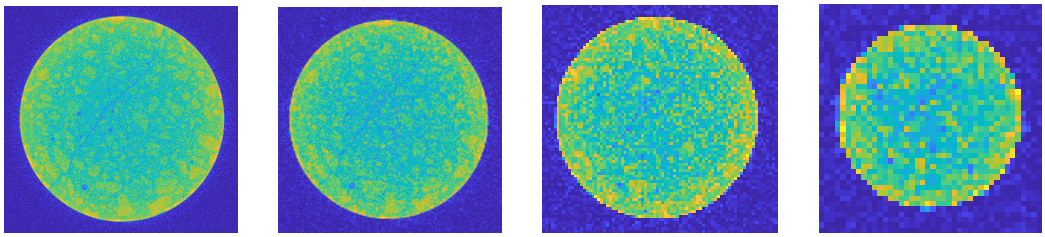
(f) Computed ALDVC, 1% to 3.8%

Figure 6.16: xz-slices of the difference image of the glass foam. White: pixels in the solid component in original and deformed image. Green: pixels only in the solid component of the initial foam in (a) and (d) and deformed by the computed displacement fields in (b), (c), (e) and (f). Magenta: pixels only in the solid component of the deformed foam. Observe the green component in the lower left corner of (e) and (f). Although the displacement field displays plausible behavior, the fault zone is not mapped correctly.

ticularly prominent in Figures 6.15(c), (f), (i) and (l). These images show the fault zones in loading direction and are therefore of special interest. Here we see that our approach calculates a strut-wise displacement, where we gain the impression that ALDVC computes only an average displacement (per subvolume). Nevertheless, the difference images in Figure 6.16 again shows the weakness of our method. Especially in this case, where the motion between 1% and 3.8% is rather large, the areas suffering from brittle fracture cannot be matched sufficiently, but still better than in ALDVC. And again, the improvement in these regions is postponed to future work.

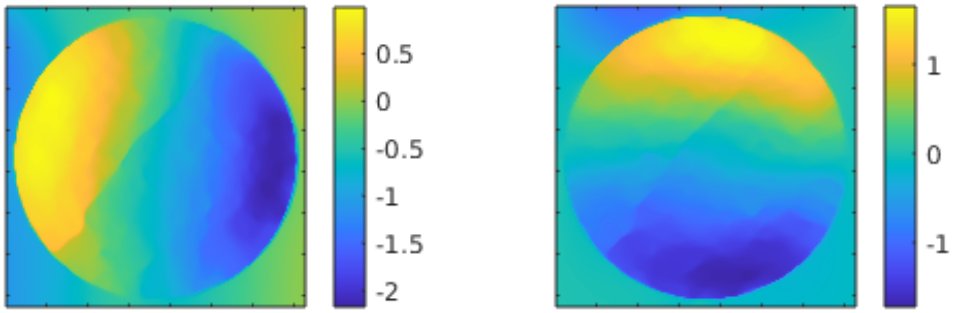
6.3 Specialized Applications of Motion Estimation

The last section of our evaluation will deal with the use of our methods beyond just plain displacement calculation. Both applications will exploit the voxel-based calculation of our computations and will show how our method can assist engineers in the assessment of their materials.



(a) Step 3 (b) Step 6 (c) Step 9 (d) Step 12

Figure 6.17: Application of morphological wavelet transform to concrete sample. Images display those stages that finish in a Cartesian voxel grid. Rescaling to equal sizes has been applied for visual impression. One can clearly see that the crack is preserved over all scales. The initial number of voxels in each direction of the quadratic slices was 648 and was then reduced to 324, 162, 81, and 41 voxels, respectively.



(a) u-component (b) v-component

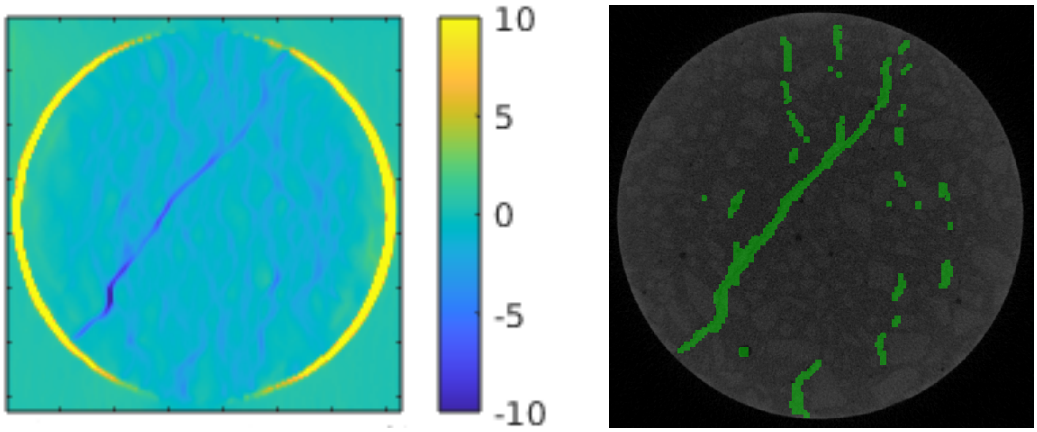
Figure 6.18: Displacement field components of concrete at downsampling stage 6.

6.3.1 Crack Detection in Concrete by Morphological Optical Flow

We start by investigating the compression of refractory concrete with our morphological approach to TV Optical Flow. Instead of using a coarse-to-fine scheme up to the original voxel grid for our calculations, we will use only four coarse steps. That is, we initially downscale the image with our morphological wavelets by 9 steps and calculate displacement on each level until we reach a quadratic voxel grid again. The four stages are therefore Cartesian, dodecahedral, cuboidal and finally Cartesian.

In Section 4.5, we already saw that morphological wavelets preserve minima, if Min-Lifting is applied. In Figure 6.17, we see the outcome of a downsampling scheme by Min-Lifting. 12 steps are applied, where we only display those stages that appear on cubic grids. We can clearly see that while loosing details concerning the aggregates in the concrete (which is the desired effect of a coarse-to-fine scheme in Optical Flow), all stages maintain their local minima in the crack area.

The calculated displacement fields between downscaling step 9 and step 6 at final step 6 can be found in Figure 6.18. Clearly, the displacement exhibits a jump in the crack area. Calculating the $(1,1)$-component of the strain $\varepsilon_{1,1}$, i.e. the derivative of u in x-direction, in fact clearly displays the crack as a local minimum, that can easily be thresholded. The strain component and the threshold computed on it can be seen in Figure 6.19.

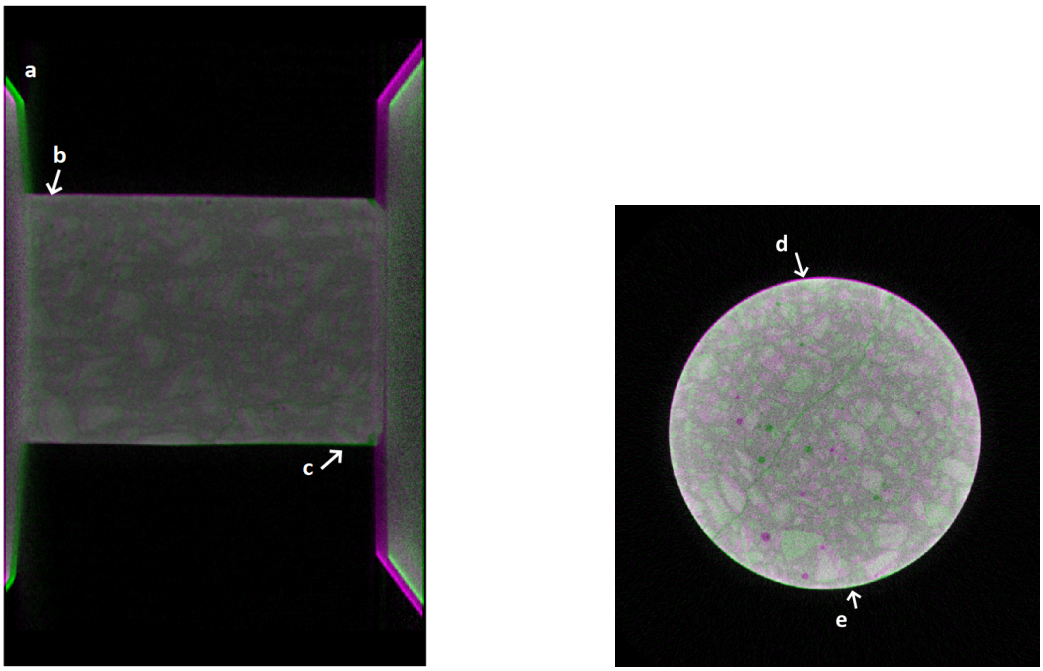


(a) $\varepsilon_{1,1}$-component (b) Detected Crack

Figure 6.19: Strain component and crack segmentation based on displacement field calculated at stage 6. In (a) negative maximal strain is indicated in blue, positive maximal strain in yellow, and zero strain in turquoise.

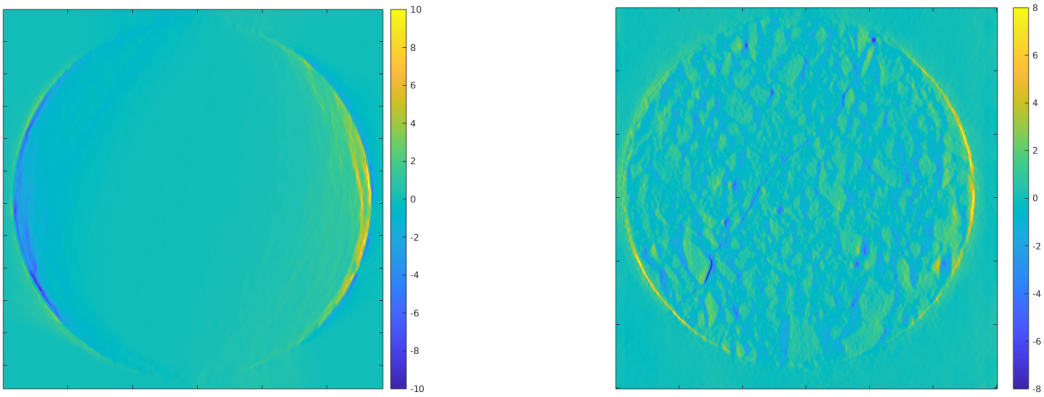
The advantages of this method are manifold. First, our approach, that preserves the important detail within all coarsening steps, can be interpreted as a reversion of the coarse-to-fine scheme of classical 2D Optical Flow – at least when applied in materials science. Recall the reasoning for coarse-to-fine schemes in 2D: only large objects will undergo large motions and those objects can still be found after smoothing and downsampling. This can also be accepted partly for in situ, as the strong, global deformation of the MMC foam in the previous section for example will also be apparent on the coarsest scale. But often the "interesting" motion in materials science is not occurring on the coarse but actually on the finest scale, such as local cracking or fracture. Luckily, such material failure will almost always result in generation of new void material, which in CT imaging *will* manifest in a local minimum. Preserving these minima up to the coarsest scale therefore enforces computation of this behavior in a very early stage of the algorithm. So, though being at a coarse resolution, we still match fine details first – a reversion of the original scheme.

This fact turns also out to be beneficial for avoiding local minima. In Figure 6.20 we see corresponding slices of the unloaded and stage 1 loaded volume, presented in an overlay image. Recall that an overlay image indicates by color, which structures can be found in one image but not in the other. First, we spot the misalignment on the very left in Figure 6.20 (a), denoted by letter a, where the mounting is supposed to coincide in both images, but the green shade indicated that it did not. Second, on top of the concrete sample we see a purple shade (letter b), where in parallel we see a green shade on the bottom (letter c). In fact, that indicates rotation of the samples towards each other. It becomes also apparent if we look at Figure 6.20 (b), where d and e again represents purple and green shade, respectively. In classical TV Optical Flow, the rotation overlays the motion induced by fracture, as seen in Figure 6.21. The $e_{1,1}$ component shows no extrema due to the crack anymore, in contrast to the approach which used morphological wavelets. Here, the influence of the rotation is also visible, but the crack can still be seen very well. In the area of materials science morphological coarse-to-fine schemes therefore perform more robust than classical approaches.



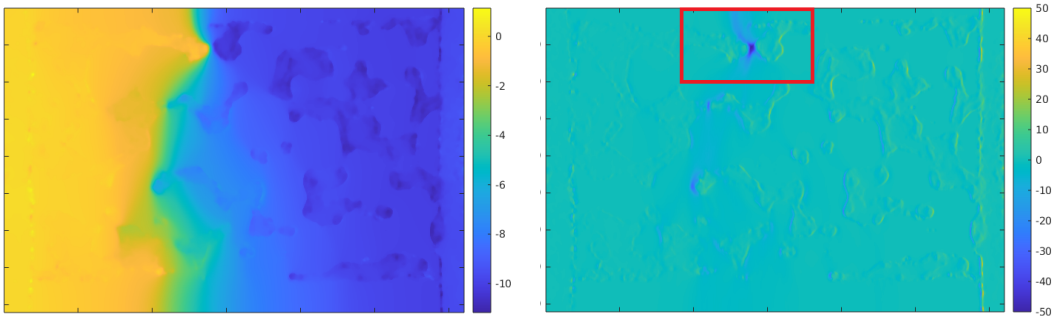
(a) xz slice (b) xy slice

Figure 6.20: Overlay of both loading stages.



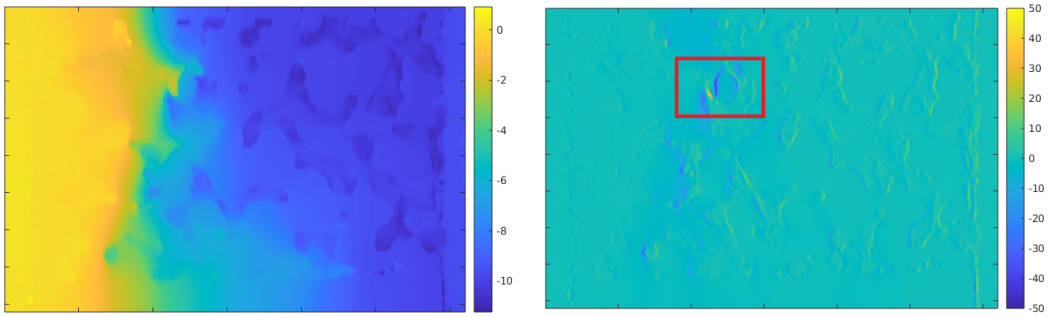
(a) $\varepsilon_{1,1}$ component based on 3DOF (b) $\varepsilon_{1,1}$ component based on TVMorph

Figure 6.21: Comparison of strains based on 3DOF and TVmorph.



(a) Sliceview of w-component 1

(b) Sliceview of $\varepsilon_{3,3}$ with local extremum 1



(c) Sliceview of w-component 2

(d) Sliceview of $\varepsilon_{3,3}$ with local extremum 2

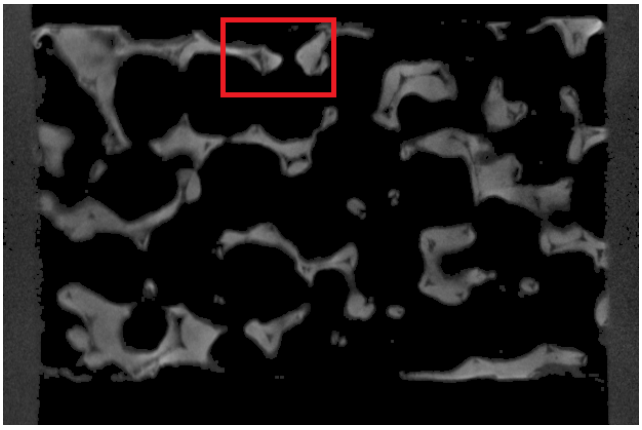
Figure 6.22: Slices of displacement and strain, positions of potential interest marked by red boxes.

Another advantage comes with the nature of the wavelets, but can be interpreted twofold: As we estimate motion on downsampled images, the complexity of the algorithm drastically decreases. Performing calculations on stage 6 as in the present example only uses 2% of the amount of the original voxels. As especially in materials science there is a trend towards resolving specimens of several meters at resolutions of micrometers, the resulting images of $10\,000^2 \times 2\,000$ voxels have to be treated in a very efficient manner. This is clearly fulfilled by our approach.

And yet another advantage stems from this interplay: As we used the coarsest levels to estimate our motion, we basically calculated displacement on compressed data. In samples as large as the ones just mentioned it is very advantageous to have a hand on algorithms that do not require full decompression of the data.

6.3.2 Early Stage Failure Identification in Foams by Optical Flow

The last example in this contribution will now shed a light on the fact why it is so important to have a voxel exact algorithm for materials that exhibit delicate behavior like foams. The material investigated here will be a fairly similar one to the MMC foam presented in Section 6.2.5. A detailed description of the compression tests of this sample can be found in [15]. Images were taken at $2, 4, 6, 10$ and 20 percent of compression. The investigators did not expect fracture already at the lowest compression rate, they therefore only investigated the sample containing 10% compression for fractured cells. We now check if the displacement computed between unloaded and loaded at 2% will give a hint on where material failure will occur.



(a) Densification

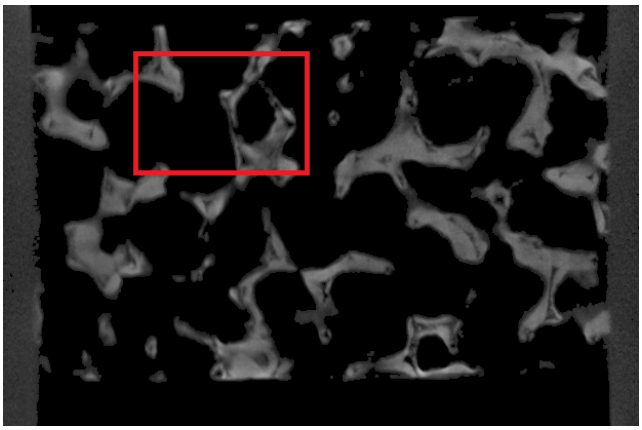
(b) Unloaded (c) 2% (d) 4% (e) 6% (f) 10%

Figure 6.23: Densification of MMC sample. (a) shows the original image with an excerpt marked by a red box. (b)-(f) show how the extracted field of view develops during the whole experiment.

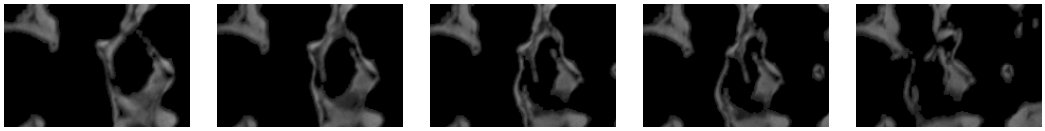
Due to the similarity of the material we use the same parameters as in Subsection 6.2.5. The sample is compressed in z-direction, we therefore calculate the strain and investigate its $(3,3)$-component, i.e. the derivative of w in z-direction. Slicewise investigation indeed brings two interesting positions to light. The respective slices of the strain can be found in Figure 6.22. Figure 6.23 now shows all corresponding positions throughout the time series. In fact, we can see two loose ends of struts colliding, which constitutes densification already at a very early stage.

The second position is even more interesting. Figure 6.24 shows the evolution in all slices, and in fact we can see that we found a breaking strut. We can even see that until the stage of 10% compression the strut is nearly completely fractured, whereas the authors of the original work struggled to observe real *cracking* at this stage at all.

We therefore conclude this chapter on evaluation of our methods by showing that calculating voxelwise displacement indeed offers great possibilities in mapping the delicate behavior of materials such as foams or fibers.



(a) Densification



(b) Unloaded (c) 2% (d) 4% (e) 6% (f) 10%

Figure 6.24: Fracture of cell of MMC sample. ensification of MMC sample. (a) shows the original image with an excerpt marked by a red box. (b)-(f) show how the extracted field of view develops during the whole experiment.

Chapter 7

Conclusion and Future Work

In this thesis we investigated motion estimation algorithms to compute displacement in materials tests. Having constituted two areas which already established methods working on volume images, we used the theory of inverse problems in Banach spaces to unify them under this roof. Inverse problems is a framework to tackle illposedness – and motion estimation can be considered as an ill-posed problem. We try to derive a three dimensional displacement field from only scalar correspondences. However, we saw that formulating the theory in Hilbert spaces is not suitable to describe the outcome of a materials test. As we are interested in assessing the durability and performance of specimens under load, we expect fracture and cracking in our images. Displacement describing this behavior will for sure consist of discontinuities and jumps – a feature that cannot be achieved if we consider displacement in Hilbert spaces. Unfortunately, the literature on algorithms overcoming these problems in *three* dimensions is very sparse. Luckily, we showed that a remedy can be found by extending total variation regularized Optical Flow to three dimensions.

We then proceeded with a thorough evaluation of our proposed extension. Motion estimation is a highly nonlinear problem and in the Optical Flow community several ways have been developed to deal with these. In this thesis, we considered the Kačanov method and primal dual approaches in detail. Primal dual algorithms treat the regularization very accurately, but still need a linearization of the data term in contrast to Kačanov's method. As it "freezes" the nonlinearities, it treats them without linearization. Especially when assessing the displacement fields, this approach turned out to be advantageous. This made it the method of choice for a comparative study on artificial displacement *and* real-world examples, which came to the conclusion that Optical Flow based methods outperform all state-of-the-art methods in motion estimation for materials tests.

We also investigated the behavior of motion estimation algorithms if large displacement can be expected. As the primal dual approaches rely on linearization, which itself relies on the assumptions of small displacement, this matter had to be rechecked. We saw that increasing the number of warps is the way to go to guarantee sufficient results in all algorithms, but once more Kačanov's method requires much less computational demand.

Another method to overcome the limitations of small displacements is to employ a coarse-to-fine strategy. By replacing the common Gaussian or Laplacian filters by a multiresolution wavelet analysis, and by choosing morphological wavelets to do so, we employed a whole new paradigm in coarse-to-fine strategies. Instead of matching the delicate behavior that we expected in our concrete sample only on the finest scale, we used the property of morphological wavelets, that they preserve local minima over all scales, to estimate cracks in

concrete already on coarse levels.

Even further, if used on the full image resolution, our voxel exact approaches were able to detect fracture in foams already at very low compression rates. Where struts only cover few voxels and their breakage is a phenomenon of equally few voxels, precise estimation algorithms are required. We showed that our extension fulfills this.

Nonetheless, our investigation opened much room for future work. Optical Flow in 2D is one of the most tackled problems in Computer Vision. However, the algorithms that we used here are fairly old, as in the recent years a trend to use learning based approaches evolved. The complexity, that comes with deep-learning in more than two dimensions and the absence of reliable training data in 3D nevertheless prohibits the use of such approaches in our applications. But there is a trend evolving in learning the regularizers of ill-posed problems. Motion estimation in materials science might also benefit from this.

Furthermore, a very appealing property of DVC always was its direct connection to FEM simulations in materials science. Having seen that the functional spaces used in DVC may produce too smooth displacement, raises the question on how to compare our motion estimation to simulations. This problem has to be evaluated not only on a computational but also on a mathematical level.

Lastly, the new paradigm introduced by morphological wavelets leaves room for many improvements. As stated, the scheme is not unique in 3D and its anisotropic nature has to be investigated further. This also applies for motion estimation on compressed data: morphological wavelets at this point are a theoretical construct and have never been employed in any compression scheme. If one wants to fully exploit their potential, these investigations cannot be avoided.

Bibliography

[1] Adrien, J., Maire, E., Gimenez, N., and Sauvant-Moynot, V.: *Experimental study of the compression behaviour of syntactic foams by in situ X-ray tomography*. Acta Materialia, 55(5):1667–1679, 2007.

[2] Aggrawal, H.O., Andersen, M.S., and Modersitzki, J.: *An image registration framework for discontinuous mappings along cracks*. In *Biomedical Image Registration*, pp. 163–173. Springer, 2020.

[3] Ambrosio, L., Coscia, A., and Maso, G.D.: *Fine properties of functions with bounded deformation*. Archive for Rational Mechanics and Analysis, 139(3):201–238, 1997.

[4] Ambrosio, L. and Tortorelli, V.M.: *Approximation of functional depending on jumps by elliptic functional via t-convergence*. Communications on Pure and Applied Mathematics, 43:999–1036, 1990.

[5] Appell, J. and Väth, M.: *Elemente der Funktionalanalysis: Vektorräume, Operatoren und Fixpunktsätze*. Vieweg, 2005.

[6] Ashby, M.F.: *Materials Selection in Mechanical Design*. Pergamon, 1992.

[7] Aubert, G. and Kornprobst, P.: *Mathematical problems in image processing: partial differential equations and the calculus of variations*, vol. 147. Springer, 2002.

[8] Baker, S. and Matthews, I.: *Lucas-Kanade 20 years on: A unifying framework*. International Journal of Computer Vision, 56(1):221–255, 2004.

[9] Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., and Szeliski, R.: *A database and evaluation methodology for optical flow*. International Journal of Computer Vision, 92(1):1–31, 2011.

[10] Ballaschk, U., Berek, H., Ehinger, D., Aneziris, C.G., and Krüger, L.: *Mg-PSZ reinforced TRIP-steel MMCs – in situ CT investigations of cruciform samples under compressive loading*. Advanced Engineering Materials, 15(7):590–599, 2013.

[11] Bar-Kochba, E., Toyjanova, J., Andrews, E., Kim, K.S., and Franck, C.: *A fast iterative digital volume correlation algorithm for large deformations*. Experimental Mechanics, 55(1):261–274, 2015.

[12] Bay, B.K.: *Methods and applications of digital volume correlation*. Journal of Strain Analysis for Engineering Design, 43:745–760, 2008.

[13] Bay, B.K., Smith, T.S., Fyhrie, D., and Saad, M.: *Digital volume correlation: three-dimensional strain mapping using X-ray tomography*. Experimental Mechanics, 39(3):217–226, 1999.

[14] Beg, M.F., Miller, M.I., Trouvé, A., and Younes, L.: *Computing large deformation metric mappings via geodesic flows of diffeomorphisms*. International Journal of Computer Vision, 61(2):139–157, 2005.

[15] Berek, H., Ballaschk, U., and Aneziris, C.G.: *In situ characterization of internal damage in TRIP-steel/Mg-PSZ composites under compressive stress using X-Ray computed tomography*. Advanced Engineering Materials, 13:1101–1107, 2011.

[16] Berek, H., Ballaschk, U., Aneziris, C.G., Losch, K., and Schladitz, K.: *The correlation of local deformation and stress-assisted local phase transformations in MMC foams*. Materials Characterization, 107:139–148, 2015.

[17] Betts, C., Balint, D., and Lin, J.: *In-situ micro-tensile testing and X-ray micro-tomography based FE modeling of open-cell metal foam struts and sandwich panels*. Procedia Materials Science, 4:197–202, 2014.

[18] Bolliger, S.A., Oesterhelweg, L., Spendlove, D., Ross, S., and Thali, M.J.: *Is differentiation of frequently encountered foreign bodies in corpses possible by Hounsfield density measurement?* Journal of Forensic Sciences, 54(5):1119–1122, 2009.

[19] Borstnar, G., Gillard, F., Mavrogordato, M., Sinclair, I., and Spearing, S.: *Three-dimensional deformation mapping of mode i interlaminar crack extension in particle-toughened interlayers*. Acta Materialia, 103:63–70, 2016.

[20] Bouterf, A., Adrien, J., Maire, E., Brajer, X., Hild, F., and Roux, S.: *Failure mechanisms of plasterboard in nail pull test determined by X-ray microtomography and digital volume correlation*. Experimental Mechanics, 56(8):1427–1437, 2016.

[21] Bouterf, A., Adrien, J., Maire, E., Brajer, X., Hild, F., and Roux, S.: *Identification of the crushing behavior of brittle foam: From indentation to oedometric tests*. Journal of the Mechanics and Physics of Solids, 98:181–200, 2017.

[22] Bredies, K. and Holler, M.: *Regularization of linear inverse problems with total generalized variation*. Journal of Inverse and Ill-posed Problems, 22(6):871–913, 2014.

[23] Bredies, K., Kunisch, K., and Pock, T.: *Total generalized variation*. SIAM Journal on Imaging Sciences, 3(3):492–526, 2010.

[24] Broit, C.: *Optimal registration of deformed images*. University of Pennsylvania, 1981.

[25] Brox, T., Bruhn, A., Papenberg, N., and Weickert, J.: *High accuracy optical flow estimation based on a theory for warping*. In *European Conference on Computer Vision*, pp. 25–36. Springer, 2004.

[26] Bruhn, A., Weickert, J., and Schnörr, C.: *Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods*. International Journal of Computer Vision, 61(3):211–231, 2005.

[27] Buljac, A., Jailin, C., Mendoza, A., Neggers, J., Taillandier-Thomas, T., Bouterf, A., Smaniotto, B., Hild, F., and Roux, S.: *Digital volume correlation: review of progress and challenges*. Experimental Mechanics, 58(5):661–708, 2018.

[28] Burger, M., Modersitzki, J., and Ruthotto, L.: *A hyperelastic regularization energy for image registration*. SIAM Journal on Scientific Computing, 35(1):B132–B148, 2013.

[29] Burger, M. and Osher, S.: *A guide to the TV zoo*. In *Level set and PDE based reconstruction methods in imaging*, pp. 1–70. Springer, 2013.

[30] Chambolle, A., Caselles, V., Cremers, D., Novaga, M., and Pock, T.: *An introduction to total variation for image analysis*. Theoretical Foundations and Numerical Methods for Sparse Recovery, 9(263-340):227, 2010.

[31] Chambolle, A. and Pock, T.: *An introduction to continuous optimization for imaging*. Acta Numerica, 25:161–319, 2016.

[32] Chateau, C., Nguyen, T.T., Bornert, M., and Yvonnet, J.: *DVC-based image subtraction to detect microcracking in lightweight concrete*. Strain, 54, 2018.

[33] Christensen, G.E., Rabbitt, R.D., and Miller, M.I.: *3D brain mapping using a deformable neuroanatomy*. Physics in Medicine & Biology, 39(3):609, 1994.

[34] Cohen, I.: *Nonlinear variational method for optical flow computation*. In *Eighth Scandinavian Conference on Image Analysis*, pp. 523–530, 1993.

[35] Cormack, A.M.: *Representation of a function by its line integrals, with some radiological applications*. Journal of Applied Physics, 34(9):2722–2727, 1963.

[36] Daubechies, I.: *Ten lectures on wavelets*. SIAM, 1992.

[37] De Zeeuw, P.: *A toolbox for the lifting scheme on quincunx grids (lisq)*. 2002.

[38] Ekeland, I. and Temam, R.: *Convex analysis and variational problems*. North-Holland American Elsevier, 1976.

[39] Elliott, J. and Dover, S.: *X-ray microtomography*. Journal of Microscopy, 126(2):211–213, 1982.

[40] Engl, H.W., Hanke, M., and Neubauer, A.: *Regularization of inverse problems*, vol. 375. Springer, 1996.

[41] Fu, Y., Lei, Y., Wang, T., Curran, W.J., Liu, T., and Yang, X.: *Deep learning in medical image registration: a review*. Physics in Medicine & Biology, 65(20):20TR01, 2020.

[42] Garcea, S., Wang, Y., and Withers, P.: *X-ray computed tomography of polymer composites*. Composites Science and Technology, 156:305–319, 2018.

[43] Gates, M., Lambros, J., and Heath, M.T.: *Towards high performance digital volume correlation*. Experimental Mechanics, 51:491–507, 2011.

[44] Goethals, W., Bultreys, T., Manigrasso, Z., Mascini, A., Aelterman, J., and Boone, M.N.: *Dynamic CT reconstruction with improved temporal resolution for scanning of fluid flow in porous media*. Water Resources Research, 58(4):e2021WR031365, 2022.

[45] Goshtasby, A.A.: *Image registration: Principles, tools and methods*. Springer, 2012.

[46] Grediac, M., Pierron, F., Avril, S., and Toussaint, E.: *The virtual fields method for extracting constitutive parameters from full-field measurements: a review*. Strain, 42(4):233–253, 2006.

[47] Grimm-Strele, H., Kabel, M., Andrä, H., Staub, S., Lienhard, J., Schweiger, T., and Herd, O.: *Efficient characterization and modelling of the nonlinear behaviour of LFT for crash simulations*. In *14th WCCM-ECCOMAS Congress 2020*, vol. 1200, 2021.

[48] Gumbsch, P. and Schöbel, A.: *Effiziente Charakterisierung und Modellierung des anisotropen Versagensverhaltens von LFT für Crashsimulation*. FAT-Schriftenreihe, (331), 2020.

[49] Hafner, D., Demetz, O., and Weickert, J.: *Why is the census transform good for robust optic flow computation?* In *Scale Space and Variational Methods in Computer Vision*, pp. 210–221. Springer, 2013.

[50] Hahn, B., Kienle-Garrido, M., Klingenberg, C., and Warnecke, S.: *Using the Navier-Cauchy equation for motion estimation in dynamic imaging*, 2020. `https://arxiv.org/abs/2009.04212`.

[51] Hajnal, J.V. and Hill, D.L.: *Medical image registration*. CRC press, 2001.

[52] Hanke, M.: *A taste of inverse problems: basic theory and examples*. SIAM, 2017.

[53] Hanke-Bourgeois, M.: *Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens*. Springer, 2009.

[54] Heid, P. and Wihler, T.P.: *A modified Kačanov iteration scheme with application to quasilinear diffusion models.* Mathematical Modelling & Numerical Analysis, 56(2), 2022.

[55] Heijmans, H.J. and Goutsias, J.: *Nonlinear multiresolution signal decomposition schemes. ii. morphological wavelets*. IEEE Transactions on Image Processing, 9(11):1897–1913, 2000.

[56] Hermann, S. and Werner, R.: *High accuracy optical flow for 3D medical image registration using the census cost function*. In *PSIVT 2013*, pp. 23–35. Springer, 2013.

[57] Hermann, S. and Werner, R.: *TV-L_1-based 3D medical image registration with the census cost function*. In *Pacific-Rim Symposium on Image and Video Technology*, pp. 149–161. Springer, 2013.

[58] Horn, B. and Schunck, B.: *Determining optical flow*. Artificial Intelligence, 17(1):185–203, 1981.

[59] Hounsfield, G.N.: *Computerized transverse axial scanning (tomography): Part 1. description of system*. The British Journal of Radiology, 46(552):1016–1022, 1973.

[60] Hubálková, J., Voigt, C., Schmidt, A., Moritz, K., and Aneziris, C.: *Comparative phenomenological study of fracture behavior of ceramic and glass foams under compressive stress using in situ X-ray microtomography*. Advanced Engineering Materials, 19(9):1700286, 2017.

[61] Hufenbach, W., Böhm, R., Gude, M., Berthel, M., Hornig, A., Ručevskis, S., and Andrich, M.: *A test device for damage characterisation of composites based on in situ computed tomography*. Composites Science and Technology, 72(12):1361–1367, 2012.

[62] Jalalzai, K.: *Some remarks on the staircasing phenomenon in total variation-based image denoising*. Journal of Mathematical Imaging and Vision, 54(2):256–268, 2016.

[63] Jiroušek, O., Jandejsek, I., and Vavřík, D.: *Evaluation of strain field in microstructures using micro-CT and digital volume correlation*. Journal of Instrumentation, 6(01):C01039, 2011.

[64] Johansson, S., Engqvist, J., Tryding, J., and Hall, S.: *3D strain field evolution and failure mechanisms in anisotropic paperboard*. Experimental Mechanics, 61(3):581–608, 2021.

[65] Jung, A.: *Offenporige, nanobeschichtete Hybrid-Metallschäume: Herstellung und mechanische Eigenschaften*. 2011.

[66] Kirsch, A.: *An introduction to the mathematical theory of inverse problems*. Springer, 1996.

[67] Landis, E. and Bolander, J.E.: *Explicit representation of physical processes in concrete fracture*. Journal of Physics D: Applied Physics, 42:214002, 2009.

[68] Le Gall, D. and Tabatabai, A.: *Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques*. In *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, pp. 761–764. IEEE, 1988.

[69] Leclerc, H., Périé, J.N., Hild, F., and Roux, S.: *Digital volume correlation: what are the limits to the spatial resolution?* Mechanics & Industry, 13(6):361–371, 2012.

[70] Leclerc, H., Périé, J.N., Roux, S., and Hild, F.: *Voxel-scale digital volume correlation*. Experimental Mechanics, 51(4):479–490, 2011.

[71] Liu, L. and Morgan, E.F.: *Accuracy and precision of digital volume correlation in quantifying displacements and strains in trabecular bone*. Journal of Biomechanics, 40(15):3516–3520, 2007.

[72] Losch, K., Schladitz, K., Ballaschk, U., Berek, H., and Aneziris, C.G.: *Interrupted in-situ compressive deformation experiments on MMC foams in an XCT: Experiments and estimation of displacement fields*. Image Analysis & Stereology, 33(2):131–145, 2014.

[73] Lucas, B.D. and Kanade, T.: *An iterative image registration technique with an application to stereo vision*. In *International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

[74] Maire, E. and Withers, P.J.: *Quantitative X-ray tomography*. International Materials Reviews, 59(1):1–43, 2014.

[75] Manandhar, S., Bouthemy, P., Welf, E., Roudot, P., and Kervrann, C.: *3D optical flow estimation combining 3D census signature and total variation regularization*. In *IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pp. 965–968, 2020.

[76] Mang, A., Gholami, A., and Biros, G.: *Distributed-memory large deformation diffeomorphic 3D image registration*. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 842–853. IEEE, 2016.

[77] Mang, A., Gholami, A., Davatzikos, C., and Biros, G.: *CLAIRE: a distributed-memory solver for constrained large deformation diffeomorphic image registration*. SIAM Journal on Scientific Computing, 41(5):C548–C584, 2019.

[78] Marigo, J.J., Maurini, C., and Pham, K.: *An overview of the modelling of fracture by gradient damage models*. Meccanica, 51(12):3107–3128, 2016.

[79] Marter, A.D., Dickinson, A.S., Pierron, F., and Browne, M.: *A practical procedure for measuring the stiffness of foam like materials*. Experimental Techniques, 42(4):439–452, 2018.

[80] Martyniuk, K., Sørensen, B.F., Modregger, P., and Lauridsen, E.M.: *3D in situ observations of glass fibre/matrix interfacial debonding*. Composites Part A: Applied Science and Manufacturing, 55:63–73, 2013.

[81] Maryamh, K., Hauch, K., Redenbach, C., and Schnell, J.: *Influence of production parameters on the fiber geometry and the mechanical behavior of ultra high performance fiber-reinforced concrete*. Structural Concrete, 22(1):361–375, 2021.

[82] Mazars, V., Caty, O., Couégnat, G., Bouterf, A., Roux, S., Denneulin, S., Pailhès, J., and Vignoles, G.L.: *Damage investigation and modeling of 3D woven ceramic matrix composites from X-ray tomography in-situ tensile tests*. Acta Materialia, 140:130–139, 2017.

[83] Mendoza, A., Schneider, J., Parra, E., Obert, E., and Roux, S.: *Differentiating 3D textile composites: a novel field of application for digital volume correlation*. Composite Structures, 208:735–743, 2019.

[84] Meyer, Y.: *Oscillating patterns in image processing and nonlinear evolution equations: the fifteenth Dean Jacqueline B. Lewis memorial lectures*, vol. 22. American Mathematical Soc., 2001.

[85] Meyers, N.G. and Serrin, J.: $H = W$. National Academy of Sciences of the USA, 51:1055–56, 1964.

[86] Modersitzki, J.: *Numerical methods for image registration*. Oxford University Press on Demand, 2004.

[87] Newman, J. and Choo, B.S.: *Advanced concrete technology*. Elsevier, 2003.

[88] Nie, Z., Li, C., Liu, H., and Yang, X.: *Deformable image registration based on functions of bounded generalized deformation*. International Journal of Computer Vision, 129(5):1341–1358, 2021.

[89] Nie, Z. and Yang, X.: *Deformable image registration using functions of bounded deformation*. IEEE Transactions on Medical Imaging, 38(6):1488–1500, 2019.

[90] Ning, H., Lu, N., Hassen, A., Chawla, K., Selim, M., and Pillay, S.: *A review of long fibre thermoplastic (LFT) composites*. International Materials Reviews, 65(3):164–188, 2020.

[91] Nogatz, T., Redenbach, C., and Schladitz, K.: *3D optical flow for large CT data of materials microstructures*. Strain, 58(3):e12412, 2022.

[92] Oliveira, F.P. and Tavares, J.M.R.: *Medical image registration: a review*. Computer Methods in Biomechanics and Biomedical Engineering, 17(2):73–93, 2014.

[93] Pan, B. and Wang, B.: *A flexible and accurate digital volume correlation method applicable to high-resolution volumetric images*. Measurement Science and Technology, 28(10):105007, sep 2017.

[94] Pan, B., Wang, B., Wu, D., and Lubineau, G.: *An efficient and accurate 3D displacements tracking strategy for digital volume correlation*. Optics and Lasers in Engineering, 58:126–135, 2014, ISSN 0143-8166.

[95] Pan, B., Wu, D., and Wang, Z.: *Internal displacement and strain measurement using digital volume correlation: a least-squares framework*. Measurement Science and Technology, 23(4):045002, 2012.

[96] Polzin, T.: *Large Deformation Diffeomorphic Metric Mappings: Theory, Numerics, and Applications*. PhD thesis, University of Lübeck, 2018.

[97] Polzin, T., Rühaak, J., Werner, R., Strehlow, J., Heldmann, S., Handels, H., and Modersitzki, J.: *Combining automatic landmark detection and variational methods for lung CT registration*. In *5th PIA 2013*, pp. 85–96, Nagoya, Japan, 2013. CreateSpace.

[98] Radon, J.: *Über die Bestimmung von Funktionen längs gewisser Mannigfaltigkeiten. Sächsische Gesellschaft der Wissenschaften math.* Phys. Klasse, Leipzig, 69:262–277, 1917.

[99] Réthoré, J., Tinnes, J.P., Roux, S., Buffière, J.Y., and Hild, F.: *Extended three-dimensional digital image correlation (X3D-DIC)*. Comptes Rendus Mécanique, 336(8):643–649, 2008.

[100] Rockafellar, R.T.: *Convex analysis*. Princeton university press, 2015.

[101] Rohr, K., Stiehl, H., Sprengel, R., Buzug, T., Weese, J., and Kuhn, M.: *Landmark-based elastic registration using approximating thin-plate splines*. IEEE Transactions on Medical Imaging, 20(6):526–534, 2001.

[102] Rolland, H., Saintier, N., and Robert, G.: *Damage mechanisms in short glass fibre reinforced thermoplastic during in situ microtomography tensile tests*. Composites Part B: Engineering, 90:365–377, 2016.

[103] Rolland, H., Saintier, N., Wilson, P., Merzeau, J., and Robert, G.: *In situ X-ray tomography investigation on damage mechanisms in short glass fibre reinforced thermoplastics: Effects of fibre orientation and relative humidity*. Composites Part B: Engineering, 109:170–186, 2017.

[104] Roux, S., Hild, F., Viot, P., and Bernard, D.: *Three-dimensional image correlation from X-ray computed tomography of solid foam*. Composites Part A: Applied science and manufacturing, 39(8):1253–1265, 2008.

[105] Rudin, L.I., Osher, S., and Fatemi, E.: *Nonlinear total variation based noise removal algorithms*. Physica D, 60:259–268, 1992.

[106] Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., and Lenzen, F.: *Variational methods in imaging*. 2009.

[107] Schneider, M., Ospald, F., and Kabel, M.: *Computational homogenization of elasticity on a staggered grid*. International Journal for Numerical Methods in Engineering, 105(9):693–720, 2016.

[108] Schöttl, L., Kolb, P., Liebig, W.V., Weidenmann, K.A., Inal, K., and Elsner, P.: *Crack characterization of discontinuous fiber-reinforced composites by using micro-computed tomography: Cyclic in-situ testing, crack segmentation and crack volume fraction*. Composites Communications, 21:100384, 2020.

[109] Scott, A., Mavrogordato, M., Wright, P., Sinclair, I., and Spearing, S.: *In situ fibre fracture measurement in carbon-epoxy laminates using high resolution computed tomography*. Composites Science and Technology, 71:1471–1477, 2011.

[110] Sket, F., Rodríguez-Hortalá, M., Molina-Aldareguía, J., Llorca, J., Maire, E., and Requena, G.: *In situ tomographic investigation of damage development in±45° carbon fibre reinforced laminates*. Materials Science and Technology, 31(5):587–593, 2015.

[111] Smith, T.S., Bay, B.K., and Rashid, M.M.: *Digital volume correlation including rotational degrees of freedom during minimization*. Experimental Mechanics, 42(3):272–278, 2002.

[112] Stock, S.: *X-ray microtomography of materials*. International Materials Reviews, 44(4):141–164, 1999.

[113] Stock, S., Guvenilir, A., Breunig, T., Kinney, J., and Nichols, M.: *Computed tomography part iii: Volumetric, high-resolution x-ray analysis of fatigue crack closure*. JOM, 47:19–23, 1995.

[114] Sweldens, W.: *The lifting scheme: A construction of second generation wavelets*. SIAM Journal on Mathematical Analysis, 29(2):511–546, 1998.

[115] Syrakos, A., Varchanis, S., Dimakopoulos, Y., Goulas, A., and Tsamopoulos, J.: *A critical analysis of some popular methods for the discretisation of the gradient operator in finite volume methods*. Physics of Fluids, 29(12):127103, 2017.

[116] Szeliski, R.: *Computer vision: algorithms and applications*. Springer, 2010.

[117] Temam, R. and Strang, G.: *Functions of bounded deformation*. Archive for Rational Mechanics and Analysis, 75(1):7–21, 1980.

[118] Tröltzsch, F.: *Optimal control of partial differential equations: theory, methods, and applications*, vol. 112. American Mathematical Soc., 2010.

[119] Trtik, P., Stähli, P., Landis, E., Stampanoni, M., and Van Mier, J.: *Microtensile testing and 3D imaging of hydrated Portland cement*. In *Proceedings 6th international conference on fracture mechanics of concrete and concrete structures (FraMCoS-VI). London: Taylor & Francis*, pp. 1277–1282, 2007.

[120] Wang, B., Pan, B., and Lubineau, G.: *Morphological evolution and internal strain mapping of pomelo peel using X-ray computed tomography and digital volume correlation*. Materials & Design, 137:305–315, 2018.

[121] Wang, L., Frost, J., Voyiadjis, G., and Harman, T.: *Quantification of damage parameters using X-ray tomography images*. Mechanics of Materials, 35(8):777–790, 2003.

[122] Wang, Y., Burnett, T.L., Chai, Y., Soutis, C., Hogg, P.J., and Withers, P.J.: *X-ray computed tomography study of kink bands in unidirectional composites*. Composite Structures, 160:917–924, 2017.

[123] Wedel, A., Pock, T., Zach, C., Bischof, H., and Cremers, D.: *An improved algorithm for TV-L_1 optical flow*. In *Statistical and geometrical approaches to visual motion analysis*, pp. 23–45. Springer, 2009.

[124] Weickert, J., Heers, J., Schnörr, C., Zuiderveld, K.J., Scherzer, O., and Stiehl, H.S.: *Fast parallel algorithms for a broad class of nonlinear variational diffusion approaches*. Real-Time Imaging, 7(1):31–45, 2001.

[125] West, J.B., Fitzpatrick, J.M., Wang, M.Y., Dawant, B.M., Maurer Jr., C.R., Kessler, R.M., Maciunas, R.J., Barillot, C., Lemoine, D., Collignon, A.M., *et al.*: *Comparison and evaluation of retrospective intermodality image registration techniques*. In *Medical Imaging 1996: Image Processing*, vol. 2710, pp. 332–347. SPIE, 1996.

[126] Yang, J., Hazlett, L., Landauer, A., and Franck, C.: *Augmented Lagrangian digital volume correlation (ALDVC)*. Experimental Mechanics, 60:1205–1223, 2020.

[127] Yang, Z., Ren, W., Sharma, R., McDonald, S., Mostafavi, M., Vertyagina, Y., and Marrow, T.: *In-situ X-ray computed tomography characterisation of 3D fracture evolution and image-based numerical homogenisation of concrete*. Cement and Concrete Composites, 75:74–83, 2017.

[128] Yermukhambetova, A., Tan, C., Daemi, S.R., Bakenov, Z., Darr, J.A., Brett, D.J., and Shearing, P.R.: *Exploring 3D microstructural evolution in Li-Sulfur battery electrodes using in-situ X-ray tomography*. Scientific Reports, 6(1):1–9, 2016.

[129] Zienkiewicz, O.C., Taylor, R.L., and Zhu, J.Z.: *The finite element method: its basis and fundamentals*. Elsevier, 2005.

Appendix A

List of Publications

- **Tessa Nogatz**, Claudia Redenbach, Katja Schladitz, 3D optical flow for large CT data of materials microstructures, *Strain*, 2022, 58(3), e12412. https://doi.org/10.1111/str.12412
- **Tessa Kuschnerus**, Claudia Redenbach, Katja Schladitz, Thomas Wagner, Friedrich Zerling, In-Situ testing with diffeomorphic mapping, July 2019, International Conference on Tomography of Materials, Cairns, Australia

Appendix B

Curriculum Vitae

SINCE 09/2021	Scientific Assistant at Mathematics Department, TU Kaiserslautern
03/2021 – 08/2021	Scientific Assistant at Fraunhofer ITWM, Image Processing Department
10/2018 – 02/2021	Scholar at Fraunhofer ITWM, Image Processing Department
SINCE 10/2018	PhD in Mathematics at Fraunhofer ITWM, Image Processing Department, and Kaiserslautern University of Technology, Germany
04/2016 – 09/2018	M.Sc. in Mathematics, University of Würzburg, Germany
10/2015 – 03/2016	M.Sc. in Mathematics, Technical University of Berlin, Germany. Change to the University of Würzburg at summer term 2016
10/2011 – 09/2015	B.Sc. in Mathematics, Technical University of Berlin, Germany

Akademischer Werdegang

SEIT 09/2021	Wissenschaftliche Mitarbeiterin im Fachbereich Mathematik, TU Kaiserslautern
03/2021 – 08/2021	Wissenschaftliche Mitarbeiterin am Fraunhofer ITWM, Abteilung Bildverarbeitung
10/2018 – 02/2021	Stipendiatin am Fraunhofer ITWM , Abteilung Bildverarbeitung
SEIT 10/2018	Doktorandin am Fraunhofer ITWM, Abteilung Bildverarbeitung, und Technischer Universität Kaiserslautern, Deutschland
04/2016 – 09/2018	M.Sc. in Mathematik, Universität Würzburg, Deutschland
10/2015 – 03/2016	M.Sc. in Mathematik, Technische Universität Berlin, Deutschland. Wechsel an die Universität Würzburg zum Sommersemester 2016
10/2011 – 09/2015	B.Sc. in Mathematik, Technische Universität Berlin, Deutschland