René Ranftl

Higher-Order Variational Methods for Dense Correspondence Problems

DOCTORAL THESIS

to achieve the university degree of
Doktor der technischen Wissenschaften

submitted to

Graz University of Technology

Supervisor

Prof. Dr. Thomas Pock
Institute for Computer Graphics and Vision
Graz University of Technology, Austria

Prof. Dr.-Ing. Andrés Bruhn
Institute for Visualization and Interactive Systems
University of Stuttgart, Germany

Graz, Austria, Sep. 2014

TO ANNA

The more I see, the less I know for sure.

John Lennon

Abstract

Dense correspondence problems, like stereo and optical flow estimation, are among the most fundamental low-level vision problems. Accurate and robust solutions of dense correspondence problems are important for many high-level tasks like navigation, video editing or scene reconstruction. Among the most successful approaches to solve dense correspondence problems are variational methods, which pose the task of finding correspondences between images as an energy minimization problem.

The success of any variational method is governed by the form of the energy, which models a prior assumption on likely solutions as well as some means to measure the quality of the solution given the observed data. In this thesis we introduce higher-order variational methods based on Total Generalized Variation. The proposed methods model the assumption of a piecewise planar world in the case of stereo and of piecewise affine motion in the case of optical flow. We show experimentally that this assumption leads to accurate results, especially in scenes depicting man-made structures. We extend the basic model using an anisotropic diffusion tensor, which allows to condition the model on image content. This leads to increased robustness and well-localized depth and motion discontinuities. Next, we introduce a non-local variant of Total Generalized Variation. This formulation can be used to incorporate soft-segmentation cues in order to model the assumption that perceptually similar regions undergo similar motion or lie on the same physical surface. Our experiments show that this approach is able to further enhance the results with respect to robustness and accuracy.

We show that non-parametric image transformations can be used in conjunction with the proposed higher-order priors in a convex optimization framework to build state-of-the-art optical flow and stereo algorithms. Specifically, extensive experiments and comparisons to other commonly used matching terms show that the Census transform serves as a robust matching term which is invariant to illumination changes, a common source of error in dense correspondence problems. For the task of optical flow estimation we propose an extension of the Census matching term, which tackles the problem of scale changes between consecutive frames. Scale changes are prevalent in cameras moving along their optical axis. This is a common pattern of movement for cameras mounted on a vehicle and can pose significant challenges for methods which do not account for this effect.

For stereo estimation we introduce a novel optimization procedure, which exploits the inherent structure of the stereo problem together with the properties of Total Generalized Variation. The approach reduces optimization of the non-convex stereo model to an alternating sequence of convex problems, which can be solved to global optimality. We show experimentally that this optimization approach is able to significantly enhance the accuracy, without changing the underlying model.

Keywords. Variational methods, Stereo, Motion, Optical flow, Total Generalized Variation

Kurzfassung

Dichte Korrespondenzprobleme, wie zum Beispiel die Schätzung von Tiefendaten und optischem Fluss, gehören zu den fundamentalen Problemen im Bereich des maschinellen Sehens. Akkurate und robuste Lösungen von dichten Korrespondenzproblemen sind wichtig für übergeordnete Probleme, zum Beispiel Navigation, das Editieren von Videos oder die Rekonstruktion einer Szene. Zu den erfolgreichsten Methoden zur Lösung von Korrespondenzproblemen gehören Variationsmethoden, die das Finden von Korrespondenzen zwischen Bildern als Energieminimierungsproblem modellieren.

Der Erfolg jeder Variationsmethode hängt von der Form der Energie ab, die einerseits eine a-priori Annahme über wahrscheinliche Lösungen modelliert und andererseits die Qualität einer Lösung anhand der vorliegenden Daten misst. In dieser Arbeit präsentieren wir Variationsmethoden höherer Ordnung, die auf einer Generalisierung der totalen Variation beruhen. Die vorgestellten Methoden modellieren die Annahme einer stückweise planaren Welt im Falle von Tiefenschätzung und die Annahme von stückweise affiner Bewegung im Fall von optischem Fluss. Wir zeigen experimentell, dass diese Annahme speziell bei von Menschenhand geschaffenen Szenen zu akkuraten Ergebnissen führt. Wir erweitern das Grundmodell mittels eines anisotropen Diffusionstensors, der es erlaubt das Modell auf den Bildinhalt zu konditionieren. Dies führt zu erhöhter Robustheit und genau ausgerichteten Tiefen- und Bewegungskanten. Weiters, führen wir eine nicht-lokale Variante der generalisierten totalen Variation ein. Diese Formulierung erlaubt es Segmentierungshinweise in das Modell einzuführen. Dies erlaubt die Modellierung der Annahme, dass perzeptuel ähnliche Regionen einer ähnlichen Bewegung unterliegen oder zur selben physikalischen Oberfläche gehören. Unsere Experimente zeigen, dass dieser Ansatz zu einer weiteren Verbesserung der Ergebnisse im Bezug auf Robustheit und Genauigkeit führt.

Wir zeigen, dass nicht-parametrische Bildtransformationen zusammen mit den zuvor erwähnten a-priori Annahmen in einem konvexen Optimierungsframework verwendet

werden können, um moderne Algorithmen für die Schätzung von optischem Fluss und Tiefe zu entwickeln. Im Speziellen, zeigen umfassende Experimente und Vergleiche zu anderen Korrelationstermen, dass die Census-Transformation äußerst robust im Bezug auf Beleuchtungsunterschied zwischen Bildern ist. Dies ist eine typische Fehlerquelle in dichten Korrespondenzproblemen. Für die Schätzung des optischen Flusses führen wir eine Erweiterung der Census-Transformation ein, welche den Korrelationsterm robust im Bezug auf den Größenunterschied von Objekten in aufeinanderfolgenden Bildern macht. Größenunterschiede sind häufig in Bildern anzutreffen, die von einer Kamera aufgenommen wurde die sich entlang ihrer optischen Achse bewegt. Dies ist zum Beispiel der Fall für Kameras, die auf einem Fahrzeug montiert sind und kann zu erheblichen Schwierigkeiten in Methoden führen, die diesen Effekt nicht berücksichtigen.

Für die Aufgabe der Tiefenschätzung führen wir ein neues Optimierungsverfahren ein, dass die spezielle Struktur des Problems und die Eigenschaften der generalisierten totalen Variation nutzt. Dieser Ansatz reduziert die Optimierung des nicht-konvexen Stereomodells auf eine alternierende Sequenz von konvexen Problemen, die global optimal gelöst werden können. Unsere Experimente zeigen, dass dieses Optimierungsverfahren die Genauigkeit signifikant erhöt, ohne das zugrunde liegende Modell zu verändern.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

The text document uploaded to TUGRAZonline is identical to the presented doctoral thesis.

Place Date Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Dissertation identisch.

Ort Datum Unterschrift

Acknowledgments

Here I have the unique opportunity to thank all the people that made this thesis, which is the culmination of many years of work and learning, possible.

First and foremost I would like to thank Thomas Pock for supervising my PhD. This thesis would not have been possible without his guidance, patience and enthusiasm. I am forever grateful for this experience. I also like to thank Andrés Bruhn for agreeing to be my second supervisor.

I want to thank Stefan Heber with whom I shared my office in the last four years for many interesting discussions and the many experiences we shared in and outside of the office. My sincere thanks also go to all the other former and current members of our group. Since our formerly small group is not so small anymore, I want to specifically thank Manuel, Markus, Yunjin and Gottfried, with whom I had a lot of overlap, both in my time at this institute as well as in our research interests. I wish to thank all my colleagues at the ICG and the members of the Reading Group, which made my time here challenging and interesting, and allowed me to catch a glimpse of the many different problems and techniques encountered in the vast field of Image Processing and Computer Vision.

I want to thank my whole family and all my friends for their support and friendship. My educational career would not have been possible without the continuous and unconditional support of my parents and my brother. Thank you! Finally, my thanks go to Anna for her love and her ability to always brighten my day.

Contents

List of Figures

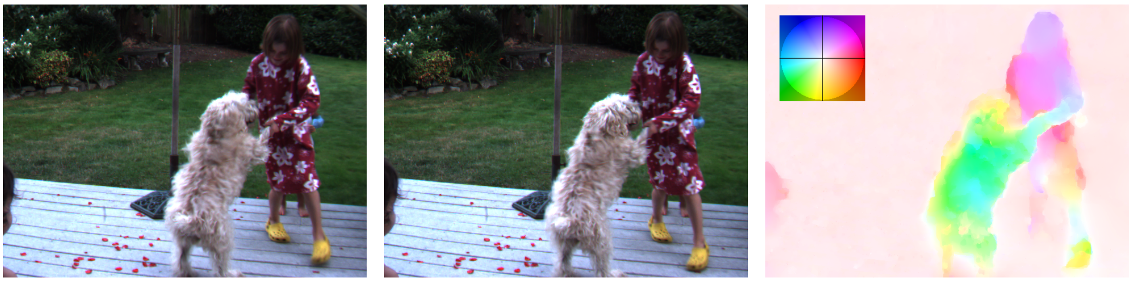
List of Tables

1 Introduction

Contents

Motion and depth perception are one of the fundamental building blocks of the human visual system. It is thus not surprising that they were early and intensively studied problems in Computer Vision. From an algorithmic point of view both motion and depth perception can be subsumed as the task of finding corresponding points between two or more images. In the case of depth estimation, the depth of a point can be recovered from the correspondence together with the geometry of the cameras used for capturing the images. In the case of motion estimation, correspondences give rise to optical flow [60], which is the motion of world points projected to the camera and is strongly related to object motion. Examples of both types of correspondence problems are shown in Figures 1.1 and 1.2.

Due to their fundamental similarity it is possible to view both problems as specific instances of a general correspondence problem. An important distinction between different types of correspondence problem is given by the categorization into *sparse* and *dense* correspondences. We talk about *sparse* correspondences if correspondences are given only for a small subset of image points, typically on the order of a few hundred to a few thousand points. In sparse correspondence problems these points are carefully selected based on the structure of their neighborhood, such that reliable correspondences can be retrieved even over large view-point variations. On the other hand, we talk about *dense* correspondences, which are the scope of this thesis, if an estimate of correspondence is given for a majority or all pixels in an image. Note that for some pixels there may not exist a corresponding pixel in the other image, due to effects like occlusion or disocclusion. The capability to provide dense estimates of correspondence is often a highly desirable property of correspondence



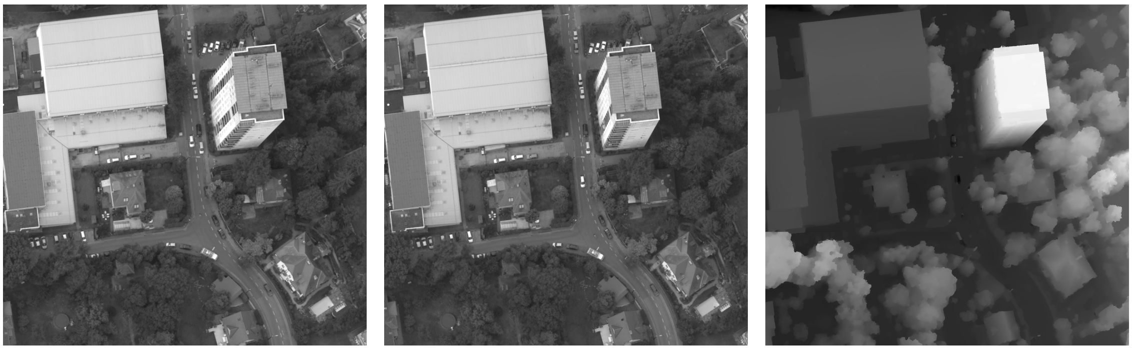
(a) First image **(b)** Second image **(c)** Optical Flow

Figure 1.1: Example of optical flow computed with one of the approaches proposed in this thesis. Two consecutive frames of a video show apparent motion. Dense optical flow is given by a two-dimensional vector that indicates the velocity of each image pixel. We use a color coding to visualize the two-dimensional flow vectors (upper left corner).

algorithms. A reliable pre-selection of sparse points is often not possible. Moreover, dense correspondences provide a richer source of information for subsequent higher-level algorithms. The majority of areas in natural images are not discriminative, since they contain little or no structure, which makes direct search for correspondence for such areas futile. Dense correspondence algorithms thus typically include, implicitly or explicitly, some form of prior assumption on the specific form of the correspondence field, which allows to resolve the ambiguities introduced by non-discriminative areas. Modeling this prior assumption in a sensible way will be a major topic of this thesis.

The visual system employs low-level cues such as motion and depth as a basic building block for many higher-level tasks. Perceptual studies indicate that motion is a fundamental cue for object recognition and grouping, especially in early developmental stages. Ostrovsky et al. [114] performed studies on congenitally blind adult individuals, which where surgically cured from their blindness. Subjects which where initially not able to recognize objects from static images consistently showed a tremendously better performance if the images were augmented with motion cues. Repeated experiments over the span of several months showed that the performance for static image recognition increased over time, which lends support to the hypothesis that motion is especially important in the early stages of development. Similar studies, carried out with infants, came to the conclusion that motion is instrumental especially for grouping localized shapes into coherent objects [2, 41, 86]. Additionally, researchers identified the key role of motion for a multitude of different everyday tasks, such as estimating the time to collision with a moving object, figure-ground segmentation, estimation of ego motion together with its importance for reliable self-locomotion and balance control as well as drawing focus to salient areas [86, 94, 108].

Similarly, depth perception is another important aspect of the human visual system. Depth cues are fundamental for tasks like navigation or grasping of objects. The fundamental mechanism for depth perception is binocular (stereo) vision, which allows to resolve



(a) First image **(b)** Second image **(c)** Disparity

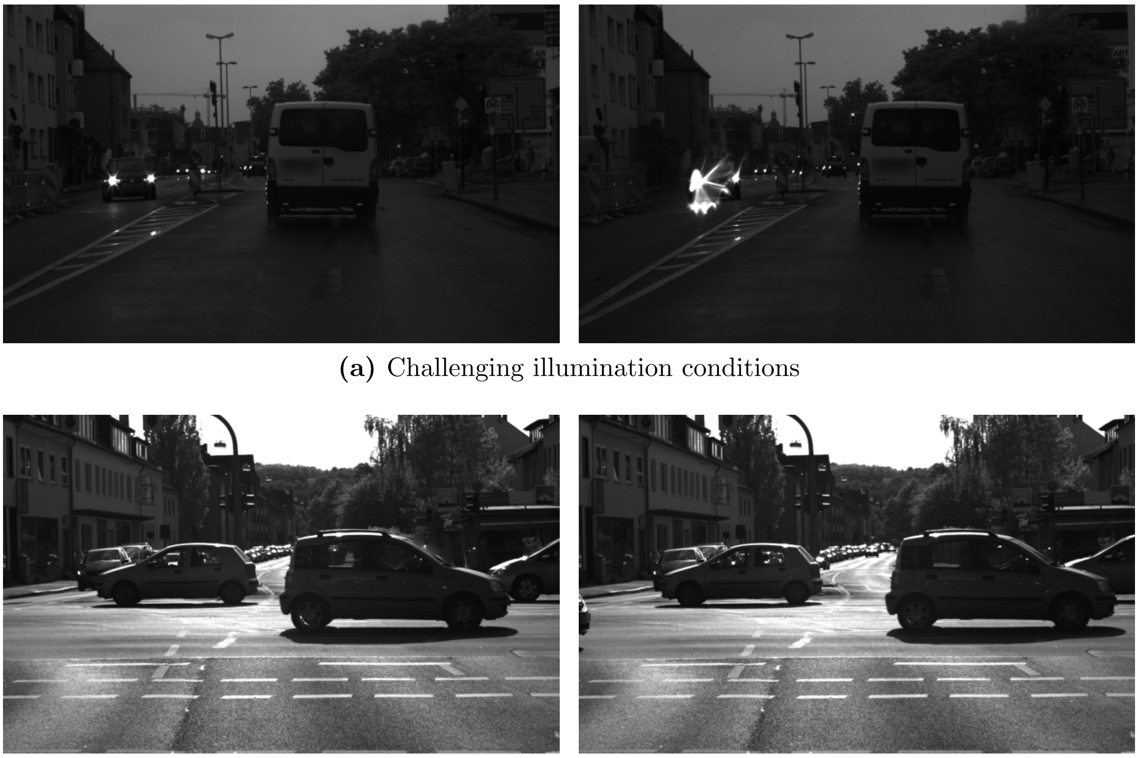
Figure 1.2: Example of stereo estimation on a pair of aerial images. Disparity is inversely proportional to depth. Near objects appear bright, whereas far objects are dark.

the depth of objects via their relative displacement in both views, the so-called parallax. The binocular setup will provide the basic setup for depth estimation in this thesis. Its main advantage is that the absolute depth can be recovered, provided that geometry of the binocular setup is known. This approach thus can be used to measure the absolute depth of an object relative to the observer, which provides rich information about a scene. Additionally to binocular cues, the visual system employs a multitude of other visual cues to estimate the depth of an object such as motion parallax, perspective, relative size or shading. While we do not address recovery of depth via these cues in this thesis, it is worth to note that the herein presented prior models are excellent models of depth and not inherently tied to the specific cues that are used for depth estimation. They can thus be easily incorporated in models that base their depth estimates on other cues.

Apart from their inherent biological motivation and relevance to high-level tasks, there are many useful applications of correspondence algorithms. Optical flow and motion estimation is an invaluable tool in video processing. Video editing tasks like artificial slow motion, frame interpolation, video stabilization and restoration of damaged frames [165], rely on optical flow estimation. These applications can tremendously benefit from robust and highly accurate optical flow, since even small errors in the optical flow can lead to distracting artifacts in the processed video sequences. Manual correction of such problems is a labor-intensive and thus costly task. The distribution of digital video, be it in the form of physical media, or as online streaming services, would not be possible without efficient video compression methods, which in turn rely on motion estimation algorithms. Similarly, stereo estimation and 3D reconstruction is an essential building block for movie production, both for the creation of 3D movies as well as for post-production purposes. On the frontier of what is currently possible, stereo and optical flow is currently finding its way into cars. Intelligent and possibly completely autonomous cars will help to reduce accidents and casualties in everyday traffic. One of the biggest hindrances for widespread deployment of such systems in cars are the robustness of currently available algorithms.

Autonomous systems have to be reliable in a multitude of adverse conditions, such as bad weather and difficult lighting conditions. Stereo and optical flow perception forms the backbone of autonomous vehicles, and while such systems have already shown impressive results [56, 179], they are still largely constraint to almost ideal conditions.

While correspondence problems are a much studied field, they are far from solved in the general setting. The problem poses inherent ambiguities, such as the difference between apparent pixel motion and true object motion, or more fundamental and low-level ambiguities, such as the aperture problem or challenging surface properties (cf. Figure 1.3). These ambiguities make the overall problem ill-posed (both on a conceptual and a mathematical level) and introduce the necessity of sophisticated models, in order to get well-defined and satisfactory solutions by an algorithm.



(a) Challenging illumination conditions

(b) Apparent motion vs. true object motion

Figure 1.3: Examples of challenges in correspondence problems. Images are part of the HCI Challenging Sequences Benchmark [102]. Top row: The overall lighting conditions, the wet, specular road and illumination artifacts from the headlights of the oncoming car pose significant challenges. Bottom row: The shadows of the car induce apparent motion, which does not correspond to true object motion.

1.1 The Relation of Stereo and Optical Flow

Both stereo estimation and optical flow can be modeled in a similar way. Both problems are dense correspondence problems, that is, the goal is to assign a correspondence between each pixel in two images. In optical flow the correspondence search space is two-dimensional, whereas in stereo the search space can be reduced to a one-dimensional space using the epipolar geometry of the cameras. In both cases, some form of prior assumption is needed in order to infer truly dense estimates, since a simple pixel- or patch-wise matching is typically non-informative for possibly a large number of pixels.

Stereo estimation can be treated as a simplified optical flow problem, which treats disparity as a continuous quantity. This approach somewhat differs from the classical approaches in the literature, which typically model disparity as a discrete quantity. The main advantages of this view are: (1) Very small disparities can be resolved, which, in the case of stereo, increases accuracy in the far-range. (2) We can employ efficient algorithms from convex optimization, in order to build real-time or near real-time stereo algorithms. Additionally, novel approaches from optical flow may be used in stereo algorithms and vice-versa. If disparities are modeled as a discrete quantity, it is not possible to directly transfer the results to optical flow, since the number of discrete labels explodes in this setting, due to the two-dimensional nature of the matching problem [62, 143]. Disadvantages are: (1) From a mathematical point of view, the matching of pixels or patches becomes more complex. The need to simplify (*i.e.* convexify) the matching term results in algorithms that may not be as good as discrete methods in recovering small details. (2) Direct modeling of occlusions, which are an inherent discrete quantity, becomes prohibitive.

We will also introduce a specialized model for stereo estimation, which is strongly related to discrete approaches, but allows for continuous valued disparity estimates. We will see that this approach can lead to higher accuracy than optical flow-based models.

1.2 A General Variational Model for Dense Correspondence

The general variational model for dense correspondence estimation consists of two parts: the matching term and the prior term.

The matching term, also called data term, relates the two input images using the estimated correspondences. The role of this term is to measure compatibility of the estimate with the evidence using photometric measures. The matching term is crucial: A bad matching term will never lead to good estimates, whereas a good matching term can minimize the need for additional prior assumptions in the model. A matching term should ideally be invariant to illumination changes and fast to evaluate. Moreover, in the case of optical flow, it should be robust to scale and rotation changes.

The prior term, also called regularization term, assigns for any possible estimate a likelihood of being a likely disparity map or motion field. This term allows to model expert knowledge of the task at hand. A simple example for a prior is the Total Variation

(TV), which assumes that solutions should be constant in most parts of the image, with a sparse set of jumps between these constant areas. Other priors enforce planarity or some form of stiffness. Priors may also be learned from a set of input-groundtruth pairs, but this approach is seldom used in optical flow and stereo (since planarity assumptions seem to be already satisfactory). A good prior should typically be easy to incorporate and solve (this is in the continuous case often synonymous with being convex), be able to model the task at hand to a satisfactory degree, while simultaneously excluding large parts of the solution space. A prior that has shown to work particularly well for dense correspondence problems is given by the Total Generalized Variation (TGV) [26]. TGV is a higher-order prior, which means that it goes beyond the classical assumption of piecewise constant solutions and for example allows for piecewise affine or piecewise quadratic solutions, which is often better suited to model the geometry of a scene or the composition of an optical flow field.

A last important point, which is not directly included in the modeling, is the question how to solve the model. Models should be efficiently solvable, in order to yield fast algorithms. In many cases this requires convexity of the model. Since optical flow and stereo are inherently non-convex problems, due to the matching term, it is important to construct convex surrogate problems in order to build efficient algorithms.

Formally, we can write a general variational model to find correspondences $u^* : \Omega \to U$ between images $I_1 : \Omega \to \Re^c$ and $I_2 : \Omega \to \Re^c$, where $\Omega \subset \Re^2$ denotes the image domain and c is the number of channels of the input images, as the following optimization problem:

$$u^* = \arg\min_u R(u; I_1) + D(u; I_1, I_2). \tag{1.1}$$

Here, we fixed I_1 as the designated reference image (*i.e.* we estimate correspondence as relative offset from the position in the reference image). This optimization problem consists of the two already mentioned parts: The regularization term $R(u; I_1)$, which imposes prior knowledge about desired solutions onto the problem. This term may also have a dependency on the reference image, which is able to give additional prior cues. An example are the probable location of jumps in the solution. The second term, $D(u; I_1, I_2)$ is the matching term, which relates the images I_1 and I_2 via the estimate u. Classical matching terms in some form or the other rely on the brightness constancy assumption.

Note that in the variational formulation we have $u : \Omega \to U$, *i.e.* u can be interpreted as a correspondence field. In the case of stereo we have $U \subseteq \Re$, whereas in the case of optical flow we have $U \subseteq \Re^2$.

For the remainder of this thesis we will be concerned with problems of the form (1.1). We will introduce different forms of the prior and the matching term, as well as efficient algorithms in order to find (approximate) minimizers u^*. All of these components are crucial for the overall performance of the estimation algorithms, both with respect to quality and with respect to speed.

1.3 Contributions and Outline

Correspondence problems can be tackled in many different ways, even while staying in the framework of global variational models. The main ingredients of any global variational method are: (1) The regularization term, which models prior knowledge about expected solutions. (2) The data term which gives a likelihood that the estimate is correct, given the observed data. (3) The optimization method, which is used to find minimizers of the model. Since the correspondence matching problem is inherently non-convex, efficient and good relaxations are needed to get satisfactory results for a given model. All of these components significantly influence the quality of the estimates. The first two components are related to problem modeling, while the last component asks the question on how to solve a given problem. This thesis presents contributions for all three components. We will show that careful modeling of both the regularization and the data term results in state-of-the-art stereo and optical flow methods. Moreover, wherever possible, we will show that choosing an appropriate optimization procedure that exploits the properties of the matching problem can significantly enhance the estimates of existing methods. The individual contributions presented in this thesis are:

Regularizers We introduce efficient higher-order regularizers based on TGV, which are well suited for modeling correspondence problems. We introduce a second-order regularizer which can be conditioned on the reference image in order to increase accuracy at motion boundaries and depth discontinuities. Moreover, we introduce a non-local second-order regularizer, called Non-Local Total Generalized Variation (NLTGV), which is able to incorporate strong prior cues and imposes smoothness on larger areas. This is especially advantageous for difficult scenes where the data term is uninformativ for large parts of the input images.

Data term We show that appropriate data terms are able to yield significantly better results than the classic brightness constancy assumption. In particular we discuss the Census transform in the context of general variational matching problems and show that this data term is well suited for complicated environments, due to its robustness to illumination changes. Moreover, we show how the Census matching term can be adapted to the special challenges that are inherent in optical flow estimation, where scale changes pose significant problems for typical patch-based data terms.

Optimization In the case of stereo matching problems we exploit the inherently one-dimensional structure and the ordering of depth values to reformulate the non-convex stereo matching problem as an alternating series of convex problems. Since each of these problems can be solved globally optimal, the overall accuracy of the method is enhanced. We compare this approach to the classical relaxation approaches based on coarse-to-fine warping and are able to show significantly better performance without changing the underlying model.

Outline The thesis is organized as follows. Chapter 2 gives an overview of basic concepts of convex optimization. We define convex sets and convex functions and discuss important related mathematical concepts. This is followed by an overview of the most important algorithms for convex optimization, where the focus lies on first-order methods, which are fast and well-suited for the problems which will be encountered later on. At the end of this chapter we introduce two classical variational models and show preliminary results of denoising and optical flow estimation.

Chapter 3 introduces the higher-order priors that will be used throughout the rest of the thesis. We introduce TGV and discuss several extension to this basic prior model, which are aimed at increasing the robustness and accuracy of the models. We further discuss other commonly encountered higher-order priors and show their qualitative differences using simple experiments.

Chapter 4 is concerned with optical flow estimation. We introduce novel robust data terms and show how they can be incorporated with the previously defined higher-order priors into state-of-the-art optical flow models. Experiments on two different benchmarks cast light on the strengths and weaknesses of the individual models.

In Chapter 5 we discuss the stereo estimation problem. We introduce the basic geometry of the stereo setup and show how depth can be recovered from correspondences. Different matching approaches are discussed and experimentally compared. We also introduce a novel approach for optimization of stereo models that incorporate a TGV prior and contrast it to the approach previously introduced in the chapter on optical flow estimation. We conclude this thesis in Chapter 6 and give a brief outlook on remaining challenges and future work.

This thesis is comprised of research which was presented in the publications [123, 124, 126]. A complete list of (co-)authored publications, including abstracts, can be found in Appendix B.

2

Convex Optimization and Variational Models in Image Processing

Contents

In this chapter we give a short overview on the field of convex analysis and convex optimization. Readers which are familiar with these topics may safely skip this chapter. Convex optimization nowadays is a mature topic, which is a de-facto standard tool in Computer Vision. This is due to the fact that many interesting problems can be modeled as, or at least approximated by, convex optimization problems. Convex analysis gives strong estimates on the properties of minimizers of such problems and convex optimization algorithms give tight theoretical bounds on the speed of convergence and thus on the quality of solutions. Most importantly they are very efficient in practical applications.

2.1 Convex Analysis

The topic of convex analysis is concerned with the characterization of convex functions and convex sets and as such forms the theoretical basis of convex optimization algorithms. A basic understanding of this foundations is of significant importance in order to apply and build convex optimization algorithms. We thus give a short self-contained overview of this topic here. A more comprehensive introduction can be found in [21, 110, 129].

2.1.1 Convex Sets

Definition 1. *A set $C \subseteq \Re^N$ is called convex, if for any two elements $x, y \in C$ and any scalar $\alpha \in [0, 1]$ the relation*

$$\alpha x + (1 - \alpha) y \in C \tag{2.1}$$

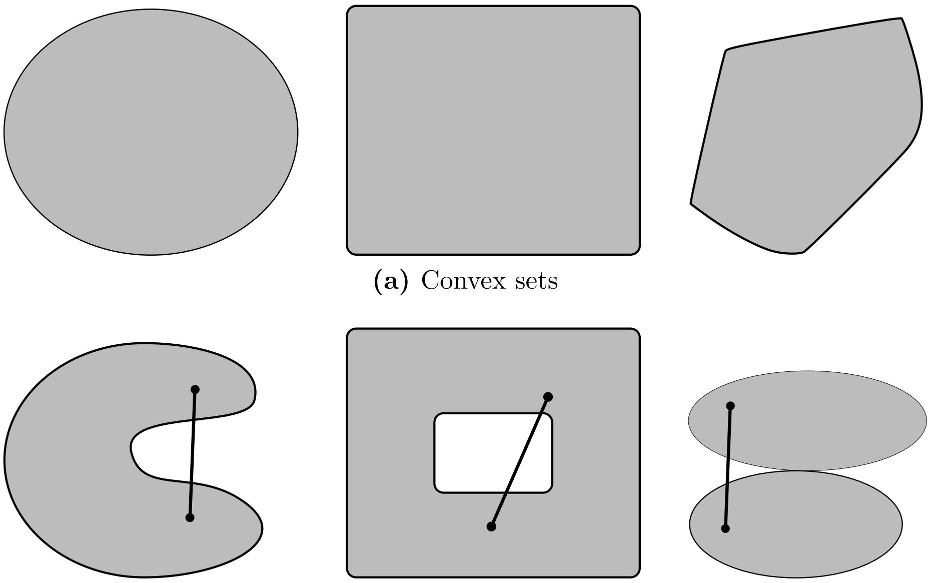
holds.

Definition 1 is a simple geometric characterization of a convex set: A set is convex if for any two points in the set, all points along the line segment joining the two points are also contained in the set. This notion can be generalized to more than two points, where it can be shown that any convex combination of points $x_n \in C$ is also contained in C:

$$\sum_n \lambda_n x_n \in C, \quad \lambda_n \geq 0. \tag{2.2}$$

Figure 2.1 shows examples of convex and non-convex sets. Note that the whole space $\Re^N$ is a convex set as is the empty set $\{\emptyset\}$. Some important transformations which preserve convexity are:

- The intersection of convex sets.
- The image of a convex set under affine transformations.



(a) Convex sets

(b) Non-convex sets

Figure 2.1: Example of convex and non-convex sets. For the non-convex examples, pairs of points x and y, which violate Definition 1, are shown.

- The Minkowski sum $C_1 + C_2 = \{x_1 + x_2 : x_1 \in C_1,\, x_2 \in C_2\}$.

An important subclass of convex sets are convex cones:

Definition 2. *A convex set $C \subset R^N$ is called a convex cone if for all $x_1, x_2 \in C$ and $\theta_1, \theta_2 \geq 0$, we have that*

$$\theta_1 x_1 + \theta_2 x_2 \in C. \tag{2.3}$$

Convex cones are an important concept in convex optimization. An example, the second-order cone, is shown in Figure 2.2. Most notably, specific instances of convex cones can be used to describe large subclasses of convex optimization problems, such as second-order cone programs or semi-definite programs, where the names imply the conic shape of the feasible regions of the programs.

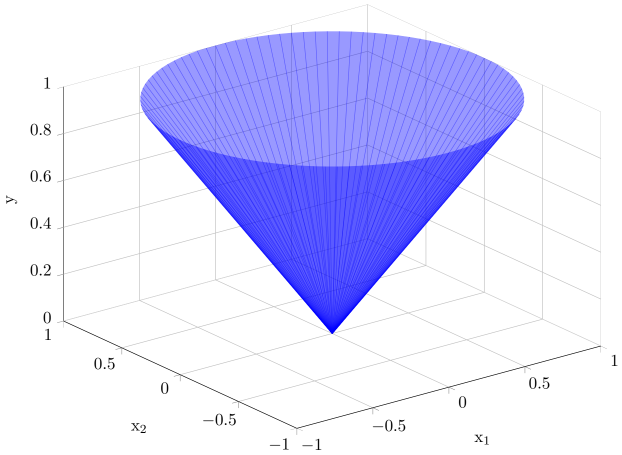


Figure 2.2: Shape of the second-order cone $K = \{(x, y) \in R^3 :\ ||x||_2 \leq y\}$. Note that the interior is part of the set K and that the cone actually extends infinitely upwards.

2.1.2 Convex Functions

Definition 3. *A function $f : C \to \Re$, with convex domain C is said to be convex, if the relation*

$$f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2) \tag{2.4}$$

holds for all $x_1, x_2 \in C$ and $\alpha \in [0, 1]$. Functions f for which inequality (2.4) *is strict are called strictly convex. A function $f(x)$ is said to be concave, if $-f(x)$ is convex.*

Definition 3 is again a geometric relation similar to the definition of convex sets: The line segment connecting any two points on the graph of a convex function lies on or above the graph of the function (or equivalently is contained in the epigraph of the function).

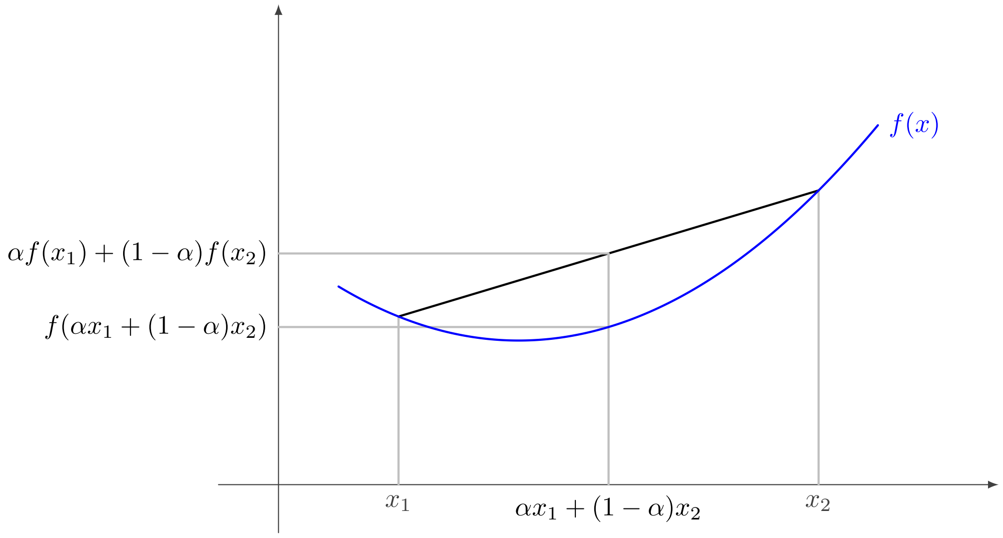


Figure 2.3: Example of a convex function. A line segment connecting any two points on the graph of the function, has to lie on or above the graph. Since in this example all points lie strictly above the graph, this function is also strictly convex.

Figure 2.3 gives a graphical example of this concept. With this geometric interpretation it is easy to see that affine functions are convex and concave.

For differentiable functions there is an equivalent gradient-based condition for convexity:

$$f(x_1) \geq f(x_2) + \langle \nabla f(x_2), x_1 - x_2 \rangle \,, \tag{2.5}$$

which needs to hold for all $x_1, x_2 \in C$, *i.e.* a convex function can be globally underestimated by its first-order approximation. An interesting fact directly follows from this relation: Consider reversing the order of the points x_1, x_2:

$$f(x_2) \geq f(x_1) + \langle \nabla f(x_1), x_2 - x_1 \rangle \,. \tag{2.6}$$

This equality still holds, since (2.5) does not assume any particular ordering of the points. Now adding (2.5) and (2.6) yields

$$\langle \nabla f(x_2) - \nabla f(x_1), x_2 - x_1 \rangle \geq 0, \tag{2.7}$$

This relation shows that the gradient of a convex function is monotone. In particular, for scalar functions it is easy to see that the gradient of a convex function necessarily is non-decreasing, and is strictly increasing if the function is strictly convex.

For non-differentiable functions the idea of characterization of a convex function by linear underestimators can be extend further by means of the so-called subdifferential.

The subdifferential at x_0 is the set of vectors v, for which the relation

$$f(x) \geq f(x_0) + \langle v, x - x_0 \rangle \tag{2.8}$$

holds for all $x \in C$. We formally write $v \in \partial f(x_0)$ for this set. For continuously differentiable convex functions the subdifferential includes exactly one element at every point x, namely the gradient. Likewise, the subdifferential reduces to the gradient for points of a convex function which are differentiable. Moreover, it can be shown that a function is convex if the subdifferential is non-emtpy everywhere, provided that its domain is convex. To see this pick any two points $x, y \in C$ and set $z = \alpha x + (1-\alpha)y \in C$, with $\alpha \in [0,1]$. For $v \in \partial f(z)$ it holds that

$$\begin{aligned} f(x) &\geq f(z) + \langle v, x - z \rangle = f(\alpha x + (1-\alpha)y) + (1-\alpha)\langle v, x - y \rangle \\ f(y) &\geq f(z) + \langle v, y - z \rangle = f(\alpha x + (1-\alpha)y) + \alpha \langle v, y - x \rangle . \end{aligned} \tag{2.9}$$

After dividing the first inequality by $1-\alpha$ and the second inequality by α, the sum of both inequalities yields

$$\alpha f(x) + (1-\alpha) f(y) \geq f(\alpha x + (1-\alpha)y), \tag{2.10}$$

which is exactly the condition for convexity given in Definition 3.

The subdifferential can be used to completely characterize the global minima of a convex function: $f(y) = \min_{x \in C} f(x)$ holds if and only if $0 \in \partial f(y)$, *i.e.* if $0 \in \partial f(y)$, we have

$$f(x) \geq f(y) + \langle 0, x - y \rangle = f(y) \Leftrightarrow f(y) \leq f(x) \quad \forall x \in C. \tag{2.11}$$

For twice continuously differential convex functions it holds that the Hessian is positive-semidefinite everywhere:

$$\nabla^2 f(x) \succeq 0, \qquad \forall x \in C. \tag{2.12}$$

The last condition often gives a simple practical test for the convexity of a given function, provided that its Hessian exists.

An important subclass of convex functions are strongly convex functions, for which the estimate

$$f(x_1) \geq f(x_2) + \langle \nabla f(x_2), x_1 - x_2 \rangle + \frac{m}{2} \left|\left| x_1 - x_2 \right|\right|^2, \quad m > 0 \tag{2.13}$$

holds. Twice continuously differential function have a Hessian whose smallest eigenvalue is equal to the modulus m and thus are positive-definite. Any strongly convex function is also strictly convex. Strong convexity gives additional structural knowledge of the functions,

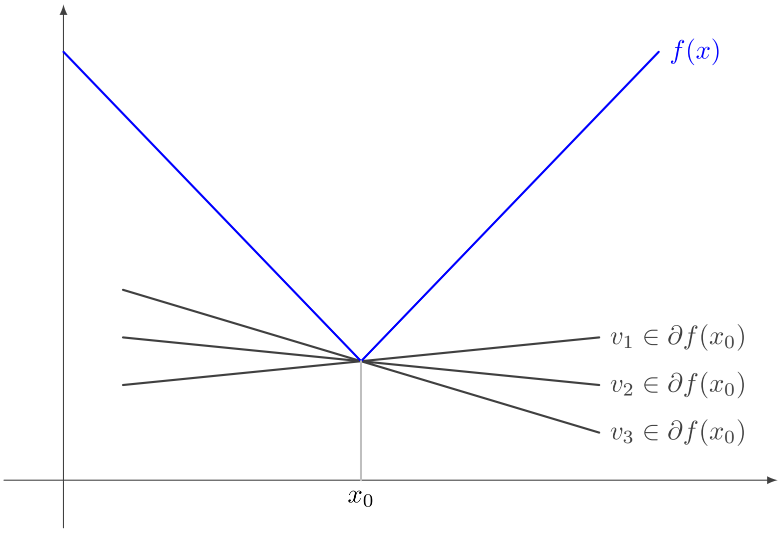

$f(x)$
$v_1 \in \partial f(x_0)$
$v_2 \in \partial f(x_0)$
$v_3 \in \partial f(x_0)$
x_0

Figure 2.4: Illustration of the subdifferential. The subdifferential of the function $f(x)$ at x_0 is defined by the set of planes passing through point x_0 with slope v, which lie below the graph of $f(x)$.

which can be exploited in optimization algorithms.

A few important transformations of convex functions which preserve convexity are:

- The sum of convex functions is convex.
- Functions remain convex under affine transformations, *i.e.*

$$g(x) = f(Ax + b), \quad A \in \Re^{M\times N}, b \in \Re^{M} \tag{2.14}$$

  is convex if $f : \Re^N \to \Re$ is convex.
- The pointwise maximum

$$g(x) = \max\{f_1(x), f_2(x)\} \tag{2.15}$$

  over convex functions is again convex.
- Every norm is convex.

2.1.3 The Convex Conjugate

The convex conjugate (also known as the Legendre-Fenchel transform) is a concept that allows to characterize functions by their slopes and intercepts instead of their points.

Definition 4. *For a function $f : C \to \Re$, the convex conjugate is defined as*

$$f^*(x^*) = \sup_{x\in C}\{\langle x^*, x\rangle - f(x)\}, \quad x^* \in C^*, \tag{2.16}$$

where C^* is the dual domain to C, which is given by

$$C^* = \{x^* : f^*(x^*) < \infty\}. \tag{2.17}$$

While this transformation is intimately related to convexity, it is not restricted to convex functions. The convex conjugate of any function is convex and lower semicontinuous. Moreover, the biconjugate

$$f^{**}(x) = \sup_{x^* \in C^*} \{\langle x, x^* \rangle - f^*(x^*)\} \tag{2.18}$$

is convex and is the convex envelope of $f(x)$, satisfying $f^{**}(x) \leq f(x)$. For closed, convex, lower semicontinuous functions the biconjugate coincides with the function itself, *i.e.* we have $f^{**}(x) = f(x)$.

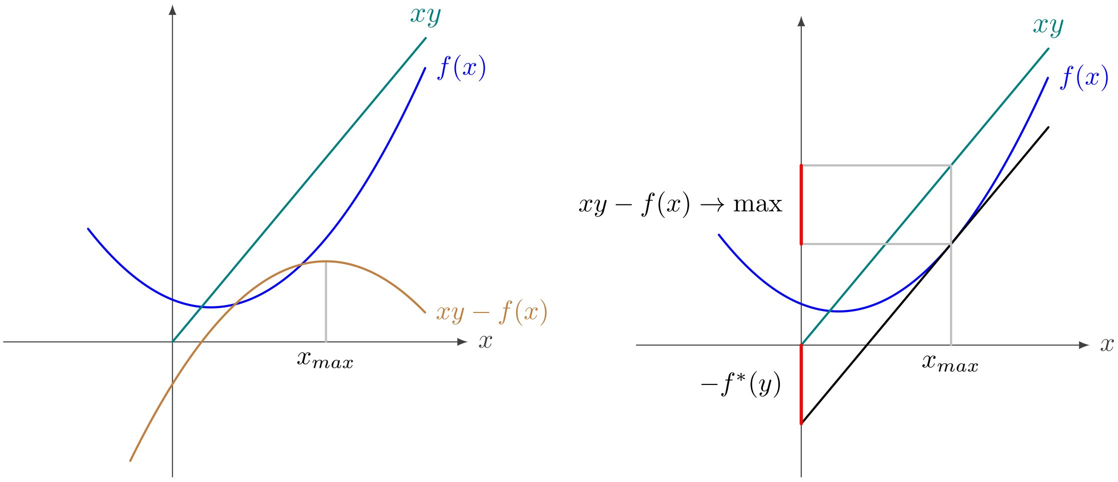


Figure 2.5: Illustration of the convex conjugate. (Left) The convex conjugate at y is defined via the maximum of $xy - f(x)$. (Right) The maximum is achieved at x_{max}, consequently the convex conjugate is given by the negative y-intercept of the function drawn in black.

The definition of the convex conjugate trivially leads to the Fenchel-Young inequality

$$f^*(x^*) + f(x) \geq \langle x^*, x \rangle \,, \tag{2.19}$$

where equality holds for closed convex functions. Using the Fenchel-Young inequality, it can be shown that the subdifferential and the convex conjugate have a close relationship for closed and convex functions. To see this choose $v \in f(\hat{x})$, then

$$\begin{aligned} f(x) \geq f(\hat{x}) + \langle v, x - \hat{x} \rangle &\Leftrightarrow \langle v, x \rangle - f(x) \leq \langle v, \hat{x} \rangle - f(\hat{x}) \\ &\Leftrightarrow f^*(v) = \sup_x \langle v, x \rangle - f(x) = \langle v, \hat{x} \rangle - f(\hat{x}). \end{aligned} \tag{2.20}$$

It follows that $f^*(v) + f(\hat{x}) = \langle v, \hat{x} \rangle$.

2.1.4 Norms

We call a function $||.|| : \Re^N \to \Re$ a norm on the vector space $\Re^N$, if it full-fills the following conditions for all $x, y \in \Re^N$ and $\alpha \in \Re$:

1. $||\alpha x|| = |\alpha| \, ||x||$
2. $||x + y|| \leq ||x|| + ||y||$
3. $||x|| = 0 \Rightarrow x = 0.$

Functions which full-fill the first and second condition, but not the third condition, are called semi-norms. From the first and second condition we have for any $\alpha \in [0, 1]$

$$\begin{aligned} ||\alpha x + (1-\alpha)y|| &\leq ||\alpha x|| + ||(1-\alpha)y|| \\ &= \alpha \, ||x|| + (1-\alpha) \, ||y|| \,, \end{aligned} \tag{2.21}$$

which shows that any (semi-)norm is convex. Conversely we have that any convex function, for which the first and the second conditions hold are semi-norms.

Some special norms which will be used frequently throughout this thesis are the Euclidean norm, the Manhattan norm and the Maximum norm respectively. Those are defined for $x \in X \subseteq \Re^N$ as

$$||x||_2 = \sqrt{\sum_{n=1}^{N} |x_n|^2}, \quad ||x||_1 = \sum_{n=1}^{N} |x_n| \,, \quad ||x||_\infty = \max(|x_1| \,, \ldots, |x_n|). \tag{2.22}$$

To simplify notation, we often use the short-hand notation $||x||$ for the Euclidean norm.

Some quantities will have additional structure, for example quantities which are given by a vector for each pixels. We denote this as $x \in \Re^{LN}$ and adopt the notation of an inner norm, which is applied per element, and an outer norm, which is applied to the resulting vector. Examples are:

$$||x||_{2,1} = \sum_{n=1}^{N} \sqrt{\sum_{l}^{L} |x_n^l|^2}, \quad ||x||_{1,1} = \sum_{n=1}^{N} \sum_{l}^{L} \left| x_n^l \right| \tag{2.23}$$

The last important concept related the norms are the dual norms. The dual norm $||.||_*$ to $||.||$ is defined as

$$||y||_* = \sup_{||x|| \leq 1} \langle x, y \rangle \,. \tag{2.24}$$

The convex conjugate of any norm $f(x) = ||x||$ is easily given in terms of the dual norm as

$$f^*(y) = \begin{cases} 0 & \text{if} \quad ||y||_* \leq 1 \\ \infty & \text{else} \end{cases} \tag{2.25}$$

2.2 Convex Optimization

Convex optimization is concerned with the optimization problems of the form

$$\min_{x \in C} f(x), \tag{2.26}$$

where $f : C \rightarrow \Re$ is a convex function and the domain $C \subseteq \Re^N$ is convex. Convex optimization problems have the important property that every local minimum is also a global minimum. Moreover, the global optimum of any strictly convex function is unique. This structure can be exploited to build efficient optimization algorithms. We will be mainly concerned with so-called large-scale first-order methods. These methods have the advantage that their convergence rate does not depend on the number of variables. This is important for the types of problems that are considered in image processing, since the optimization problems may involve millions of variables. On the downside, such methods are only able to yield results of medium accuracy. This is seldom a problem, since solutions of very high accuracy are rarely needed in image processing, however.

This section aims to give a brief overview of the most important first-order algorithms, which are relevant to this thesis.

2.2.1 Gradient Descent

Gradient descent is one of the most widely used approaches for the optimization of smooth functions. For convex optimization problems, convergence to a global optimum can be rigorously proven. Gradient descent can be applied to optimization problems of the form

$$\min_{x \in \Re^N} f(x), \tag{2.27}$$

where $f(x)$ is a smooth function. The iterations of this method are shown in Algorithm 2.1.

```
1.  Choose a starting point $x_0 \in \Re^N$, set $k = 0$
2.  While not converged
    $x_{k+1} = x_k - h_k \nabla f(x_k)$
    $k = k + 1$
```

Algorithm 2.1: Gradient descent

If $f(x)$ is convex and has Lipschitz-continuous gradient, *i.e.* the relation

$$||\nabla f(x) - \nabla f(y)|| \leq L \, ||x - y|| \, , \tag{2.28}$$

holds, it can be shown that Gradient Descent converges for a constant step-size choice of $h_k = h \in (0, \frac{2}{L})$ with a rate of $\mathcal{O}(\frac{1}{k})$. For strongly convex functions with modulus m, better convergence rates can be obtained by choosing $h \in (0, \frac{2}{m+L}]$.

Gradient descent is seldom used on convex functions in practice, since it is quite slow. It has been shown that the optimal convergence rate for first order methods on smooth convex functions is $\mathcal{O}(\frac{1}{k^2})$ [110], which is far better than the convergence rate of Gradient Descent. Nesterov's Accelerated Gradient [109] method is a simple modification, which reaches this theoretically optimal convergence rate. The iterations are shown in Algorithm 2.2. The scheme uses an additional overrelaxation step in order to reach the optimal convergence rate. This requires to store historical updates, and results in a doubled memory demand (which is seldom problematic in practice). It is also important to note that in contrast to Gradient Descent the iterates x_k of the accelerated method are not monotonically decreasing and often show periodic phases of increase and decrease.

```
1.  Choose a starting point $y_1 = x_0 \in \Re^N$, $t_1 = 1$, set $k = 1$
2.  While not converged
        $x_k = y_k - \frac{1}{L}\nabla f(y_k)$
        $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
        $y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1})$
        $k = k + 1$
```

Algorithm 2.2: Nesterov's Accelerated Gradient method

2.2.2 Proximal Methods

Proximal Methods are a class of convex optimization algorithms, which use the so-called proximal operator on the objective function in order to minimize the objective. They allow for a larger class of convex function to be optimized than the gradient-based approaches discussed earlier. Many well-known algorithms can be understood as proximal methods.

The basic building block of Proximal Methods is the proximal operator: The proximal operator of a convex function $f(x)$ evaluated at a point v is given by

$$\mathbf{prox}_f(v) = \arg\min_x \left\{ \frac{1}{2} \, ||x - v||^2 + f(x) \right\} . \tag{2.29}$$

This operation can be understood in some sense as the generalization of the projection operation. Consider the problem of projecting a point v onto the convex set C:

$$\mathbf{proj}_C(v) = \arg\min_{x \in C} \left\{ \frac{1}{2} ||x - v|| \right\}. \tag{2.30}$$

This can be equivalently expressed as an unconstrained problem using the indicator function of the set C,

$$I_C(v) = \begin{cases} 0 & \text{if } v \in C \\ \infty & \text{else ,} \end{cases} \tag{2.31}$$

as

$$\mathbf{proj}_C(v) = \arg\min_{x} \left\{ \frac{1}{2} ||x - v||^2 + I_C(x) \right\}, \tag{2.32}$$

which is exactly the proximal operator applied to the indicator function of the set C.

The proximal operator is strongly convex, thus its solution is unique. Moreover, the fixed point of the proximal operator is a minimizer of $f(x)$, *i.e.*

$$x^* = \mathbf{prox}_f(x^*) \Leftrightarrow f(x^*) \leq f(x) \quad \forall x \in C \tag{2.33}$$

The proximal operator is strongly related to the Moreau envelope of $f(x)$:

$$M_f(v) = \min_{x} \left\{ \frac{1}{2} ||x - v||^2 + f(x) \right\}, \tag{2.34}$$

which is the infimal convolution of $f(x)$ with the function $g(x) = \frac{1}{2} ||x||^2$:

$$(g \star_{\text{inf}} f)(v) = \inf_{x} \{ g(v - x) + f(x) \}. \tag{2.35}$$

M_f can be interpreted as a regularization of f. Both functions share the same set of minimizers, but M_f is continuously differentiable, even if f is not [117], which makes its optimization considerably easier. The proximal operator exactly returns the point, where (2.34) achieves its unique minimum. In fact, evaluation of the proximal operator can be understood as a gradient step on M_f [117].

The fixed point property (2.33) already points to a simple optimization algorithm, the so-called proximal point algorithm, which is simply given by iterating

$$x^{k+1} = \mathbf{prox}_{\lambda f}(x^k). \tag{2.36}$$

This scheme is not always useful in practice, however, since minimizing the original function $f(x)$ is usually not harder than solving the proximal operator repeatedly. The concept of proximal operators becomes mostly useful in the presence of additional structure

```
1.  Choose a starting point $x_0 \in \Re^N$, set $k = 0$
2.  While not converged
    $x_{k+1} = \mathbf{prox}_{h_k f}(x_k - h_k \nabla g(x_k))$
    $k = k + 1$
```

Algorithm 2.3: Proximal Gradient algorithm

in the optimization problem. Consider the following problem:

$$\min_{x \in \Re^N} f(x) + g(x), \tag{2.37}$$

where $g(x)$ is convex with L-Lipschitz-continuous gradient and $f(x)$ is convex, but possibly non-smooth. Problems of this form can be minimized using the Proximal Gradient algorithm (Algorithm 2.3), which can be understood as a generalization of the Gradient Descent method. It is interesting to note that this algorithm shows the same convergence rate and step-size rule as Gradient Descent, despite the presence of a non-smooth term $f(x)$. If $f(x)$ is an indicator function of a convex set, the Proximal Gradient algorithm is equivalent to Projected Gradient Descent.

Most importantly, the algorithm again can be accelerated to achieve optimal convergence rate $\mathcal{O}(\frac{1}{k^2})$. This algorithm, which is listed in Algorithm 2.4 is best known as Fast Iterative Shrinkage Thresholding Algorithm (FISTA) [12] in the literature.

Note that for the Proximal Gradient algorithm as well as for FISTA one fundamental assumption is that the proximal operator is easy to evaluate. In practice this means that a closed-form solution is available. If no closed-form solution is available efficient algorithms can sometimes still be build, if the optimization problem posed by the proximal operator is sufficiently simple.

```
1.  Choose a starting point $y_1 = x_0 \in \Re^N$, $t_1 = 1$, set $k = 1$
2.  While not converged
    $x_k = \mathbf{prox}_{\frac{1}{L}f}(y_k - \frac{1}{L}\nabla g(y_k))$
    $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
    $y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1})$
    $k = k + 1$
```

Algorithm 2.4: Fast Iterative Shrinkage Thresholding Algorithm (FISTA)

2.2.2.1 Dykstra algorithm

The prevalent approach for optimization using proximal methods is based on splitting the problem in tractable sub-parts. The Proximal Gradient algorithm 2.3 illustrates this, where the problem is split into a smooth part and a non-smooth part, which can then be handled independently during optimization. A similar approach can be adopted for projecting a point p onto the intersection of convex sets C_1 and C_2:

$$\min_{x \in C_1 \cap C_2} \frac{1}{2} \|x - p\|^2 \tag{2.38}$$

Since the intersection of convex sets is again convex, this problem is convex overall. Using indicator functions, this problem can be rewritten as the unconstrained problem

$$\min_{x} \frac{1}{2} \|x - p\|^2 + I_{C_1}(x) + I_{C_2}(x). \tag{2.39}$$

It is easy to see that (2.39) is in fact the proximal operator $\mathbf{prox}_{I_{C_1}+I_{C_2}}(p)$. Unfortunately the proximal operator of a sum of functions is in general not easy to evaluate. Nonetheless, on can adopt a splitting approach, where the projections are carried out in an alternating manner. One particularly efficient approach is called the Dykstra Algorithm [23] (not to be confused with the more famous Dijkstra's Shortest Path algorithm), which is listed in Algorithm 5.2. Its iterations alternatingly project leading points onto the sets C_1 and C_2. While this algorithm is mostly used for the projection onto convex sets, it can also be used with arbitrary proximal operators in order to minimize composite functions with a quadratic term. This variant is a special case of Douglas-Rachford splitting for composite functions [40].

```
1.  Set $x_0 = p$, $p_0 = 0$, $q_0 = 0$, set $k = 0$
2.  While not converged
        $y_k = \mathbf{prox}_{I_{C_1}}(x_k + p_k)$
        $p_{k+1} = x_k + p_k - y_k$
        $x_{k+1} = \mathbf{prox}_{I_{C_2}}(y_k + q_k)$
        $q_{k+1} = y_k + q_k - x_{k+1}$
        $k = k + 1$
```

Algorithm 2.5: Dykstra algorithm

2.2.3 Primal-Dual Method

The primal-dual method [36, 52, 72] is an algorithm, which is particularly well suited for the type of problems, which are handled in this thesis. It can be applied to problems,

```
1.  Set $x_0 = \bar{x}_0 \in \Re^N$, $y_0 \in \Re^M$, set $k = 0$, choose $\theta \in [0,1]$
2.  While not converged
    $y_{k+1} = \mathbf{prox}_{\sigma f^*}(y_k + \sigma A\bar{x}^k)$
    $x_{k+1} = \mathbf{prox}_{\tau g}(x_k - \tau A^* y^{k+1})$
    $\bar{x}_{k+1} = x_{k+1} + \theta(x_{k+1} - x_k)$
    $k = k + 1$
```

Algorithm 2.6: Primal-Dual algorithm

which admit the following structure:

$$\min_x f(Ax) + g(x), \tag{2.40}$$

where $f : \Re^M \to [0, \infty)$ and $g : \Re^N \to [0, \infty)$ are proper, convex and lower-semicontinuous functions and $A \in \Re^{M \times N}$ is a linear operator. Note that we do not require smoothness of the problem, although it an be exploited to achieve a faster convergence rate if it is present in at least one of the functions. This general structure allows to handle a large class of convex optimization problems, which are often encountered in image processing. Using the biconjugate of $f(Ax)$, the minimization problem can be rewritten as a convex-concave saddle-point problem, the so called primal-dual formulation:

$$\min_{x \in \Re^N} \max_{y \in \Re^M} \langle Ax, y \rangle + g(x) - f^*(y), \tag{2.41}$$

where $f^*(y)$ is the convex conjugate of f. The associated dual problem is given by

$$\max_{y \in \Re^M} -g^*(-A^*y) - f^*(y). \tag{2.42}$$

The linear operator A^* is the adjoint operator to A, *i.e.* it full-fills the relation

$$\langle Ax, y \rangle = \langle x, A^*y \rangle, \tag{2.43}$$

and is for the case of a real-valued matrix A simply given by the matrix transpose.

The primal-dual method directly operates on the primal-dual formulation (2.41) and is summarized in Algorithm 2.6. The iterations can be interpreted as alternating proximal gradient steps, which maximize the objective with respect y and minimize with respect to x, respectively. Moreover the algorithm includes an additional overrelaxation step, which is crucial for the convergence of the procedure. The algorithm is guaranteed to converge for step-sizes $\tau\sigma \, ||A||^2 < 1$ with a convergence rate of $\mathcal{O}(\frac{1}{k})$, in terms of the restricted primal-dual gap

$$\mathcal{G}_{B_1 \times B_2}(x, y) = \max_{y' \in B_2} \langle y', Ax \rangle - f^*(y') + g(x) - \min_{x' \in B_1} \langle y, Ax' \rangle - f^*(y) + g(x'), \qquad (2.44)$$

with $B_1 \times B_2 \subset \Re^N \times \Re^M$. This is the optimal convergence rate for non-smooth convex problems [110]. Additionally, if $g(x)$ is strongly convex, the algorithm can be accelerated to a convergence rate of $\mathcal{O}(\frac{1}{k^2})$ [37].

2.2.3.1 Preconditioning

The primal-dual method requires knowledge about the operator norm $||A||$ of the linear operator A in order to find step-sizes which guarantee convergence. In many applications, this norm is difficult to estimate. Pock and Chambolle [120] introduced a practical diagonal preconditioning scheme to find step-sizes τ and σ which automatically lead to convergence. Moreover, the preconditioning often leads to faster convergence if the operator A is badly scaled. The scheme uses a local per-variable step-size, which is given by

$$\tau_j = \frac{1}{\sum_{i=1}^{M} |A_{i,j}|^{2-\alpha}}, \qquad \sigma_i = \frac{1}{\sum_{j=1}^{N} |A_{i,j}|^{\alpha}}, \qquad \alpha \in [0, 2] \qquad (2.45)$$

The step-size for each variable can thus be computed simply by summing over the rows and columns of the operator A, respectively. This scheme is highly practical and can often be computed without explicit formation of the operator A.

2.2.4 Other Methods

There are many first-order methods for non-smooth and smooth convex optimization, which exploit the structure of specific instances of the problems at hand. It is beyond the scope of this work to give a comprehensive list. An excellent overview of the most important methods can be found in [40].

A completely different class of algorithms is given by Interior Point methods [168]. Different from first-order methods their scope are small to medium scale problems, which are solved to high accuracy. Due to these properties they are not as often encountered in image processing when compared to first-order methods. The basic form that is handled by interior point methods are linear objectives with convex feasible regions of a particular form, such as second-order cones or the space of semi-definite matrices. Like Interior Point methods in linear programming, the problem is reduced to a series of unconstrained convex problems by modeling the feasible region using barrier functions. The subproblems are solved using Newton-type methods, which exhibit rapid convergence, but do not scale for problems with a large amount of variables. The solutions of the subproblems describe a trajectory which lies strictly in the interior of the feasible region, hence the name Interior Point methods.

Despite their problems with large-scale problems, Interior Point methods can be a viable tool for rapid prototyping in image processing, since the algorithms are applicable to large subclasses of convex problems such as second-order cone or semi-definite programs. As a matter of fact all of the optimization problems in this thesis can be represented as standard second-order cone programs. Most importantly, there exist automatic tools [65] to convert optimization problems into standard forms, which can then be solved using off-the-shelf interior point solvers like SeDuMi [144].

2.3 Variational Models

The models in this thesis are defined in a continuous setting. As such, they act on functions instead of vectors. While any implementation on a digital computer requires the discretization, it is nonetheless advantageous to first define and analyze the model in the continuous setting. In this section we will introduce basic concepts of variational models along with a fundamental model for image restoration, the ROF model [130]. We will discuss some basic properties of this model and how the model can be discretized and implemented.

2.3.1 Preliminaries

Throughout the rest of this thesis, bold symbols denote elements in the domain of a function:

$$\mathbf{x} = \left(x^1, \dots, x^N\right)^T \in \Omega. \tag{2.46}$$

For images we thus have $\mathbf{x} = (x^1, x^2)^T$. For one-dimensional functions we default to non-bold notation $x^1 = x \in \Omega \subseteq \Re$.

Let us first introduce a concept, which will be crucial in order to allow for a rigorous treatment of non-smooth functions. A *distribution* T_u on a function $u : \Omega \to \Re$ with $\Omega \subseteq R^N$ is defined as

$$\langle T_u, \phi \rangle = \int_\Omega \phi(\mathbf{x}) u(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \quad \phi \in \mathcal{C}_c^\infty(\Omega), \tag{2.47}$$

where the test function ϕ is defined on the space of infinitely differentiable functions with compact support on Ω (it is zero outside of Ω). This mapping is well-defined if u is locally integrable, *i.e.*

$$\int_M |u| \, \mathrm{d}\mathbf{x} < \infty, \; \forall M \subset \Omega, \; M \text{ compact}. \tag{2.48}$$

Distributions can be used to generalize the concept of differentiation to functions which are not differentiable in the classical sense. Using integration by parts and noting that ϕ

has compact support, we have

$$\left\langle \frac{\partial T_u}{\partial x^i}, \phi \right\rangle = \int_\Omega \frac{\partial T_u}{\partial x^i} \phi \, \mathrm{d}\mathbf{x} = - \int_\Omega T_u \frac{\partial \phi}{\partial x^i} \, \mathrm{d}\mathbf{x}. \tag{2.49}$$

$\frac{\partial T_u}{\partial x_i}$ is called the *distributional derivative* of u and coincides with the classical derivative if u is smooth, but is also well defined for any locally integrable u. Using these definitions we can define the distributional gradient of a non-scalar function u analogous the gradient, by gathering the partial distributional derivatives into a single vector:

$$Du = \left(\frac{\partial T_u}{\partial x^1}, \dots \frac{\partial T_u}{\partial x^N} \right)^T, \tag{2.50}$$

For smooth functions we have $\nabla u = Du$.

2.3.2 Total Variation and the ROF Model

Total Variation (TV) is among the most successful and widely used priors in Computer Vision. TV was most notably used as a natural image prior for denoising in the seminal work by Rudin, Osher and Fatemi [130]. Later, TV was also used successfully as a prior for optical flow estimation [176].

In its most well-known form TV is defined as

$$TV(u) = \int_\Omega |Du|, \tag{2.51}$$

where the operator D denotes the distributional gradient defined in the previous subsection. The set of functions $u \in L^1(\Omega)$ with $TV(u) < \infty$ is called the space of functions with bounded variation $BV(\Omega)$.

Alternatively, TV is often equivalently stated in its dual form

$$TV(u) = \sup \left\{ - \int_\Omega u \, \mathrm{div} \, p \, \mathrm{d}\mathbf{x} \;\middle|\; p \in C_c^1(\Omega, \Re^N), |p(\mathbf{x})| \leq 1 \; \forall \mathbf{x} \in \Omega \right\}, \tag{2.52}$$

where C_c^1 denotes the space of continuously differentiable functions with compact support. Both definitions are well-defined even for non-differentiable functions if u is locally integrable. It is important to note that typically the absolute value in (2.51) and (2.52) is understood to be the Euclidean norm, *i.e.*

$$|(q_1, \dots, q_N)^T| = \sqrt{\sum_{i=1}^N q_i^2}\,. \tag{2.53}$$

Any other norm leads to a valid definition of the Total Variation as well, however. Note that the norms arising in (2.51) and (2.52) are dual to each other. They coincide in the case of the Euclidean norm, since this norm is self-dual (*i.e.* the dual norm of the Euclidean norm is again the Euclidean norm). We will stick to the Euclidean norm in large parts of this work and make it explicit whenever we deviate from this definition of TV.

It is easy to see that TV is a convex function and that

$$TV(\alpha u) = |\alpha| TV(u). \tag{2.54}$$

Thus as was shown in Section 2.1.4 it follows that TV is a semi-norm. Moreover TV is rotationally invariant, if the norm to measure the magnitude of the gradient is chosen to be the Euclidean norm (2.53). An interesting property of TV is that it has a convenient geometric interpretation: $TV(u)$ measures the total length of the level-sets of u.

TV-based models encode the prior assumption that their solution should have sparse gradients, *i.e.* solutions are composed of constant regions with a sparse set of jumps between them. In the case of stereo estimation this assumption directly translates into the assumption of a fronto-parallel world, which is too restrictive for many real-world scenes. Nonetheless, TV is still an important prior, since it is well understood in theory and in practice, and its variations are still used extensively. A seminal model that uses TV as prior is the model by Rudin, Osher and Fatemi [130], the so-called ROF model. This model is able to remove Gaussian noise, while preserving dominant edges in the image. It can be understood as an edge-preserving non-linear filter and is given by

$$\min_u \frac{\lambda}{2} \int_\Omega (u(\mathbf{x}) - f(\mathbf{x}))^2 \, \mathrm{d}\mathbf{x} + \int_\Omega |Du|. \tag{2.55}$$

This model has been extensively used and analyzed in image processing. We will use the ROF model to exemplify a few issues regarding the actual implementation of variational models on digital computers. We will show the discretization of the model and apply algorithms from the previous section to solve the model efficiently.

Discretization and Optimization The first step for the implementation is discretization of the continuous model. Throughout this work we assume a rectangular image domain, which is laid out on a grid of size $N_x \times N_y$ with discretization steps Δx and Δy. Locations in this grid are indexed by pairs of integers (i, j), *i.e.* the discrete domain can be described by

$$G^\Delta = \{(i\Delta x, j\Delta y) : (0,0) \leq (i,j) < (N_x, N_y)\}. \tag{2.56}$$

Thus the discrete image is given by $u : G^\Delta \to \Re$. We define the discrete gradient that approximates Du as

$$(\nabla u)_{i,j} = \begin{pmatrix} (\delta_x u)_{i,j} \\ (\delta_y u)_{i,j} \end{pmatrix}, \tag{2.57}$$

which is defined via the finite forward differences:

$$(\delta_x u)_{i,j} = \begin{cases} (u_{i+1,j} - u_{i,j})/\Delta x & \text{if} \quad i < N_x - 1 \\ 0 & \text{else} \end{cases}$$
$$(\delta_y u)_{i,j} = \begin{cases} (u_{i,j+1} - u_{i,j})/\Delta y & \text{if} \quad j < N_y - 1 \\ 0 & \text{else} \end{cases} \tag{2.58}$$

Using this definition the adjoint operator acting on $p : G^\Delta \to \Re^2$ is given by

$$(\nabla^* p)_{i,j} = -((\bar{\delta}_x p^1)_{i,j} + (\bar{\delta}_y p^2)_{i,j}), \tag{2.59}$$

with finite backward differences

$$(\bar{\delta}_x p^1)_{i,j} = \begin{cases} (p^1_{i,j} - p^1_{i-1,j})/\Delta x & \text{if} \quad 0 < i < N_x - 1 \\ p^1_{i,j}/\Delta x & \text{if} \quad i = 0 \\ -p^1_{i-1,j}/\Delta x & \text{if} \quad i = N_x - 1 \end{cases}$$
$$(\bar{\delta}_y p^2)_{i,j} = \begin{cases} (p^2_{i,j} - p^2_{i,j-1})/\Delta y & \text{if} \quad 0 < j < N_y - 1 \\ p^2_{i,j}/\Delta y & \text{if} \quad j = 0 \\ -p^2_{i,j-1}/\Delta y & \text{if} \quad j = N_y - 1 \end{cases} \tag{2.60}$$

Using these definitions, we can define the discrete ROF model as

$$\min_u \frac{\lambda}{2} \sum_{i,j} (u_{i,j} - f_{i,j})^2 + \sum_{i,j} |(\nabla u)_{i,j}|_2 \, . \tag{2.61}$$

To allow for uncluttered notation we can stack all quantities into vectors $u, f \in \Re^{N_x N_y}$ and write

$$\min_u \frac{\lambda}{2} ||u - f||_2^2 + ||\nabla u||_{2,1} \, , \tag{2.62}$$

where the matrix $\nabla \in \Re^{2N_x N_y \times N_x N_y}$ is a sparse matrix implementing the forward differences.

Model (2.62) is a convex optimization problem, which can be solved using an appropriate solver. We will exemplify here the two most widely used algorithms for solving this model: The primal-dual algorithm 2.6 and FISTA 2.4. Let us start with the primal-dual algorithm. We can derive a convex-concave saddle-point formulation using the dual formulation (2.52) of TV:

$$\min_u \max_{||p||_\infty \leq 1} \frac{\lambda}{2} ||u - f||_2^2 + \langle \nabla u, p \rangle = \min_u \max_{||p||_\infty \leq 1} \frac{\lambda}{2} ||u - f||_2^2 + \langle u, \nabla^* p \rangle \, , \tag{2.63}$$

where $p \in \Re^{2N_xN_y}$ and we adopt the short-hand notation $||p||_{2,\infty} = ||p||_\infty$.

In this form Algorithm 2.6 is directly applicable. We identify $g(u) = \frac{\lambda}{2}\,||u-f||_2^2$ and $f^*(p) = I_{||p||_\infty \leq 1}$. Consequently the iterations of the primal.-dual algorithm are given by

$$\begin{cases} p_{k+1} = \mathbf{prox}_{I_{||p||_\infty \leq 1}}(p_k + \sigma\nabla\bar{u}_k) \\ u_{k+1} = \mathbf{prox}_{\tau g}(u_k - \tau\nabla^* p_{k+1}) \\ \bar{u}_{k+1} = 2u_{k+1} - u_k \end{cases} . \tag{2.64}$$

For this model the proximity operators are very easy to evaluate. For the dual variable $\hat{p}$, the operator reduces to a simple element-wise projection onto the Euclidean norm ball:

$$\left(\mathbf{prox}_{I_{||p||_\infty \leq 1}}(\hat{p})\right)_{i,j} = \mathbf{prox}_{I_{|p|_2 \leq 1}}(\hat{p}_{i,j}) = \frac{\hat{p}_{i,j}}{\max(1, |\hat{p}_{i,j}|_2)}. \tag{2.65}$$

For the primal variables, we have to solve element-wise a scalar quadratic minimization problem, which can be solved in closed form:

$$(\mathbf{prox}_{\tau g}(\hat{u}))_{i,j} = \arg\min_u \frac{1}{2}\,||u - \hat{u}_{i,j}||_2^2 + \frac{\lambda\tau}{2}\,||u - f_{i,j}||_2^2 = \frac{\hat{u}_{i,j} + \lambda\tau f_{i,j}}{1+\lambda\tau} \tag{2.66}$$

This algorithm has a convergence rate of $\mathcal{O}(\frac{1}{k})$ in terms of the primal-dual gap, but can be accelerated to $\mathcal{O}(\frac{1}{k^2})$ using only minor modifications to the step-sizes and the overrelaxation step [36].

On the other hand it is also possible to apply FISTA to the dual problem. Problem (2.63) admits a closed form solution with respect to u. For any p the first-order optimality condition with respect to u is given by

$$u^* = f - \frac{1}{\lambda}\nabla^* p, \tag{2.67}$$

Substituting into (2.63) and multiplying by λ yields

$$\max_{||p||_\infty \leq 1} -\frac{1}{2}\,||\nabla^* p||^2 + \lambda\,\langle f, \nabla^* p\rangle\,. \tag{2.68}$$

By adding the term $||\lambda f||^2$ and multiplying by -1, this can be written more compactly as the minimization problem

$$\min_{||p||_\infty \leq 1} \frac{1}{2}\,||\nabla^* p - \lambda f||^2 \tag{2.69}$$

It is now easy to see that this is a convex optimization problem, which can be written as the sum of a smooth function and a non-smooth function by using an indicator function:

$$\min_{p} \frac{1}{2} ||\nabla^* p - \lambda f||^2 + I_{||p||_\infty \leq 1}(p). \tag{2.70}$$

Thus FISTA can be applied to this problem and the basic iterations are given by

$$\begin{cases} p_{k+1} = \mathbf{prox}_{I_{||p||_\infty \leq 1}}(\bar{p}_k - \frac{1}{L}\nabla(\nabla^* \bar{p}_k - \lambda f)) \\ \bar{p}_{k+1} = p_k + \frac{t_k - 1}{t_{k+1}}(p_k - p_{k-1}), \end{cases} \tag{2.71}$$

where t_k is computed as shown in Algorithm 2.4. After convergence, the primal solution can simply be recovered using (2.67). This algorithm has a convergence rate of $\mathcal{O}(\frac{1}{k^2})$ in terms of the dual energy (2.69), which is again the theoretically optimal rate of first-order algorithms for this type of problem.

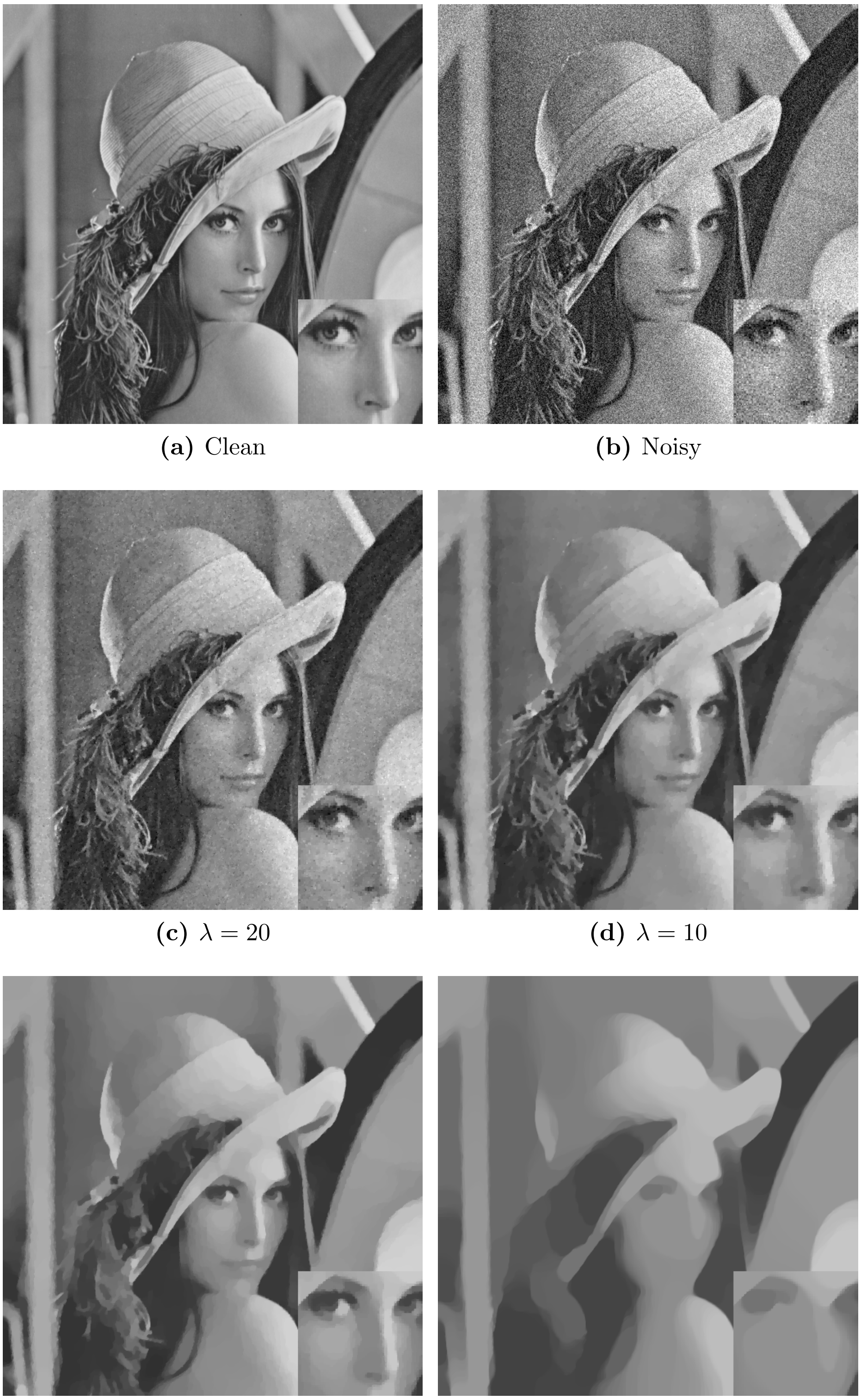
Figure 2.6 shows an example of ROF denoising together with close-ups of a sub-region of the image. We vary the magnitude of the regularization parameter λ. It can be seen that with stronger regularization more noise is removed, but also fine details are lost. The close-ups show that the resulting images are composed of piecewise constant regions. This effect is known as staircasing and is an artifact of the TV regularizer. The higher-order regularizers which will be discussed in this work are explicitly designed to address this problem, and relax the assumption of piecewise constancy. Especially for the task of optical flow and stereo estimation this is not purely a cosmetic, visual difference, since higher-order priors can generally allow for more plausible solutions if the data is ambiguous.

2.3.3 A Variational Model for Correspondence Problems

We are now ready to state a variational model for correspondence problems: Consider two input images $I_1, I_2 : \Omega \to \Re$. The correspondence problem is to find a displacement field $\mathbf{u} = (u, v)^T : \Omega \to \Re^2$, which warps every pixel from I_1 to I_2, *i.e.* we have

$$I_1(\mathbf{x} + \mathbf{u}) = I_2(\mathbf{x}) \tag{2.72}$$

This equality, which models brightness constancy, already gives a hint at how to find the unknown displacement field in principle. The field, assuming that there are no occlusions or illumination changes, should satisfy (2.72) everywhere. It is obvious that the correspondence problem is severely underconstrained in this form: There is only one measurement for two unknowns, the solution therefore is highly ambiguous. In the case of stereo estimation one can exploit camera geometry to reduce the two dimensional problem to a one dimensional problem. However, there mighty still be many fields $\mathbf{u}$ that fulfill (2.72). To resolve these ambiguities we can incorporate additional smoothness assumptions. A prior that models a smoothness assumption was already introduced: TV, which can be used to encode that solutions shall be piece-wise constant. Note also that (2.72) can never



(a) Clean

(b) Noisy

(c) $\lambda = 20$

(d) $\lambda = 10$

(e) $\lambda = 5$

(f) $\lambda = 1$

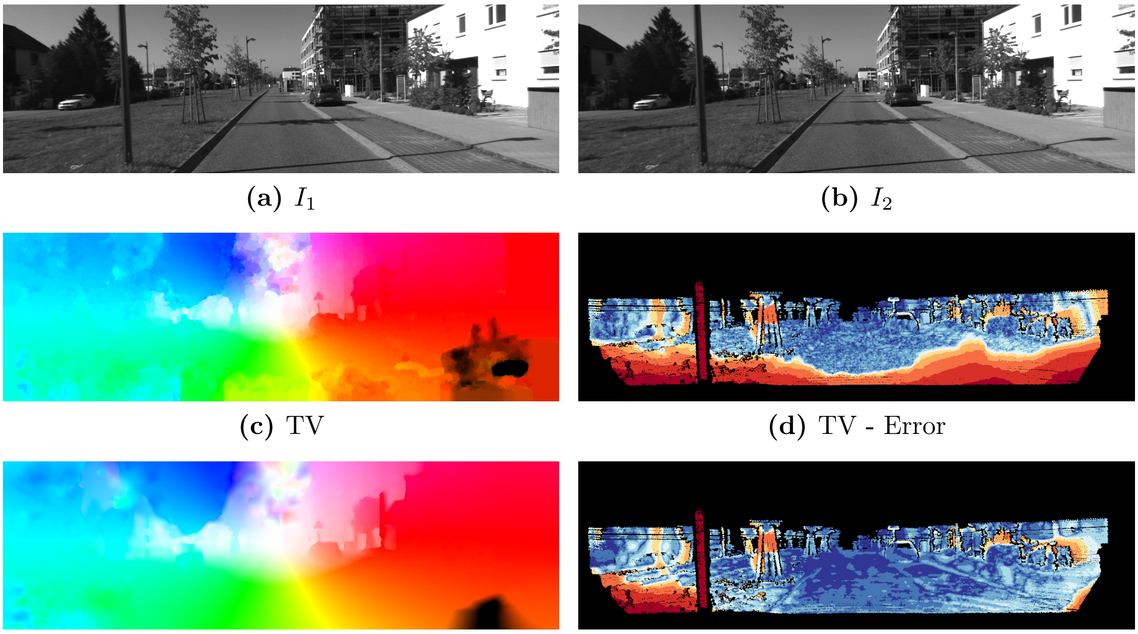
Figure 2.6: Denoising using the ROF model for different regularization parameters. Smaller values of λ correspond to stronger regularization. As the influence of the regularization term gets higher, progressively more details are lost. The image is composed of piecewise constant regions, which is an artifact of the TV regularization.

be full-filled exactly everywhere in the image in practical applications (except if there is no displacement), since some pixels may have no correspondence at all due to occlusions. Moreover, in realistic scenarios the brightness between the two images may change due to environmental effects or automatic adjustments of the camera. If we relax the hard constraint to account for these problems and introduce a prior, for example TV, we can formulate the correspondence problem as the variational problem

$$\min_{\mathbf{u}} TV(u) + \int_{\Omega} \Phi(I_1(\mathbf{x}+\mathbf{u}) - I_2(\mathbf{x}))\, \mathrm{d}\mathbf{x}. \tag{2.73}$$

Here Φ is a non-negative penalty function, which is in the most classical form simply the quadratic function or the absolute value and $TV(\mathbf{u})$ is an appropriate extension of TV to vector-valued functions. This model is in general non-convex, due to the matching term even when the penalty function Φ is convex, since $h(\mathbf{u}) = I_1(\mathbf{x}+\mathbf{u})$ may be arbitrarily shaped. We will see in Chapters 4 and 5 how this can be handled.

Both the prior and the data term have a large influence on the quality of the estimates. Choosing the correct prior for an application can make a large difference, as is shown in Figure 2.7. In this example, the piecewise constancy assumption of TV is too strict. Relaxing this assumption to piecewise affine functions, can lead to qualitatively much better estimates, especially under challenging conditions. The next chapter will be devoted to priors which can model such assumptions.



(a) I_1

(b) I_2

(c) TV

(d) TV - Error

(e) TGV

(f) TGV - Error

Figure 2.7: Qualitative influence of the prior.

3

Priors for Optical Flow and Stereo

Contents

A well-known and often unwanted effect of models that incorporate Total Variation (TV) is that solutions of these models show so-called staircasing artifacts, *i.e.* the solutions are piecewise constant. In this section we discuss higher-order priors that can be used in stereo and optical flow estimation and aim to avoid the problem of staircasing. We introduce one of the main building blocks of our stereo and optical flow algorithms, Total Generalized Variation (TGV), and show how this prior can be extended in various ways in order to enhance the prior with additional knowledge. In view of the general correspondence model (1.1), this section is concerned with the term $R(u; I_1)$. Some of the priors only take the correspondence field u into account, *i.e.* we have a regularizer $R(u)$ which is independent of the reference image I_1.

3.1 Total Generalized Variation

TGV [26] is a non-trivial generalization of TV. Its aim is to measure not only jumps in the solution, but also higher-order variations like kinks. In its most general form TGV of order k is defined as

$$TGV_\alpha^k(u) = \sup \left\{ \int_\Omega u \operatorname{div}^k q \,\mathrm{d}\mathbf{x} \;\middle|\; q \in \mathcal{C}_\mathrm{c}^k(\Omega, \mathrm{Sym}^k(\Re^2)), \right.$$
$$\left. \left\| \operatorname{div}^l q \right\|_\infty \le \alpha_l,\ l = 0, \dots, k-1 \right\}, \tag{3.1}$$

where $\mathrm{Sym}^k(\Re^2)$ is the space of symmetric tensor fields of order k on $\Re^2$ and $\mathcal{C}_\mathrm{c}^k(\Omega, \mathrm{Sym}^k(\Re^2))$ denotes the space of k-times continuously differentiable, compactly supported functions defined on this tensor space. Finally, $\alpha_l > 0$ are user-defined scalars. Analogous to (2.52), we will refer to this form of TGV as the dual form. It is easy to see hat TGV of order 1 is equivalent to the scaled Total Variation, *i.e.* $TGV_\alpha^1(u) = \alpha\, TV(u)$. The most important properties of TGV and their role in practical implementations are [26] :

1. TGV is a semi-norm and thus convex and one-homogeneous. The convexity enables to use efficient convex solvers for models that involve TGV as a prior. The semi-norm property can be used to analyze the space of functions for which TGV is well-defined.

2. For locally integrable functions u, TGV of order k is equal to zero if and only if u is a polynomial of order $k-1$. As a consequence, solutions of models that involve this prior tend to be piecewise polynomials of order $k-1$. For $k \geq 2$, this is clearly less restrictive than the piecewise constancy assumption of TV and especially useful if desirable solutions can be described in such a way.

3. TGV is rotationally invariant. This is a property that is desirable for many applications, since in most cases rotated versions of an image are equally likely than the non-rotated version. Rotational invariance is most important after discretization, where it can be shown that priors with rotational invariance do not show strong metrication artifacts, since there is less bias to align solutions with the discrete image grid.

4. TGV and TV are equivalent for piecewise constant functions. Thus TGV-based models may give the same piecewise constant solution as a TV-based model, if such a solution is supported by the data. If the data on the other hand supports more smoothness, TGV-based models may also recover solutions which are not piecewise constant. As a result we do not expect to get significantly worse results using TGV over TV, if piecewise constancy is already a good prior assumption for a task, but significantly better results if this assumption is violated.

Note that the order k of TGV has to be chosen by in advance. We will focus on TGV of order 2 in this work, since it allows to encode piecewise affinity of solutions, which is a good prior assumption for correspondence problems and provides a good trade-off between computational complexity and expressive power of the prior. Using (3.1), TGV of order 2 can be written as

$$\begin{aligned} TGV_\alpha^2(u) = \sup \Big\{ \int_\Omega u \,\mathrm{div}^2 q \,\mathrm{d}\mathbf{x} \;\Big|\; & q \in \mathcal{C}_\mathrm{c}^2(\Omega, \mathrm{Sym}^2(\Re^2)), \\ & ||q||_\infty \leq \alpha_0, \quad ||\mathrm{div}\, q||_\infty \leq \alpha_1 \Big\}. \end{aligned} \tag{3.2}$$

$\mathrm{Sym}^2(\Re^2)$ is the space of symmetric 2×2 matrices. Since the symmetry is not strictly required, we can define a slightly different variant of TGV as

$$TGV_\alpha^2(u) = \sup \Big\{ \int_\Omega u \operatorname{div}^2 q \,\mathrm{d}\mathbf{x} \;\Big|\; q \in \mathcal{C}_\mathrm{c}^2(\Omega, \Re^{2\times 2}), \\ ||q||_\infty \le \alpha_0, \quad ||\operatorname{div} q||_\infty \le \alpha_1 \Big\}. \tag{3.3}$$

We will compare both versions later in this section in an empirical denoising experiment, but will mostly use the non-symmetric variant throughout this work.

We will now derive an alternative form of TGV which will be more convenient to use in practice. We call this, analogous to the two equivalent definitions of TV, the primal form.

Proposition 1. *The primal form of* (3.3) *is given by*

$$TGV_\alpha^2(u) = \min_w \Big\{ \alpha_1 \int_\Omega |Du - w| + \alpha_0 \int_\Omega |Dw| \Big\}, \tag{3.4}$$

where $w : \Omega \to \Re^2$.

Proof. From duality of the ℓ_1-norms in (3.4), we get

$$TGV_\alpha^2(u) = \min_w \sup_{\substack{||p||_\infty \le \alpha_1 \\ ||q||_\infty \le \alpha_0}} \int_\Omega (Du - w) \cdot p + \int_\Omega Dw \cdot q. \tag{3.5}$$

(3.5) is a saddle-point problem, which is linear in w, p and q. Morover, the dual variables p and q lie in a closed convex set and are bounded. Thus a saddle-point exists [129, Corollary 37.3.1] and the minimum and the supremum can be interchanged [129, Corollary 37.3.2]:

$$TGV_\alpha^2(u) = \sup_{\substack{||p||_\infty \le \alpha_1 \\ ||q||_\infty \le \alpha_0}} \min_w \Big\{ -\int_\Omega u \operatorname{div} p \,\mathrm{d}\mathbf{x} - \int_\Omega w \cdot (\operatorname{div} q + p) \,\mathrm{d}\mathbf{x} \Big\}. \tag{3.6}$$

w can be understood as a Lagrange multiplier for the constraint $\operatorname{div} q + p = 0$. Minimizing with respect to w and introducing an indicator function

$$I_X(x) = \begin{cases} 0 & \text{if} \quad x \in X \\ \infty & \text{else} \end{cases},$$

we get

$$TGV_\alpha^2(u) = \sup_{\substack{||p||_\infty \le \alpha_1 \\ ||q||_\infty \le \alpha_0}} -\int_\Omega u \operatorname{div} p \,\mathrm{d}\mathbf{x} - I_{\{p = -\operatorname{div} q\}}(p, q). \tag{3.7}$$

It is now easy to see that the indicator function can be eliminated by substituting $p = -\operatorname{div} q$, which yields

$$TGV_\alpha^2(u) = \sup_{\substack{||\operatorname{div} q||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \int_\Omega u \operatorname{div}^2 q \, \mathrm{d}\mathbf{x}, \tag{3.8}$$

which is exactly the dual definition of TGV (3.3). □

Remark 1. A similar result can be derived for the symmetric variant (3.2). the primal formulation then involves a symmetrized gradient operator $(\mathcal{E}w)(x) : \Re^2 \to \operatorname{Sym}(\Re^{2\times 2})$:

$$TGV_\alpha^2(u) = \min_w \Big\{ \alpha_1 \int_\Omega |Du - w| + \alpha_0 \int_\Omega |\mathcal{E}w| \Big\}. \tag{3.9}$$

The symmetric definition stems mainly from the consideration that higher-order derivatives are symmetric, *i.e.* the higher-order derivatives do not depend on the order in which partial differentiation is carried out, and that w is in fact strongly related to the second derivatives of u.

The primal form (3.4) is more convenient in practice than (3.3), since the constrained set in the dual form is very complicated due to the constraints on the divergence of the dual variable. Projection onto this set can not be carried out in closed-form, which slows down the first-order algorithms shown in the previous section. Moreover, the primal form allows for an intuitive interpretation [124] of how this prior acts on a function u: Before the variation of the image u is measured, a vector field w, which is forced to have low variation, is subtracted from the gradient. Affine functions will lead to a constant field w. This (locally) leads to a combined cost of zero for such functions under TGV, which explains the tendency of models involving TGV of second order to produce piecewise affine solutions.

To get a better understanding on how TGV acts on a function and how TV compares, consider the following one-dimensional examples, which where adapted from [26]. An in-depth treatment of the properties of TGV in the context of one-dimensional signals can be found in [115]. All examples are defined on the domain $\Omega =]0, 1[$. Graphical examples of the functions under consideration are depicted in Figure 3.1.

We first analyze a simple linear function:

$$u_{lin}(x) = hx + b \tag{3.10}$$

For this function we have

$$TV(u_{lin}) = \int_\Omega |Du_{lin}| = \int_0^1 \left| u'_{lin} \right| \, \mathrm{d}x = |h|\,. \tag{3.11}$$

Thus TV incurs a cost for linear functions, which is proportional to the slope and the domain of the function. In contrast, for TGV_α^2, we have

$$TGV_\alpha^2(u) = \sup_{\substack{||q'||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \int_0^1 uq'' \,\mathrm{d}x = \sup_{\substack{||q'||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \int_0^1 u''q \,\mathrm{d}x = 0, \tag{3.12}$$

since the second derivative vanishes for a linear function. Another interesting example is a step function

$$u_{stp} = \begin{cases} b_1 & \text{if} \qquad x < c, \quad c \in \,]0,1[\\ b_2 & \text{if} \qquad x > c. \end{cases} \tag{3.13}$$

Using the dual definition of TV, integration by parts and the fact that p is compact on Ω, we get

$$\begin{aligned} TV(u_{stp}) &= \sup_{||p||_\infty \leq 1} \int_0^c u_{stp}p' \,\mathrm{d}x + \int_c^1 u_{stp}p' \,\mathrm{d}x \\ &= \sup_{||p||_\infty \leq 1} [u_{stp}p]_0^c - \int_0^c u'_{stp}p \,\mathrm{d}x + [u_{stp}p]_c^1 - \int_c^1 u'_{stp}p \,\mathrm{d}x \\ &= \sup_{||p||_\infty \leq 1} (b_1 - b_2)p(c) = |b_1 - b_2|. \end{aligned} \tag{3.14}$$

For TGV we can use the same approach to get

$$\begin{aligned} TGV_\alpha^2(u_{stp}) &= \sup_{\substack{||q'||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \int_0^c u_{stp}q'' \,\mathrm{d}x + \int_c^1 u_{stp}q'' \,\mathrm{d}x \\ &= \sup_{\substack{||q'||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} [u_{stp}q']_0^c - \int_0^c u'_{stp}q' \,\mathrm{d}x + [u_{stp}q']_c^1 - \int_c^1 u'_{stp}q' \,\mathrm{d}x \\ &= \sup_{\substack{||q'||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} (b_1 - b_2)q'(c) = \alpha_1|b_1 - b_2|. \end{aligned} \tag{3.15}$$

Thus $TGV_\alpha^2(u_{stp}) = \alpha_1 \, TV(u_{stp})$. Both priors measure the height of the jump.

Let us now consider a piecewise linear function:

$$u_{pl}(x) = \begin{cases} h_1x + b_1 & \text{if} \qquad x < c, \quad c \in \,]0,1[\\ h_2x + b_2 & \text{if} \qquad x > c \end{cases}. \tag{3.16}$$

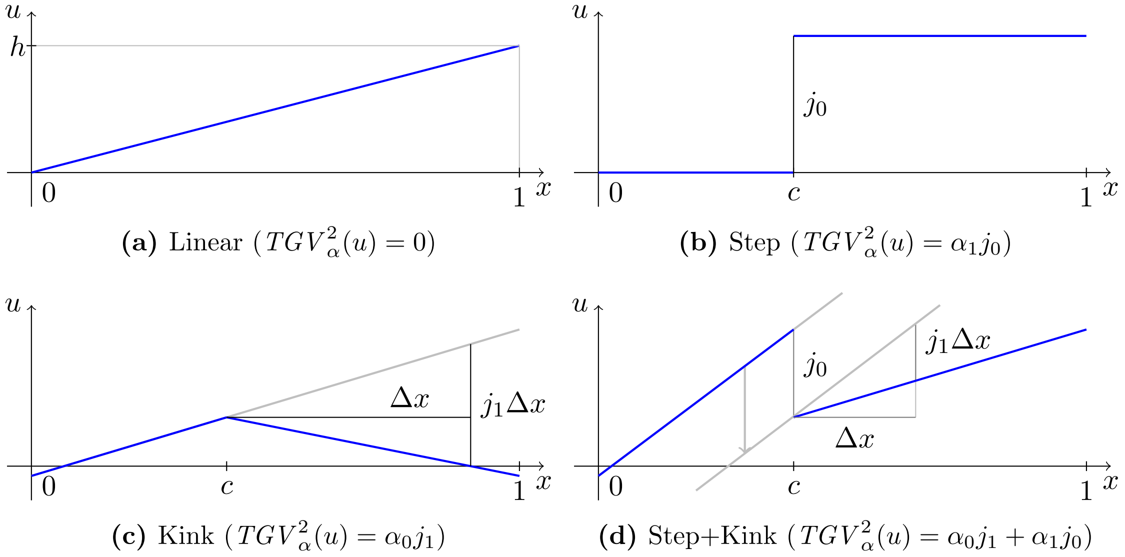
An example of this function is shown in Figure 3.1d. The Total Variation of this function is given by

$$\begin{aligned} TV(u_{pl}) &= \int_0^c |h_1| \,\mathrm{d}x + \int_c^1 |h_2| \,\mathrm{d}x + \int_0^c |(b_1 - b_2)\delta(x-c)| \,\mathrm{d}x \\ &= c\,|h_1| + (1-c)\,|h_2| + |b_1 - b_2|\,, \end{aligned} \tag{3.17}$$

TV measures the jumps between the linear pieces and also integrates the slopes of the pieces. As a consequence TV penalizes not only jumps but also deviations from piecewise constancy. In contrast, for TGV_α^2, we have from [26] for the choice $\alpha_0, \alpha_1 > 0, \alpha_0/\alpha_1 < 1/2$

$$TGV_\alpha^2(u_{pl}) = \alpha_1 |h_1 c + b_1 - h_2 c - b_2| + \alpha_0 |h_1 - h_2|. \tag{3.18}$$

TGV measures the height of the jump between the linear pieces, as well as their absolute difference in slope. Moreover the linear parts themselves incur no cost. This shows that in some sense TGV of second-order measures jump discontinuities as well as second-order discontinuities (kinks). Generally, TGV of order k measures higher-order discontinuities of up to order $k-1$. Each scalar weighting factor α_l is related to an order of the discontinuity and can be used to balance the influence between them.



(a) Linear ($TGV_\alpha^2(u) = 0$) **(b)** Step ($TGV_\alpha^2(u) = \alpha_1 j_0$)

(c) Kink ($TGV_\alpha^2(u) = \alpha_0 j_1$) **(d)** Step+Kink ($TGV_\alpha^2(u) = \alpha_0 j_1 + \alpha_1 j_0$)

Figure 3.1: One-dimensional examples for TGV_α^2

Discretization In order to implement TGV_α^2 we again have to define discrete analogs to the continuous derivative operators. We build on the same finite-differences discretization as previously shown in Section 2.3.2 for the case of TV and assume a rectangular grid of size $N_y \times N_x$. A discretized version of the primal form of TGV^2_α can then be written as

$$TGV^2_\alpha(u) = \min_{w_{i,j} \in \Re^2} \alpha_1 \sum_{i,j} \left|(\nabla u)_{i,j} - w_{i,j}\right|_2 + \alpha_0 \sum_{i,j} \left|(\boldsymbol{\nabla} w)_{i,j}\right|_2 , \tag{3.19}$$

where the operator $\boldsymbol{\nabla}$ is a discrete approximation to the Jacobian, which is defined slightly differently for the symmetric and the asymmetric variants of TGV^2_α. For asymmetric TGV^2_α we have

$$(\boldsymbol{\nabla}^A w)_{i,j} = \begin{pmatrix} (\delta_x w^1)_{i,j} & (\delta_y w^1)_{i,j} \\ (\delta_x w^2)_{i,j} & (\delta_y w^2)_{i,j} \end{pmatrix} . \tag{3.20}$$

For the symmetric case we simply symmetrize the off-diagonal entries of the Jacobian operator:

$$(\boldsymbol{\nabla}^S w)_{i,j} = \begin{pmatrix} (\delta_x w^1)_{i,j} & \frac{(\delta_y w^1)_{i,j} + (\delta_x w^2)_{i,j}}{2} \\ \frac{(\delta_y w^1)_{i,j} + (\delta_x w^2)_{i,j}}{2} & (\delta_y w^2)_{i,j} \end{pmatrix} . \tag{3.21}$$

It is again convenient to rewrite (3.19) in terms of matrix-vector products by stacking u into a vector of length $M = N_x N_y$:

$$TGV^2_\alpha(u) = \min_{w \in \Re^{2M}} \alpha_1 \left|\left|\nabla u - w\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w\right|\right|_{2,1} , \tag{3.22}$$

where the linear operators for the asymmetric and the symmetric case are given by

$$\boldsymbol{\nabla}^A = \begin{pmatrix} \nabla & 0 \\ 0 & \nabla \end{pmatrix} \in \Re^{4M \times 2M}, \qquad \boldsymbol{\nabla}^S = \begin{pmatrix} \nabla_x & 0 \\ 0 & \nabla_y \\ \nabla_y & \nabla_x \end{pmatrix} \in \Re^{3M \times 2M}. \tag{3.23}$$

Optimization Now consider a functional with TGV^2_α prior and some generic convex data term $g : \Re^M \to \Re$. The data term g could be for example a simple quadratic data term as in the ROF model (c.f. Section 2.3.2).

$$\min_u TGV^2_\alpha(u) + g(u) = \min_{u,w} \alpha_1 \left|\left|\nabla u - w\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w\right|\right|_{2,1} + g(u). \tag{3.24}$$

Using dual norms this can be reformulated equivalently as a convex-concave saddle-point problem:

$$\min_{u,w} \max_{p \in P, q \in Q} \langle \nabla u - w, p \rangle + \langle \boldsymbol{\nabla} w, q \rangle + g(u). \tag{3.25}$$

The set P describes for each pixel (i,j) the two-dimensional unit ball scaled by α_1:

$$P = \left\{ p = (p^1, p^2) \in \Re^{2 \times M} : \left|p_{i,j}\right|_2 \leq \alpha_1 \right\} \tag{3.26}$$

The set Q is defined slightly differently depending on whether the asymmetric or the symmetric variant of TGV_α^2 is considered. For asymmetric TGV_α^2 the set Q is given by the four-dimensional scaled unit ball, whereas for the symmetric variant it is given by the three-dimensional scaled unit ball:

$$\begin{aligned} Q^A &= \left\{ q = (q^1, q^2, q^3, q^4) \in \Re^{4 \times M} : |q_{i,j}|_2 \leq \alpha_0 \right\} \\ Q^S &= \left\{ q = (q^1, q^2, q^3) \in \Re^{3 \times M} : |q_{i,j}|_2 \leq \alpha_0 \right\}. \end{aligned} \tag{3.27}$$

It is easy to see that in any case the set is convex. Moreover, it is defined point-wise which allows for efficient projections onto the sets. The saddle-point problem (3.25) can be optimized using the primal-dual algorithm which was already introduced in the previous section. The iterations of the algorithm are shown in Algorithm 3.1. The algorithm in this form can be applied to any model with TGV_α^2 regularization and a convex data term g. If the proximal operator of g can be evaluated efficiently, we can also expect an overall efficient algorithm. The primal-dual algorithm converges for step-sizes $\tau\sigma \left|\left|A\right|\right|^2 \leq 1$, where $\left|\left|A\right|\right|^2$ is the operator norm of A. To derive this norm let us first rewrite the saddle-point problem (3.25) into the canonical primal-dual form (2.41):

$$\min_{u,w} \max_{p \in P, q \in Q} \left\langle \begin{bmatrix} \nabla & -I \\ 0 & \boldsymbol{\nabla} \end{bmatrix} \begin{pmatrix} u \\ w \end{pmatrix}, \begin{pmatrix} p \\ q \end{pmatrix} \right\rangle + g(u). \tag{3.28}$$

It is now easy to see that in order to find the step-sizes τ and σ it is necessary to estimate the operator norm of

$$A = \begin{bmatrix} \nabla & -I \\ 0 & \boldsymbol{\nabla} \end{bmatrix}. \tag{3.29}$$

```
1.  Set $u_0 = \bar{u}_0 \in \Re^M$, $w_0 = \bar{w}_0 \in \Re^{2M}$
2.  Set $p \in P$, $q \in Q$, set $k = 0$
3.  While not converged
        $p_{k+1} = \mathbf{proj}_{p \in P}(p_k + \sigma(\nabla \bar{u}_k - \bar{w}_k))$
        $q_{k+1} = \mathbf{proj}_{q \in Q}(q_k + \sigma \boldsymbol{\nabla} \bar{w}_k)$
        $u_{k+1} = \mathbf{prox}_{\tau g}(u_k - \tau \nabla^* p_{k+1})$
        $w_{k+1} = w_k - \tau(\boldsymbol{\nabla}^* q_{k+1} - p_{k+1})$
        $\bar{u}_{k+1} = 2u_{k+1} - u_k$
        $\bar{w}_{k+1} = 2w_{k+1} - w_k$
        $k = k + 1$
```

Algorithm 3.1: Primal-dual algorithm for optimization of a TGV_α^2-based model.

We have

$$||A||_2^2 = ||A^*||_2^2 = L^2$$

and by the definition of the operator norm for any vector u, w it holds that

$$||Av||_2 \leq L\,||v||_2\,, \quad v = \begin{pmatrix} u \\ w \end{pmatrix} \tag{3.30}$$

A quick way to get a rough estimate of L is to use the following decomposition:

$$||A||_2 \leq \sqrt{||A||_1\,||A||_\infty} \leq \sqrt{15}. \tag{3.31}$$

It turns out, however, that we can get a better estimate by having a closer look at the structure of this operator. We now assume the symmetric variant of TGV, the asymmetric case can be handled analogously. We first decompose the operator into a sum over the individual image pixels:

$$\begin{aligned} ||Av||_2^2 = \sum_{i,j} &(u_{i+1,j} - u_{i,j} - w_{i,j}^1)^2 + (u_{i,j+1} - u_{i,j} - w_{i,j}^2)^2 \\ &+ (w_{i+1,j}^1 - w_{i,j}^1)^2 + (w_{i,j+1}^1 - w_{i,j}^1)^2 \\ &+ (w_{i+1,j}^2 - w_{i,j}^2)^2 + (w_{i,j+1}^2 - w_{i,j}^2)^2. \end{aligned} \tag{3.32}$$

Using Jensen's inequality the individual squares in this sum can be estimated as

$$(\sum_n x_n)^2 \leq n \sum_n x_n^2, \tag{3.33}$$

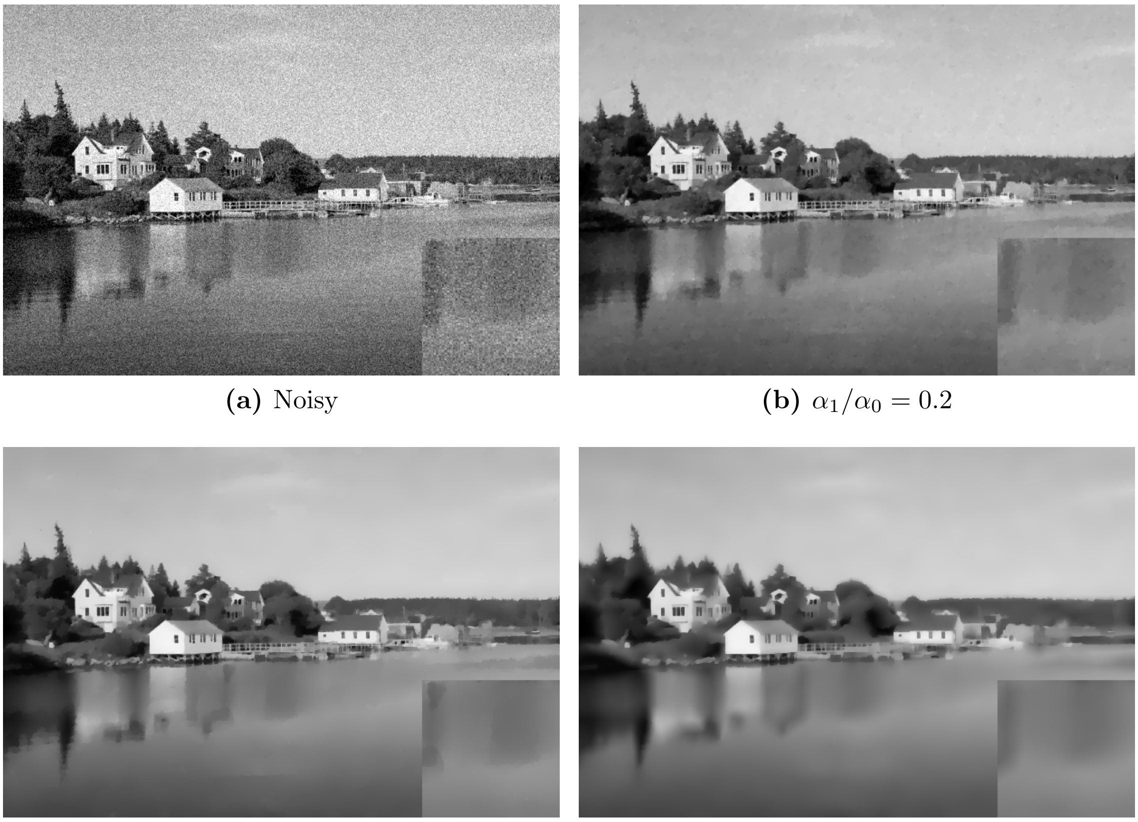
which yields the estimate

$$\begin{aligned} ||Av||_2^2 \leq \sum_{i,j} &3\Big((u_{i+1,j})^2 + (u_{i,j+1})^2 + (w_{i,j}^1)^2 + (w_{i,j}^2)^2\Big) + 6(u_{i,j})^2 + \\ &2\Big((w_{i+1,j}^1)^2 + (w_{i,j+1}^1)^2 + (w_{i+1,j}^2)^2 + (w_{i,j+1}^2)^2 + 2(w_{i,j}^1)^2 + 2(w_{i,j}^2)^2\Big) \\ &\leq 12\,||v||_2^2 \end{aligned} \tag{3.34}$$

Thus, Algorithm 3.1 is guaranteed to converge for stepsizes of the form $\tau = \frac{c}{\sqrt{12}}, \sigma = \frac{1}{c\sqrt{12}}$, where $c > 0$ can be chosen arbitrarily.

Finally, Figure 3.2 shows an example of TGV-based denoising with a quadratic data term and compares different ratios of the weights α_0 and α_1. The results are governed by the ratio between those weights. Higher ratios lead to smoother results, whereas low ratios lead to results similar to a TV-based model.

Symmetric vs. Asymmetric In this section we show the practical difference between the symmetric and the asymmetric variant of TGV_α^2. We generate piecewise affine test


(a) Noisy

(b) $\alpha_1/\alpha_0 = 0.2$

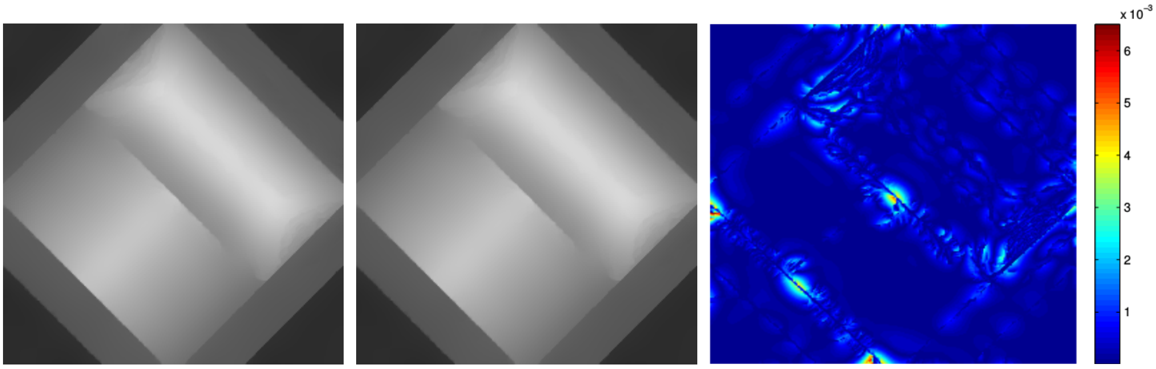
(c) $\alpha_1/\alpha_0 = 0.3$

(d) $\alpha_1/\alpha_0 = 0.5$

Figure 3.2: Denoising with TGV. The behaviour is governed by the ratio of α_1 to α_0. For small ratios jumps are penalized less severely, which results in a behaviour similar to TV.

images and apply a modest amount of additive white Gaussian noise. The test images are then denoised using a quadratic data term as in the ROF model 2.62 and TGV_α^2 in its symmetric and asymmetric variant as prior. We use the generic primal-dual algorithm 3.1 for TGV_α^2 based models, where the proximal operator with respect to the data term is given by (2.66).

Figure 3.3 shows a visual comparison of both approaches. The visual differences between both images are so small that they are indistinguishable by the naked eye. A comparison in terms of absolute differences between both variants is shown in Figure 3.3c, where it can be seen that the maximum per pixel difference is on the order of 10^{-3}. We choose the asymmetric variant for the remainder of this work, since it better fits to the formulation of Non-Local TGV. The main drawback of the asymmetric variant is its slightly larger memory footprint due to the 4-dimensional dual variable q. For GPU-based implementations this problem is mitigated by the requirement of word-aligned vector variables, which results in the need to store the dual variable as an array of 4-dimensional float vectors to get optimal performance even for the symmetric variant.



(a) Symmetric **(b)** Asymmetric **(c)** Difference image

Figure 3.3: Comparison of asymmetric and symmetric TGV.

3.1.1 Image-Driven TGV

Priors can be further enhanced by conditioning them on additional input cues. In the case of dense corresondence problems, we can exploit the fact that depth or optical flow edges may only appear at locations where there are intensity edges in the reference image. This assumption can be easily incorporated into the TGV prior by either replacing the isotropic norm in (3.4) by an anisotropic variant, or by modifying the gradient operators appropriately [124]: We define the Image-Driven Total Generalized Variation (ITGV) by introducing a positive-definite diffusion tensor $M^{\frac{1}{2}} : \Re^2 \to \Re^{2\times 2}$:

$$ITGV^2_\alpha(u) = \min_{w\in\mathbb{R}^2}\Big\{\alpha_1 \int_\Omega |M^{\frac{1}{2}}(Du - w)| + \alpha_0 \int_\Omega |Dw|\Big\}. \tag{3.35}$$

It is important to note that the diffusion tensor is typically spatially varying and depends on some guidance image I. We drop this explicit spatial dependence to allow for uncluttered notation.

Proposition 2. *The dual form of $ITGV^2_\alpha$ is given by*

$$ITGV^2_\alpha(u) = \sup_{\substack{||\mathrm{div}\, q(x)||_{M^{-1}(\mathbf{x})} \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \int_\Omega u \,\mathrm{div}^2 q \,\mathrm{d}\mathbf{x}, \tag{3.36}$$

were $||x||_Q = \langle x, Qx\rangle^{\frac{1}{2}}$.

Proof. We follow the same argument as in Proposition 1. The primal-dual form of $ITGV^2_\alpha$ is given by

$$ITGV^2_\alpha(u) = \sup_{\substack{||p||_\infty \leq \alpha_1 \\ ||q||_\infty \leq \alpha_0}} \min_w \Big\{ -\int_\Omega u \,\mathrm{div}\, M^{\frac{1}{2}} p \,\mathrm{d}\mathbf{x} - \int_\Omega w \cdot (\mathrm{div}\, q + M^{\frac{1}{2}} p)\,\mathrm{d}\mathbf{x}\Big\}. \tag{3.37}$$

Optimality with respect to w is reached if $p = -M^{-\frac{1}{2}} \operatorname{div} q$ holds. By substituting p and noting that

$$\left\langle Q^{-\frac{1}{2}} x, Q^{-\frac{1}{2}} x \right\rangle^{\frac{1}{2}} = ||x||_{Q^{-1}}\,, \tag{3.38}$$

the dual version (3.36) follows. □

We again employ a one-dimensional example to show the effects of such a data-driven parametrization of the regularizer. The one-dimensional case can be reduced to the form

$$ITGV_\alpha^2(u) = \sup_{\substack{||q'(x)||_2 \le \alpha_1 m(x) \\ ||q||_\infty \le \alpha_0}} \int_0^1 u q'' \, \mathrm{d}x. \tag{3.39}$$

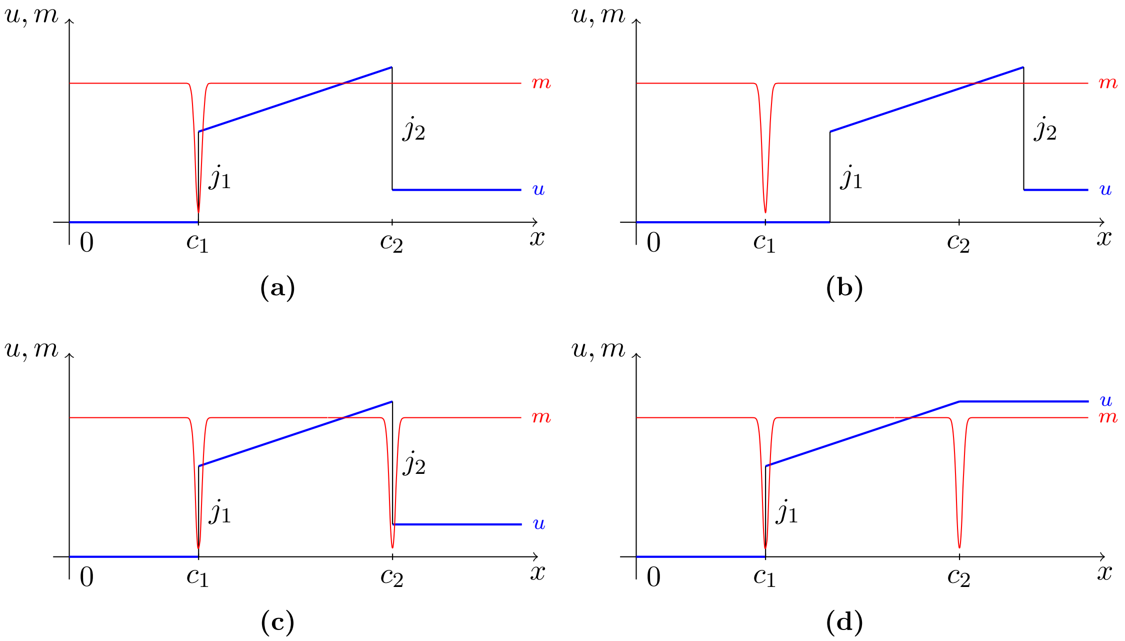
If we consider the piecewise linear function u_{pl} and assume that $m(x) > 2\alpha_0/\alpha_1$ and smooth on the sub-intervals $]0, c[$ and $]c, 1[$ then

$$\begin{aligned} ITGV_\alpha^2(u_{pl}) &= \sup_{\substack{||q'(x)||_2 \le \alpha_1 m(x) \\ ||q||_\infty \le \alpha_0}} q'(c)(h_2 c + b_2 - h_1 c - b_1) + q(c)(h_1 - h_2) \\ &= \alpha_1 m(c) \left| h_2 c + b_2 - h_1 c - b_1 \right| + \alpha_0 \left| h_1 - h_2 \right|, \end{aligned} \tag{3.40}$$

by applying integration by parts twice and using the compactness of q. With this approach it is obviously possible to steer the cost of jumps according to the function $m(x)$. High values of $m(c)$ incur a high cost for a jump at c, which will result in the tendency to produce no jump in the solution. Conversely low values incur similar costs as without the weighting and will allow for jumps at this location. Figure 3.4 shows a graphical example with a piecewise linear function u, that has two constant pieces and a middle piece with slope k. For a general guidance function m we have

$$ITGV_\alpha^2(u) = \alpha_1 m(c_1) \left| j_1 \right| + \alpha_1 m(c_2) \left| j_2 \right| + 2\alpha_0 |k|. \tag{3.41}$$

In Figure 3.4a the guidance function is constructed in a way such that jumps which are not near c_1 incur a high cost. This is the case for the jump at c_2. Any model that minimizes $ITGV_\alpha^2$ will thus prefer to smooth out the jump at c_2 and exactly localize the jump at c_1. Figure 3.4b shows the same example for a shifted version of u. In this case both jumps incur high cost, which indicates that this solution is unlikely. With the guidance function shown in Figure 3.4c, both jumps get assigned a low cost, and thus are likely to be in the solution of a model that minimizes this regularizer. The guidance function does not influence the second-order part: The example in Figure 3.4d in fact gives a lower cost than 3.4c, since both show the same kink, but Figure 3.4d completely lacks the jump. For very small $m(c_2)$, however, the costs of both configurations are approximately equal. A one-dimensional denoising example is shown in Figure 3.5, where the guidance function was constructed based on the jumps in the groundtruth signal. It can be seen that *ITGV* provides sharper jumps and is better able to reconstruct the piecewise linear groundtruth signal.



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

Figure 3.4: Example of *ITGV* on a piecewise linear function with different guidance functions.

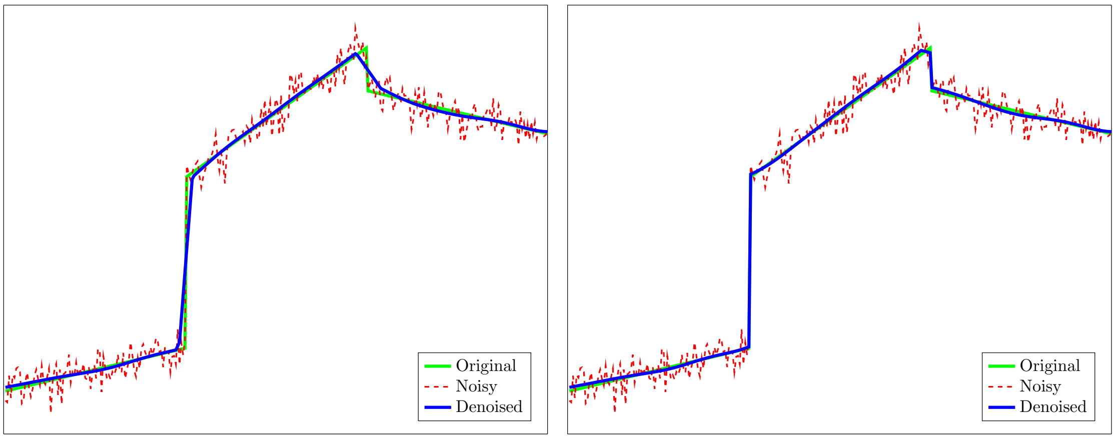
In the two-dimensional case the behavior can also be steered along a specific direction by forming the diffusion tensor $M^{\frac{1}{2}}$ appropriately. If we choose a diagonal matrix with entries which are equal, we reflect a prior for a jump discontinuity which is equally likely in both directions. A more sophisticated choice is a positive-definite matrix, which allows to prefer certain directions (e.g. the directions of the eigenvectors via the magnitude of the corresponding eigenvalues of the operator). One such choice is given by the Nagel-Enkelmann operator [107]:

$$M^{\frac{1}{2}} = \exp(-\gamma \left|\nabla I\right|_2^{\beta}) n n^T + n^{\perp} {n^{\perp}}^T, \tag{3.42}$$

where I is a guidance image, n is related to the gradient of I as

$$n(x) = \begin{cases} \frac{\nabla I}{\left|\nabla I\right|_2} & \text{if } \left|\nabla I\right|_2 \geq \epsilon \\ (1,0)^T & \text{else} \end{cases}.$$

$n^{\perp}$ is a unit vector perpendicular to n. Finally, γ, $\beta > 0$ are user-chosen parameters and ϵ is a small, positive threshold . This choice of diffusion tensor anisotropicaly scales



(a) *TGV* **(b)** *ITGV*

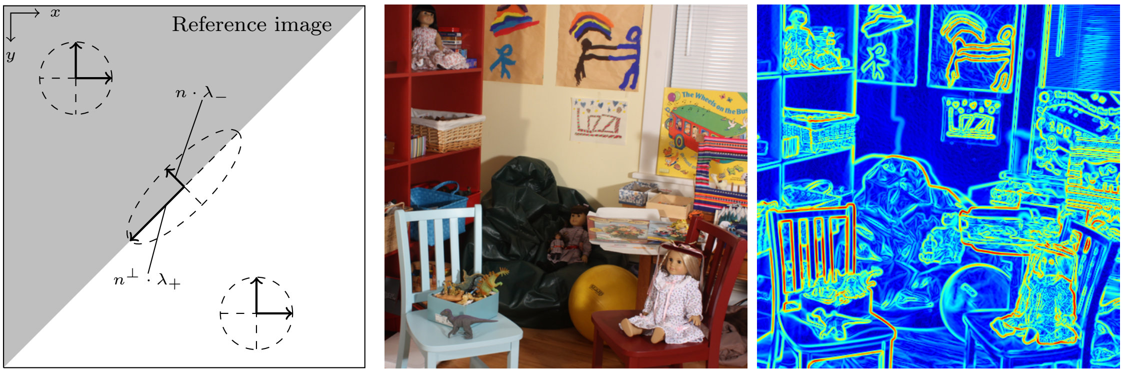
Figure 3.5: Comparison of 1D denoising with *ITGV* and *TGV*. The guidance function was set to one everywhere, except near the location of the jumps, where it was set close to zero. *ITGV* results in sharper jumps. For small jumps in the groundtruth data, *TGV* tends to approximate the jump by a series of kinks (right jump), whereas *ITGV* can successfully recover a jump.

the first-order term based on the image content: For areas with small gradients in the guidance image the matrix is approximately the identity matrix, thus the regularizer shows no preference for a particular direction. In areas with a large gradient $M^{\frac{1}{2}} \approx n^{\perp}{n^{\perp}}^{T}$, which has eigenvectors n and $n^{\perp}$ and associated eigenvalues 0 and 1, respectively. As a consequence the jump term $Du - v$ is attenuated if it lies in the direction of the image gradient (*i.e.* accross the guidance edge). Directions perpendicular to the gradient (*i.e.* along the guidance edge) are still (approximately) subject to the full cost. In a energy minimization framework, this results in solutions which are smooth along edges and sharp across edges. An important factor in practice is also that in stereo or optical flow models this data-driven approach leads to much better localized edges, since the data term is not the only source of information for the location of discontinuities.

A schematic example of this operator is shown in Figure 3.6. We also show the degree of anisotropy for an example image measured as the ratio of eigenvalues $\frac{\lambda_+}{\lambda_-}$.

3.1.2 Non-Local TGV

The inherent locality of the previously discussed priors is often problematic when large areas have ambiguous data terms. In this case the regularizer has to be able to effectively propagate information over those areas. It is often possible to build stronger priors for matching problems by incorporating large areas of interest and conditioning them on the reference image [166].



(a) Schematic overview

(b) Anisotropy. Warmer colors denote higher anisotropy.

Figure 3.6: Graphical illustration of the Nagel-Enkelmann operator. (a) On an edge, the eigenvector associated to the smaller eigenvalue λ_- points across the edge (skewed ellipse). In homogeneous regions the both eigenvalues are equal to one (circles). (b) Anisotropy of the operator plotted as $\frac{\lambda_+}{\lambda_-}$. The anisotropy is high along image edges and low in homogeneous regions.

Non-Local Total Variation (NLTV) [61] for example can be defined as:

$$NLTV(u) = \int_\Omega \int_\Omega \alpha(\mathbf{x}, \mathbf{y}) |u(\mathbf{x}) - u(\mathbf{y})| \, \mathrm{d}\mathbf{y} \, \mathrm{d}\mathbf{x}. \tag{3.43}$$

Here, the non-negative support weights $\alpha(\mathbf{x}, \mathbf{y})$ allow to incorporate additional prior information into the regularization term, *i.e.* $\alpha(\mathbf{x}, \mathbf{y})$ can be used to strengthen the regularization in large areas, which is especially useful in the presence of ambiguous data terms. Variants of this regularizer have been successfully applied to the task of optical flow estimation [91, 145, 166].

Motivated by $NLTV$, a non-local extension of the TGV^2 regularizer was introduce in [123]:

Definition 5. *Let $u : \Omega \to \Re$, $w : \Omega \to \Re^2$ and $\alpha_0, \alpha_1 : \Omega \times \Omega \to \Re^+$ be support weights. We define the Non-Local Second-Order Total Generalized Variation regularizer $NLTGV(u)$ as*

$$\begin{aligned} NLTGV(u) = \min_w \int_\Omega \int_\Omega \alpha_1(\mathbf{x}, \mathbf{y}) \, |u(\mathbf{x}) - u(\mathbf{y}) - \langle w(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle| \, \mathrm{d}\mathbf{y} \, \mathrm{d}\mathbf{x} \\ + \sum_{d=1}^{2} \int_\Omega \int_\Omega \alpha_0(\mathbf{x}, \mathbf{y}) \left| w^d(\mathbf{x}) - w^d(\mathbf{y}) \right| \mathrm{d}\mathbf{y} \, \mathrm{d}\mathbf{x}, \end{aligned} \tag{3.44}$$

where vector components are denoted by super-scripts, i.e. $w(\mathbf{x}) = (w^1(\mathbf{x}), w^2(\mathbf{x}))^T$.

The reasoning behind this definition is as follows: Considering a point $\mathbf{x} \in \Omega$, the expression $u(\mathbf{x}) - \langle w(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle$ defines a plane through the point $(\mathbf{x}, u(\mathbf{x}))$, with normal vector $(w(\mathbf{x}), -1)^T$. Consequently the inner integral of the first expression,

$$\int_\Omega \alpha_1(\mathbf{x}, \mathbf{y}) |u(\mathbf{x}) - u(\mathbf{y}) - \langle w(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle \,| \,\mathrm{d}\mathbf{y}, \tag{3.45}$$

measures the total deviation of u from the plane at the point $\mathbf{x}$, weighted by the support function α_1. A one-dimensional example of this first-order part is visualized in Figure 3.7. The outer integral evaluates this deviation at every point in the image. This term can be understood as a linearization of u around a point $\mathbf{x}$. The linearization is not constant, *i.e.* as we are interested in a field w which minimizes the total deviations from the (in the continuous setting infinitely many) local planes, the normal vector $w(\mathbf{x})$ can vary, although not arbitrarily as the term

$$\sum_{d=1}^{2} \int_\Omega \int_\Omega \alpha_0(\mathbf{x}, \mathbf{y}) \left| w^d(\mathbf{x}) - w^d(\mathbf{y}) \right| \,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{x} \tag{3.46}$$

forces the field w to have low Non-Local Total Variation itself. Intuitively (3.44) assigns low values to functions u which can be well approximated by affine functions.

Analogous to TGV, we now derive primal-dual and dual representations of (3.44), which will later serve as the basis for the optimization of functionals that incorporate this regularizer.

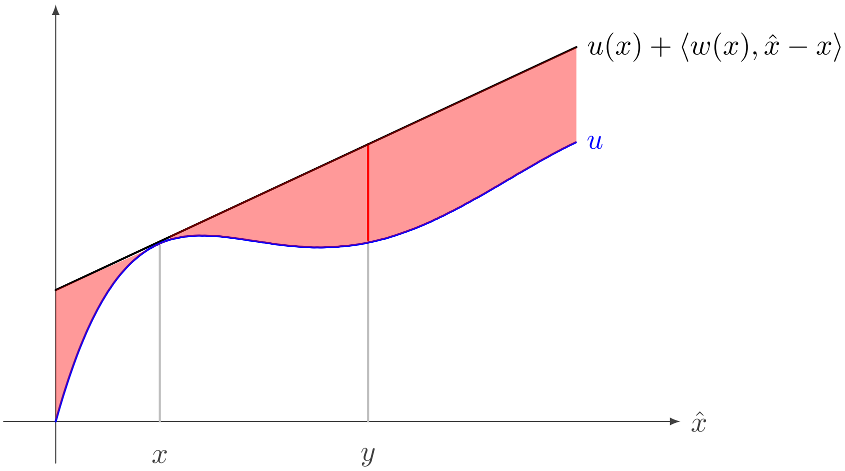


Figure 3.7: One-dimensional visualization of the first-order part of Non-Local Total Generalized Variation (NLTGV). The difference between the linearization and the original function u (visualized in red) is integrated over the whole domain of u.

Proposition 3. *The dual of* (3.44) *is given by*

$$\begin{aligned} NLTGV(u) = &\sup_{\substack{|p(\mathbf{x},\mathbf{y})|\leq\alpha_1(\mathbf{x},\mathbf{y})\\ |q^d(\mathbf{x},\mathbf{y})|\leq\alpha_0(\mathbf{x},\mathbf{y})}} \int_\Omega \left(\int_\Omega \{p(\mathbf{x},\mathbf{y}) - p(\mathbf{y},\mathbf{x})\}\,\mathrm{d}y\right) u(x)\mathrm{d}x \\ s.t. \quad &\int_\Omega q^d(\mathbf{x},\mathbf{y}) - q^d(\mathbf{y},\mathbf{x})\,\mathrm{d}\mathbf{y} = \int_\Omega p(\mathbf{x},\mathbf{y})(x^d - y^d)\,\mathrm{d}\mathbf{y} \quad \forall d \in \{1,2\} \end{aligned} \tag{3.47}$$

Proof. Dualizing the ℓ_1-norms in (3.44) yields

$$\begin{aligned} NLTGV(u) = &\min_w \sup_{|p(\mathbf{x},\mathbf{y})|\leq\alpha_1(\mathbf{x},\mathbf{y})} \int_\Omega\int_\Omega (u(\mathbf{x}) - u(\mathbf{y}) - \langle w(\mathbf{x}), \mathbf{x}-\mathbf{y}\rangle)\cdot p(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y} \\ &+ \sum_{d=1}^2 \sup_{|q^d(\mathbf{x},\mathbf{y})|\leq\alpha_0(\mathbf{x},\mathbf{y})} \int_\Omega\int_\Omega (w^d(\mathbf{x}) - w^d(\mathbf{y}))\cdot q^d(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y} \\ &= \min_w \sup_{\substack{|p(\mathbf{x},\mathbf{y})|\leq\alpha_1(\mathbf{x},\mathbf{y})\\ |q^d(\mathbf{x},\mathbf{y})|\leq\alpha_0(\mathbf{x},\mathbf{y})}} \int_\Omega \left(\int_\Omega \{p(\mathbf{x},\mathbf{y}) - p(\mathbf{y},\mathbf{x})\}\;\mathrm{d}\mathbf{y}\right) u(\mathbf{x})\,\mathrm{d}\mathbf{x} \\ &+ \sum_{d=1}^2 \int_\Omega \left(\int_\Omega \left\{q^d(\mathbf{x},\mathbf{y}) - q^d(\mathbf{y},\mathbf{x}) + p(\mathbf{x},\mathbf{y})(y^d - x^d)\right\}\;\mathrm{d}\mathbf{y}\right) w^d(\mathbf{x})\,\mathrm{d}\mathbf{x}. \end{aligned} \tag{3.48}$$

By taking the minimum with respect to w we arrive at the dual form. □

We will now show two basic properties of *NLTGV*:

Proposition 4. *$NLTGV(u)$ is a semi-norm.*

Proof. To show this statement, consider that the supremum in (3.47) is taken over linear functions with additional linear constraints on p and q. It is well-known that the supremum over linear functions is convex [21]. Since the constraints on p and q form a linear and thus convex set, $NLTGV(u)$ is convex. Moreover it is easy to see from (3.47) that $J(u)$ is positive one-homogeneous. As a consequence the triangle inequality holds, which establishes the semi-norm property. □

Proposition 5. *$NLTGV(u) = 0$ if and only if u is affine.*

Proof. Assume that u is affine, *i.e.* $u(\mathbf{x}) = \langle a, \mathbf{x}\rangle + b, \quad a \in \Re^2$. By plugging into (3.44) it is easy to see that the minimum is attained at $w(\mathbf{x}) = a$. As a consequence we have $J(u) = 0$. Conversely assume that $NLTGV(u) = 0$. In any case this requires that

$$\sum_{d=1}^2 \int_\Omega\int_\Omega \alpha_0(\mathbf{x},\mathbf{y}) \left|w^d(\mathbf{x}) - w^d(\mathbf{y})\right|\;\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{x} = 0, \tag{3.49}$$

which implies that $w(x) = c \in \Re^2$, $\forall \mathbf{x} \in \Omega$. Consequently

$$\min_c \int_\Omega \int_\Omega \alpha_1(\mathbf{x}, \mathbf{y}) \left| u(\mathbf{x}) - u(\mathbf{y}) - \langle c, \mathbf{x} - \mathbf{y} \rangle \right| \mathrm{d}\mathbf{y} \, \mathrm{d}\mathbf{x} = 0, \tag{3.50}$$

if and only if $u(\mathbf{x})$ is of the form $u(\mathbf{x}) = \langle a, \mathbf{x} \rangle + b$ and hence affine. □

Since the properties in Proposition 4 are shared by TGV and $NLTGV$, it can be expected that both behave qualitatively similar when used in an energy minimization framework. The main advantage of $NLTGV$ is the larger support size and the possibility to enforce additional prior knowledge using the support weights α_1 and α_0. This is especially advantageous for optical flow and stereo estimation, where support weights can be readily computed from a reference image, in order to allow better localization of motion and depth boundaries and to resolve ambiguities. Akin to [166] the support weights α_1 and α_0 can be used to incorporate soft-segmentation cues into the regularizer, e.g. in the case of optical flow or stereo estimation it is possible to locally define regions which are forced to have similar motion or lie on the same plane based on the reference image, respectively. We compute support weights based on color similarities and spatial proximity as

$$\alpha_1(\mathbf{x}, \mathbf{y}) = \frac{1}{Z(\mathbf{x})} \exp(-\tfrac{||I_1(\mathbf{x}) - I_1(\mathbf{y})||}{w_c}) \exp(-\tfrac{||\mathbf{x} - \mathbf{y}||}{w_p}), \quad \alpha_0(\mathbf{x}, \mathbf{y}) = c\alpha_1(\mathbf{x}, \mathbf{y}), \tag{3.51}$$

where w_c and w_p are user-chosen parameters that allow to weight the influence of the individual terms and $Z(\mathbf{x})$ ensures that the support weights sum to one. An example of patches and their associated support weights is shown in Figure 3.8. It is important to

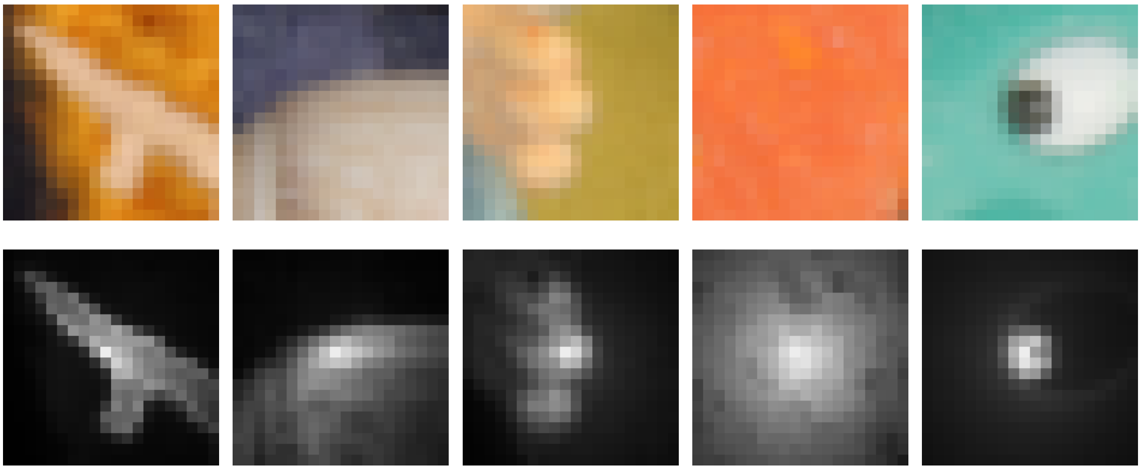
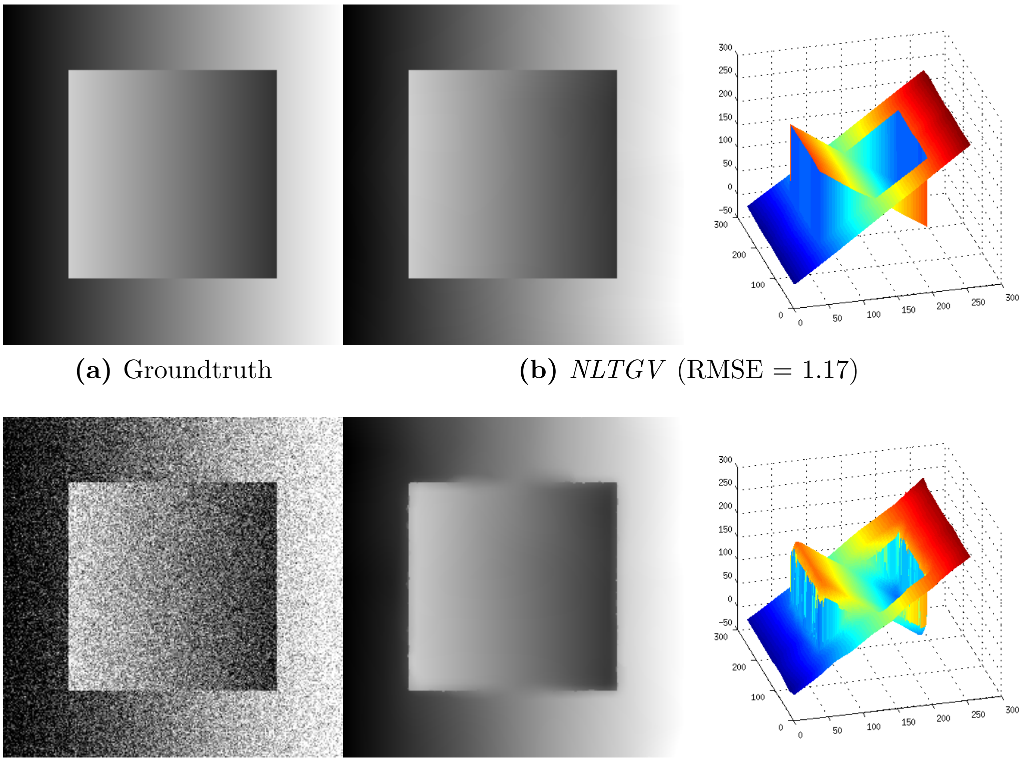


Figure 3.8: Examples of guidance patches (top row) and their associated support weights (bottom row). Pixels with similar color get high weight and thus are forced to approximately lie on the same plane. The influence of the weight is regulated also by spatial proximity, which decays exponentially.

note that the exponential decay of the support weights in terms of proximity is not a pure modeling choice. It also enables a tractable implementation, since models which feature support weights that are non-zero over very large areas quickly lead to intractable models. If the support weights are vanishing outside of a certain region they can simply be ignored in the final model, which enables efficient minimization of non-local models in practice.

Figure 3.9 shows a synthetic experiment which demonstrates the qualitative behavior of $NLTGV$. We denoise a piecewise linear function using a quadratic data term with TGV and $NLTGV$, respectively. We assume prior knowledge of jumps in order to compute the support weights and set $\alpha_1(\mathbf{x}, \mathbf{y}) = 1$ if there is no discontinuity between $\mathbf{x}$ and $\mathbf{y}$ and $\alpha_1(\mathbf{x}, \mathbf{y}) = 0.1$ otherwise. Support weights outside of a 5×5 window were set to zero. While prior knowledge of jumps is not available in real denoising problems, similar support weights can be easily derived in optical flow estimation from the input images. It can be seen that $NLTGV$ nearly perfectly reconstructs the original image, while TGV has problems with accurate localization of the discontinuities.



(a) Groundtruth

(b) $NLTGV$ (RMSE = 1.17)

(c) Noisy

(d) TGV (RMSE = 5.59)

Figure 3.9: Comparison of $NLTGV$ and TGV for the denoising of a synthetic image. $NLTGV$ is able to perfectly reconstruct the groundtruth image. TGV tends to oversmooth jumps.

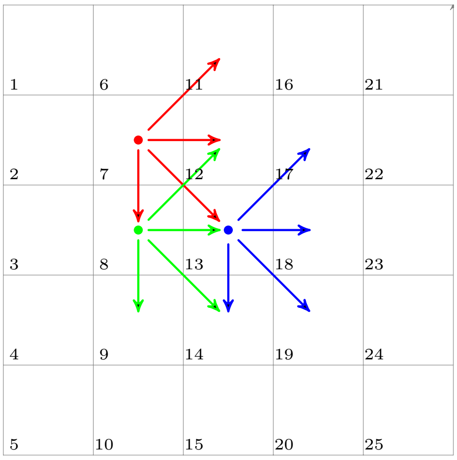


Figure 3.10: Construction of discrete *NLTGV*. The example shows the connections for three pixels (red, green, blue) and their connections to neighboring pixels, assuming a patch size of 3 × 3. Numbers in the grid denote pixel indices k.

Discretization For the discretization of NTLGV we adopt the convention that the data is stacked column-wise into a vector and the integers k, l index specific elements in this vector. In order to allow for a simpler notation, we introduce a signed distance $d_{kl} = (d^1_{kl}, d^2_{kl})^T = l_k - l_l$, where $l_k = (x^1(k), x^2(k))^T$ is the location of pixel k in the image plane. Using these conventions, the spatially discrete Non-Local Total Generalized Variation of $u \in \Re^K$ is given by

$$NLTGV(u) = \min_w \sum_k \sum_{l>k} \left[(\alpha_1)_{kl} |(u_k - u_l + d_{kl} w_k)| + (\alpha_0)_{kl} \, ||w_k - w_l||_1\right], \tag{3.52}$$

where $w \in \Re^{2K}$.

Let $p^{kl} \in \Re$ and $q^{kl} \in \Re^2$ be the dual variable associated to the connection of pixels k and l. The discretized model can be written in its primal-dual formulation as

$$NLTGV(u) = \min_w \max_{\substack{|p_{kl}| \leq (\alpha_1)_{kl} \\ ||q_{kl}||_\infty \leq (\alpha_0)_{kl}}} \sum_k \sum_{l>k} \left[(u_k - u_l + d_{kl} w_k) \cdot p_{kl} + (w_k - w_l) \cdot q_{kl}\right] \tag{3.53}$$

Remark 2. In order to prevent double counting of edges we set the support weights in (3.53) to zero for all $y^1(k) \leq x^1(l)$ or $(y^2(k) \leq x^2(l)) \wedge (y^1(k) \leq x^1(l))$, where $\mathbf{x}$ and $\mathbf{y}$ are points in the image plane and superscripts denote their horizontal and vertical coordinates, respectively. This convention is implicitly handled by the inner sum $\sum_{k>l}$, since it only runs over pixels which have not been traversed previously. An example of this construction is shown in Figure 3.10.

3.1.3 Vectorial TGV

Many practical applications deal with vectorial data. Image restoration algorithms need to be able to handle color images, for example. For correspondence problems vectorial regularizes are also of interest, since optical flow is a two-dimensional quantity. The straight-forward approach to extend scalar regularizers to vectorial data is to simply apply the regularizer on each channel individually. Correlation between channels is not exploited in this approach and it thus may lead to artifacts.

For TV there have been proposed several valid, but distinct, vectorial extensions [19, 27, 63], which aim to exploit the inter-channel correlations. Note that there is no *correct* extension to the vectorial setting, since the results and the form of a good prior for vectorial data strongly depend on the application, or more accurately on the properties of the vector space for which it is defined. TGV has also been extended to the vectorial case: Bredies [25] defines a vectorial variant of TGV_α^2, which acts on multi-channel images $u : \Omega \to \Re^C$:

$$VTGV_\alpha^2(u) = \sup \left\{ \int_\Omega \sum_{c=1}^{C} u_c \operatorname{div}^2 q_c \,\mathrm{d}\mathbf{x} \;\middle|\; q \in \mathcal{C}_\mathrm{c}^k(\Omega, \operatorname{Sym}^2(\Re^2))^C, \right.$$
$$\left. \left|\left| \operatorname{div}^l q \right|\right|_{\infty,*,l} \leq \alpha_l,\ l = 0, 1 \right\}. \tag{3.54}$$

An important feature of this definition is that it preserves the most important properties of TGV [25]. $||.||_{\infty,*,l}$ are generic dual tensor norms, which need to be defined based on the properties of the multi-channel image. Bredies [25] defines these norms via the Euclidean norm over the image channels. This results in a tensor variant of the Frobenius norm in the dual. Based on (3.54), the discrete primal vectorial TGV_α^2 is given by:

$$VTGV_\alpha^2(u) = \min_w \alpha_1 \left|\left| \nabla u - w \right|\right|_{o,1} + \alpha_0 \left|\left| \boldsymbol{\nabla} v \right|\right|_{o,1}, \tag{3.55}$$

where $||.||_{o,1}$ denotes the dual norm to the discrete variant of $||.||_{\infty,*}$ and the linear operators are replicated for each color channel. By choosing the Euclidean norm over color channels, we get:

$$||v||_{o,1} = \sum_{i,j} \sqrt{\sum_{c=1}^{C} \sum_{k=1}^{K} (v_c^k)_{i,j}}\,, \tag{3.56}$$

where K corresponds to the number of components per pixel ($i.e.$ $K = 2$ for the first-order term and $K = 3$ or $K = 4$ for the second-order term in symmetric and asymmetric TGV, respectively).

The choice of the Euclidean norm is arbitrary and other norms are possible. Miyata [105] argues that due to the averaging in the Euclidean norm over the channels, directional information is lost. This information is valuable in color image processing, where true jumps in the image are mostly present in all color channels, whereas jumps in a single channel can be mostly attributed to noise. To account for this he proposes to use the ∞-norm instead of the Euclidean norm:

$$||v||_{o,1} = \sum_{i,j} \sum_{k=1}^{K} \left| \max_{c} |v_c^k| \right|. \tag{3.57}$$

The numerical experiments [105] on color image denoising indeed show that this variant shows less artifacts around color edges. This definition is less useful for optical flow, since the strong assumption on the correlation of the channels (i.e. the x- and y-directions of the flow) does not hold for this type of data.

3.2 Other Higher-Order Priors

TGV is a versatile convex higher-order prior, but of course not the only one that was proposed in the literature. One way to define a prior which accounts for higher-order derivatives is to directly penalize the second-derivatives of a sufficiently smooth function along with the first derivatives. This gives rise to the TV-TV^2 regularizer [115], which can be defined as

$$\Phi(u) = \alpha TV(u) + \beta TV^2(u), \tag{3.58}$$

with

$$TV^2(u) = \int_\Omega |D^2 u|, \tag{3.59}$$

where D^2u the distributional Hessian. The scalars α and β allow to trade-off the influence of both terms. Related higher-order priors where only the distributional Hessian is considered were also proposed earlier [101]. Another variant that considers the Hessian was proposed by Danielsson et al. [43] and later also used for optical flow estimation [152]. Recently, Lellmann et al. [95] introduced and analyzed a regularizer based on a non-local Hessian and gradient, which shares some conceptual similarities with Non-Local TGV:

$$(G_u^*(\mathbf{x}), H_u^*(\mathbf{x})) = \arg \min_{G_u, H_u} \frac{1}{2} \int_{\Omega - \{\mathbf{x}\}} \left(u(\mathbf{x} - \mathbf{z}) - u(\mathbf{x}) - G_u^T \mathbf{z} - \frac{1}{2} \mathbf{z}^T H_u \mathbf{z} \right)^2 \alpha(\mathbf{z}) \, \mathrm{d}\mathbf{z},$$

where the integration domain is given by $\Omega - \{\mathbf{x}\} = \{\mathbf{y} - \mathbf{x} : \mathbf{y} \in \Omega\}$.

The goal of all of these regularizers is similar to TGV, *i.e.* their main aim is to reduce staircasing and allow for piecewise affine solutions. In contrast to TGV they are strictly focused on the second-order case and can not be generalized to arbitrary order k. Moreover they exhibit a few undesirable properties and visual artifact, which can be seen in the comparison in Figure 3.11.

Chambolle and Lions [35] considered the inf-convolution of two convex functionals

$$\mathrm{INF}(u) = \min_{u_1+u_2=u} \int_\Omega |Du_1| + \alpha \int_\Omega |D^2 u_2|, \tag{3.60}$$

which is strongly related to TGV_α^2. To see this connection simply set $u_1 = u - u_2$ and substitute to get

$$\mathrm{INF}(u) = \min_{u_2} \int_\Omega |D(u-u_2)| + \alpha \left|D^2 u_2\right|. \tag{3.61}$$

The difference to TGV_α^2 is now easy to see: The distributional gradient in the first-order term is applied to $u - u_2$ not only to u. Moreover, the second term measures the second-variation of u_2. Not surprisingly, the inf-convolution regularizer behaves qualitatively similar to TGV_α^2, but may exhibit some undesirable artifacts such as residual staircasing (Figure 3.11), which are not present in TGV.

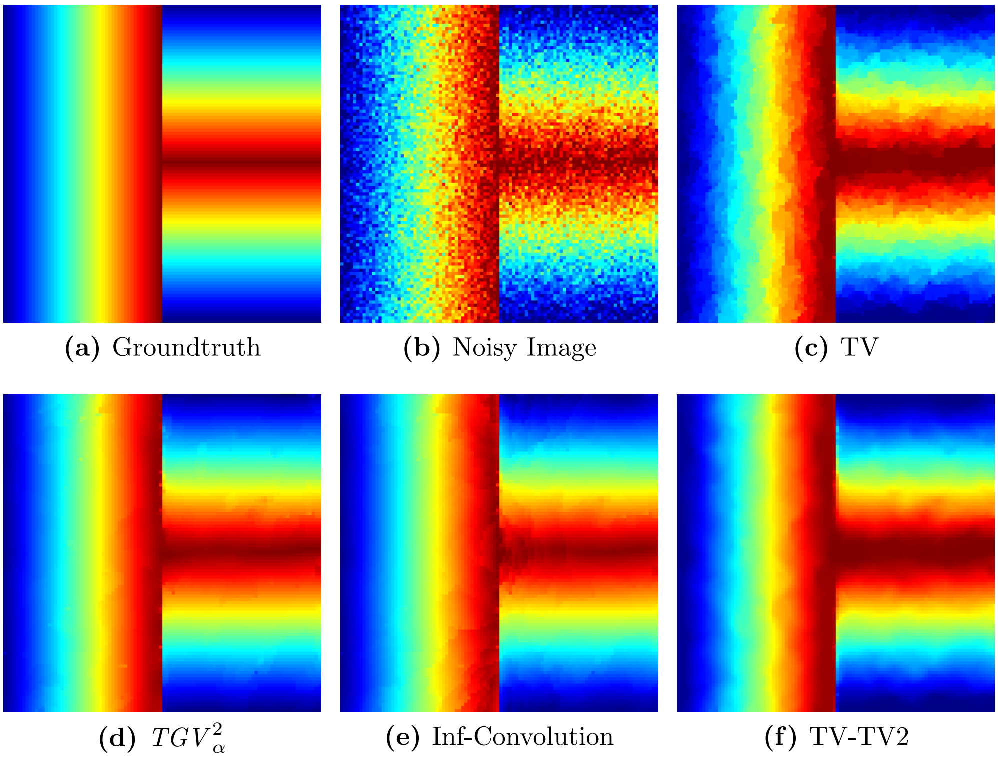
Ishikawa [84] proposed to use the Absolute Gaussian Curvature as a prior for stereo estimation. Gestalt psychological considerations and preliminary experiments indeed show that this may be an excellent prior for stereo data. Since the Gaussian Curvature is non-convex, the work resorted to a discretization of the disparity space and Simulated Annealing for the optimization of a stereo model involving this prior, which is known to be inaccurate and slow.

3.3 Discussion

TGV offers a flexible way to model higher-order variations in variational models. In this section we introduced the basic TGV prior along with several variants and extensions. We have shown how the continuous prior can be discretized and optimized using a theoretically optimal convex minimization algorithm. Our preliminary experiments on piecewise affine functions have clearly shown that TGV is indeed able to model such functions. Moreover, it was shown that the introduction of additional information in the form of a guidance image can significantly improve the prior. What remains open is the question if stereo or optical flow data is indeed well modeled by the proposed higher-order priors.

Stereo: TGV (of order two) in a variational stereo algorithm softly enforces the assumption of a piecewise planar world. Most scenes, especially in man-made environments, can be very well approximated using this assumption.

One important point for TGV is that the planarity assumption is not a hard constraint. Planar areas emerge automatically, wherever this is supported by the data. This, in principle, allows to preserve details in the estimates, and does not require a preliminary segmentation of the scene into a set of planar regions. This feature may also be detrimental if the data is contradictory over large areas, however.



(a) Groundtruth **(b)** Noisy Image **(c)** TV

(d) TGV_α^2 **(e)** Inf-Convolution **(f)** TV-TV2

Figure 3.11: Denoising a piecewise affine image using different higher-order priors. We use a colormap to emphasize the differences between the methods.

Optical Flow: TGV is able to enforce piecewise affine flow fields, which model the flow of bodies undergoing rigid motions well. The assumption encoded by TGV is thus again well-justified. Non-rigid motions can be approximated using a set of smaller rigid motions, which again is supported by the TGV prior. We will later show in the experimental section that TGV as an optical flow prior is especially strong in the presence of motions, which significantly differ from the classical assumption of piecewise constant motion.

4 Optical Flow

Contents

Optical flow is the apparent motion between images acquired at different times. In general, it is a matching problem and closely related to the stereo matching problem. Optical flow may be induced by moving objects, by a moving camera or both. In contrast to stereo estimation, in optical flow estimation, pixels may move freely along both image dimensions, not only along a epipolar line. As such optical flow is generally regarded as a harder problem than the stereo matching problem. Also, except for a few special cases, modeling optical flow estimation as a discrete labeling problem is not feasible since the computational complexity is very high due to a very large label space. Consequently, most optical flow models are continuous in nature and many can be understood as variants of the seminal Horn-Schunk model [80].

In this section we introduce the general optical flow problem. We discuss adaptions of data terms to the special circumstances of this problem. Different variants of the Total Generalized Variation (TGV) regularizers will be used and evaluated for this task and we will show how such models can be optimized in practice.

4.1 Related Work

Optical Flow and motion estimation is a long standing research topic in Computer Vision. Arguably the two most influential early works on optical flow estimation are the work by

Lucas and Kanade [100] on feature tracking and the work on dense optical flow computation by Horn and Schunk [80]. The following section includes a historic overview of optical flow methods, which can be traced back to these works. Moreover, we try to broadly categorize the approaches into four categories, based on their fundamental philosophy:

1. Local Methods: Pixels only locally contribute to the flow estimate. Many models in this category are variants of [100]. They are versatile and can be adapted to a variety of different tasks apart from optical flow estimation.

2. Continuous Global Methods: The flow estimate at each location depends on the estimate of all other locations. Models in this category are typically variational models or directly based on PDEs and are embedded into a coarse-to-fine framework. They are heavily inspired by the Horn-Schunk model and currently constitute the majority of models.

3. Discrete Global Methods: Optical flow estimation is modeled as a discrete labeling problem. This approach often provides higher accuracy but results in very high computational complexity.

4. Interpolation-based Methods: A recent development, which tries to eliminate the coarse-to-fine approach by interpolating from high-quality sparse initial matches. These approaches sometimes include continuous or discrete models as additional refinement steps.

For many models no hard categorization is possible, since ideas from any category are often applicable to models in other categories.

Local Methods In contrast to a pixel-based optical flow approach, the aim of the Lucas-Kanade algorithm is to find corresponding patches between two images. The incorporation of patches instead of single pixels into the optical flow estimate allows to resolve ambiguities which are inherent to the optical flow problem and also provides robustness to some degree of noise. The method is local in the sense that each patch is considered for itself, without regard to the other patches in the image. Due to the locality only patches which exhibit some degree of structure can be correlated, thus the Lucas-Kanade algorithm is only able to provide optical flow estimates around image edges. Completely homogeneous regions can not be resolved, since there is no mechanism to propagate information from the edges into these regions. The original formulation, which is directly based on the brightness of the input images, and correlates patches under a least-squares criterion, has been modified and extended in several ways over the years. These modification range from changing the motion model from purely translational motion to affine motion [137], selection of image features which are easy to track using a local method [137], modification of the numerical algorithm [6, 137], changing the composition of the warping process [6, 68, 139] and modification of the least-squares matching criterion to a more robust matching criterion [48]. A comprehensive overview of a modern implementation of the Lucas-Kanade method can be found in a series of papers by Baker et al. [4–8].

Local methods have the inherent advantage that they are highly data-parallel and can be extremely efficient [119]. Tao et al. [151] introduced a (practically) sub-linear time algorithm, which computes dense optical flow estimates using a local approach. In contrast to most other algorithms, their approach is not based on an iterative energy-minimization scheme, but purely on local probabilistic reasoning.

Continuous Global Methods The Horn-Schunk model [80] relies on the pixel-based brightness constancy assumption for optical flow estimation and augments this simple criterion with a smoothness assumption on the flow field. The smoothness assumption again resolves ambiguities and provides robustness to noise. In contrast to the Lucas-Kanade algorithm, the Horn-Schunk method is able to provide optical flow estimates also in homogeneous regions, since the smoothness assumption can propagate information to these regions. As a result, the optical flow estimates resulting from this model are dense, every pixel has a corresponding estimate of its apparent motion. This is often a highly desirable property, since it gives more flexibility to any algorithm that is based on the optical flow estimates. The model is a global variational model, *i.e.* every pixel of the flow estimate depends implicitly on all other pixels.

The Horn-Schunk model forms the basis of many modern optical flow algorithms and is also the basis of the algorithms presented in the following sections. The three main ingredients of the model are: (1) a dataterm for correlating the input images. (2) a smoothness assumption, which resolves ambiguities and propagates information to homogeneous regions. (3) the optimization procedure used to solve the model. It is important to note that the choice of the dataterm and the smoothness assumption is tightly interwoven with the choice of optimization procedure. All of the individual components have a large impact on the final performance. Again, countless enhancements have been proposed to these basic components, which try to tackle different shortcomings of the original model. Black and Anandan proposed to use robust penalty functions in the data term to gain robustness to outliers [15, 16]. Brox et al. [28] used a linear combination of brightness and gradient constancy to increase robustness with respect to illumination changes. Later, Xu et al. [169] introduced a method that locally selects between gradient and brightness constancy and show that this approach can lead to improved accuracy. Similarly, Bruhn and Weickert proposed to separately penalize brightness and gradient constancy using a robust penalty function [30], which favors solutions where at least one of the two assumptions hold. Higher-order constancy assumptions based on the Hessian and Laplacian have been incorporated by Papenberg et al. [116]. Bruhn et al. [31] combine a global model with the Lucas-Kanade algorithm, which results in a modified patch-based data term.

Another source of error in the classical Horn-Schunk model is the smoothness term, which tends to oversmooth motion boundaries. A typical modification to tackle this problem is the use of robust penalizers for the smoothness assumption. One of the first approaches was to use the Total Variation (TV) norm as an edge-preserving regularizer [39, 92]. To this day TV and smooth approximations of TV are among the most popular regularizers for optical flow estimation [28, 30, 162, 164, 176]. TV has the advantage of being convex, which simplifies optimization of the model, provided that the dataterm is also convex. Nonetheless non-convex penalizers have been successfully incorporated into optical flow models [15, 73, 138]. Further extensions consist of higher-order priors in order to better model the flow fields. Trobin et al. [152] introduce a convex second-order prior which penalizes deviations from affine functions. Demetz et al. [45] use a regularizer that is directly based on the Hessian of the flow components. The potential of conditioning the regularizer on the image content has been explored early on. Nagel and Enkelmann [107] proposed an anisotropic regularization term, which can be conditioned on a guidance image. A scalar spatially varying weighting of the regularization was proposed in [1]. Other approaches try to adapt the regularization directly to the optical flow estimate [135, 138].

Continuous global models are typically embedded into a coarse-to-fine warping framework in order to support the recovery of large motions. The drawback of this strategy is that motion of small structures can not be recovered reliably. There have been some efforts to circumvent this problem. Brox and Malik introduced descriptor matching into a global model [29] to recover large motions of small structures. Braux et al. [24] generalized this approach and also incorporated line-based features into a TGV-based global model. Xu et al. [169] refine the estimates in the coarse-to-fine pyramid using a discrete global model, which is solved using fusion moves.

Discrete Global Methods Discrete global methods model optical flow estimation as a discrete labeling problem. The task is to assign to each pixel a label which is chosen from a discrete set of possible labels. Each label corresponds to a specific flow vector. Since the set of possible flow vectors is typically very large, these algorithms are computationally very demanding. On the positive site, they allow for very flexible data terms and completely circumvent the problems introduced by a coarse-to-fine warping strategy.

The main challenge in these approaches is to handle the potentially huge label space efficiently. One approach is to exploit the specific structure of the optical flow label space, which can be modeled as a product label space. The resulting problem is a combinatorial optimization problem, which can be solved or approximated using different approaches, like convex relaxations [62, 143] or mean-field inference [156]. Goldstein et al. [64] present a functional lifting approach for the optical flow problem, which allows to compute globally optimal solutions. All of these approaches are interesting from a mathematical point of view, but are currently computationally too expensive for practical applications.

Move-making algorithms try to find a middle-ground between the accuracy of discrete models and the speed of continuous models. The basic idea of these approaches is to iteratively fuse proposals (which are provided by an external process) under a global model using a series of discrete binary problems. Trobin et al. [153] used this idea together with a convex relaxation of the binary subproblems. Lempitsky et al. [96] solved the binary problems using Quadratic Pseudo-Boolean Optimization (QPBO) [20].

The last large class of models in this category is given by layer-based models. These approaches are inherently related to motion segmentation, *i.e.* they try to decompose the motion field into a set of regions of similar motions. Such models support strong reasoning about geometric relationships and occlusions and typically result in discrete-continuous energies. Sun et al. [146, 147] decompose the scene into a small set of layers and jointly reason about their motion and geometric relationship. Unger et al. [154] jointly model motion segmentation and optical flow estimation. They use fine-grained segmentation and show how occlusion estimation procedures can be incorporated directly into the model.

Interpolation-based Methods Interpolation-based methods interpolate dense optical flow from a sparse set of initial matches. Ren et al. [127] use an approach similar to the Lucas-Kanade method, which additionally groups flow vectors based on the content of the reference image. They propose to interpolate dense optical flow from the irregularly sampled feature locations using a Delaunay triangulation and use linear interpolation in the interior of the resulting triangles. Leordeanu et al. [97] use a custom sparse matching approach on large patches, which are subsequently interpolated to a complete dense optical flow and refined using a classical continuous model. Revaud et al. [128] use sophisticated sparse matches [164] and propose to interpolate a dense flow field in an edge-preserving way. They again use a continuous global model to refine the results in a post-processing step.

4.2 A Basic Optical Flow Model

In this section we describe a prototypical optical flow model, which will form the basis for various enhancements in the subsequent sections of this chapter. We introduce fundamental components of variational optical flow models, such as the optical flow constraints and the coarse-to-fine warping strategy, and discuss their properties.

The fundamental assumption in optical flow models is the so-called brightness constancy assumption. It models that fact that under ideal conditions, pixels do not change their brightness under motion. Given two input images $I_1 : \Omega \to \Re$ and $I_2 \in \Omega \to \Re$ and the motion $\mathbf{u} : \Omega \to \Re^2$, this assumption can be formally stated as

$$I_1(\mathbf{x} + \mathbf{u}) = I_2(\mathbf{x}). \tag{4.1}$$

Deviations from this assumptions are called the warping error. It is natural to directly use this relationship to model optical flow estimation, by measuring how much this relation is violated by a given flow field:

$$D(I_1, I_2, \mathbf{u}) = \int_\Omega \Phi(I_1(\mathbf{x} + \mathbf{u}) - I_2(\mathbf{x}))\, \mathrm{d}x. \tag{4.2}$$

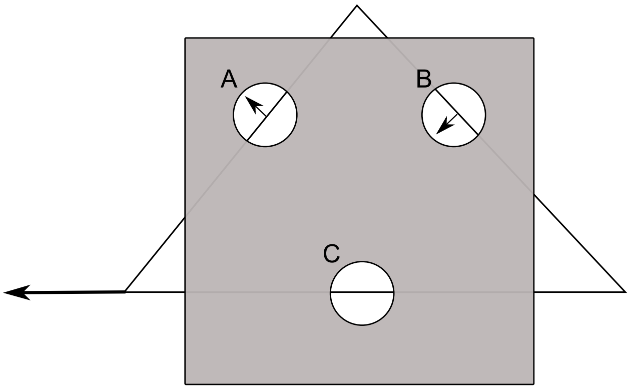


Figure 4.1: An illustration of the aperture problem. The triangle is moving to the left. Viewed through apertures A and B, the apparent motion is contradictory and does not correspond to the true motion. No movement can be observed through aperture C. In general only movement perpendicular to an edge can be perceived trough a small aperture.

The function Φ is typically a ℓ_2- or ℓ_1-norm, but also more robust non-convex functions can be used. (4.2) shows a complex relation between the (unknown) flow field $\mathbf{u}$ and the overall matching score $D(I_1, I_2, \mathbf{u})$: $\mathbf{u}$ arises in the argument of I_1. This warping operation will be in general non-convex. This form also prevents direct analytical treatment of the data term in a continuous model. The standard approach to remedy these problems is to perform a first-order Taylor expansion of the image $I_1(\mathbf{x}+\mathbf{u})$ around some displacement $\mathbf{u}_0$:

$$I_1(\mathbf{x} + \mathbf{u}) \approx I_1(\mathbf{x} + \mathbf{u}_0) + (\mathbf{u} - \mathbf{u}_0)^T \nabla I_1|_{\mathbf{x}+\mathbf{u}_0} \tag{4.3}$$

By setting $I_t(\mathbf{x}) = I_1(\mathbf{x} + \mathbf{u}_0) - I_2(\mathbf{x})$, this leads to the Optical Flow Constraint (OFC)

$$I_t + (\mathbf{u} - \mathbf{u}_0)^T \nabla I_1 = 0, \tag{4.4}$$

which forms the basis of many optical flow models. The *OFC* is an equation in two unknowns and thus underdetermined. It is thus not possible to unambiguously find a flow field from this equation alone. The fact that many different flows can fulfill the *OFC* is known as the aperture problem. This is a well-known physical phenomenon and is illustrated in Figure 4.1. The aperture problem shows that the problem of optical flow estimation is ill-posed. Clearly additional information is necessary to find reasonable flow fields. In a variational model this additional information is explicitly modeled using regularization terms, which typically model a prior assumption on the form of a flow field. In a variational model the *OFC* is integrated by penalizing deviations from the constraint:

$$D(I_1, I_2, \mathbf{u}, \mathbf{u}_0) = \int_\Omega \Phi(I_t + (\mathbf{u} - \mathbf{u}_0)^T \nabla I_1) \, \mathrm{d}\mathbf{x} \tag{4.5}$$

Together with a regularization term $R(\mathbf{u})$, the general variational optical flow model, which

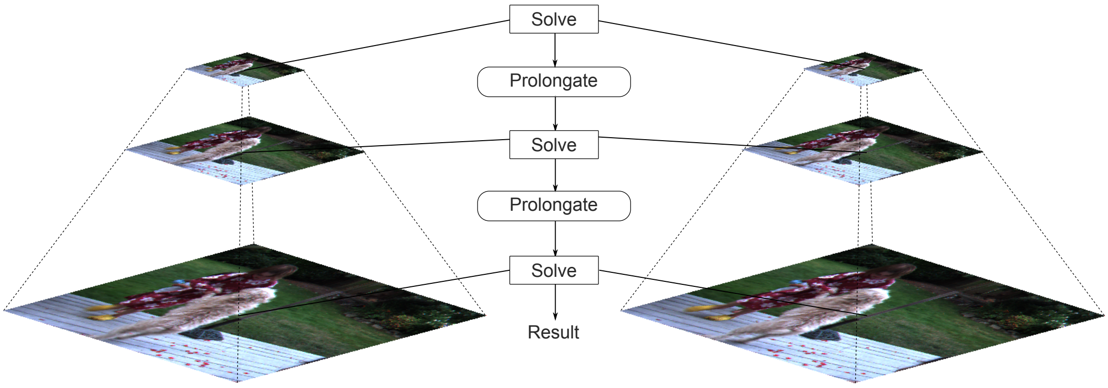


Figure 4.2: An illustration of the coarse-to-fine approach.

was already introduced in the introductory chapter, is given by

$$\min_{\mathbf{u}} R(\mathbf{u}) + D(I_1, I_2, \mathbf{u}, \mathbf{u}_0) \tag{4.6}$$

This formulation has clear practical merits over (4.2): The *OFC* is linear and thus convex. Moreover (4.5) is differentiable, if Φ is differentiable and it is convex if Φ is convex. A well-designed regularization term $R(\mathbf{u})$ gives additional information, which allows to resolve the ambiguities introduced by the aperture problem. As a consequence the model can be efficiently treated in modern optimization frameworks, for appropriate choices of the penalizer Φ and regularization term R. Of course there is a price to pay for this tractability: The Taylor expansion is an approximation to the originally warped image $I_1(\mathbf{x}+\mathbf{u})$, which is only valid in the vicinity of $\mathbf{u}_0$. If the flow $\mathbf{u}$ is far away from $\mathbf{u}_0$, a data term based on the *OFC* does not reliably measure the warping error. This problem can be solved by an iterative re-linearization: The optical flow model (4.6) is solved for an initial flow field $\mathbf{u}_0$ (which is typically equal to zero). The image is again linearized around the solution and the model is again solved. This process repeats iteratively until some stopping criterion is met.

This process allows for larger optical flows to be found but easily gets stuck in a non-optimal solution. As an additional remedy the whole process is thus integrated into a coarse-to-fine framework. The input images are subsampled to form an image pyramid (typically in a Gaussian scale space). The optical flow is computed on the subsampled input images, where the computation is started on the coarsest level. The solution of the coarse level is propagated to the next finer level by upsampling the optical flow and starting linearization at this initial solution. This process is repeated until the finest level (the original image size) is reached. An overview of this approach is shown in Figure 4.2. The coarse-to-fine approach results in fast convergence, since on the higher, computational demanding levels a good initial solution is already available. Larger motion can be found, provided that the image pyramid is computed up to a sufficiently coarse level and that the scale space is sufficiently fine sampled. A major down-side is that due to the subsampling, fine motion details, like the motion of thin structure or small objects, cannot be captured.

4.2.1 Classical Instances of the General Model

We now briefly examine three different classical optical flow models, which can be understood as instances of the general variational model 4.6. We start from the very early Horn-Schunk model, and show two different more advanced models, which can be viewed as robust variants of this model and are very similar in spirit, but are nonetheless solved using fundamentally different strategies, namely the Euler-Lagrange framework and convex optimization.

4.2.1.1 Horn-Schunk

The oldest instance of our prototypical variational optical flow model is given by the seminal Horn-Schunk model. This model already included all the main building blocks that were described so far. The Horn-Schunk model is given by a quadratic penalization of the optical flow constraint and a quadratic penalization of the gradients of the flow field. Its specific form allows for a simple mathematical treatment in terms of optimization: Since both terms in the model are quadratic and convex, the resulting optimization problem is an instance of Thikhonov regularization, which allows for a closed-form solution, by solving a linear system of equations. The model is given by

$$\min_{(u,v)} \int_\Omega (|\nabla u|^2 + |\nabla v|^2)\,\mathrm{d}\mathbf{x} + \lambda \int_\Omega |I_t + (u - u_0)\nabla_x I_1 + (v - v_0)\nabla_y I_1|^2\,\mathrm{d}\mathbf{x}, \tag{4.7}$$

where we explicitly split the flow $\mathbf{u} = (u, v)^T$ into its horizontal and vertical component. This model can be discretized analogous to the ROF model shown in Section 2.3.2. After discretization we have

$$\min_{(u,v)} ||\nabla u||^2 + ||\nabla v||^2 + \lambda\, ||I_t + \mathrm{diag}(\nabla_x I_1)(u - u_0) + \mathrm{diag}(\nabla_y I_1)(v - v_0)||^2\,, \tag{4.8}$$

where we used the operator diag(x) for notational convenience, which generates a $N_x N_y \times N_y N_y$ diagonal matrix with its argument $x \in \Re^{N_x N_y}$ on the main diagonal. All other quantities are given by column vectors of dimension $N_x N_y$ which result from stacking the images.

Using the first-order optimality conditions the unique optimal solution necessarily fulfills the linear relation

$$\begin{aligned} \nabla^T\nabla u + \lambda \hat{I}_x(I_t + \hat{I}_x(u - u_0) + \hat{I}_y(v - v_0)) = 0 \\ \nabla^T\nabla v + \lambda \hat{I}_y(I_t + \hat{I}_x(u - u_0) + \hat{I}_y(v - v_0)) = 0, \end{aligned} \tag{4.9}$$

where we used the short-hand notation $\hat{I}_x = \mathrm{diag}(\nabla_x I_1)$ and $\hat{I}_y = \mathrm{diag}(\nabla_y I_1)$. This can be written more compactly as

$$\underbrace{\begin{pmatrix} \nabla^T\nabla + \lambda\hat{I}_x^2 & \lambda\hat{I}_x\hat{I}_y \\ \lambda\hat{I}_x\hat{I}_y & \nabla^T\nabla + \lambda\hat{I}_y^2 \end{pmatrix}}_{A} \begin{pmatrix} u \\ v \end{pmatrix} = \lambda \underbrace{\begin{pmatrix} \hat{I}_x(\hat{I}_x u_0 + \hat{I}_y v_0 - I_t) \\ \hat{I}_y(\hat{I}_x u_0 + \hat{I}_y v_0 - I_t) \end{pmatrix}}_{b} . \tag{4.10}$$

Excluding a pathological case, matrix A is positive-definite [104], thus the system has a unique solution and the matrix is invertible. In practice the system of equations will be large but sparse, in which case iterative methods for the solution of linear systems can be used [76].

Figure 4.3 shows an exemplary result obtained from the Horn-Schunk model, which makes its main properties immediately obvious: The model gives dense and smooth flow fields, but motion boundaries are not well preserved. This behavior stems from the quadratic penalization of the gradients in the penalizer. Large jumps at motion boundaries are disproportional penalized, which results in a strong smoothing effect. The second drawback is that deviations from the *OFC* constraint can not be robustly handled, again due to the quadratic penalization. In any realistic scenario there will be strong deviations of the *OFC* for occluded or disoccluded pixels, since in these regions there are pixels in one frame, which have no corresponding pixel in the second frame.

4.2.1.2 The Euler-Lagrange framework

It was already noted that the quadratic penalization is the main source of error in the Horn-Schunk model. It is thus natural to replace the quadratic penalization with a more robust function. This results in a model that is able to recover sharper motion boundaries and also is more robust at pixels where the assumptions of the data term are heavily violated. Note, however, that this comes for a price: Deviating from the quadratic penalization renders optimization of the models much harder, since the optimality conditions will in general be non-linear. There are two prevalent approaches to handle optimization of the resulting functionals: Solutions based on the Euler-Lagrange framework, where the necessary conditions (the Euler-Lagrange equations) for optimality of the continuous model are discretized and then typically solved using iterative linearizations. The second approach is to discretize the functional itself and apply (convex) optimization algorithms to find a solution. The Euler-Lagrange framework offers the advantage that arbitrary smooth penalizers can be used [180]. Convexity is typically not a driving issue in this framework. On the other hand the classical Euler-Lagrange framework requires smoothness of the functional, which can be detrimental to the sharpness of the motion boundaries. This is not the case for the convex optimization framework, which allows for non-smooth penalizations, but requires convexity of the penalizers. Moreover, convex optimization approaches can be parallelized trivially, which allows for easy implementation on graphics processing units. The Euler-Lagrange approach can also be parallelized but requires more sophisticated implementations [66].

To exemplify the Euler-Lagrange framework, we briefly describe a simplified variant of the approach by Brox et al. [28]. They propose to replace the quadratic penalizers with robust functions:

$$\min_{(u,v)} \int_\Omega \Phi(|\nabla u|^2 + |\nabla v|^2)\,\mathrm{d}\mathbf{x} + \lambda \int_\Omega \Phi(|I_1(\mathbf{x}+\mathbf{u}) - I_2(\mathbf{x})|^2)\,\mathrm{d}\mathbf{x}. \tag{4.11}$$

The model uses the original non-linear brightness constancy as data term. Brox et al. [28] propose to use the Charbonnier penalty $\Phi(x^2) = \sqrt{s^2+\varepsilon^2}$ as subquadratic penalizer, which results in an overall convex model. The smoothness term can be interpreted as a smooth approximation to TV, whereas the data term can be interpreted as a smooth approximation to the ℓ_1-norm. The necessary conditions for a optimum of this continuous model are given by the Euler-Lagrange equations:

$$\begin{aligned}\Phi'(I_t^2)\cdot I_x I_t - \lambda \operatorname{div}(\Phi'(|\nabla u|^2 + |\nabla v^2|)\nabla u) = 0\\ \Phi'(I_t^2)\cdot I_y I_t - \lambda \operatorname{div}(\Phi'(|\nabla u|^2 + |\nabla v^2|)\nabla v) = 0,\end{aligned} \tag{4.12}$$

where I_x and I_y denote partial derivatives and I_t is the difference between the second image and the first image after warping:

$$I_x = \partial_x I_1(\mathbf{x}+\mathbf{u}), \qquad I_y = \partial_y I_1(\mathbf{x}+\mathbf{u}), \qquad I_t = I_1(\mathbf{x}+\mathbf{u}) - I_2(\mathbf{x}). \tag{4.13}$$

Equation (4.12) constitutes a non-linear Partial Differential Equation (PDE), which will be subsequently discretized, and iteratively solved based on linearizations of the non-linear *PDE*. There are two-sources of non-linearity: First, the warping operation I_t is non-linear, but can linearized using a first-order Taylor expansion. Second, the functions Φ' are also non-linear. In order to handle both non-linearities Brox et al. use nested fixed-point iterations.

$$\begin{aligned}(\Phi')_D^{k,l}\cdot(I_x^k(I_t^k + I_x^k\,\mathrm{d}u^{k,l+1} + I_y^k\,\mathrm{d}v^{k,l+1})) - \lambda \operatorname{div}((\Phi')_S^{k,l}\nabla(u^k + \mathrm{d}u^{k,l+1})) = 0\\ (\Phi')_D^{k,l}\cdot(I_y^k(I_t^k + I_x^k\,\mathrm{d}u^{k,l+1} + I_y^k\,\mathrm{d}v^{k,l+1})) - \lambda \operatorname{div}((\Phi')_S^{k,l}\nabla(v^k + \mathrm{d}v^{k,l+1})) = 0,\end{aligned} \tag{4.14}$$

where k and l are iteration counters for the outer and inner fixed point iterations, respectively. The update is written in terms of increments

$$u^k = u^k + \mathrm{d}u^{k,l+1}, \qquad v^k = v^k + \mathrm{d}v^{k,l+1} \tag{4.15}$$

and the term I_t was linearized using a first-order Taylor expansion:

$$I_t^{k+1} \approx I_t^k + I_x^k\,\mathrm{d}u^{k,l+1} + I_y^k\,\mathrm{d}u^{k,l+1}. \tag{4.16}$$

The terms $(\Phi')_D^{k,l}$ and $(\Phi')_S^{k,l}$ are short-hand for:

$$\begin{aligned}(\Phi')_D^{k,l} &= \Phi'((I_t^k + I_x \, \mathrm{d}u^{k,l} + I_y^k \, \mathrm{d}v^{k,l})^2)\\(\Phi')_S^{k,l} &= \Phi'(|\nabla(u^k + \mathrm{d}u^{k,l})|^2 + |\nabla(v^k + \mathrm{d}v^{k,l})|^2).\end{aligned} \tag{4.17}$$

The system (4.14) is linear in the unknown increments $\mathrm{d}u^{k,l+1}$ and $\mathrm{d}u^{k,l+1}$ and thus can be solved using any solver for linear systems. For the spatial discretization of the gradient and divergence operators, the standard finite differences scheme from Section 2.3.2 can be used. Example results of this model can be seen in 4.3. Obviously, motion boundaries are well preserved and outliers in the data term can be handled, which overall leads to significantly better results.

4.2.1.3 TV-L1

The last classical model is given by the TV-L1 model [176]. As the name already indicates, this model uses the non-smooth Total Variation as a prior and penalizes deviations from the optical flow constraint using the ℓ_1-norm:

$$\min_{(u,v)} \int_\Omega \sqrt{|Du|^2 + |Dv|^2} + \lambda \int_\Omega |I_t + (u - u_0)\nabla_x I_1 + (v - v_0)\nabla_y I_1| \, \mathrm{d}\mathbf{x}. \tag{4.18}$$

This model is very similar to the previous model, which was solved using the Euler-Lagrange framework. The main difference is the use of the non-smooth L1 norm for both the smoothness and the data term (*i.e.* this model can be recovered from (4.11) by setting $\Phi(s^2) = \sqrt{s^2}$ and direct use of the *OFC* in the data term). The model is convex, which enables the usage of convex optimization algorithms to solve this model.

The first step for the solution if this model is to discretize the functional, which yields

$$\min_{\mathbf{u}=(u,v)} \sum_{i,j} |(\nabla \mathbf{u})_{i,j}|_2 + \lambda \left|(I_t)_{i,j} + (\mathbf{u}_{i,j} - (\mathbf{u}_0)_{i,j})^T (\nabla I_1)_{i,j}\right|, \tag{4.19}$$

where we set $(\nabla \mathbf{u})_{i,j} = ((\nabla u)_{i,j}, (\nabla v)_{i,j})^T$. The original work [176] used a quadratic splitting approach for the solution of this model, which allowed for the iterative splitting into a ROF model and a pixel-wise optimization problem, which can be solved in closed form. Since then, advancements in convex optimization made it possible to directly solve the non-smooth model. We will adopt the later approach, since it comes with convergence guaranties and will also be the basis for solving models throughout the rest of this thesis. The basic idea is, like for the ROF model in Section 2.3.2, to rewrite (4.19) as an equivalent convex-concave saddle-point problem, which can then be solved using the primal-dual algorithm 2.6. To achieve this, we rewrite (4.19) using the dual definition of TV:

$$\min_{\mathbf{u}} \max_{p \in P} \sum_{i,j} (u_{i,j} \operatorname{div} p_{i,j}^u + v_{i,j} \operatorname{div} p_{i,j}^v) + \lambda \left|(I_t)_{i,j} + (\mathbf{u}_{i,j} - (\mathbf{u}_0)_{i,j})^T (\nabla I_1)_{i,j}\right|, \tag{4.20}$$

where the feasible set of the dual variables is given by

$$P = \{p \in \Re^{4N_xN_y},\, p_{i,j} = ((p^u)_{i,j}^T, (p^v)_{i,j}^T)^T : \sqrt{|(p^u)_{i,j}|_2^2 + |(p^v)_{i,j}|_2^2} \leq 1\}, \tag{4.21}$$

i.e. there are four dual variables per pixel, which are feasible if they lie in the four-dimensional Euclidean unit ball. Note that in this formulation the dual variables of the flow components are coupled via the constraint (4.21). This is because the model (4.18) was defined via an isotropic vectorial variant of TV [27]. An alternative definition could be to penalize the components of the flow independently, e.g. by defining the model as

$$\min_{(u,v)} \int_\Omega (|Du| + |Dv|) + \lambda \int_\Omega |I_t + (u - u_0)\nabla_x I_1 + (v - v_0)\nabla_y I_1|\, \mathrm{d}x. \tag{4.22}$$

This would yield the same objective function (4.20), but a different feasible set for the dual variables:

$$P = \{p \in \Re^{4N_xN_y},\, p_{i,j} = ((p^u)_{i,j}^T, (p^v)_{i,j}^T)^T : |(p^u)_{i,j}|_2 \leq 1,\ |(p^v)_{i,j}|_2 \leq 1\}, \tag{4.23}$$

It is now possible to again apply the primal-dual algorithm to solve the saddle-point problem:

$$\begin{cases} p_{k+1} = \mathbf{prox}_{p\in P}(p_k + \sigma\nabla\bar{\mathbf{u}}_k) \\ (u_{k+1}, v_{k+1}) = \mathbf{prox}_{\tau g}(u_k - \tau\nabla^* p^u_{k+1},\ v_k - \tau\nabla^* p^v_{k+1}) \\ \bar{\mathbf{u}}_{k+1} = 2\mathbf{u}_{k+1} - \mathbf{u}_k, \end{cases} \tag{4.24}$$

where we have $g(\mathbf{u}) = \lambda\sum_{i,j}\left|(I_t)_{i,j} + (\mathbf{u}_{i,j} - (\mathbf{u}_0)_{i,j})^T(\nabla I_1)_{i,j}\right|$. It remains to show how the proximal operators can be evaluated. For the dual variables it suffices to project the variables to the four-dimensional unit ball:

$$p = \mathbf{prox}_{p\in P}(\hat{p}) \Leftrightarrow p_{i,j} = \frac{\hat{p}_{i,j}}{\max\left(1, \sqrt{\left|\hat{p}^u_{i,j}\right|_2^2 + \left|\hat{p}^v_{i,j}\right|_2^2}\right)}. \tag{4.25}$$

The proximal operator with respect to the data term is a pointwise optimization problem in two variables:

$$\mathbf{u} = \mathbf{prox}_{\tau g}(\hat{\mathbf{u}}) \Leftrightarrow \mathbf{u}_{i,j} = \arg\min_{\mathbf{u}} \frac{1}{2\tau}\,||\mathbf{u} - \hat{\mathbf{u}}_{i,j}||^2 + \lambda|(I_t)_{i,j} + (\mathbf{u} - (\mathbf{u}_0)_{i,j})^T(\nabla I_1)_{i,j}|,$$

which can be reduced to an equivalent one-dimensional optimization problem. To see this let $\mathbf{n} = \frac{(\nabla I_1)_{i,j}}{|(\nabla I_1)_{i,j}|_2}$ be the direction of the image gradient and $\mathbf{n}^\perp$ be a vector perpendicular to $\mathbf{n}$. $\mathbf{u}$ can be expressed in terms of a update along the directions spanned by the vectors

$\mathbf{n}$ and $\mathbf{n}^\perp$:

$$\mathbf{u}_{i,j} = \hat{\mathbf{u}}_{i,j} + \delta\mathbf{n} + \delta^\perp\mathbf{n}^\perp, \tag{4.26}$$

Plugging into (4.25) it is easy to see that the magnitude of update along the direction perpendicular to the gradient $\delta^\perp$ necessarily is equal to zero (since $\delta^\perp\mathbf{n}^\perp \cdot \nabla I_1 = 0$ for all $\delta^\perp$ and the quadratic term forces the minimum to lie as near as possible to $\hat{\mathbf{u}}$). Thus, the proximal operator can be equivalently stated as a one dimensional optimiziation problem of the variable δ along the direction $\mathbf{n}$:

$$\min_{\delta} \frac{1}{2\tau}\delta^2 + \lambda|\rho(\hat{\mathbf{u}}_{i,j}) + \delta\,|(\nabla I_1)_{i,j}|_2\,|, \tag{4.27}$$

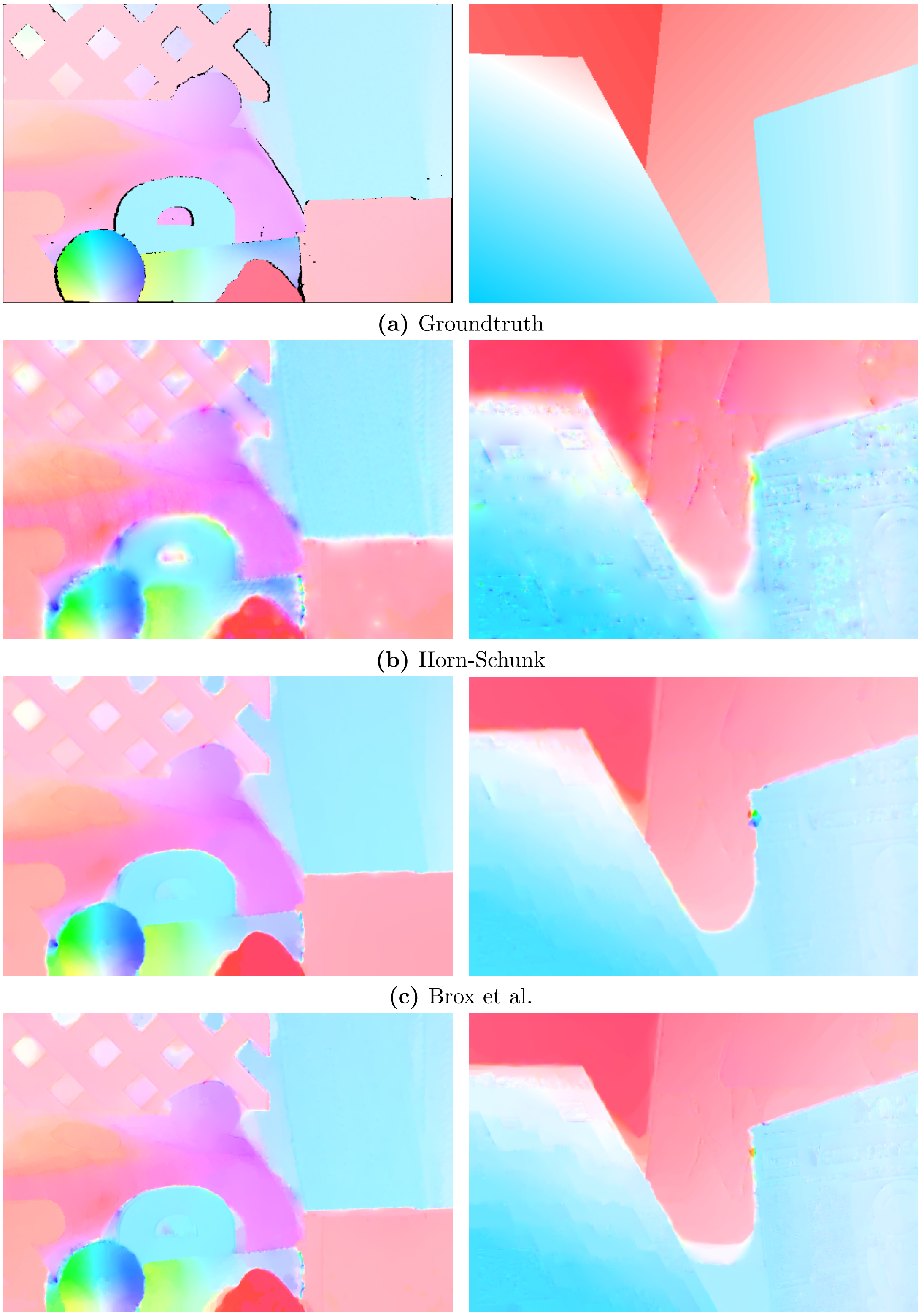
where $\rho(\mathbf{u}) = (I_t)_{i,j} + (\mathbf{u} - (\mathbf{u}_0)_{i,j})^T(\nabla I_1)_{i,j}$. This optimization problem can be solved using a soft-thresholding operation. After solving for δ and substituting the result into (4.26) the final update scheme is given by

$$\mathbf{u} = \mathbf{prox}_{\tau g}(\hat{\mathbf{u}}) \Leftrightarrow \mathbf{u}_{i,j} = \hat{\mathbf{u}}_{i,j} + \begin{cases} \tau\lambda(\nabla I_1)_{i,j} & \text{if} \quad \rho(\hat{\mathbf{u}}_{i,j}) < -\lambda\tau\,|(\nabla I_1)_{i,j}|_2^2 \\ -\tau\lambda(\nabla I_1)_{i,j} & \text{if} \quad \rho(\hat{\mathbf{u}}_{i,j}) > \lambda\tau\,|(\nabla I_1)_{i,j}|_2^2 \\ -\rho(\hat{\mathbf{u}}_{i,j})\frac{(\nabla I_1)_{i,j}}{|(\nabla I_1)_{i,j}|_2^2} & \text{if} \quad |\rho(\hat{\mathbf{u}}_{i,j})| \le \lambda\tau\,|(\nabla I_1)_{i,j}|_2^2 \end{cases} \tag{4.28}$$

One very practical property of the convex optimization approach is that it is trivially parallelizable, since all operations in each iterations of the primal-dual algorithm are purely local. This allows for simple and real-time capable implementations on graphics processing units [176].

4.2.1.4 Discussion

Figure 4.3 shows a comparison of the classical models we have described. Evidently the model based on the Euler-Lagrange framework as well as the TV-L1 model provide much cleaner and sharper results than the Horn-Schunk model. Both models qualitatively provide similar results, which is not surprising since they optimize very similar objective functions. Qualitatively TV-L1 provides slightly sharper motion boundaries. The Euler-Lagrange-based approach on the other hand benefits from the smooth penalizer, which results in a less strong assumption of piecewise constancy and consequently less staircasing in areas where this assumption is violated.



(a) Groundtruth

(b) Horn-Schunk

(c) Brox et al.

(d) TV-L1

Figure 4.3: Example results for classical optical flow models.

4.3 Enhancements

Multiple enhancements are possible to the variational optical flow model from the preceding section. In the light of this work, the most obvious enhancement is given by replacing the simple regularizer by one that better models a typical optical flow and also can be conditioned on additional input modalities. The robustness of the estimation can be significantly enhanced by using more robust, but also more complex, data terms. In this section we will describe both variants and the adaptions which are necessary to integrate them into the model.

4.3.1 Complex Data Terms

In practical situation the brightness constancy assumption is almost always violated for at least some parts of the image. Algorithms, which directly rely on this assumption will necessarily fail to provide robust results. Other sources of optical flow error include violation of the Lambertian assumption on specular surfaces, violation of the 1-1 mapping assumption due to occlusions and artifacts in the input images due to noise or physical distortions. There are different approaches to tackle violation of the basic assumptions, which can be broadly classified as: (1) Pre-processing of the input images, (2) explicit modeling of the error (e.g. the illumination change), (3) robustification of the data term.

Wedel et al. [162] proposed a pre-processing step, called structure-texture decomposition, where the input images are decomposed into a low-frequency structure part and a high-frequency texture part. Since differences in intensity can be assumed to be of low-frequency or even globally constant in some cases, most of these changes will be contained in the structural part. The intensity image thus is replaced by the texture image, or a blended version of the structure and texture images, in order to increase robustness to illumination changes. Another preprocessing approach is to derive photometric invariants from color input images [103, 155]. Preprocessing approaches are appealing, since they can be applied to existing models with minimum effort.

Chambolle and Pock [36] explicitly modeled illumination changes in a variational model using an additive compensation in the data term. Similar to [162] the fundamental assumption is that the illumination change is of low frequency, which is imposed using a Total Variation penalization of the compensation term. Modeling illumination change as an additional unknown results in a larger model and also is somewhat constraint to specific forms of illumination changes (in this case additive illumination changes are modeled). In a similar spirit, Demetz et al. [45] learn the basis of a Brighness Transfer Function (BTF) from training data and incorporate online estimation of the coefficients of the BTF into a variational optical flow model.

The last approach to handle challenging illumination conditions is to replace the brightness constancy assumption by a more robust correlation measure. Steinbrücker et al. [142] used the truncated ℓ_1-norm for measuring deviations from brightness constancy, which is more robust to outliers. Their optimization scheme is based on a successive quadratic relaxation approach [176], which allows for arbitrary matching terms. Werlberger et al. [166] proposed to use quadratic approximations of the truncated normalized cross correlation in a coarse-to-fine warping framework. The brightness constancy assumption can be replaced by, or augmented with, a gradient constancy assumption [28, 169] to gain robustness to illumination changes. Also various measures based on higher-order derivatives have been proposed for matching [116].

In practice the introduction of complex data terms, which are robust to typical problems encountered in optical flow estimation, has shown to be a promising approach [59]. The incorporation of complex data terms automatically raises the question on how to integrate them into the model, without sacrificing overall tractability. A flexible approach was introduced in [166], where the complete data term is approximated using a second-order Taylor expansion. The basis of continuous optical flow models is a convex approximation of the non-convex data term:

$$\int_\Omega \rho(\mathbf{x}, \mathbf{u}(\mathbf{x}), I_1, I_2)\,\mathrm{d}\mathbf{x} \approx \int_\Omega \hat{\rho}(\mathbf{x}, \mathbf{u}(\mathbf{x}), I_1, I_2)\,\mathrm{d}\mathbf{x}. \tag{4.29}$$

Similar to the linearization approach presented in the previous section, the pointwise matching cost ρ can be locally approximated using a second-order Taylor expansion:

$$\begin{aligned}\rho(\mathbf{x}, \mathbf{u}(\mathbf{x})) \approx& \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x})) + (\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x}))^T \nabla \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x})) \\ &+ \frac{1}{2}(\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x}))^T \nabla^2 \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x}))(\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x})) = \hat{\rho}(\mathbf{x}, \mathbf{u}(\mathbf{x})).\end{aligned} \tag{4.30}$$

The explicit dependence on the input images was dropped for notational simplicity. Approximating the data term directly allows for arbitrary data terms and makes this approach very flexible. Again, due to the inaccuracies introduced by the approximation, models which use this approach to handle the data term need to be embedded into a coarse-to-fine warping framework to be able to handle large displacements.

Since the Hessian $\nabla^2 \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x}))$ is not guaranteed to be positive semi-definite, which would lead to a non-convex data term, in practice a positive semi-definite approximation $H \approx \nabla^2 \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x}))$ is used to ensure tractability of the model. A simple approach is to use a diagonal matrix with non-negative approximations of the second-derivatives on the main diagonal:

$$H = \begin{pmatrix} \max(0, \rho_{xx}(\mathbf{x}, \mathbf{u}_0(\mathbf{x}))) & 0 \\ 0 & \max(0, \rho_{yy}(\mathbf{x}, \mathbf{u}_0(\mathbf{x}))) \end{pmatrix}. \tag{4.31}$$

This form guarantees that the matrix H is positive semi-definite and thus the approximation $\hat{\rho}(\mathbf{x}, \mathbf{u}(\mathbf{x}))$ is convex. Since the second-derivatives ρ_{xx}, ρ_{yy} and ρ_{xy} are typically not available for complex data terms analytically, they need to be approximated using finite differences. Some data terms may not be twice-differentiable everywhere. While in this case the derivatives are not well-defined, the proposed scheme serves as guideline and motivation to get a reasonable quadratic approximation based on finite differences.

Another, slightly more accurate variant to ensure convexity of the approximation is to first compute the full, possibly not positive-definite, Hessian

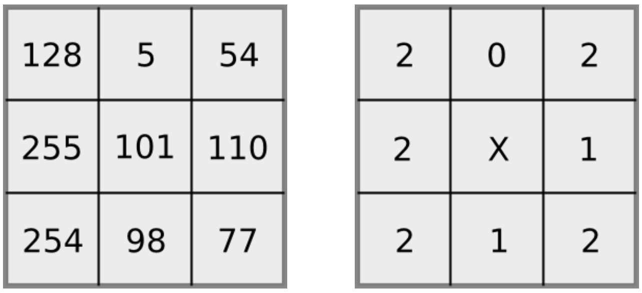
$$\nabla^2 \rho(\mathbf{x}, \mathbf{u}_0(\mathbf{x})) = \begin{pmatrix} \rho_{xx}(\mathbf{x}, \mathbf{u}_0(x)) & \rho_{xy}(\mathbf{x}, \mathbf{u}_0(x)) \\ \rho_{yx}(\mathbf{x}, \mathbf{u}_0(x)) & \rho_{yy}(\mathbf{x}, \mathbf{u}_0(x)) \end{pmatrix} \tag{4.32}$$

and subsequently project it to the space of positive semi-definite matrices. This can be achieved using an eigendecomposition and clamping the eigenvalues to ensure that they are non-negative [165]. While the resulting approximation to the Hessian will in general be better using this approach, it incurs higher computational cost than (4.31) and only in some cases results in significantly better optical flow results [165].

Now that we are able to use arbitrary data terms in an optical flow model, the main question is, which matching cost to use in practice. While there is a multitude of different terms (which will be discussed in more detail in the following chapter on stereo estimation), one of the most successful is given by the Census transform [175]. The Census transform is a popular approach to gain robustness against illumination changes in optical flow [106, 124, 157] on realistic data sets. Most of the top performing methods on the KITTI optical flow benchmark [59] incorporate this matching term. The principal idea is to generate a binary or ternary representation, called Census signature, of an image patch and measures patch similarity using the Hamming distance between signatures.

Let us define the per-pixel Census assignment function for an image $I : \Omega \rightarrow \Re$ as:

$$C_\varepsilon(I, \mathbf{x}, \mathbf{y}) = \operatorname{sgn}(I(\mathbf{x}) - I(\mathbf{y}))\mathbb{1}_{|I(\mathbf{x})-I(\mathbf{y})|>\varepsilon} + 1, \tag{4.33}$$



128	5	54
255	101	110
254	98	77

(a) Input patch

2	0	2
2	X	1
2	1	2

(b) Census transform

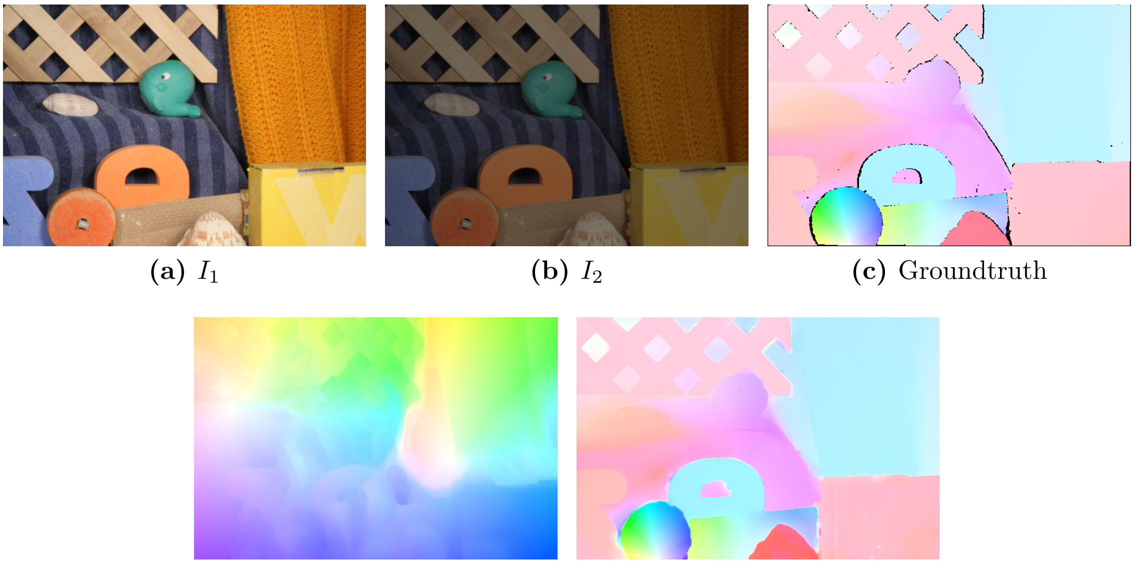
Figure 4.4: Example of the Census transform $C_\varepsilon(I, \mathbf{x}, \mathbf{y})$ applied to a 3×3 patch. The center value of the input patch (a) is taken as reference value. To each pixel in the patch one of three values is assigned, based on their relative graylevel magnitude when compared to the center pixel. (b) The Census transform provides a robust description of the patch in terms of a ternary string, which for this example is given by '20212122' (assuming the upper left pixel as starting point and clock-wise gathering of entries). This example assumes $\varepsilon = 20$.

which assigns to the pixel at location $\mathbf{y}$ one of the values $\{0,1,2\}$ based on the value of the pixel $\mathbf{x}$. An example of this transformation applied to a 3×3 patch is shown in Figure 4.4. Given two images I_1, I_2 and a flow field $\mathbf{u}:\Omega\rightarrow\Re^2$, the Census matching cost of the flow $\mathbf{u}$ is defined via the Hamming distance of the two strings as

$$\rho_c(\mathbf{x},\mathbf{u}(\mathbf{x}),I_1,I_2) = \frac{1}{|\mathcal{B}|-1}\int_\Omega \mathbb{1}_{C_\varepsilon(I_1,\mathbf{x},\mathbf{y})\neq C_\varepsilon(I_2,\mathbf{x}+\mathbf{u}(\mathbf{x}),\mathbf{y}+\mathbf{u}(\mathbf{x}))}\mathcal{B}(\mathbf{x}-\mathbf{y})\,\mathrm{d}\mathbf{y}, \tag{4.34}$$

where $\mathcal{B}$ denotes a box filter, which defines the size of the matching window, and normalization by $|\mathcal{B}|-1$ provides a normalization between 0 and 1.

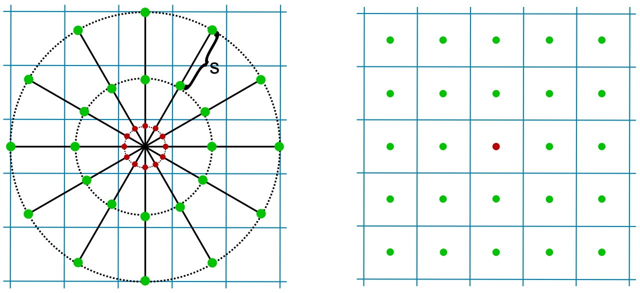
This data term is highly non-convex and non-smooth. Different variants have been proposed to still incorporate it into variational models in the literature. [106, 124] proposed a simple direct linearization of the data term based on finite differences and additionally ensured non-negativity of the approximation by taking the absolute value of the approximation. The resulting data term is convex and can be solved using a shrinkage operation, when used in the context of the primal-dual algorithm. [157] linearized the warped image and showed that a non-convex proximal mapping based on this term can be solved in closed form. While they propose to use the primal-dual algorithm, in this approach the overall energy remains non-convex. As a result this strategy is not rigorously defined from an optimization point of view. Hafner et al. [67] showed strong relations of Census matching to the gradient constancy assumption. They use a smooth approximation and



(a) I_1 **(b)** I_2 **(c)** Groundtruth

(d) Brightness constancy **(e)** Census

Figure 4.5: Census vs. brightness constancy. We simulated a multiplicative brightness change in one of the images. The model based on brightness constancy completely fails, whereas the Census data term is still able to correctly recover the optical flow.



(a) SCensus **(b)** Census

Figure 4.6: Example of the proposed sampling strategy analogous to a 5x5 census transform. The center value is computed by averaging the sampling positions on the inner most ring (red). A ternary string of length 24 is generated from the sampling positions on the outer rings (green).

optimize the overall model based on the Euler-Lagrange framework. Here we propose to approximate the data term using the quadratic approximation (4.30). Figure 4.5 compares a model with TGV regularization and the classical brightness constancy assumption to a model which uses the Census data term on a pair of images with illumination changes.

4.3.2 Scale-Robust Census Transform

Classical patch-based matching approaches are problematic when scale changes between two images occur, since the patch in the first image will capture different features than the patch in the second image. This is a common motion pattern in many applications, especially where a forward facing camera is mounted on a mobile platform, for example a car or a robot. If one knew the amount of scale change, a simple remedy to this problem would be to appropriately rescale the patch, such that the local appearance is again the same. Unfortunately the scale change in optical flow estimation is unknown a-priori.

To this end we draw ideas from SIFT descriptor matching under scale changes in order to alleviate these problems: Consider SIFT descriptors h^1 and h^2 computed from two images I_1 and I_2 at points p^1 and p^2 respectively. Hassner et al. [71] showed that if descriptors are sampled at different scales s_i and the *min-dist* measure, which is defined as

$$\min_{i,j} dist(h^1_{s_i}, h^2_{s_j}), \tag{4.35}$$

is used as matching score, it is possible to obtain accurate matches even under scale changes. Since SIFT descriptors are based on distributions of image gradients and [67] has shown a strong relationship of the Census transform to an anisotropic gradient constancy assumptions, it is reasonable to assume that a similar strategy might be applicable to Census transform matching.

We define a variant of the Census transform, which is easily amenable for multi-scale resampling, by using radial sampling instead of a window-based sampling strategy. An example of this sampling strategy is shown in Figure 4.6. We sample radially around the center point. Samples from the inner ring are averaged and serve as the basis value for generating the Census string, *i.e.* the average takes the role of the center pixel when compared to the standard Census transform. To generate the Census string, samples on the outer ring are compared to the average value. All samples are extracted using bilinear interpolation, whenever a sampling point is not in the center of a pixel. This strategy allows simple rescaling of the descriptor, which is important for an efficient implementation. The radial sampling shares similarities to Local Binary Patterns [112].

Formally, we fix some radial discretization step $\theta = \frac{2\pi}{K}$ and a radius r and introduce scale depended coordinates $\hat{\mathbf{x}} = (\hat{x}^1, \hat{x}^2)^T$

$$\hat{x}^1(k, s, r) = x^1 + rs\cos(k\theta), \qquad \hat{x}^2(k, s, r) = x^2 + rs\sin(k\theta) \tag{4.36}$$

We define the difference between the average value of the inner ring $r_i = \frac{s}{4}$ and the l-th sample from an outer ring r as

$$f(I, \hat{x}, l, s, r) = \tfrac{1}{K}\sum_{k=1}^{K}(G_s * I)(\hat{\mathbf{x}}(k, s, \tfrac{s}{4})) - (G_s * I)(\hat{\mathbf{x}}(l, s, r)), \tag{4.37}$$

where G_s denotes a Gaussian kernel with variance s. Analogous to the Census assignment function (4.33) we define the scale-dependent Census assignment function as

$$C_\varepsilon^s(I, \mathbf{x}, l, r) = \operatorname{sgn}(f(I, \mathbf{x}, l, s, r))\mathbb{1}_{|f(I,\mathbf{x},l,s,r)|>\varepsilon}, \tag{4.38}$$

This definition allows to compare descriptors at different scales s_1 and s_2 using the Hamming distance:

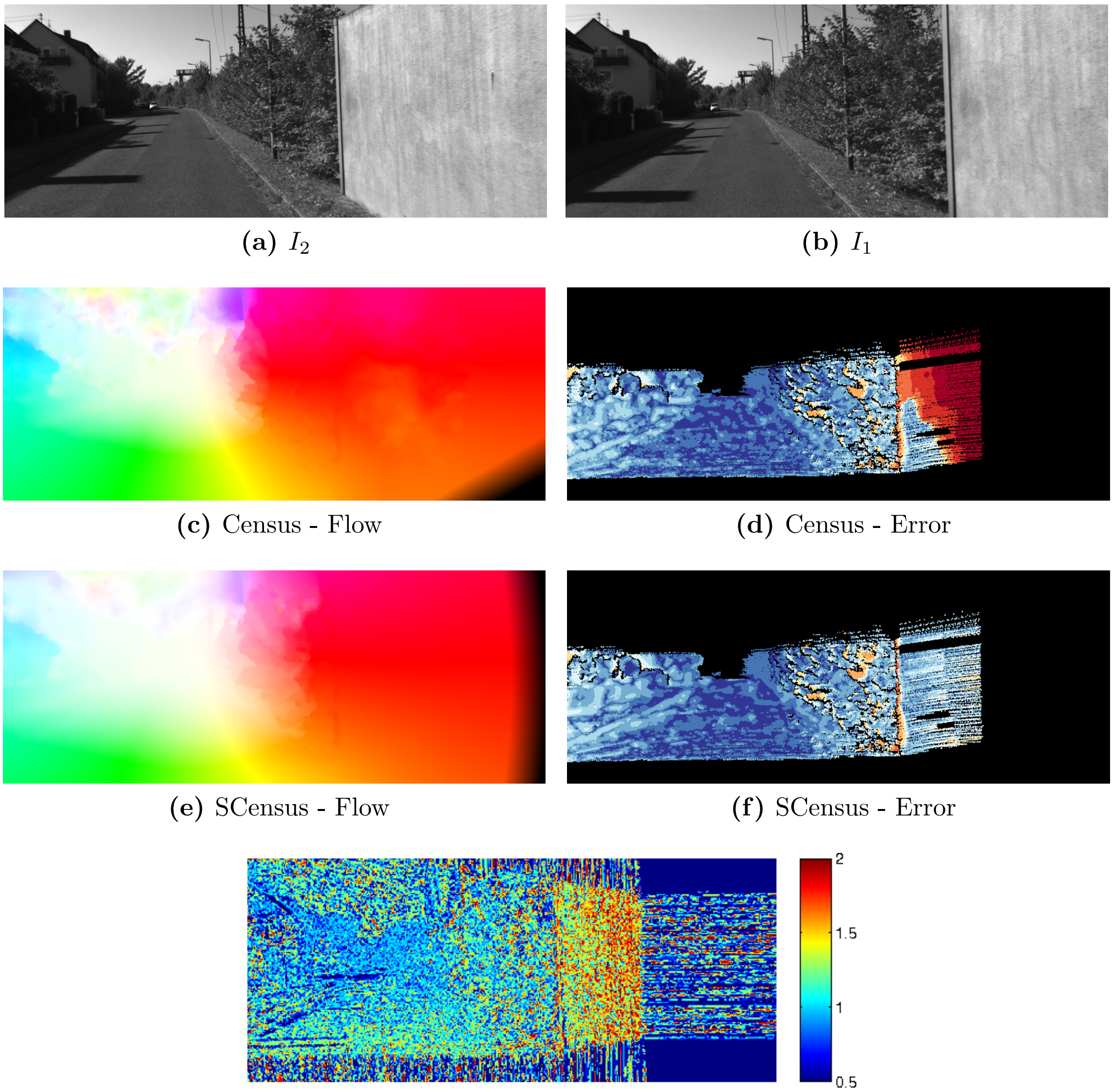
$$\rho_{s_2}^{s_1}(\mathbf{x}, \mathbf{u}(\mathbf{x}), I_1, I_2) = \sum_{l=1}^{L}\sum_{r=1}^{R}\mathbb{1}_{C_\varepsilon^{s_1}(I_1,\mathbf{x},l,r)\neq C_\varepsilon^{s_2}(I_2,\mathbf{x}+\mathbf{u}(\mathbf{x}),l,r)}. \tag{4.39}$$

By introducing the *min-dist* measure we finally arrive at the scale-robust Census data term:

$$\rho(\mathbf{x}, \mathbf{u}(\mathbf{x}), I_1, I_2) = \min_{s_1,s_2}\rho_{s_2}^{s_1}(\mathbf{x}, \mathbf{u}(\mathbf{x}), I_1, I_2). \tag{4.40}$$

While this data term is highly non-linear and non-convex, it can still be easily integrated into our continuous model using the convex quadratic approximation (4.30).

In practice we fix the scale in the first first image to the original scale and compute $\rho_{s_2}^1$ for a number of scales s_2. This definition is slightly biased toward forward motion, but is also able to handle moderate scale changes in the other direction. Figure 4.7 shows the qualitative behavior of the scale-robust data term in areas that undergo a strong scale



(a) I_2

(b) I_1

(c) Census - Flow

(d) Census - Error

(e) SCensus - Flow

(f) SCensus - Error

(g) Selected Scale

Figure 4.7: Example behaviour of the Census dataterm and the scale-robust Census dataterm. The wall to the right undergoes a strong scale change. Census fails in these areas. Using scale-robust Census we are able to find a correct flow field.

change. It can be seen that the proposed data term is able to successfully choose the correct scale on many points, which allows the global model to achieve accurate results.

4.3.3 Higher-Order Priors

Now that we have discussed enhancements to the data term we can turn our attention to the enhancement of the optical flow prior. The basic optical flow model, which was introduced in the beginning of this chapter, models the prior based on the gradients of the flow fields. A TV-based prior supports jumps in the optical flow field but has a strong assumption of piecewise constant flow fields. This not only results in staircasing but may also be detrimental in areas where the data term is ambiguous, like on slanted surfaces with a homogeneous texture. Higher-order priors can remedy this problem. We already introduced TGV and different variants which can be conditioned on the reference image. All these higher-order priors are convex, as a consequence they can be easily incorporated into our general optical flow model.

For sake of completeness, in this subsection we discuss how a optical flow model with the quadratic approximation of the (scale-robust) Census transform and a higher-order prior can be optimized. We also compare the different variants of the higher-order priors discussed in the preceding chapter visually. A quantitative comparison follows at the end of this chapter.

Total Generalized Variation We have already seen in Section 3, how a general TGV-based model with convex data terms can be optimized (see Algorithm 3.1). Now consider the following optical flow model

$$\min_{(u,v)} TGV_\alpha^2(u) + TGV_\alpha^2(v) + \lambda \int_\Omega \hat{\rho}(\mathbf{x}, (u(\mathbf{x}), v(\mathbf{x}))^T, I_1, I_2)\, \mathrm{d}\mathbf{x}, \tag{4.41}$$

where the flow component $\mathbf{u}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}))^T$ are individually penalized (there is now coupling among horizontal and vertical displacements in the prior) and the data term $\hat{\rho}$ is of the form (4.30). We assume the same discretization on a regular grid as shown in Chapter 3, which results in the spatially discrete model:

$$\begin{aligned}\min_{u,w^u,v,w^v} \alpha_1 \left|\left|\nabla u - w^u\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w^u\right|\right|_{2,1} + \alpha_1 \left|\left|\nabla v - w^v\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w^v\right|\right|_{2,1} +\\ \lambda \sum_{i,j} \hat{\rho}((i,j)^T, ((u)_{i,j}, (v)_{i,j})^T, I_1, I_2)\end{aligned} \tag{4.42}$$

Algorithm 3.1 can be applied to solve this model using some minor modifications: First, the gradient descent steps and projections which are related to the prior are decoupled in the variables u and v. They thus can be performed independently. Second, the proximal operator $\mathbf{prox}_{\tau g}(\hat{\mathbf{u}})$ needs to be specified. This term is dependend on both displacement directions and is thus coupled in general. The iterations of this algorithm are shown in Algorithm 5.3. The proximal mapping reduces point-wise to a quadratic program, which can be easily solved in closed-form. We have

$$\mathbf{u} = \mathbf{prox}_{\tau g}(\hat{\mathbf{u}}) \Leftrightarrow \mathbf{u}_{i,j} = \arg\min_{\mathbf{u}} \frac{1}{2} \left|\left|\mathbf{u} - \hat{\mathbf{u}}_{i,j}\right|\right|^2 + \lambda\tau\hat{\rho}((i,j)^T, \mathbf{u}, I_1, I_2). \tag{4.43}$$

We will use the short-hand notation $\mathbf{u}_{i,j} = \mathbf{prox}_{\tau\hat{\rho}}(\hat{\mathbf{u}}_{i,j})$ to denote this pointwise proximal operator for the remainder of this thesis.

```
1.  Set $u = \bar{u} \in \Re^M$, $v = \bar{v} \in \Re^M$, $w^u \in \Re^{2M}$, $w^v \in \Re^{2M}$
2.  Set $p^u, p^v \in P$, $q^u, q^v \in Q$, set $k = 0$
3.  While not converged
        $p^u_{k+1} = \mathbf{proj}_{p \in P}(p^u_k + \sigma(\nabla \bar{u}_k - \bar{w}^u_k))$
        $p^v_{k+1} = \mathbf{proj}_{p \in P}(p^v_k + \sigma(\nabla \bar{v}_k - \bar{w}^v_k))$
        $q^u_{k+1} = \mathbf{proj}_{q \in Q}(q^u_k + \sigma \boldsymbol{\nabla} \bar{w}^u_k)$
        $q^v_{k+1} = \mathbf{proj}_{q \in Q}(q^v_k + \sigma \boldsymbol{\nabla} \bar{w}^v_k)$
        $(u_{k+1}, v_{k+1}) = \mathbf{prox}_{\tau g}((u_k - \tau \nabla^* p^u_{k+1}, v_k - \tau \nabla^* p^v_{k+1}))$
        $w^u_{k+1} = w^u_k - \tau(\boldsymbol{\nabla}^* q^u_{k+1} - p^u_{k+1})$
        $w^v_{k+1} = w^v_k - \tau(\boldsymbol{\nabla}^* q^v_{k+1} - p^v_{k+1})$
        $\bar{u}_{k+1} = 2u_{k+1} - u_k$
        $\bar{v}_{k+1} = 2v_{k+1} - v_k$
        $\bar{w}^u_{k+1} = 2w^u_{k+1} - w^u_k$
        $\bar{w}^v_{k+1} = 2w^v_{k+1} - w^v_k$
        $k = k + 1$
```

Algorithm 4.1: Primal-Dual algorithm for optimization of (4.41).

Now, by substituting (4.30) and dropping constant terms, the proximal operator can be equivalently written as

$$\mathbf{u}_{i,j} = \arg\min_{\mathbf{u}} \frac{1}{2\lambda\tau} \left|\left|\mathbf{u} - \hat{\mathbf{u}}_{i,j}\right|\right|^2 + \frac{1}{2}\mathbf{u}^T H_{i,j}\mathbf{u} + (\nabla\rho_{i,j} - H_{i,j}(\mathbf{u}_0)_{i,j})^T\mathbf{u}, \tag{4.44}$$

where $H_{i,j}$ is a positive semi-definite approximation to the Hessian of the data term and $\nabla\rho_{i,j}$ is the spatial gradient at pixel (i, j), respectively. This is a convex quadratic program in two unknowns. We can state it in the standard form of quadratic programming by expanding the quadratic norm term and again dropping constants:

$$\begin{aligned}
&\arg\min_{\mathbf{u}} \frac{1}{2\lambda\tau}(\mathbf{u}^T\mathbf{u} - 2\hat{\mathbf{u}}^T_{i,j}\mathbf{u}) + \frac{1}{2}\mathbf{u}^T H_{i,j}\mathbf{u} + (\nabla\rho_{i,j} - H_{i,j}(\mathbf{u}_0)_{i,j})^T\mathbf{u} \\
&= \arg\min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T(H_{i,j} + \frac{1}{\lambda\tau}Id)\mathbf{u} + (\nabla\rho_{i,j} - H_{i,j}(\mathbf{u}_0)_{i,j} - \frac{1}{\lambda\tau}\hat{\mathbf{u}}_{i,j})^T\mathbf{u} \\
&= \arg\min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T A\mathbf{u} + b^T\mathbf{u} \qquad (4.45)
\end{aligned}$$

The a minimum of (4.45) is given by the solution of the linear system

$$A\mathbf{u} = -b. \tag{4.46}$$

For the diagonal approximation (4.31) the solution again decouples in the two unknowns, whereas for the full approximation (4.32) the inverse of the matrix A can be explicitly and efficiently formed.

Image-driven Total Generalized Variation We now replace the standard TGV regularizer by the image-driven variant discussed in Section 3.1.1, *i.e.* we consider the model

$$\min_{(u,v)} ITGV(u) + ITGV(v) + \lambda \int_\Omega \hat{\rho}(\mathbf{x}, (u(\mathbf{x}), v(\mathbf{x}))^T, I_1, I_2)\, \mathrm{d}\mathbf{x}. \tag{4.47}$$

Again, assuming the standard discretization on a rectangular grid, we can apply the primal-dual algorithm using some small modifications. The difference between optimization of (4.41) and (4.47) is that an additional linear operator is applied in the descent and ascent steps of the algorithm, which reflects the conditioning on the image content I_1. Moreover, it is convenient to replace the constant step-sizes τ and σ with the preconditioning scheme discussed in Section 2.2.3.1, since they would otherwise have to be estimated for every input image I_1 separately.

Using the discretization on a grid of size $N_x \times N_y$, the optical flow model with an image-driven TGV prior reads:

$$\begin{aligned}\min_{u,w^u,v,w^v} \alpha_1 \left|\left| M^{\frac{1}{2}}(\nabla u - w^u)\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w^u\right|\right|_{2,1} + \alpha_1 \left|\left| M^{\frac{1}{2}}(\nabla v - w^v)\right|\right|_{2,1} + \alpha_0 \left|\left|\boldsymbol{\nabla} w^v\right|\right|_{2,1} + \\ \lambda \sum_{i,j} \hat{\rho}((i,j)^T, ((u)_{i,j}, (v)_{i,j})^T, I_1, I_2),\end{aligned} \tag{4.48}$$

where the entries of the operator $M^{\frac{1}{2}} \in \Re^{2M \times 2M}$ are given by

$$M^{\frac{1}{2}} = \begin{pmatrix} M_a & M_c \\ M_c & M_b \end{pmatrix} = \begin{pmatrix} \mathrm{diag}(m_a) & \mathrm{diag}(m_c) \\ \mathrm{diag}(m_c) & \mathrm{diag}(m_b) \end{pmatrix}. \tag{4.49}$$

It is assumed that the first M rows of ∇ correspond to finite differences in x-direction and the last M rows of ∇ correspond to finite differences in y-direction. Different orderings are of course possible, but the form of $M^{\frac{1}{2}}$ has to be adapted accordingly.

The coefficients $m_a, m_b, m_c \in \Re^M$ are given according to the anisotropic diffusion tensor defined in (3.42):

$$\begin{pmatrix} (m_a)_{i,j} & (m_c)_{i,j} \\ (m_c)_{i,j} & (m_b)_{i,j} \end{pmatrix} = \exp(-\gamma \left|(\nabla I_1)_{i,j}\right|_2^\beta) n_{i,j} n_{i,j}^T + n_{i,j}^\perp n_{i,j}^{\perp^T}. \tag{4.50}$$

Note that the operator $M^{\frac{1}{2}}$ has a block diagonal structure and that it is symmetric. Problem (4.48) can be transformed into a saddle-point structure by rewriting the norms in terms of duality:

$$\min_{\substack{u,w^u\\ v,w^v}} \max_{\substack{p^u \in P, q^u \in Q\\ p^v \in P, q^v \in Q}} \left\langle M^{\frac{1}{2}}(\nabla u - w^u), p^u \right\rangle + \langle \boldsymbol{\nabla} w^u, q^u \rangle + \left\langle M^{\frac{1}{2}}(\nabla v - w^v), p^v \right\rangle + \langle \boldsymbol{\nabla} w^v, q^v \rangle$$
$$+ \lambda \sum_{i,j} \hat{\rho}((i,j)^T, ((u)_{i,j}, (v)_{i,j})^T, I_1, I_2). \tag{4.51}$$

Application of the primal-dual algorithm with preconditioning to this saddle-point problem yields the iterations shown in Algorithm 4.2. The algorithm includes additional diagonal preconditioning matrices Σ and T, which ensure convergence of the algorithm and can be computed efficiently on the fly. This approach was already discussed in Section 2.2.3.1 for general linear operators. For this specific instance, the preconditioning matrices can be decomposed into submatrices based on the variables they are acting on (denoted by subscripts). The preconditioning matrices can be derived by summing the coefficients of the linear operators along the horizontal and vertical directions, respectively. We assume a preconditioning exponent of $\alpha = 0$, which results in the following diagonal entries in the preconditioning matrices for the variable u and v:

$$\frac{1}{(T_{u,v})_{k,k}} = ((m_a)_{i,j} + (m_c)_{i,j})^2 + ((m_b)_{i,j} + (m_c)_{i,j})^2$$
$$+ (m_a)^2_{i-1,j} + (m_b)^2_{i,j-1} + (m_c)^2_{i-1,j} + (m_c)^2_{i,j-1}, \quad T_{u,v} \in \Re^{M \times M}, \tag{4.52}$$

where k is the linear index corresponding to pixel i, j. For the second-order variables w, we need to discriminate between the components related to horizontal and vertical derivatives (w has two entries per pixel!):

$$\begin{aligned} (T_w)_{k,k} = &\quad ((m_a)^2_{i,j} + (m_c)^2_{i,j} + 4)^{-1} \\ (T_w)_{k+M,k+M} = &\quad ((m_b)^2_{i,j} + (m_c)^2_{i,j} + 4)^{-1}, \quad T_w \in \Re^{2M \times 2M} \end{aligned} \tag{4.53}$$

Finally due to the choice of α, computing the preconditioning matrices for the dual steps amounts to merely counting the number of entries in each row of the operator, which results in constant dual step sizes:

$$\Sigma_p = \frac{1}{5} Id, \qquad \Sigma_q = \frac{1}{2} Id \tag{4.54}$$

In a practical implementation, the linear operators ∇, $M^{\frac{1}{2}}$ and the preconditioning matrices need not to be formed explicitly. All operations can be performed locally. Finite differences can be computed locally in an explicit form (see explicit computation in Section 2.3.2) and reweighted by multiplication with the local 2×2 matrix that is given by the diffusion tensor (4.50). The step-sizes which result from the preconditioning matrices can also be computed explicitly. This locality enables efficient implementations on graphics processing units, with only moderate overhead when compared to the baseline *TGV* model 4.41.

```
1.  Set $u = \bar{u} \in \Re^M$, $v = \bar{v} \in \Re^M$, $w^u \in \Re^{2M}$, $w^v \in \Re^{2M}$
2.  Set $p^u, p^v \in P$, $q^u, q^v \in Q$, set $k = 0$
3.  While not converged
        $p^u_{k+1} = \mathbf{proj}_{p\in P}(p^u_k + \Sigma_p M^{\frac{1}{2}}(\nabla \bar{u}_k - \bar{w}^u_k))$
        $p^v_{k+1} = \mathbf{proj}_{p\in P}(p^v_k + \Sigma_p M^{\frac{1}{2}}(\nabla \bar{v}_k - \bar{w}^v_k))$
        $q^u_{k+1} = \mathbf{proj}_{q\in Q}(q^u_k + \Sigma_q \boldsymbol{\nabla} \bar{w}^u_k)$
        $q^v_{k+1} = \mathbf{proj}_{q\in Q}(q^v_k + \Sigma_q \boldsymbol{\nabla} \bar{w}^v_k)$
        $(u_{k+1}, v_{k+1})^T_{i,j} = \mathbf{prox}_{\tau_{i,j}\lambda\hat{\rho}}((u_k - T_{uv}\nabla^* M^{\frac{1}{2}} p^u_{k+1}, v_k - T_{uv}\nabla^* M^{\frac{1}{2}} p^v_{k+1})^T_{i,j})$
        $w^u_{k+1} = w^u_k - T_w(\boldsymbol{\nabla}^* q^u_{k+1} - M^{\frac{1}{2}} p^u_{k+1})$
        $w^v_{k+1} = w^v_k - T_w(\boldsymbol{\nabla}^* q^v_{k+1} - M^{\frac{1}{2}} p^v_{k+1})$
        $\bar{u}_{k+1} = 2u_{k+1} - u_k$
        $\bar{v}_{k+1} = 2v_{k+1} - v_k$
        $\bar{w}^u_{k+1} = 2w^u_{k+1} - w^u_k$
        $\bar{w}^v_{k+1} = 2w^v_{k+1} - w^v_k$
        $k = k + 1$
```

Algorithm 4.2: Primal-Dual algorithm for optimization of (4.47).

Non-Local Total Generalized Variation Non-Local Total Generalized Variation can be expected to yield improved results over *ITGV* since larger areas can interact and it is possible to model segmentation cues directly into the prior. Consider the optical flow model with Non-Local *TGV* prior:

$$\min_{(u,v)} NLTGV(u) + NLTGV(v) + \lambda \int_\Omega \hat{\rho}(\mathbf{x}, (u(\mathbf{x}), v(\mathbf{x}))^T, I_1, I_2)\, \mathrm{d}\mathbf{x}, \tag{4.55}$$

The discretization of the *NLTGV* term for scalar functions u was already shown in Section 3.1.2. For the two-dimensional flow model, we introduce a signed distance matrix

$$D_{kl} = \begin{pmatrix} d^1_{kl} & d^2_{kl} & 0 & 0 \\ 0 & 0 & d^1_{kl} & d^2_{kl} \end{pmatrix} \in \Re^{2\times 4}, \tag{4.56}$$

where $d_{kl} = (d^1_{kl}, d^2_{kl})^T = l_k - l_l$ again denotes the difference in spatial location between pixels k and l. Let $\mathbf{p}_{kl} \in \Re^2$ and $\mathbf{q}_{kl} \in \Re^4$ be the dual variable associated to the connection of pixels k and l. Since it differs from our previous notation, it is important to note that this grouping is in terms of the flow components, *i.e.* the first component in $\mathbf{p}_{kl}$ is related to the horizontal displacement u and the second component is related to the vertical displacement v. Analogously for each $\mathbf{q}_{kl}$ the first two components are related to horizontal displacement and the last two components are related to vertical displacement. The discretized model can be compactly written in its primal-dual formulation as

$$\begin{aligned}\min_{\mathbf{u},\mathbf{w}} \max_{\substack{||\mathbf{p}_{kl}||_\infty \le (\alpha_1)_{kl} \\ ||\mathbf{q}_{kl}||_\infty \le (\alpha_0)_{kl}}} \sum_k \sum_{l>k} [(\mathbf{u}_k - \mathbf{u}_l + D_{kl}\mathbf{w}_k) \cdot \mathbf{p}_{kl} + (\mathbf{w}_k - \mathbf{w}_l) \cdot \mathbf{q}_{kl}] \\ + \lambda \sum_k \hat{\rho}(l_k, (\mathbf{u})_k, I_1, I_2).\end{aligned} \tag{4.57}$$

The primal-dual algorithm applied to this saddle-point problem is shown in Algorithm 4.3. We specified the iterations on the pixel level, since the linear operators arising from *NLTGV* are very complex. The feasible sets for the dual variables are given pointwise as

$$\begin{aligned}P_{kl} &= \{\mathbf{p}_{kl} \in \Re^2 : ||\mathbf{p}_{kl}||_\infty \le (\alpha_1)_{kl}\} \\ Q_{kl} &= \{\mathbf{q}_{kl} \in \Re^4 : ||\mathbf{q}_{kl}||_\infty \le (\alpha_0)_{kl}\}.\end{aligned} \tag{4.58}$$

A projection onto P_{kl} can be achieved by clamping each component to the interval $[-(\alpha_1)_{kl}, (\alpha_1)_{kl}]$. Projection onto Q_{kl} can be handled analogously. The algorithm again features a preconditioning scheme via the step-sizes τ_w, τ_u and σ_p, σ_q. Since the spatially varying weights are captured by the convex sets Q and P, these are constant whoever and fully determined by the support size of the regularizer. If we consider a window of size h, then

$$\begin{aligned}(T_u)_{k,k} &= (\tau_u)_k = (4((h-1)/2)((h-1)/2+1))^{-1} \\ (T_w)_{k,k} &= (\tau_w)_k = (8((h-1)/2)((h-1)/2+1))^{-1} \\ (\Sigma_p)_{kl,kl} &= (\sigma_p)_{kl} = (2 + ||d_{kl}||^2)^{-1} \\ (\Sigma_q)_{kl,kl} &= (\sigma_q)_{kl} = \frac{1}{2}\end{aligned} \tag{4.59}$$

The support weights are computed as bilateral weights as shown in (3.51). Other choices, such as geodesic distances [42], are possible, as long as the weights remain non-negative.

4.4 Experiments

In this section we evaluate the performance of the proposed models on two challenging data sets. The models were implemented using CUDA; all experiments were conducted on a Geforce 780Ti GPU. We use a scale factor of 0.8 for the coarse-to-fine pyramid and 15 warps per pyramid level. For the scale-robust data term we evenly sample 7 scales

```
1. Set $\mathbf{u} = \bar{\mathbf{u}} \in \Re^{2M}$, $\mathbf{w} \in \Re^{4M}$
2. Set $\mathbf{p}_{kl} \in P_{kl}$, $\mathbf{q}_{kl} \in Q_{kl}$, set $n = 0$
3. While not converged
   $\mathbf{p}_{kl}^{n+1} = \mathbf{proj}_{\mathbf{p}_{kl} \in P_{kl}}(\mathbf{p}_{kl}^{n} + (\sigma_p)_{kl}(\bar{\mathbf{u}}_k^n - \bar{\mathbf{u}}_l^n + D_{kl}\bar{\mathbf{w}}_k^n))$
   $\mathbf{q}_{kl}^{n+1} = \mathbf{proj}_{\mathbf{q}_{kl} \in Q_{kl}}(\mathbf{q}_{kl}^{n} + (\sigma_q)_{kl}(\bar{\mathbf{w}}_k^n - \bar{\mathbf{w}}_l^n))$
   $\mathbf{u}_{k}^{n+1} = \mathbf{prox}_{(\tau_v)_k \lambda \hat{\rho}}(\mathbf{u}_k^n - (\tau_v)_k \sum_{k>l}(\mathbf{p}_{kl}^{n+1} - \mathbf{p}_{lk}^{n+1}))$
   $\mathbf{w}_{k}^{n+1} = \mathbf{w}_k^n - (\tau_w)_k \sum_{k>l}(\mathbf{q}_{kl}^{n+1} - \mathbf{q}_{lk}^{n+1} + D_{kl}^T \mathbf{p}_{kl}^{n+1})$
   $\bar{\mathbf{u}}_{k}^{n+1} = 2\mathbf{u}_k^{n+1} - \mathbf{u}_k^n$
   $\bar{\mathbf{w}}_{k}^{n+1} = 2\mathbf{w}_k^{n+1} - \mathbf{w}_k^n$
   $n = n + 1$
```

Algorithm 4.3: Primal-Dual algorithm for optimization of (4.57). The algorithm is formulated in terms of local per-variable updates.

between 0.5 and 2 in both image. For *NLTGV*, we fix $w_p = 2$, which gives a good trade-off between accuracy and computational complexity. The remaining parameters were adapted for each benchmark individually. With these settings the runtimes of *TGV* and *ITGV* are approximately 4 seconds, whereas the *NLTGV*-based models take approximately 16 seconds. Faster runtimes, for the price of reduced accuracy, can be achieved by using a smaller pyramid factor or less warps.

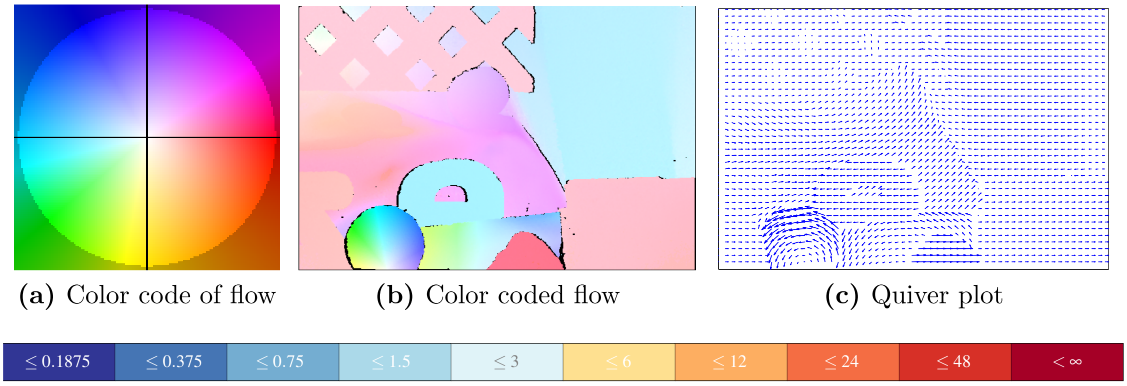
Throughout our experiments we use the average End-Point Error (EPE) in pixels as evaluation metric, which is defined as the average distance of the estimated displacement from the groundtruth displacement $\mathbf{u}_{gt}$:

$$\text{EPE}(\mathbf{u}) = \frac{1}{N_x N_y} \sum_{i,j} ||\mathbf{u}_{i,j} - (\mathbf{u}_{gt})_{i,j}||_2 \,. \tag{4.60}$$

For visualization we follow the methodology of the Middlebury benchmark [9], which visualizes optical flow as a color-coded image, where hue indicates direction and saturation indicates magnitude of the flow. Figure 4.8a shows this color-coding together with an example of a color coded flow in Figure 4.8b. An alternative visualization as a subsampled quiver plot is shown in Figure 4.8c. The color-coding of error images is shown in Figure 4.8d, where color codes pixels with *EPE* below a certain threshold.

4.4.1 KITTI Benchmark

The KITTI Benchmark [59] features an optical flow benchmark with realistic sequences taken from a moving car. The data set is split into a training set of 194 images pairs and a test set of 195 image pairs. Optical flow groundtruth on this benchmark was computed directly from groundtruth stereo data, which was obtained using a Velodyne laserscanner,



(a) Color code of flow **(b)** Color coded flow **(c)** Quiver plot

(d) Color code of error images

Figure 4.8: Color-coding.

and by projecting a 3D point cloud to a temporally neighboring frame. We use the training set, where groundtruth optical flow is available, to show the influence of Non-Local TGV as well as the scale-robust data term. As a baseline model we use standard TGV with gradient constancy assumption. We compare different combinations of regularizers and data terms: Standard TGV (TGV), Image-Driven Total Generalized Variation (ITGV) as well as Non-Local Total Generalized Variation (NLTGV). Each regularizer was evaluated in combination with the gradient constancy assumption (GRAD), normalized cross correla-

		GRAD	NCC	CENSUS	SCENSUS
TGV	2px	12.52	10.23	9.17	8.73
	3px	10.20	7.72	6.84	6.59
	4px	8.83	6.48	5.78	5.53
	5px	7.85	5.71	5.12	4.84
	EPE	2.40	1.77	1.66	1.54
ITGV	2px	11.36	8.81	8.11	7.97
	3px	9.27	6.67	6.27	6.01
	4px	8.10	5.63	5.38	5.04
	5px	7.28	4.98	4.82	4.44
	EPE	3.88	2.71	1.64	1.48
NLTGV	2px	10.19	8.46	7.58	**7.35**
	3px	8.12	6.31	5.74	**5.50**
	4px	7.06	5.35	4.90	**4.59**
	5px	6.24	4.72	4.34	**4.00**
	EPE	1.92	1.52	1.44	**1.32**

Table 4.1: Average error in % for different models and different error thresholds on the KITTI NOC-training set.

	Out-Noc [%]		Out-All [%]		Avg-Noc [px]	Avg-All [px]	RT [s]
	3px	2px	3px	2px			
NLTGV-SC	**5.93**	**7.64**	**11.96**	**14.55**	1.6	3.8	16
DDR-DF [163]	6.03	8.23	13.08	16.01	1.6	**2.7**	60
TGV2ADCS [24]	6.20	8.04	15.15	17.87	1.5	4.5	12
DataFlow [157]	7.11	9.16	14.57	17.41	1.9	5.5	180
EpicFlow [128]	7.19	9.53	16.15	19.47	**1.4**	3.7	15
DeepFlow [164]	7.22	9.31	17.79	20.44	1.5	5.8	17

Table 4.2: Average error on the KITTI test set for error thresholds $3px$ and $2px$. Suffixes "Noc" and "All" refer to errors evaluated in non-occluded and all regions, respectively. "RT" denotes runtime in seconds.

tion (NCC) (exact expressions for these terms will be given in the upcoming chapter), the Census data term (CENSUS), as well as the scale-robust Census data term (SCENSUS) discussed in this chapter. For all data terms we used the quadratic approximation in order to arrive at a convex model. The Census, NCC and $NLTGV$ window sizes were set to 5×5. We used a subset of the training set (20% of the images) to find optimal values for the remaining parameters of each method using grid-search. Since the groundtruth flow fields in this data set are not pixel-accurate, we follow the officially suggested methodology of evaluating the percentage of pixels, which have End-Point Error above some threshold [59].

Table 4.1 shows a quantitative comparison between all possible pairings of data terms and regularizers. It shows the percentage of wrong pixels for different standard error thresholds, as well as the average endpoint error. It is evident from this table that a more sophisticated regularizer leads to smaller errors in all considered error metrics. $ITGV$ consistently outperforms TGV and $NLTGV$ consistently outperforms $ITGV$, regardless of the type of data term used. This can be attributed to better behaviour around motion boundaries and in the case of $NLTGV$ also due to a larger support size that is included locally for regularization. Secondly, it can be seen that gradient constancy is consistently outperformed by more advanced data terms. NCC already shows large improvements. The Census transform again gives an improvement over NCC. Last, the scale-robust Census transform outperforms all other data terms. This is not surprising, since forward motion is the dominant motion in this data set. Finally, Figures 4.9 and 4.10 show qualitative results on this benchmark. All examples where computed with the SCENSUS data term. Both figures show that the large support size of $NLTGV$ is able to overcome artifacts, like large occluded areas or a moderate amount of highlights on the windshield. Figure 4.11 shows a failure case of the proposed methods. Illumination artifacts which are present on large parts of the image lead to conflicting estimates of the data term. This can not be overcome by any of the proposed priors. Such cases will always be problematic for two-frame optical flow methods, since they highlight the difference between apparent motion and true object motion.

The combination of *NLTGV* with the scale-robust Census transform leads to a state-of-the art optical flow method, as is indicated by the results on the KITTI test data set in Table 4.2. The proposed method *NLTGV*-SC was at the time of writing ranked first among two-frame optical flow methods. In general we can conclude that *TGV*-based models are an excellent choice for applications similar to the KITTI benchmark: Estimating optical flow from a moving vehicle in man-made environments. For this setup *TGV*-based regularizer provide strong prior cues for the task at hand and can be paired with robust data terms.

Influence of the support weights for NLTGV So far the support weights for the *NLTGV* regularizer were given by normalized bilateral weights computed from gray value images. We repeat the experiment for the model with *NLTGV* regularizer and SCENSUS data term with unnormalized support weights. The general equation used for the support weights is given by

$$\alpha_1(\mathbf{x},\mathbf{y}) = \frac{1}{Z(\mathbf{x})}\exp(-\tfrac{||I_1(\mathbf{x})-I_1(\mathbf{y})||}{w_c})\exp(-\tfrac{||\mathbf{x}-\mathbf{y}||}{w_p}), \quad \alpha_0(\mathbf{x},\mathbf{y}) = c\alpha_1(\mathbf{x},\mathbf{y}). \tag{4.61}$$

For normalized support weights, $Z(x)$ ensures that all weights in a patch sum to one. This normalization factor is given by

$$Z(\mathbf{x}) = \int_\Omega \alpha_1(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{y}. \tag{4.62}$$

For unnormalized support weights we simply set $Z(\mathbf{x}) = 1$.

Table 4.3 shows the average errors on the KITTI training data set. Unnormalized support weights give a slightly lower error rate. The difference, however, is striking visually, where it is evident that motion boundaries which have corresponding high-contrast edges in the reference image are sharper for unnormalized support weights. Motion boundaries are more strongly blurred, however, if the corresponding edge is of low-contrast (cf. Figure 4.12) in the case of unnormalized weights. This is not surprising, since the normalization balances the influence of an edge, based on the local structure of the patch under consideration.

Figure 4.13 shows an evaluation of different window sizes and color weights w_p on the accuracy. We used the Middlebury Optical Flow benchmark [9] for this evaluation. Figure 4.13a shows the average *EPE* for an increasing spatial parameter w_p, where the window size was set to $2w_p + 1$. It can be seen that the error decreases up to a window size of 7×7 and slowly rises after that. Figure 4.13b shows the influence of the color parameter. There is a sweet spot between being too strict in terms of color/intensity distance, which may result in isolated regions and being too lenient, which does not allow for an effective grouping based on color.

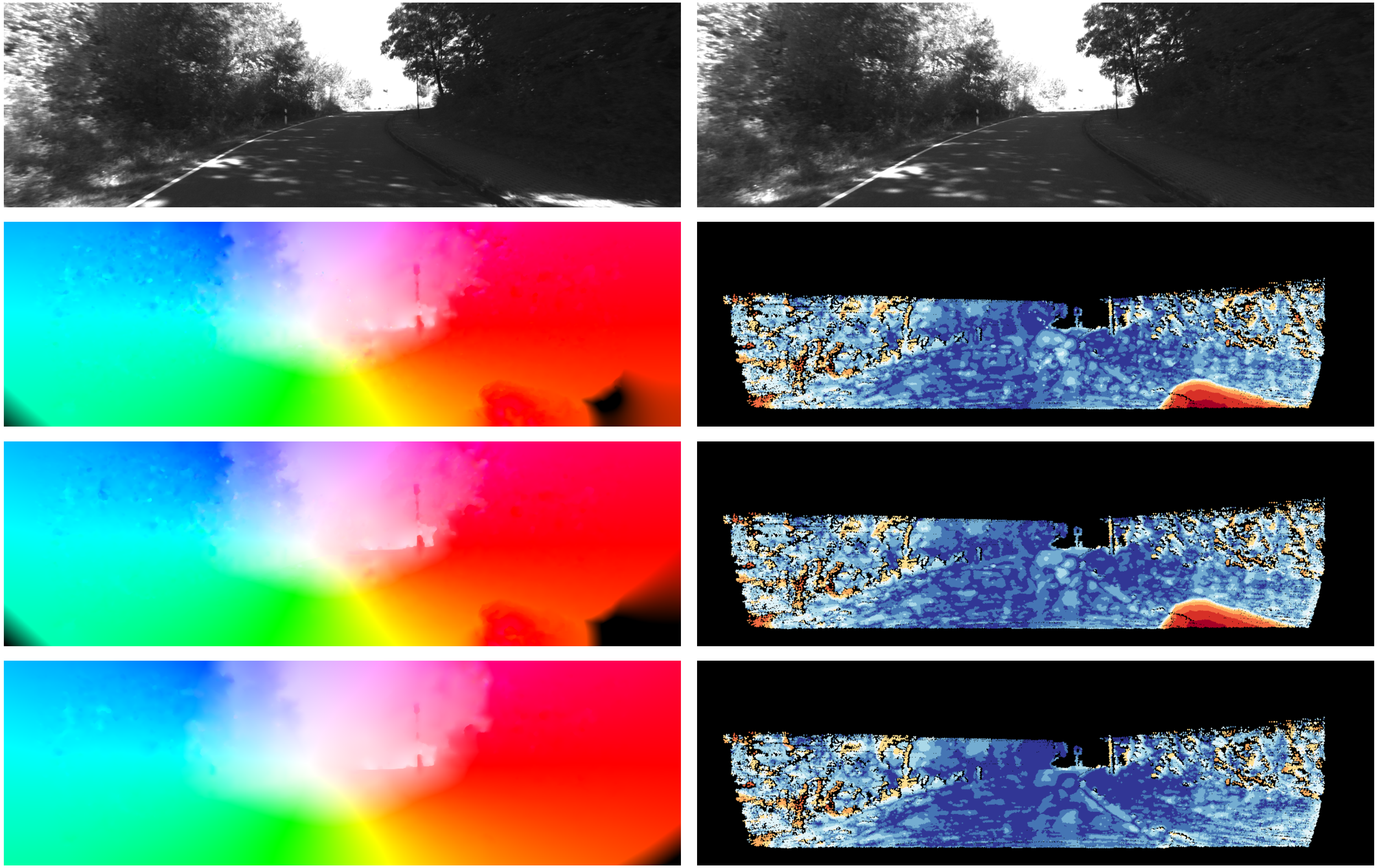


Figure 4.9: Comparison between *TGV*, *ITGV* and *NLTGV*. *NLTGV* can successfully extrapolate the flow in the challenging occluded area in the lower right of the reference image. From top to bottom: Input images, *TGV*, *ITGV*, *NLTGV*. From left to right: Flow and corresponding error.

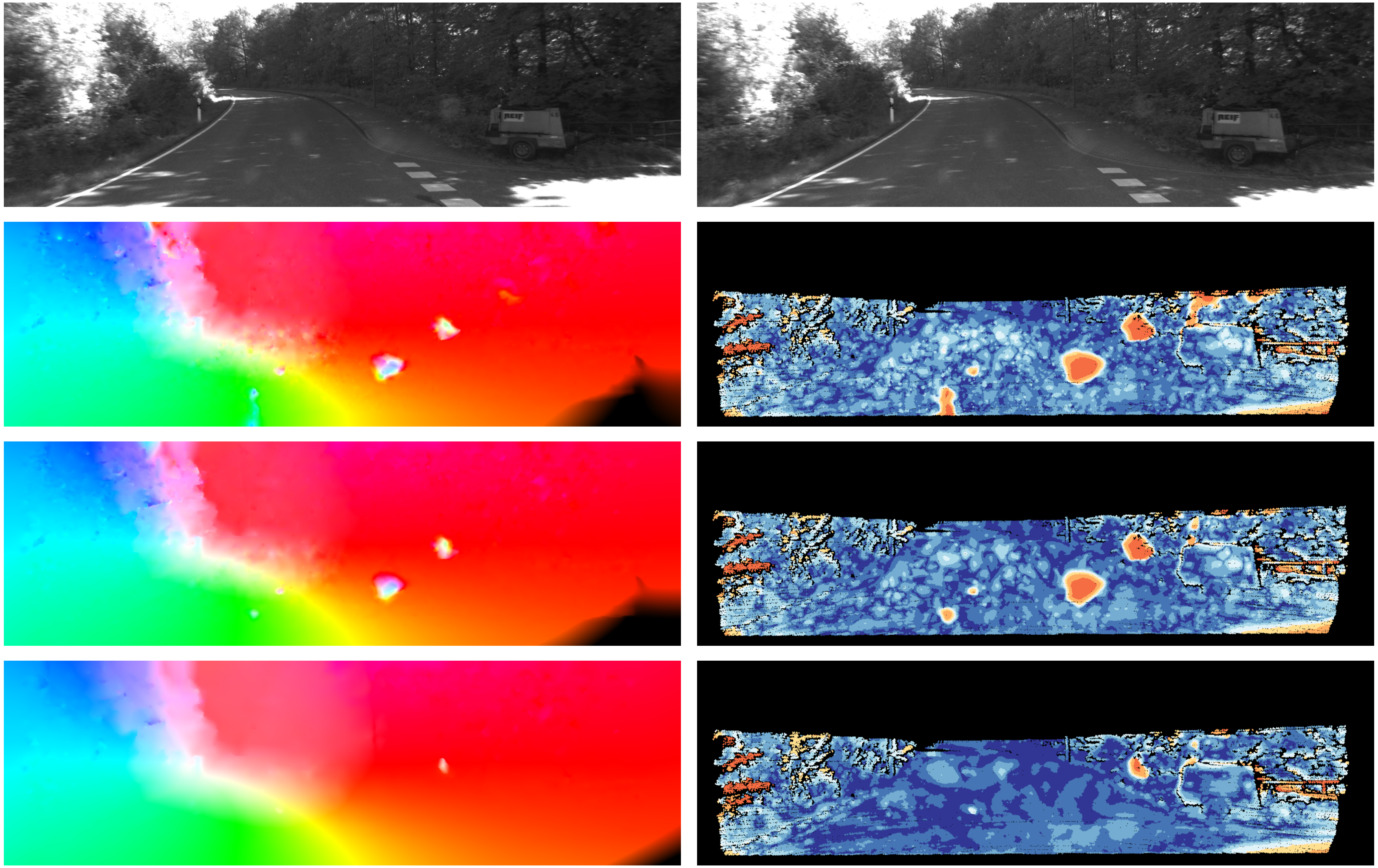


Figure 4.10: Comparison between *TGV*, *ITGV* and *NLTGV*. Note that the error introduced by the highlights on the windshield are less pronounced for *NLTGV* and *ITGV*, when compared to the *TGV* result. From top to bottom: Input images, *TGV*, *ITGV*, *NLTGV*. From left to right: Flow and corresponding error.

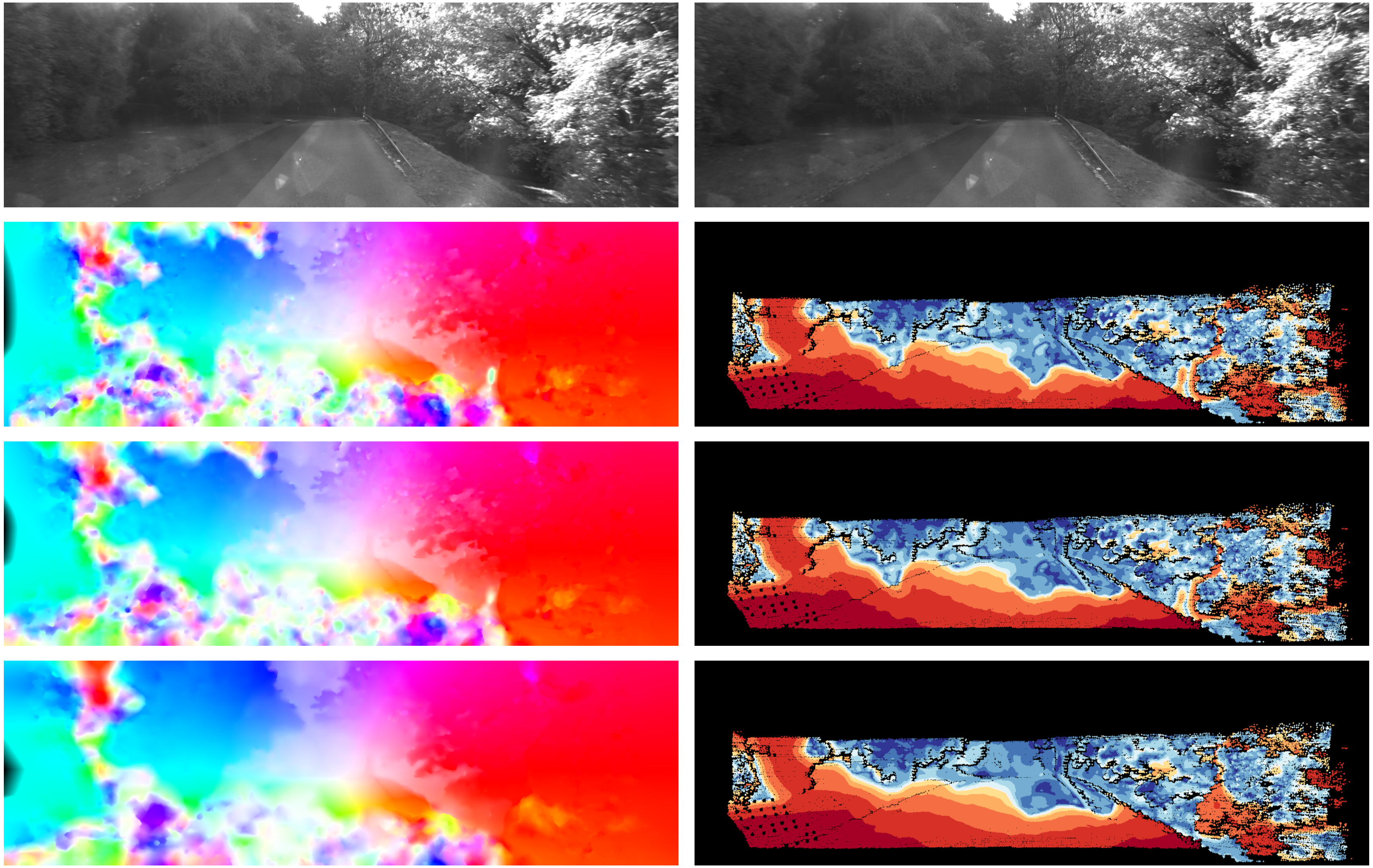
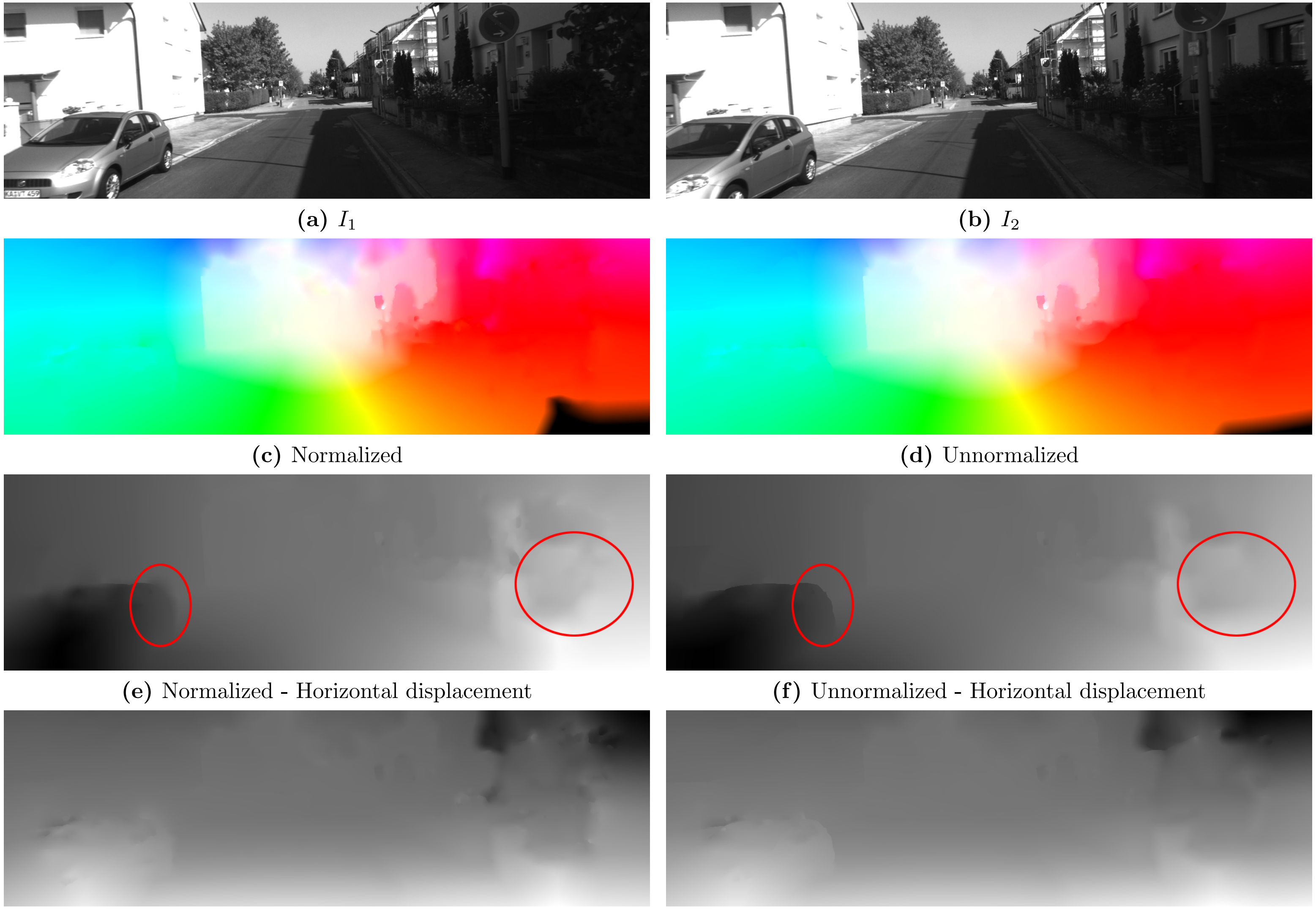


Figure 4.11: Failure case. Due to the dominant highlights on the windshield the data term completely fails for large areas. None of the regularizers is able to overcome such problems. From top to bottom: Input images, *TGV*, *ITGV*, *NLTGV*. From left to right: Flow and corresponding error.



(a) I_1

(b) I_2

(c) Normalized

(d) Unnormalized

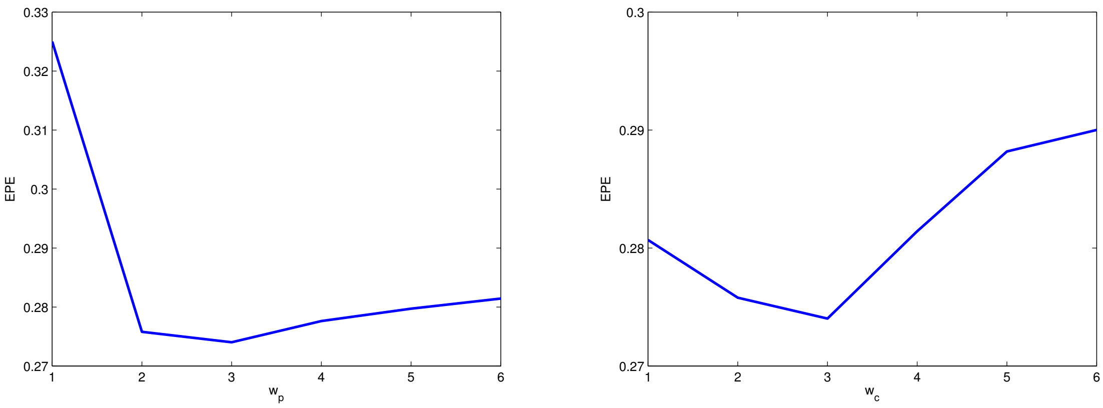
(e) Normalized - Horizontal displacement

(f) Unnormalized - Horizontal displacement

(g) Normalized - Vertical displacement

(h) Unnormalized - Vertical displacement

Figure 4.12: Comparison between unnormalized and normalized support weights.



(a) Proximity parameter **(b)** Color parameter

Figure 4.13: Evaluation of *NLTGV* parameters.

4.4.2 Sintel Benchmark

The synthetic Sintel Benchmark [32] features large motion, challenging illumination conditions and specular reflections. In our evaluation we use the *final* sequence, which additionally contains motion blur and atmospheric effects. We use two image pairs from each subsequence of the training set to find optimal parameters and report the average *EPE* as error measure.

Since this benchmark features sequences of strongly varying difficulty, we show a breakdown of the error on its individual subsequences in Table 4.4. It can be seen that *TGV* and *ITGV* perform similar on this data set, with *ITGV* being slightly more robust for very challenging sequences. *NLTGV* on the other hand clearly leads to the best results on sequences of low to moderate difficulty. The failures on very difficult scenes are typically due to very large motions and artifacts like strong motion blur. Algorithms which are based on coarse-to-fine warping completely fail on such sequences.

Table 4.5 show results on the Sintel test set. We see an improvement over the *TGV*-based model [157] and an *NLTV*-based model (*NLTV*-SC). As already noted, the most critical regions for the overall error are high-velocity regions, which are problematic in purely coarse-to-fine-based methods. Hence, it is not surprising that methods which integrate some form of sparse matching [97, 164] work better than classical coarse-to-fine-based approaches on this dataset. A-priori matches could be easily integrated into our

	2px	3px	4px	5px	EPE
Normalized	7.35	5.50	4.59	4.00	1.32
Unnormalized	**7.18**	**5.36**	**4.45**	**3.88**	**1.30**

Table 4.3: Comparison of different strategies for computing the *NLTGV* support weights.

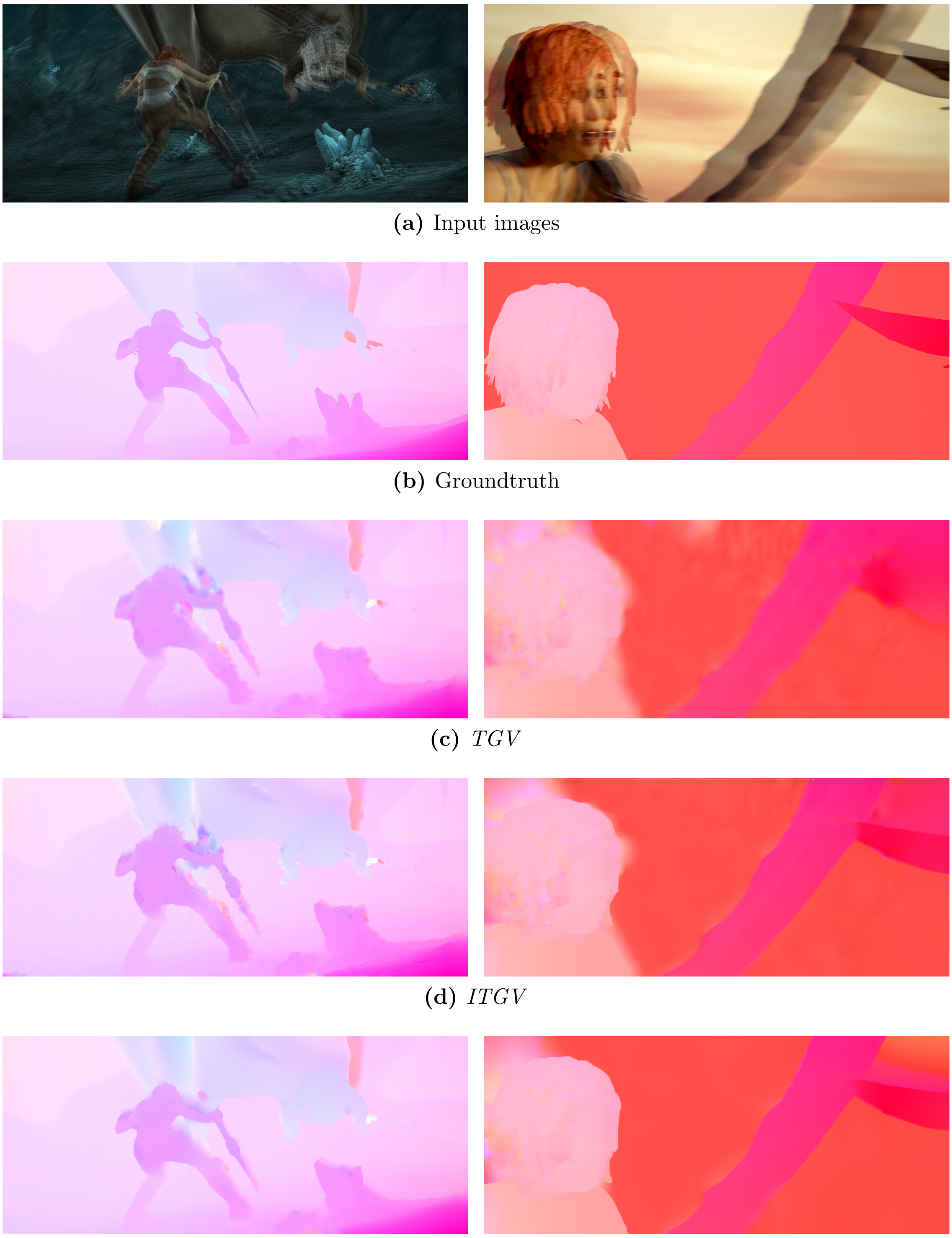
	TGV	ITGV	NLTGV
alley1	0.47	0.47	**0.36**
alley2	0.43	0.44	**0.32**
ambush2	43.16	**41.97**	43.57
ambush4	34.38	**33.99**	34.15
ambush5	13.84	**12.74**	14.20
ambush6	**22.92**	25.40	22.96
ambush7	1.74	**1.41**	1.50
bamboo1	0.56	0.56	**0.46**
bamboo2	1.61	1.95	**1.32**
bandage1	0.83	0.82	**0.72**
bandage2	0.69	0.68	**0.58**
cave2	14.60	**13.86**	14.93
cave4	6.94	6.78	**5.95**
market2	1.47	1.53	**1.18**
market5	**20.36**	23.07	21.88
market6	6.05	5.65	**4.26**
mountain1	0.98	0.85	**0.70**
shaman2	0.46	0.43	**0.34**
shaman3	0.78	0.71	**0.49**
sleeping1	0.16	0.17	**0.09**
sleeping2	0.14	0.16	**0.08**
temple2	4.00	4.11	**3.59**
temple3	20.03	**18.89**	19.30
Average	6.77	6.75	**6.61**

Table 4.4: Average *EPE* in pixels on the Sintel training data set with different regularizers. The data term was fixed to SCENSUS. *NLTGV* outperforms *TGV* and *ITGV* consistently on sequences of small to medium difficulty. *ITGV* and *TGV* are slightly more robust on sequences of high difficulty, where the coarse-to-fine warping completely fails.

model [24]. We leave such an extension for future work. Finally, Figures 4.14 and 4.15 show a qualitative comparison between *TGV* and *NLTGV* on this benchmark.

Rank	Method	EPE all	s0-10	s10-40	s40+
1	EpicFlow [128]	6.469	1.180	4.000	38.687
4	DeepFlow [164]	7.212	1.284	4.107	44.118
21	NLTGV-SC	8.746	1.587	4.780	53.860
23	DataFlow [157]	8.868	1.794	5.294	52.636
28	NLTV-SC	9.855	1.202	4.757	64.834

Table 4.5: Average *EPE* for a selection of different models on the Sintel test set. The columns “sA-B” refer to *EPE* over regions with velocities between A and B.



(a) Input images

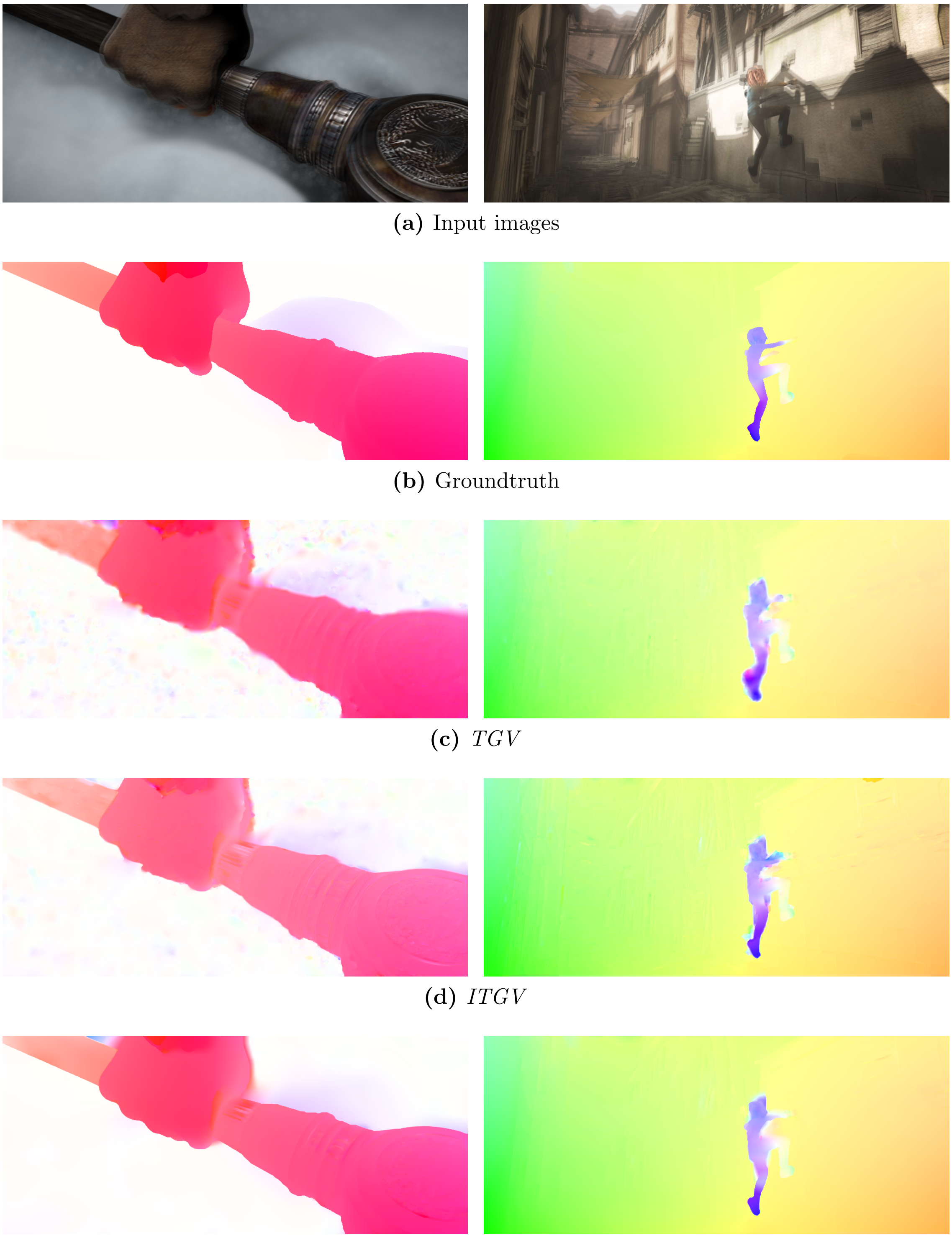
(b) Groundtruth

(c) *TGV*

(d) *ITGV*

(e) *NLTGV*

Figure 4.14: Results on Sintel dataset.



(a) Input images

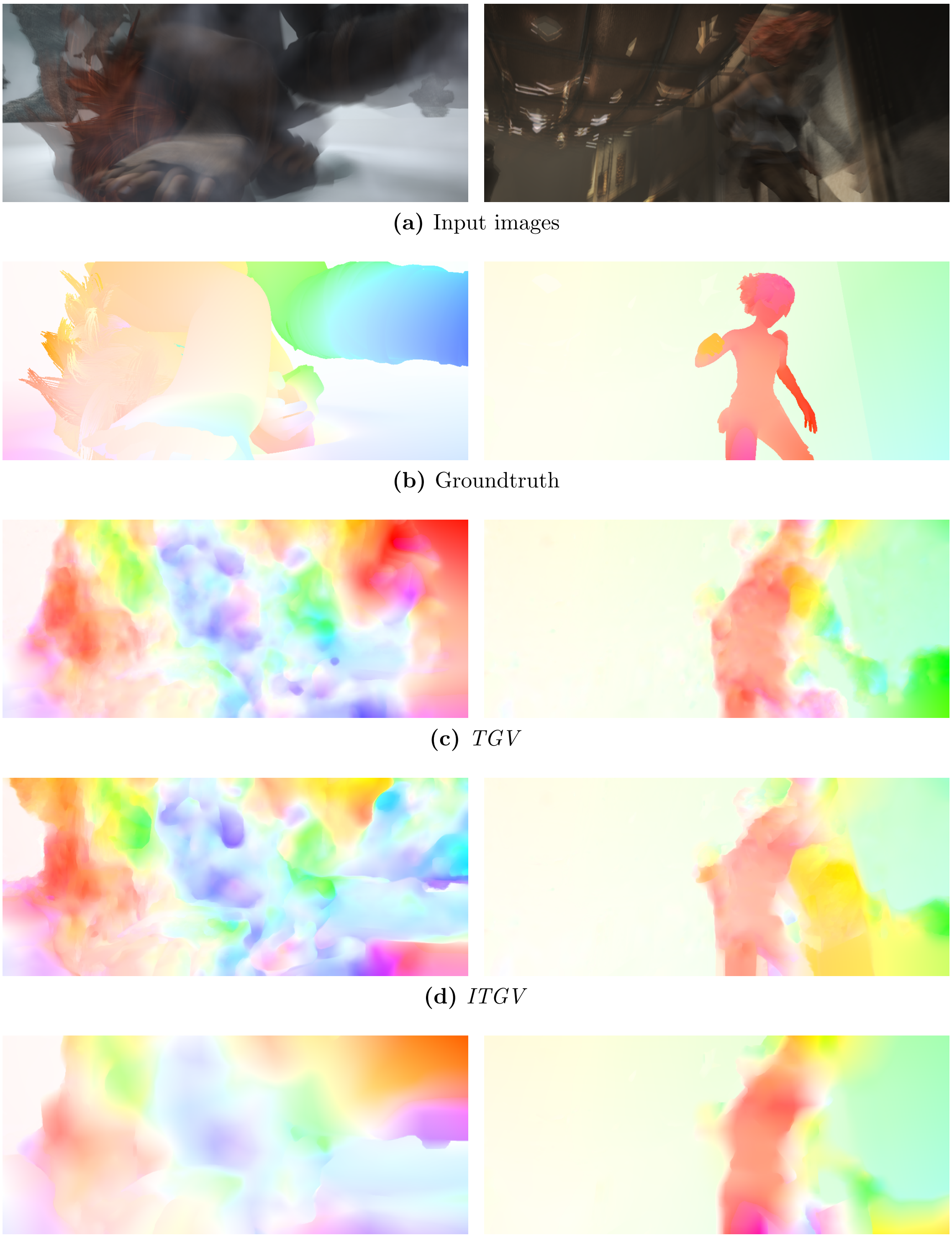
(b) Groundtruth

(c) *TGV*

(d) *ITGV*

(e) *NLTGV*

Figure 4.15: Results on Sintel dataset.



(a) Input images

(b) Groundtruth

(c) *TGV*

(d) *ITGV*

(e) *NLTGV*

Figure 4.16: Failure cases on Sintel. Sequences with very large motions and dominant occlusions can not be resolved by the coarse-to-fine approach.

4.5 Discussion

In this chapter we have introduced various robust higher-order variational models for optical flow estimation. Census-based data terms provide robust matching results, the proposed second-order models have been shown to provide state-of-the-art results on challenging optical flow benchmarks. It is notable that TGV and its proposed extensions perform especially well on the realistic KITTI dataset. One of the main reason is that the motion which is mainly featured in this data set is very well modeled by the assumption of a piecewise affine optical flow field. On the synthetic Sintel benchmark, which features a wider array of motions, we also observe an increase in robustness and accuracy, but not to such a large degree. This benchmark highlights one of the main drawbacks of coarse-to-fine based models: they have problems with very large motions, especially of small structures. Examples of such failure cases are shown in Figure 4.16. The problems get only worse if additional adverse conditions, such as atmospheric effects or large untextured regions, are present in the images.

5 Stereo

Contents

The task of stereo matching is to estimate the depth of scene points from two images viewing a scene from different angles. This concept can be extended to multiple images, each viewing the scene from different angles. Typically, two slightly shifted cameras are used to acquire stereo images simultaneously. Note, however that all principles presented here still ably to a single camera viewing a static scene from different positions at different times, provided that the two positions of the camera are known.

Variational stereo methods give an estimate of scene depth for every point. The stereo problem is intrinsically a matching problem, in order to give an estimate of scene depth it is necessary to find matching pixels or patches between the input images. While this problem is severly ill-posed, a few geometric considerations from the setup of the cameras viewing the scene can simplify the overall matching problem. In this section we give an overview of state-of-the-art variational methods. We introduce the geometric principles governing multi-view camera setups, and show how variational methods can be used to exploit the special structure of the stereo matching problem.

5.1 Epipolar Geometry

Images of stereo cameras full-fill an interesting relationship, which can be explicitly described using epipolar geometry. Points x in one camera are constrained to lie on a line in the second camera, the so-called epipolar line. This relationship reduces the stereo correspondence problem to a one-dimensional matching problem along the epipolar lines, which is arguably simpler than the optical flow problem. In this section we aim to give a brief overview on camera geometry and especially the geometry of binocular camera setups. A comprehensive treatment can be found in [70].

Let us first consider a single pinhole camera model with intrinsic calibration matrix

$$K = \begin{pmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{pmatrix}, \tag{5.1}$$

where f is the focal length (determining the distance of the image plane from the camera center) and (p_x, p_y) is the principal point in the image plane (the origin of the image coordinate system). If we assume that the camera is placed in the center of the world coordinate system and the camera coordinate system is aligned with the world coordinate system, the projection of a world point $\mathbf{X} = (X, Y, Z)^T$ to the image plane of the camera can be described in homogeneous coordinates as:

$$\mathbf{x} = \begin{pmatrix} zu \\ zv \\ z \end{pmatrix} = K \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = K\mathbf{X}, \tag{5.2}$$

where u and v are the coordinates in the image plane. The perspective transformation incurs a loss of information since any point $t\mathbf{X}$ along the ray between the camera center and the object point leads to the same projected coordinates u and v. Obviously it is not possible to directly recover the depth of a point from a single general image. The 3D information is invariably lost (although 3D geometry can sometimes be recovered, provided that the image admits some special structure [90]).

For cameras, which are not aligned with the world coordinate frame, we may introduce the so-called extrinsic camera parameters, R and t which bring the coordinate systems into alignment:

$$\mathbf{x} = K[R|t] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}. \tag{5.3}$$

Here $R \in SO(3)$ is a rotation matrix and $t = -R\mathbf{C} \in \Re^3$ is a translation vector, where

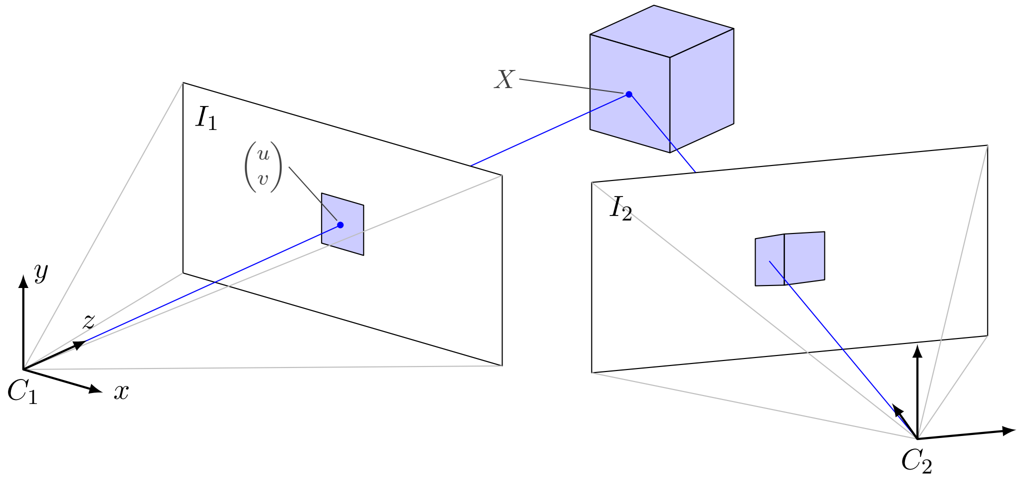


Figure 5.1: Binocular stereo vision. A 3D point on a surface is imaged by two cameras. Given two matching pixels (the intersection of the blue rays with the image plane) the 3D point can be triangulated.

$\mathbf{C} \in \Re^3$ is the camera center in the world coordinate frame. The matrix $P = K[R|t] \in \Re^{3\times4}$ is the projection matrix and fully describes the projective transformation of any 3D point to the image plane assuming an ideal pinhole camera. Projected points in the image plane can be back-projected to a ray $\mathbf{X}(\lambda)$, which passes trough the camera center and the point in the image plane:

$$\mathbf{X}(\lambda) = P^{+}\mathbf{x} + \lambda\mathbf{C}, \tag{5.4}$$

where P^{+} denotes the pseudo-inverse of the projection matrix P.

Now consider two cameras $\mathbf{C}_1$ and $\mathbf{C}_2$ with projection matrices P_1 and P_2 viewing the same scene (cf. Figure 5.1). Since both cameras view the same scene there will be corresponding points. If these correspondences are known, it is possible to reconstruct the original 3D point $\mathbf{X}$ using triangulation: The 3D point necessarily is at the location where the back-projected rays intersect. It is now obvious that two cameras in principle suffice for 3D reconstruction of a scene (up to some scale), provided that one can establish correspondences between the images of the cameras.

So far we have established that 3D reconstruction is fundamentally a correspondence problem, similar to the optical flow problem. One important difference is that the search space in the stereo problem can be reduced to a one-dimensional space by exploiting the inherent geometry of the camera setup. A fundamental observation is that if two camera view the same scene point, the projections in the camera planes can not be at arbitrary points. Instead one can deduce from the geometry of the cameras alone that for any given point in one camera, the projection in the second camera is constrained to be on a specific

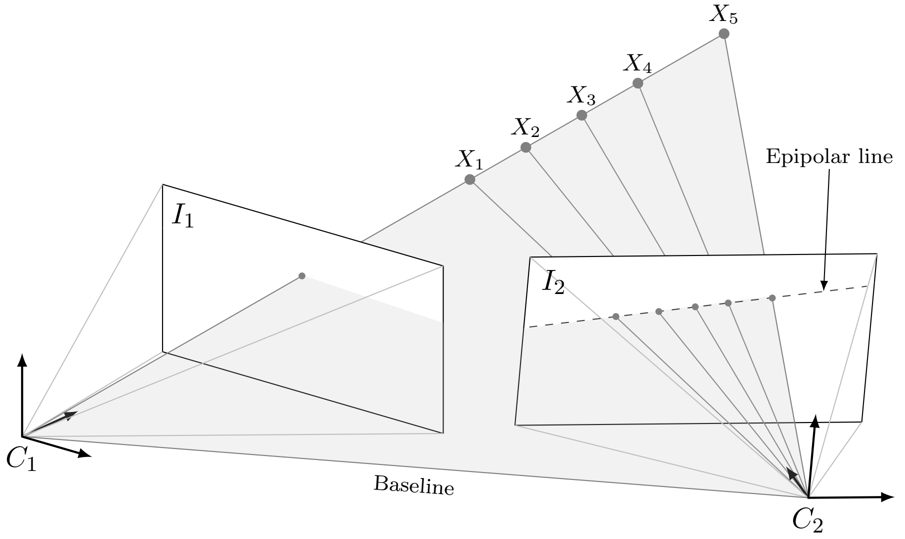


Figure 5.2: Epipolar geometry.

line. To see this consider the back-projection of the point $\mathbf{x}_1$ from the first camera:

$$\mathbf{X}(\lambda) = P_1^+\mathbf{x}_1 + \lambda\mathbf{C}_1. \tag{5.5}$$

Now consider the two world points point $P_1^+\mathbf{x}_1$ and $\mathbf{C}_1$ which both lie on this back-projected ray. The second-camera images these points as $\tilde{\mathbf{x}}_\mathbf{2} = P_2P_1^+\mathbf{x}_1$ and $\mathbf{e}_2 = P_2\mathbf{C}_1$, respectively. $\mathbf{e}_2$ is called the epipole, which is the image of the first camera center in the second camera (we can analogously define a point $\mathbf{e}_1$, which is the image of the second camera center in the first camera). Now consider the line joining the points $\tilde{\mathbf{x}}_\mathbf{2}$ and $\mathbf{e}_2$, which is given by

$$\mathbf{l}_2 = \mathbf{e}_2 \times \tilde{\mathbf{x}}_\mathbf{2}. \tag{5.6}$$

This line is denoted as epipolar line. In this representation a point $\mathbf{y}$ lies on the line if and only if $\mathbf{l}_2^T\mathbf{y} = 0$. By rewriting the cross product in matrix form, we arrive at the relation

$$\mathbf{l}_2 = [\mathbf{e}_2]_\times P_2P_1^+\mathbf{x}_1 = F\mathbf{x}_1. \tag{5.7}$$

The matrix F is called the fundamental matrix. It is important to note that the choice of world point $P_1^+\mathbf{x}_1$ is arbitrary, any other point on the ray leads to the same epipolar line $\mathbf{l}_2$ (cf. Figure 5.2). To see this simply plug in an arbitrary point on the ray

$$[\mathbf{e}_2]_\times P_2(P_1^+\mathbf{x}_1 + \lambda\mathbf{C}_1) = [\mathbf{e}_2]_\times P_2P_1^+\mathbf{x}_1 + \lambda\underbrace{[\mathbf{e}_2]_\times\mathbf{e}_2}_{=\mathbf{0}} = \mathbf{l}_2 \tag{5.8}$$

This shows that any point on the ray $\mathbf{X}(\lambda)$ is constrained to lie on a line in the second image, irrespective of its distance to the first camera. Thus in order to establish correspondences in the stereo estimation problem, it suffices to search for correspondences along the epipolar line. Moreover, the epipolar line is completely defined by the fundamental matrix F, which itself is completely defined by the projection matrices of the two cameras. A fundamental relation that corresponding points necessarily full-fill is given by

$$\mathbf{x}_2^T F \mathbf{x}_1 = 0. \tag{5.9}$$

This can be easily seen from geometric principles, e.g.the back-projected rays of the points $\mathbf{x}_1$ and $\mathbf{x}_2$ intersect in some world point $\tilde{\mathbf{X}}$, whos projection in the second-camera lies on the epipolar line $F\mathbf{x}_1$ (and vice-versa).

The fundamental matrix is a 3×3 matrix of rank 2, which has seven degrees of freedom. It can be estimated linearly from 8 correspondences [69], no knowledge of the camera calibration is necessary. The projection matrices can be recovered from the fundamental matrix up to a projective ambiguity. However, if the camera calibrations are known, the fundamental matrix can be specialized to the so-called essential matrix [99]:

$$E = K_2^T F K_1 = [\mathbf{t}]_\times R. \tag{5.10}$$

The essential matrix has only 5 degrees of freedom (due to scale ambiguity). Moreover, the projection matrices of the cameras can be recovered from the essential matrix up to some unknown scale.

5.1.1 Rectification

Rectification is the process of reprojecting the images of two cameras to a common image plane. This transformation results in horizontal epipolar lines, which is more convenient for the subsequent matching stage. Rectification is tantamount with setting up two virtual cameras which are related by a pure horizontal motion along the line connecting the optical centers of the two cameras (the so-called baseline). As a result the stereo correspondence problem is simplified to a search along the horizontal scanline.

The basic step of rectification algorithms is to rotate the cameras around their centers such that their image planes are parallel to the baseline. Moreover, the intrinsic parameters are adjusted such that the image planes are co-planar and corresponding points lie on the same scanline. It is desirable to minimize the amount of distortion that is introduced by the rectifying tranformation. While there are multiple different ways to achieve a rectification, we briefly explain the simple approach by Fusiello et al. [58]. Consider two cameras described by the projection matrices

$$P_1 = K_1[R_1| - R_1\mathbf{C}_1], \qquad P_2 = K_2[R_2| - R_2\mathbf{C}_2]. \tag{5.11}$$

The goal of the rectification algorithm is to find new projection matrices

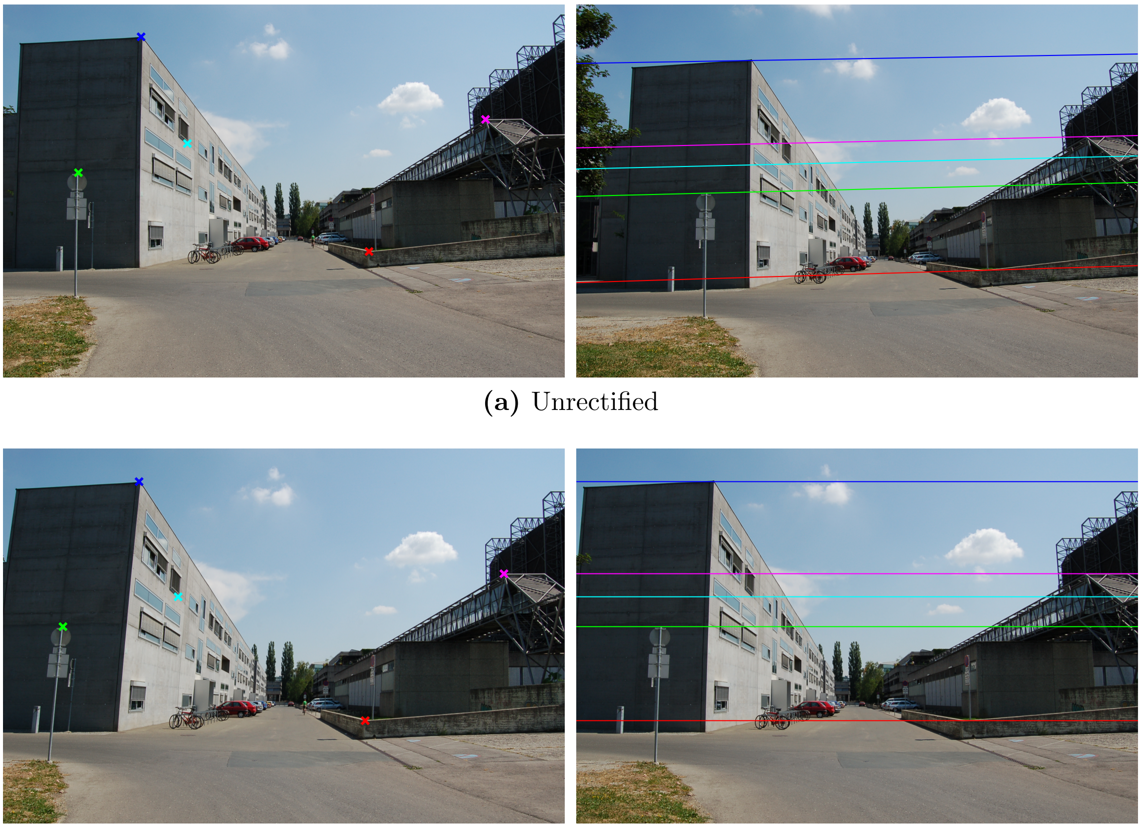
$$\tilde{P}_1 = \tilde{K}[\tilde{R}| - \tilde{R}\mathbf{C}_1], \qquad \tilde{P}_2 = \tilde{K}[\tilde{R}| - \tilde{R}\mathbf{C}_2], \tag{5.12}$$

which fulfill the previously mentioned requirements. The rotation matrix $\tilde{R}$ is chosen as

$$\tilde{R} = \begin{pmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{pmatrix}, \tag{5.13}$$

where the individual vectors $\mathbf{r}$ denote the axes of the new coordinate frame and are given by

$$\mathbf{r}_1 = \frac{\mathbf{C}_1 - \mathbf{C}_2}{||\mathbf{C}_1 - \mathbf{C}_2||}, \qquad \mathbf{r}_2 = \mathbf{k} \times \mathbf{r}_1, \qquad \mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2, \tag{5.14}$$



(a) Unrectified

(b) Rectified

Figure 5.3: Example of image rectification. A few salient image points in the left image together with their corresponding epipolar lines in the right image are shown. Note that in the rectified pair, the epipolar lines are horizontal and on the same scanline as the points to be matched in the left image.

where $\mathbf{k}$ is some arbitrary unit vector. The new intrinsic parameters $\tilde{K}$ can be set arbitrarily (it is only important that they are the same for both cameras), but are computed by averaging the original intrinsic parameters in practice.

The final rectifying tranformation, which transform points from the original camera to the new rectified camera, *i.e.* the homography $\tilde{\mathbf{x}}_\mathbf{1} = T_1\mathbf{x}_1$, can be computed from the old and the new camera matrices by

$$T_1 = \tilde{Q}_1 Q_1^{-1}, \tag{5.15}$$

where $\tilde{P}_1 = [\tilde{Q}_1|\tilde{\mathbf{q}}_\mathbf{1}]$ and $P_1 = [Q_1|\mathbf{q}_\mathbf{1}]$, *i.e.* $\tilde{Q}_1$ and Q_1 are 3×3 submatrices of the old and new projection matrix, respectively. The rectifying transformation for the right camera can be computed analogously.

Figure 5.3 shows an example of stereo rectification. Note that in the unrectified setup the epipolar lines are slanted and the corresponding point-epipolar pairs do not lie on the same scanline. After rectification, the epipolar lines are parallel and coincide with the horizontal scanline.

The described rectification process will fail if the original cameras are related by a pure forward translation [58], and result in strongly distorted rectified images if there is a dominant forward translation between the cameras. While such a setup will not arise in a classical binocular stereo system, it may arise in the case of a single forward moving camera. For such cases it is often still possible to perform a rectification using more sophisticated methods [113].

5.1.2 Computing Depth from Rectified Image Pairs

For rectified image pairs we have shown that the stereo problem reduces to a matching problem along the horizontal scanline. It remains to show how depth can be recovered from this setup. Consider the rectified stereo setup shown in Figure 5.4. A 3D world point $\mathbf{X} = (X, Y, Z)^T$ is imaged in the left and right cameras at horizontal positions x_1 and x_2 and at vertical position y, respectively. If we assume the first camera (the reference camera) to be in the origin of the world-coordinate system, then from similar triangles, we get the relations

$$\frac{Z}{f} = \frac{X}{x_1}, \qquad \frac{Z}{f} = \frac{X-b}{x_2}, \qquad \frac{Z}{f} = \frac{Y}{y}, \tag{5.16}$$

where the length of the horizontal translation between the cameras is denoted by the baseline b. By rearranging it is possible to compute the position $\mathbf{X}$ of the world point:

$$Z = \frac{bf}{x_1 - x_2}, \qquad X = \frac{x_1 Z}{f}, \qquad Y = \frac{yZ}{f}. \tag{5.17}$$

The defining quantity is the horizontal displacement $d_x = x_1 - x_2$ between the images of the world point in the two cameras. We call this quantity the disparity, which is typically directly the sought after quantity in two-view stereo matching algorithms. It is inversely proportional to the depth of a world point and offers the benefit that it is a natural quantity to search for correspondence, since it can be directly estimated by correlating pixels or patches at uniformely spaced positions on the horizontal scanline.

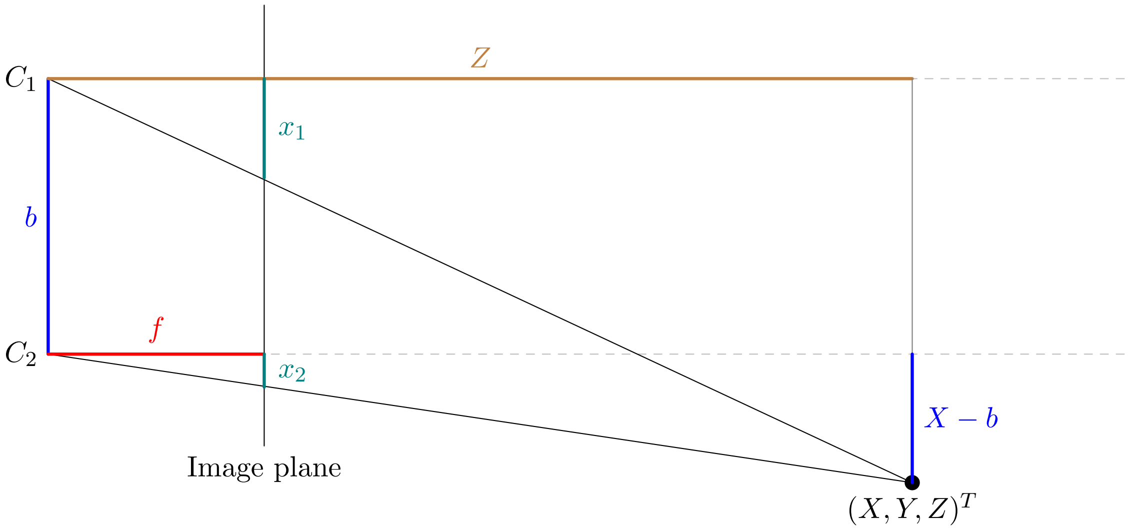


Figure 5.4: Recovery of depth from disparity. We assume that the world coordinate system coincides with the coordinate system of camera C_1 (the reference view).

5.2 Related Work

Stereo models can be similarly categorized into different classes of models as optical flow models. In view of the most prevalent approaches, however, it becomes obvious that in contrast to optical flow, the vast majority of models for stereo matching are discrete, *i.e.* the matching problem is modeled as a discrete labeling problem. This is due to the fact that for common image sizes and stereo setups, discrete modeling is computationally feasible, since the label space is only one-dimensional. Since large displacements are not an inherent problem in discrete models, they are especially appealing for stereo setups with a large baseline. We will later introduce a variational model that is motivated and formulated in continuous space, but reduces to a discrete problem in the actual implementation.

Based on the dominant approaches, we distinguish three classes of stereo algorithms:

1. Local methods

2. Global methods

3. Semi-Global methods

Again, it is often not possible to perform a hard categorization of any given algorithm, since many of them draw ideas from the different categories (in particular, local methods typically directly form the basis of global and semi-global methods).

Local Methods Local methods are the earliest approaches to compute depth from binocular camera setups [98]. Local methods match patches or pixels along the epipolar line at some predefined discrete intervals (disparities). They then assign to each pixels the disparity giving the lowest matching error/cost. The accuracy and robustness can be increased by aggregating costs in some neighborhood of a pixel. This can be achieved by simple averaging [50] or median filtering or by more sophisticated edge-aware approaches [81].

The most popular correlation methods will be discussed in the following section. Apart from the used correlation method, local methods may differ in the way how the cost is aggregated in order to increase robustness to outliers and allow for well-aligned depth boundaries and how the results are post-processed. Yoon and Kweon [174] propose to us a bilateral weighting scheme to increase the accuracy along depth boundaries. Zhang et al. [178] propose an efficient cross-based edge-aware aggregations scheme based on integral images. Einecke and Eggert [51] relax the inherent assumption of fronto-parallelism in the cost aggregation stage, which allows for larger correlation windows even in the presence of slanted surfaces.

Since local methods do not explicitly model global smoothness, the results are often noisy and ambiguous areas are not handled well. On the positive side, local methods are very fast and easy to implement. Moreover, the local matching often forms the basis of global or semi-global methods, it is thus of significant interest to find robust and accurate local methods.

Global Methods Global methods on the other hand also have a global smoothness assumption. Global methods themselves again can be divided into discrete and continuous models. Discrete models model disparity as a discrete quantity. As such local methods form the basis of these type of global models. Discrete models have the advantage that fairly general energies, both in terms of the prior and the data term, can be considered. But this comes for the price of large memory demand and computational complexity (this is especially striking if the number of possible disparities is large). Continuous models on the other hand are comparatively fast and have low memory demand, but are not as flexible when it comes to the form of the energy.

There is a large body of work on discrete stereo models. A classical paradigm is to directly model disparity as a label and solve the resulting combinatorial optimization problem using Graph Cuts [88] or Belief Propagation [54, 118, 148, 149, 173]. Another popular approach is to use external proposals and perform a series of fusion moves to iteratively optimize the global model [96, 167]. Recent state-of-the-art approaches abandon the pixel-wise labeling in favor of higher-level reasoning on super-pixels or small segments of the scene [170, 171]. Other recent state-of-the-art models include more frames into the estimate and perform a simultaneous estimation of optical flow and stereo [172] or directly model the 3D scene flow and reproject it to get stereo and optical flow estimates [158, 159]. Most of these methods rely on standard stereo approaches like Semi-Global Matching [77] to generate accurate initial proposals, which are then refined.

Continuous global models are less popular for the task of stereo matching, since they tend to miss small-scale structure and large disparity ranges may pose significant challenges to the optimization procedure. Nonetheless there are a few examples of these models: Wei et al. [163] used a continuous Markov Random Field (MRF) formulatiom for stereo and optical flow, with a learned regularization term. Heise et al. [74] incorporated the PatchMatch stereo algorithm [18] into a continuous model. A classical optical flow-style stereo model was proposed in [124] and later augmented with a discrete search procedure [93].

Semi-Global Methods It is worth to note that there are also Semi-Global Methods, which try to find a middle ground between the speed of local methods and the accuracy of global methods. The most prevalent approach is Semi-Global Matching (SGM), which has proven to be a fast and robust method in real-world applications [56]. There are various modifications and extensions to this approach which aim to enhance the robustness and accuracy of *SGM* [75, 78, 82, 140].

A last class of algorithms, which can be considered semi-global in natures, is based on the PatchMatch algorithm [11]. These algorithms perform randomized propagation of disparities based on a smoothness assumption, but do not optimize a global energy. The PatchMatch algorithm has been adapted to stereo by Bleyer et al. [18]. They greedily propagate local plane parametrizations across the image. This approach was later successfully combined with global models [13, 74].

5.3 Data terms

This section introduces the most important matching terms for stereo estimation. We will later evaluate these correlation measures in the context of a global model. For the stereo matching problem we assume rectified image pairs. Thus the displacement vector is given by $\mathbf{d} = (d_x, 0)^T$.

Absolute differences and Related data terms The simplest matching criterion is given by performing a simple pixel-wise test

$$\rho_{bright}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \left| I_1(\mathbf{x} + \mathbf{d}) - I_2(\mathbf{x}) \right| . \tag{5.18}$$

This is a reasonable matching criterion under ideal conditions, but will quickly fail otherwise. A simple approach to get more robust estimates is to consider a patch centered around the pixel of interest and to apply the very same matching criterion by summing over all pixels in the patch. This gives the Sum of Absolute Differences (SAD) cost:

$$\rho_{\mathrm{SAD}}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \sum_{\hat{\mathbf{x}} \in P(\mathbf{x})} \left| I_1(\hat{\mathbf{x}} + \mathbf{d}) - I_2(\hat{\mathbf{x}}) \right|, \tag{5.19}$$

where $P(\mathbf{x})$ denotes the patch centered at coordinate $\mathbf{x}$.

SAD is more robust than pixel-wise absolute differences, but introduces an additional artifact, known as foreground fattening: Pixels along a depth edge get wrongly assigned to the foreground. This effect is visible in any patch-based approach, which does not explicitly account for this effect (for example through a adaptive aggregation scheme [178]), but can often be mitigated in a later stage using a global model that supports an image-driven regularization. The absolute value can be replaced for example by a quadratic function (although the sum of quadratic differences is less robust than SAD). In any case, direct measurements on pixels are prone to fail if there are slight variations between the input images.

A more robust data term can be computed by matching image gradients instead of pixel values:

$$\rho_{grad}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \left|\left| \nabla I_1(\mathbf{x} + \mathbf{d}) - \nabla I_2(\mathbf{x}) \right|\right|_2. \tag{5.20}$$

This measure is robust to illumination changes, but results in more noise in the estimates (which is hardly a problem in global models, since they are robust to noise). Again it is possible to get more robust estimates by summing the gradients in a patch around the pixel of interest.

Normalized Crosscorrelation A measure that is more robust than brightness or gradient-based cost functions is given by the normalized cross correlation [53]. Normalized crosscorrelation is a window-based method, where windows from the left and right images are correlated using the equation:

$$\mathrm{NCC}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \frac{\sum_{\hat{\mathbf{x}} \in P(\mathbf{x})} (I_1(\hat{\mathbf{x}} + \mathbf{d}) - \mu_1)(I_2(\hat{\mathbf{x}}) - \mu_2)}{\sqrt{\sum_{\hat{\mathbf{x}} \in P(\mathbf{x})} |I_1(\hat{\mathbf{x}} + \mathbf{d}) - \mu_1|^2 |I_2(\hat{\mathbf{x}}) - \mu_2|^2}}, \tag{5.21}$$

where μ_1 and μ_2 denote the mean gray value of the patch under consideration in I_1 and I_2 respectively, *i.e.* we have

$$\mu_1 = \frac{1}{|P(\mathbf{x})|} \sum_{\hat{\mathbf{x}} \in P(\mathbf{x})} I_1(\hat{\mathbf{x}} + \mathbf{d}) \qquad \mu_2 = \frac{1}{|P(\mathbf{x})|} \sum_{\hat{\mathbf{x}} \in P(\mathbf{x})} I_2(\hat{\mathbf{x}}) \tag{5.22}$$

The resulting values are in the range $[-1, 1]$, where a correlation value of 1 denotes perfect correlation. Since we are formulating the stereo matching problem as a minimization problem, the NCC-based matching term is given by

$$\rho_{\text{NCC}}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \min(1, 1 - \text{NCC}(I_1, I_2, \mathbf{x}, \mathbf{d})), \tag{5.23}$$

where the minimum operations ensures that patches with negative correlations are assigned a constant high cost. The normalized crosscorrelation is invariant to affine transformations of the intensities of the image patch, which provides additional robustness when compared to brightness or gradient-based cost functions.

Mutual Information Mutual Information is a statistical quantity that describes the mutual relation between two random variables. It is defined via the joint entropy of the images I_1^w and I_2, where I_1^w is the image I_1 after warping:

$$\text{MI}(I_1^w, I_2) = H(I_1^w) + H(I_2) - H(I_1^w, I_2), \tag{5.24}$$

where the joint entropy is defined as

$$H(I_1^w, I_2) = -\int_0^1 \int_0^1 P_{I_1^w, I_2}(i_1, i_2) \log P_{I_1^w, I_2}(i_1, i_2) \, \mathrm{d}i_1 \, \mathrm{d}i_2. \tag{5.25}$$

$P_{I_1^w, I_2}$ denotes the joint probability of images I_1^w and I_2 and the entropies $H(I_1^w)$ and $H(I_2)$ are defined via the marginal distributions of the joint probability $P_{I_1^w, I_2}$. Similar images have high Mutual Information, whereas dissimilar images have low Mutual Information.

Egnal [49] computed the mutual information between patches for stereo matching. This approach provides robustness to illumination changes, but is expensive to evaluate in practice. Kim et al. [87] showed how Mutual Information can be used for the stereo matching problem, without resorting to a patch-based approach. They use the fact that $H(I_2)$ does not depend on the disparities and assume that $H(I_1^w)$ is almost constant, these terms can thus be neglected in an energy minimization/maximization framework. They then use a first-order Taylor approximation to rewrite the Mutual Information as an integration over pixels in contrast to the integration over the joint distribution (5.25), which again allows to build a cost volume: The normalized joint histogram can be defined as

$$P_f^0(i_1, i_2) = \frac{1}{N_x N_y} \sum_{\mathbf{x}} T[(i_1, i_2) = (I_1^w(\mathbf{x}), I_2(\mathbf{x}))], \tag{5.26}$$

where $T[a = b]$ is 1 if a equals b and 0 otherwise. The histogram is smoothed using a Gaussian kernel of (user-defined) variance σ to obtain an estimate of the joint probability

$$P_f(i_1, i_2) = P_f^0(i_1, i_2) * g_\sigma(i_1, i_2). \tag{5.27}$$

Define

$$D(i_1, i_2) = -\frac{1}{N_x N_y} \log P_f(i_1, i_2) * g_\sigma(i_1, i_2), \tag{5.28}$$

then the per-pixel mutual information cost is given by

$$\rho(I_1, I_2, \mathbf{x}, \mathbf{d}) = D(I_1(\mathbf{x} + \mathbf{d}), I_2(\mathbf{x})). \tag{5.29}$$

This can again be used to build a cost volume for stereo matching. Hirschmüller [77] argued that the marginal entropies should not be assumed to be constant (the constancy assumptions fails in the presence of occlusions) and also included the computation of the marginal distributions in an analogous way.

Rank transform The Rank transform [175] is a non-parametric transformation of an image. It encodes the relative magnitude of the center pixel of a patch by determining the rank among its neighbors, *i.e.* the number of pixels in the patch which are smaller than the center pixels. The Rank transform is morphological invariant, two patches will be assigned the same rank if they are related by a monotonically increasing function of the gray values. Formally, the rank transform of an image I at position x is given by

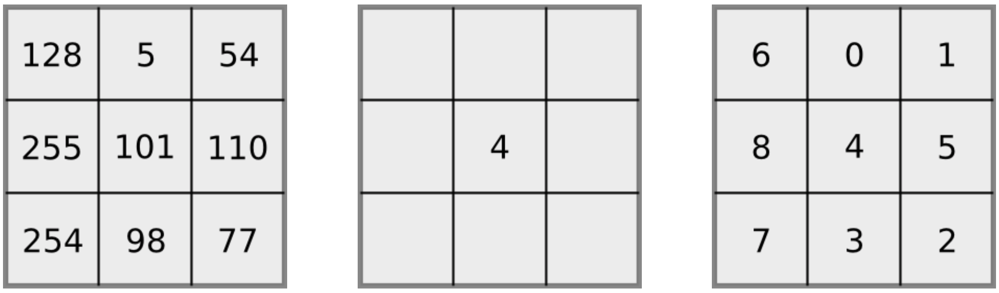
$$R(I; \mathbf{x}) = |\{\mathbf{y} \in P(\mathbf{x}) \mid I(\mathbf{y}) < I(\mathbf{x})\}|, \tag{5.30}$$

where $P(\mathbf{x})$ again denotes the patch centered at position $\mathbf{x}$. After both input images have been transformed using the Rank transform, absolute differences or the sum of absolute differences are used as correlation measure.

Demetz et al. [44] introduced an extension to the Rank transform, called Complete Rank transform. The approach is to compute a signature of each patch by considering the ranking of each pixel in the patch, where again rank is determined by the number of pixels with smaller gray value than the pixel under consideration. Figure 5.5 shows an example of this approach. This signature again is morphologically invariant, but in contrast to the Rank transform, it retains the local structure of the patch and thus is more discriminative when used for subsequent correlation.

Census transform Another non-parametric transformation is given by the Census transform, which was already introduced in the context of optical flow estimation. The Census transform [175] is well-known for its robustness to illumination changes and is frequently used in stereo algorithms. For global continuous models Müller *et al.* [106] recently showed its usefulness in the context of optical flow estimation.

Similar to the Rank transform and the Complete Rank transform, the Census cost is morphologically invariant. It is interesting to note that the Complete Rank Transform can be interpreted as $|P(x)|$ Census transforms, each using a different reference pixel in the patch under consideration [44]. As such the Census transform takes a middle-ground



128	5	54
255	101	110
254	98	77

(a) Input patch

	4	

(b) RT

6	0	1
8	4	5
7	3	2

(c) CRT

Figure 5.5: Example of Rank transform (RT) and Complete Rank transfom (CRT). The resulting CRT signature for this patch is '601523784' (assuming the upper left pixel as starting point and clock-wise gathering of entries).

between the Rank transform and the Complete Rank transform. Moreover it can be implemented very efficiently on modern computing architectures using only comparisons and bitwise manipulations.

Modifications Patch-based matching approaches can be made more robust using a simple truncation scheme:

$$\rho_T(I_1, I_2, \mathbf{x}, \mathbf{d}) = \min(t, \rho(I_1, I_2, \mathbf{x}, \mathbf{d})). \tag{5.31}$$

This approach is especially useful for brightness and gradient-based matching, but also enhances the robustness of normalized crosscorrelation.

Matching is often carried out at integer disparities, which limits the theoretical accuracy of these approaches since they are subject to artifacts introduced by the discrete nature of the image sensor. The straightforward approach to increase the accuracy to subpixel ranges is to sample at fractional disparities, where the warped image is interpolated at non-integer disparities. This approach, while simple, has the drawback that the costvolume gets larger, which increases the overall computational load not only for the matching stage but also for subsequent smoothing stages. Birchfeld and Tomasi [14] introduced an approach to increase the subpixel accuracy of discrete matching approaches by resampling the disparity at subpixel locations and taking the minimum of the local neighborhood as cost for the integer disparities. The resulting costvolume has the same dimensions as the costvolume sampled only at integer locations, but is less prone to introduce artifacts which stem from pixel discretization. The basic approach is given by

$$\rho_{BT}(I_1, I_2, \mathbf{x}, \mathbf{d}) = \min_{a \in A}(\rho(I_1, I_2, \mathbf{x}, \mathbf{d} + a), \rho(I_2, I_1, \mathbf{x}, \mathbf{d} - a)), \tag{5.32}$$

where A denotes a set of subpixel shifts along the scanline, e.g.

$$A = \{[0,0]^T, [\tfrac{1}{2},0]^T, [-\tfrac{1}{2},0]^T\}. \tag{5.33}$$

Birchfeld and Tomasi [14] showed how this modified matching term can be computed for brightness-based matching with only a small constant overhead. Note, however, that this approach can in principle be applied to any patch-based matching cost, although it results in increased computational complexity. In many cases, however, the computational complexity of subsequent stages far surpasses the complexity of the matching stage, so that this increase is not of concern. In practical applications it is sometimes useful to also resample not only along the scanline but also in vertical direction [126] in order to increase robustness with respect to slight calibration error (which result in not perfectly horizontal scanlines):

$$A = \{[0,0]^T, [\tfrac{1}{2},0]^T, [-\tfrac{1}{2},0]^T, [0,\tfrac{1}{2}]^T, [0,-\tfrac{1}{2}]^T\}. \tag{5.34}$$

Another modification is to use linear combinations of different, complementary data terms. Sun et al. [150] empirically observed that a combination of Census matching and absolute differences results in increased robustness.

5.4 A Lifting-based Higher-Order Model

The epipolar geometry of a stereo setup constraints the possible matching locations between pixels to the epipolar line. The unconstrained two-dimensional matching problem can thus be reduced to a one-dimensional matching problem. In this section we introduce a model that exploits this one-dimensionality and the specific physical meaning of disparity in order to find accurate solutions [126]. The basic building block is a convex reformulation of the non-convex stereo matching problem.

We focus on models with second-order Total Generalized Variation (TGV) regularization, as this is the most widely used and also the simplest instance of TGV (besides TV), *i.e.* we consider functionals of the form

$$\min_{u,w} \overbrace{\alpha \int_\Omega |Dw|_\Gamma + \underbrace{\int_\Omega |Du - w|_\Sigma}_{}}^{E_1(w|u)} \underbrace{{} + \lambda \int_\Omega \rho(u)\, \mathrm{d}\mathbf{x}}_{E_2(u|w)}, \tag{5.35}$$

where $u : \Omega \to \Re$ and $w : \Omega \to \Re^2$, D is the distributional derivative, which is also well defined for discontinuous functionals, and the norms are defined as $|x|_M = \langle x, Mx\rangle^{\frac{1}{2}}$, M symmetric and positive definite. The introduction of the operator M will later allow us to easily incorporate anisotropic edge-weighted diffusion into the model. Note that for $\Gamma = I$ and $\Sigma = I$, the definition reduces to the standard definition of second-order TGV [26]. We will assume throughout the rest of this section that the data term $\rho(u)$ is non-convex. An extension of this basic formulation to higher-order instances of TGV is straight-forward, as it only involves a modified version of subproblem $E_1(w|u)$, the main difficulty of the model lies in $E_2(u|w)$, since it includes the non-convex data term. We will shortly see, however, that the special structure of the regularizer together with the ordering properties of the disparity space still allow for solutions of this problem.

Our main observation is as follows: It is possible to decompose problem (5.35) into the two subproblems $E_1(w|u)$ and $E_2(u|w)$. Let the pair (u^*, w^*) be a stationary point of (5.35), then it is obvious that the relation

$$u^* = \arg\min_u E_2(u|w^*) \tag{S1}$$

$$w^* = \arg\min_w E_1(w|u^*) \tag{S2}$$

holds, *i.e.* given w^* it is possible to deduce u^* by solving a possibly simpler subproblem and vice versa. Note that (S1) is a non-convex problem, while (S2) is a convex problem, which is equivalent to a generalized vectorial TV-L1 denoising problem [111]. This observation points to an iterative scheme for finding approximate solutions to (5.35):

$$\begin{aligned} u^{n+1} &= \arg\min_u E_2(u|w^n) \\ w^{n+1} &= \arg\min_w E_1(w|u^{n+1}). \end{aligned} \tag{A1}$$

Note that by definition we have $E(u^n, w^n) \geq E(u^{n+1}, w^n) \geq E(u^{n+1}, w^{n+1})$ and $0 \leq E(u, w) < \infty$, $\forall(u, w)$, therefore the procedure will converge in the functional value, although not necessarily to a global optimum.

The update steps in (A1) already constitute the basic iterations of the proposed algorithm for optimizing (5.35), which can be interpreted as a non-smooth block-descent scheme. It remains to show how to solve the individual subproblems in each step.

5.4.1 Global Solutions for First-Order Models

The subproblem $E_2(u|w)$ is a non-convex variational problem with a non-convex data term and a convex regularization term. It was shown by Pock et al. [121] that problems with this special structure can be solved globally optimal using the framework of calibrations, provided that the regularization term admits some additional structure. The basic idea is to lift the problem to a higher-dimensional space, where a globally optimal solution to the original problem can be computed. A similar construction was proposed in the context of discrete graphical models by Ishikawa [83]. We will shortly discuss both variants in the following and establish their relation, before we proceed to the actual optimization of $E_2(u|w)$.

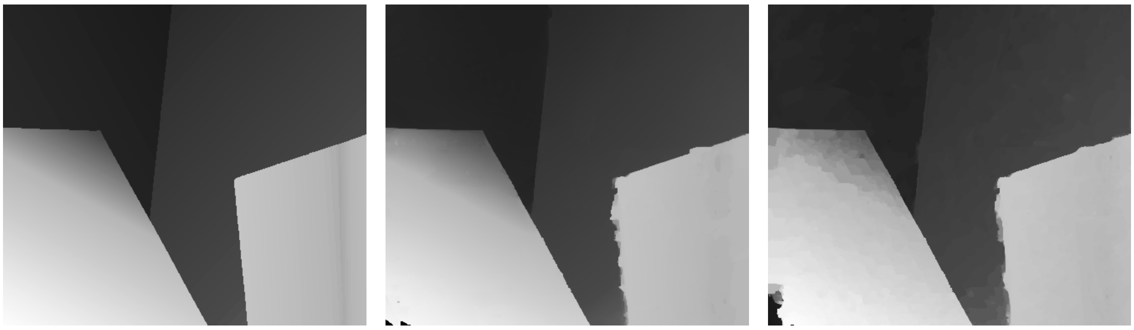

(a) Groundtruth **(b)** *TGV* **(c)** *TV*

Figure 5.6: Comparison between the proposed lifting-based higher-order model and a lifting-based model with TV prior. The higher-order model is able to capture the piecewise affine nature of the scene, leading to an almost perfect estimate. The TV-based model on the other hand shows strong staircasing on slanted surfaces.

5.4.1.1 Variational Construction

Consider a TV-based stereo problem with some arbitrary non-convex data term:

$$\min_u \int_\Omega |Du| + \lambda \int_\Omega \rho(\mathbf{x}, u(\mathbf{x}))\, \mathrm{d}\mathbf{x}. \tag{5.36}$$

Pock et al. [122] showed how this non-convex problem can be equivalently formulated as a convex problem using a functional lifting approach. The basic idea is to decompose the unknown function u in terms of its sub-level sets. Consider the indicator function of the t-th upper level $\mathbf{1}_{\{u>t\}} : \Omega \to \{0,1\}$, which is defined as

$$\mathbf{1}_{\{u>t\}}(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad u(\mathbf{x}) > t \\ 0 & \text{else} \end{cases} \quad . \tag{5.37}$$

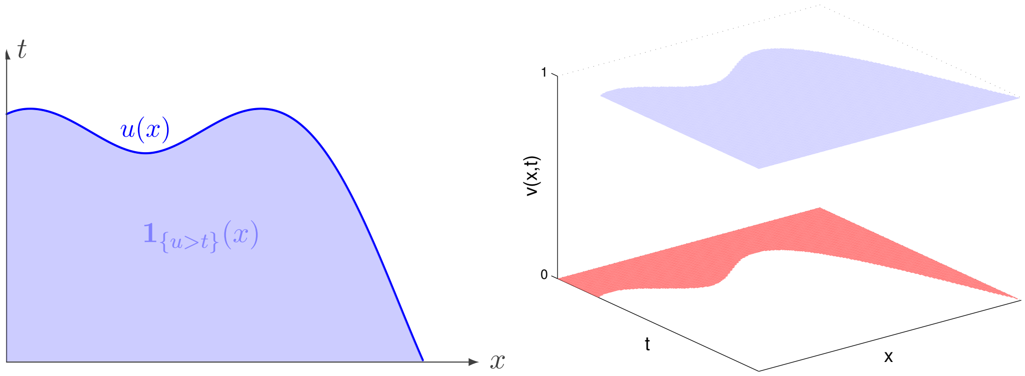
We define a binary function on the joint image and level-set domain $\Omega \times T$

$$v(\mathbf{x}, t) = \mathbf{1}_{\{u>t\}}(\mathbf{x}). \tag{5.38}$$

Assume without loss of generality that disparities lie in the interval $T = [0, t_{\max}]$ (if the minimum disparity is not equal to zero, the label space can be simply shifted accordingly). By the definition of upper level sets, for any u which takes values in T, it must hold that its corresponding function $v(\mathbf{x}, \mathbf{t})$ fulfills $v(\mathbf{x}, 0) = 1$ and $v(\mathbf{x}, t_{\max}) = 0$ and is monotone decreasing in t . Thus the function v is restricted to lie in the set

$$D = \{v : \Omega \times T \to \{0,1\} \mid v(\mathbf{x}, 0) = 1,\, v(\mathbf{x}, t_{\max}) = 0, \quad D_t v \leq 0\} \tag{5.39}$$

For a level-set representation $v(\mathbf{x}, t)$ its corresponding function $u(\mathbf{x})$ can be recovered by



(a) $\mathbf{1}_{\{u>t\}}(x)$ **(b)** $v(x,t)$

Figure 5.7: Example of the subgraph indicator function $\mathbf{1}_{\{u>t\}}(\mathbf{x})$ and corresponding function $v(\mathbf{x},t)$ for a one-dimensional function. (a) The original function and its subgraph indicator $\mathbf{1}_{\{u>t\}}(\mathbf{x})$. (b) The corresponding lifted function $v(\mathbf{x},t)$, which is equal to one wherever $u(\mathbf{x}) > t$ (blue) and zero otherwise (red).

integrating along the level-set variable t:

$$u(\mathbf{x}) = \int_T v(\mathbf{x},t)\,\mathrm{d}t. \tag{5.40}$$

It is important to note that both $u(\mathbf{x})$ and $v(\mathbf{x},t)$ are two different representations of the same underlying function. A one-dimensional example for this representation is shown in Figure 5.7.

The core observation of [122] now is that by rewriting energy (5.36) in terms of the function v, it is possible to arrive at an equivalent problem with convex objective function but non-convex domain (due to the binary domain of v):

$$\min_{v \in D} \int_{\Omega\times T} |D_x v| + \rho(\mathbf{x},t)|D_t v|. \tag{5.41}$$

This approach uses a fundamental relation between the Total Variation of a function and the Total Variation of its level-sets, known as the (generalized) co-area formula [34, 55]:

$$\int_\Omega |D_x u| = \int_{\Omega\times T} |D_x v|. \tag{5.42}$$

The data term can also be decomposed in terms of the level sets as

$$\int_\Omega \rho(\mathbf{x},u(\mathbf{x}))\,\mathrm{d}\mathbf{x} = \int_{\Omega\times T} \rho(\mathbf{x},t)\delta(u(\mathbf{x})-t)\,\mathrm{d}\mathbf{x} = \int_{\Omega\times T} \rho(\mathbf{x},t)|D_t v|, \tag{5.43}$$

where we denote by D_x the usual spatial distributional derivative and by D_t the distributional derivative in direction t. It is now easy to see the equivalence between (5.41) and (5.36). For every valid level set indicator function the energy of the corresponding function u can be computed directly on the level sets, without explicit recovery of u. Moreover the objective is convex, the only remaining non-convexity is due to the binary constraints, which ensure that v indeed is a valid level set indicator. A further relaxation of the binary domain to the interval $[0, 1]$, yields an overall convex functional. The most remarkable property of this relaxation is that exact minimizers of the non-convex problem can be recovered by thresholding the solution of the relaxed problem [38, 122].

5.4.1.2 Ishikawa's Method

Ishikawa [83] earlier showed a similar lifting approach for the stereo problem based on the formalism of discrete *MRF*s. The fundamental idea of lifting the label space to a higher-dimensional space by using a (discrete) level set representation for the unkown labeling and exploiting the ordering properties of the stereo label space is also leveraged in this approach. The fundamental difference is that here the stereo problem is treated as an inherently discrete problem. The practical differences between the discrete construction and the variational model are not as large as one might think, however. Since the previously presented variational lifting approach boils down to an effectively discrete label space after discretization of the variational model, the resulting energies share strong similarities. In fact, for certain types of regularizers, namely the ones which can be decomposed into a sum over pairwise pixel interactions, the discretized energy of the variational model and the *MRF* energy of the Ishikawa construction are exactly equivalent. The last difference is that the model is solved using a (also globally optimal) combinatorial solver instead of using convex optimization.

Let us introduce the general notion of a pairwise *MRF*: A pairwise *MRF* is defined on an undirected graph $G = (V, E)$ consisting of a set of nodes V and a set of edges E. The nodes V can be labeled with values from some predefined label set. The edges model interactions between nodes. One can define an energy for some labeling u on this graph as

$$E(u) = \sum_{k \in V} g_k(u_k) + \sum_{(k,l) \in E} h_{(k,l)}(u_k, u_l). \tag{5.44}$$

We assume that u_k takes values in some discrete set (the label space). Minimization of such a discrete energy is typically intractable, depending on the specific form of the graph, the label space and the pairwise potentials $h_{(k,j)}$. Global minimization of *MRF*s which include loops and a label space with more than two labels is in general NP-hard [17]. The two label case is tractable provided that the pairwise potentials are submodular [89], *i.e.* they fulfill the condition:

$$h_{(k,j)}(0, 1) + h_{(k,j)}(1, 0) \geq h_{(k,l)}(1, 1) + h_{(k,l)}(0, 0). \tag{5.45}$$

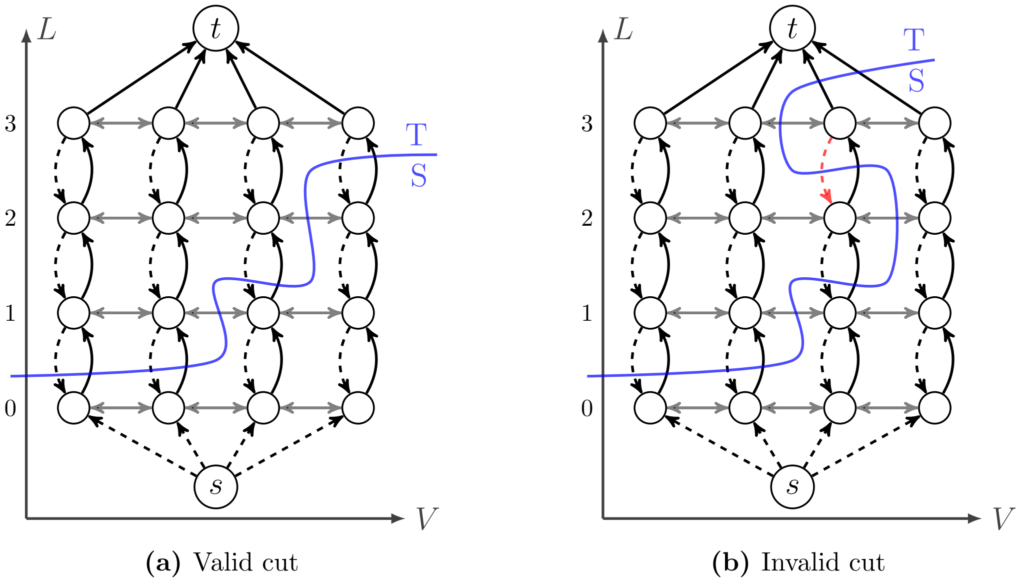


Figure 5.8: Example of Ishikawa's graph construction for a chain *MRF* of length 4 and 4 labels. Dashed edges correspond to edges with infinite capacity. Horizontal edges correspond to pairwise potentials, vertical solid edges correspond to unary potentials. (a) A valid cut cuts each column exactly once. (b) An invalid cut has infinite energy, since an edge with infinite capacity is cut (red).

The prevalent approaches for solving submodular energies are combinatorial min-cut/max-flow algorithms [22]. Other tractable energies are multi-label energies on graphs without loops (trees) or two-label energies on loopy graphs which admit the additional structural property of outer-planarity [136].

Ishikawa's approach considers energies of the special form

$$E(u) = \sum_{k \in V} g_k(u_k) + \sum_{(k,l) \in E} \alpha_{kl} f(u_k - u_l), \qquad u_k \in L, \tag{5.46}$$

where $\alpha_{kl} \geq 0$, u_k takes values in the discrete set $L = \{0, \ldots, T\}$ and the function $f : \Re \to \Re$ is assumed to be convex. An important choice here is that that the pairwise potential only depends on differences $u_k - u_l$ of their labels. The idea now is to perform minimization of the multi-label energy (5.46) by constructing an auxiliary binary graph, in which the minimum cut energy is equivalent to the minimum energy of (5.46). This is achieved by lifting the label space to a higher-dimensional space and introducing a directed graph construction, which ensures that the minimum cut solution represents a discrete level set indicator function.

A graphical illustration for a one-dimensional *MRF* is shown in Figure 5.8. The key ingredients are constraint edges (dashed) which ensure that each column is cut exactly once. Data edges (black) directly correspond to the unary cost of assigning a specific label to a node. Finally, penalty edges (gray) can be directly computed from the individual pairwise terms in (5.46). This construction ensures that every cut corresponds to a valid configuration $\hat{u}$. Moreover, the energy of the cut (the sum over the edge capacities which where cut), is equal to the energy $E(\hat{u})$. Thus by finding a minimum cut (a cut with minimum cost), it is possible to find the labeling with minimum energy. It turns out that the graph is not only binary but also submodular, thus finding this minimum cut is indeed possible in polynomial time using combinatorial max-flow/min-cut algorithms.

Relation to the Variational Approach Consider the following instance of the discrete model (5.46):

$$E(u) = \sum_{k \in V} g_k(u_k) + \lambda \sum_{(k,l) \in E} |u_k - u_l|, \qquad u_k \in L, \tag{5.47}$$

We adopt a discrete variant of the subgraph representation (5.40), *i.e.* $u_k = \sum_{i \in L \cup \{T+1\}} v_{k,i}$, with $v_{k,0} = 1$, $v_{k,T+1} = 0$ and $v_{k,i+1} \leq v_{k,i}$, where we extended the label space with an additional label $T + 1$. Then it is easy to check that the energy can be equivalently written as the binary energy

$$E(v) = \sum_{i \in L} \sum_{k \in V} g_k(i) |v_{k,i+1} - v_{k,i}| + \lambda \sum_{i \in L \cup \{T+1\}} \sum_{(k,l) \in E} |v_{k,i} - v_{l,i}|, \quad v_{k,i} \in \{0, 1\}. \tag{5.48}$$

This is a spatially discrete variant of the variational model (5.41), if we assume anisotropic Total Variation and that the edge set E contains edges between direct horizontal and vertical neighborhoods in a 4-connected grid graph. It remains to show that this energy can be represented as a minimum cut problem in a directed graph. The key in the construction are again the constraints, since (5.48) is not submodular due to the data term. Constraints can be incorporated by including infinite capacity edges (since cuts which include these edges have infinity energy). The first constraint $v_{k,0} = 1$ can be realized by infinite capacity edges from the source to all $v_{k,0}$. The second constraint, monotonicity along the label space, is equivalent to stating that each path from node $v_{k,0}$ to $v_{k,T+1}$ can be cut exactly once. This constraint can be realized by introducing constraint edges from $v_{k,i+1}$ to $v_{k,i}$. Having the constraints in place, the data cost is counted exactly once per column k, namely where $v_{k,i} - v_{k,i+1} = 1$, we thus add an edge from $v_{k,i}$ to $v_{k,i+1}$ with capacity $g_k(i)$. Finally, for each neighbor in the image grid we have the smoothness penalty

$$f(v_{k,i}, v_{l,i}) = \begin{cases} \lambda & \text{if} \quad (v_{k,i} = 0 \wedge v_{l,i} = 1) \vee (v_{k,i} = 1 \wedge v_{l,i} = 0) \\ 0 & \text{if} \quad (v_{k,i} = 1 \wedge v_{l,i} = 1) \vee (v_{k,i} = 0 \wedge v_{l,i} = 0). \end{cases} \tag{5.49}$$

This can be realized by adding an edge with capacity λ from $v_{k,i}$ to $v_{l,i}$ and an edge with the same capacity in the reverse direction. Finally note that all nodes $v_{k,T+1}$ are constrained to be equal to 0, they can thus be merged into the sink t, which makes they constraint edges from $v_{k,T+1}$ to $v_{k,T}$ redundant. They can thus be removed. The resulting graph is exactly equal to Ishikawa's graph, who more directly arrived at this graph, without explicitly definition of the auxiliary *MRF* (5.48).

Remark 3. It is possible to arrive at an alternative graph representation by exploiting the monotonicity property. The data edges in fact can be represented as unary potentials:

$$\begin{aligned}\sum_{i\in L} g_k(i)|v_{k,i+1}-v_{k,i}| &= \sum_{i\in L} g_k(i)(v_{k,i}-v_{k,i+1})\\ &= \sum_{i\in L} v_{k,i}(g_k(i)-g_k(i-1)),\end{aligned} \tag{5.50}$$

with $g_k(-1)=0$. Thus these terms can be expressed as unary terms, which means that instead of the data edges, they can also be modeled as edges from node s to $v_{k,i}$ with capacity $g_k(i)-g_k(i-1)$.

5.4.2 Solving the Non-Convex Part

It turns out that the idea of lifting the functional to a higher-dimensional space in order to arrive at a tight convex relaxation of the original problem can be extended to a larger class of problems [121]. The key insight is that

$$\int_{\Omega\times T} |Dv(\mathbf{x},t)| + \rho(\mathbf{x},t)|D_t v(\mathbf{x},t)| \tag{5.51}$$

represents an interfacial energy of the boundary Γ_u of the function u. The framework [121] shows that in order to find a minimizer u^* of functionals of the form

$$\min_u \int_\Omega f(\mathbf{x},u(\mathbf{x}),Du), \tag{5.52}$$

where $f(x,u(x),Du)$ is assumed to be convex in the last argument, we can solve the auxiliary problem

$$\min_{v\in\mathcal{C}} \sup_{\phi\in\mathcal{K}} \int_{\Omega\times\Re} \phi\cdot Dv, \tag{5.53}$$

where the convex sets $\mathcal{C}$ and $\mathcal{K}$ are given by

$$\mathcal{C}=\left\{v\in BV(\Omega\times\Re;[0,1]) : \lim_{t\to-\infty} v(\mathbf{x},t)=1, \quad \lim_{t\to\infty} v(\mathbf{x},t)=0\right\}$$

and

$$\begin{aligned}\mathcal{K}=\{\phi=(\phi_x,\phi_t)\in C_0(\Omega\times\Re;\Re^d\times\Re) :&\\ \phi_t(\mathbf{x},t)\geq f^*(\mathbf{x},t,\phi_x(\mathbf{x},t)), \forall\mathbf{x},t\in\Omega\times\Re\}&.\end{aligned} \tag{5.54}$$

Here f^* denotes the convex conjugate of the function f with respect to its last argument. The sets $\mathcal{C}$ and $\mathcal{K}$ are defined point-wise. The intuition behind this formulation again is that instead of minimizing u directly, one represents the energy in terms of characteristic functions of the upper level sets $\mathbf{1}_{\{u>t\}}(\mathbf{x})$. Again the function $v : \Omega \times \Re$, represents this characteristic function which is indicated by the constraint set $\mathcal{C}$. The duals ϕ represent a flux through the boundary Γ_u of u. Given a minimizer v^* the corresponding minimizer u^* can as before be recovered by integrating v over the lifted dimension: $u^*(\mathbf{x}) = \int_\Re v^*(\mathbf{x}, t)dt$. This formulation is very general, the specific form of the convex regularization term only influences the set $\mathcal{K}$. Pock et al. [121] derived the set $\mathcal{K}$ for Quadratic, TV, Huber and Lipschitz regularization terms.

Let us establish the connection of this formulation to the first-order model (5.51). Using duality (5.51) can be equivalently written as

$$\sup_{\substack{|\phi_x(\mathbf{x},t)|_2 \leq 1 \\ |\phi_t(\mathbf{x},t)| \leq \rho(\mathbf{x},t)}} \int_{\Omega \times T} \langle D_x v(\mathbf{x}, t), \phi_x(\mathbf{x}, t)\rangle + \langle D_t v(\mathbf{x}, t), \phi_t(\mathbf{x}, t)\rangle \tag{5.55}$$

Since $v(\mathbf{x}, t)$ needs to be monotone in direction t we have $D_t v(\mathbf{x}, t) \leq 0$. As a consequence the supremum with respect to $\phi_t(\mathbf{x}, t)$ will be reached if $\phi_t(\mathbf{x}, t)$ is non-positive. We can thus give an equivalent formulation as

$$\sup_{\substack{|\phi_x(\mathbf{x},t)|_2 \leq 1 \\ \phi_t(\mathbf{x},t) \geq -\rho(\mathbf{x},t)}} \int_{\Omega \times T} \langle Dv(\mathbf{x}, t), \phi(\mathbf{x}, t)\rangle \, , \tag{5.56}$$

where $\phi(\mathbf{x}, t) = (\phi_x(\mathbf{x}, t), \phi_t(\mathbf{x}, t))^T$ and $Dv(\mathbf{x}, t) = (D_x v(\mathbf{x}, t), D_t v(\mathbf{x}, t))^T$. The objective of this problem is equal to the objective function of the general problem (5.53). Now consider the general formulation (5.53). We need to specify the convex set $\mathcal{K}$, which in turn is specified via the convex conjugate of the function

$$f(\mathbf{x}, t, p) = |p(\mathbf{x})|_2 + \rho(\mathbf{x}, t). \tag{5.57}$$

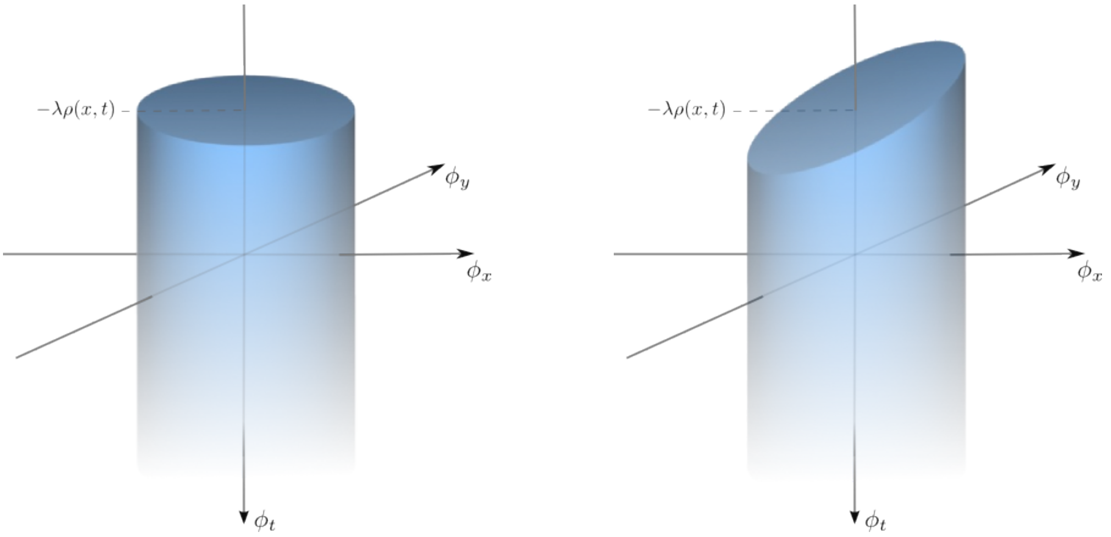
Note that this function is related to the original, non-convex problem (5.36). The convex conjugate is given by

$$\begin{aligned} f^*(\mathbf{x}, t, \phi_x) &= -\rho(\mathbf{x}, t) + \sup\{\langle p(\mathbf{x}), \phi_x(\mathbf{x}, t)\rangle - |p(\mathbf{x})|_2\} \\ &= \begin{cases} -\rho(\mathbf{x}, t) & \text{if} \quad |\phi_x(\mathbf{x}, t)|_2 \leq 1 \\ \infty & \text{else} \end{cases} \end{aligned} \tag{5.58}$$

It follows that the feasible set of the dual variables can be written as

$$\begin{aligned} \mathcal{K} = \{\phi = (\phi_x, \phi_t) \in C_0(\Omega \times \Re; \Re^d \times \Re) : \\ \phi_t(\mathbf{x}, t) \geq -\rho(\mathbf{x}, t), |\phi_x(\mathbf{x}, t)|_2 \leq 1, \forall \mathbf{x}, t \in \Omega \times \Re\} \, , \end{aligned} \tag{5.59}$$

which is equivalent to the feasible set of the duals in (5.56).



(a) *TV* **(b)** *TGV*

Figure 5.9: The feasible set $\mathcal{K}$ for (a) TV and (b) TGV.

In problem $E_2(u|w)$, the regularization term is similar to TV regularization, with the difference that a constant vector is subtracted from the gradient, before the absolute value is measured. We identify $f(\mathbf{x},t,p) = |p(\mathbf{x}) - w^n(\mathbf{x})|_\Sigma + \lambda\rho(\mathbf{x},t)$. The convex conjugate with respect to p is

$$\begin{aligned} f^*(\mathbf{x},t,\phi_x) &= \sup_{p(x)} \Big\{ \langle \phi_x(\mathbf{x},t), p(\mathbf{x})\rangle - |p(\mathbf{x}) - w^n(\mathbf{x})|_\Sigma \Big\} - \lambda\rho(\mathbf{x},t) \\ &= \sup_{q} \Big\{ \langle \phi_x(\mathbf{x},t), q\rangle - |q|_\Sigma \Big\} + \langle \phi_x(\mathbf{x},t), w^n(\mathbf{x})\rangle - \lambda\rho(\mathbf{x},t) \\ &= \begin{cases} \langle \phi_x(\mathbf{x},t), w^n(\mathbf{x})\rangle - \lambda\rho(\mathbf{x},t), & \text{if} \quad |\phi_x(\mathbf{x},t)|_{\Sigma^{-1}} \le 1 \\ \infty, & \text{else.} \end{cases} \end{aligned} \tag{5.60}$$

The resulting set $\mathcal{K}$ is illustrated in Figure 5.9(b). The feasible set for TV regularization is shown in Figure 5.9(a). It can be seen that for problem $E_2(u|w)$ the feasible set is slightly more complicated than in the TV case. While for TV the set is given by the interior of a cylinder with radius 1, which is bounded from below by a vertical plane centered at $(0,0,-\lambda\rho(\mathbf{x},t))^T$, the set in the TGV case is bounded from below by plane that includes the point $(0,0,-\lambda\rho(\mathbf{x},t))^T$ but can be arbitrarily oriented (in fact the normal of this plane is given by $(w^n,-1)^T$). This makes projection onto this set slightly harder, as a closed-form solution is no longer available.

Discretization and Optimization. In order to solve (5.53) it is necessary to discretize the domain $\Omega \times \Re$ of the continuous functions v and ϕ. For the sake of simplicity let us only consider the case $\Omega \subset \Re \times \Re$, higher-dimensional cases can be derived analogously.

We discretize on a three-dimensional grid of size $N_x \times N_y \times N_t$ with discretization steps Δx, Δy and Δt:

$$G^\Delta = \{(i\Delta x, j\Delta y, k\Delta t) : (0,0,0) \leq (i,j,k) < (N_x, N_y, N_t)\}\,. \tag{5.61}$$

Here the triple (i,j,k) denotes the location in the grid.

For numerical reasons we replace the vector field ϕ_x, with a rotated version $\Sigma^{\frac{1}{2}}\phi_x$, which leads to a simplification of the convex set $\mathcal{K}^\Delta$, without changing the formulation.

The feasible sets for the discrete version of (5.53) are then given by

$$\mathcal{C}^\Delta = \left\{v^\Delta \in [0,1]^{N_x N_y N_t} : v^\Delta_{i,j,0} = 1, v^\Delta_{i,j,N_t-1} = 0\right\} \tag{5.62}$$

and

$$\begin{aligned}\mathcal{K}^\Delta = \Big\{\phi^\Delta = (\phi^\Delta_x, \phi^\Delta_y, \phi^\Delta_t) \in \Re^{3N_x N_y N_t} :&\\ (\phi^\Delta_t)_{i,j,k} + \lambda(\rho)_{i,j,k} \geq \left\langle (\phi^\Delta_x, \phi^\Delta_y)^T_{i,j,k}, \Sigma^{\frac{1}{2}} w_{i,j}\right\rangle\,,&\\ |(\phi^\Delta_x, \phi^\Delta_y)^T_{i,j,k}|_2 \leq 1, \quad \forall (i,j,k) \in G^\Delta\Big\}\,.&\end{aligned} \tag{5.63}$$

In order to discretize the differential operator D, we use forward differences with Neumann boundary conditions. Furthermore we allow Σ to vary locally, which allows us to incorporate image-driven TGV regularization similar to [124] into the framework, *i.e.* we define a linear operator $\nabla_\Sigma : \Re^{N_x N_y N_t} \to \Re^{3N_x N_y N_t}$, with

$$(\nabla_\Sigma v^\Delta)_{i,j,k} = \begin{pmatrix} \Sigma^{\frac{1}{2}}_{i,j} & & 0 \\ & & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} (\delta_x v^\Delta)_{i,j,k} \\ (\delta_y v^\Delta)_{i,j,k} \\ (\delta_t v^\Delta)_{i,j,k} \end{pmatrix} \tag{5.64}$$

and

$$(\delta_x v^\Delta)_{i,j,k} = \begin{cases} (v^\Delta_{i+1,j,k} - v^\Delta_{i,j,k})/\Delta x & \text{if} \quad i < N_x - 1 \\ 0 & \text{else} \end{cases} \tag{5.65}$$

$$(\delta_y v^\Delta)_{i,j,k} = \begin{cases} (v^\Delta_{i,j+1,k} - v^\Delta_{i,j,k})/\Delta y & \text{if} \quad i < N_y - 1 \\ 0 & \text{else} \end{cases} \tag{5.66}$$

$$(\delta_t v^\Delta)_{i,j,k} = \begin{cases} (v^\Delta_{i,j,k+1} - v^\Delta_{i,j,k})/\Delta t & \text{if} \quad i < N_t - 1 \\ 0 & \text{else.} \end{cases} \tag{5.67}$$

Operator (5.64) reduces to the standard discretization of a gradient operator, if $\Sigma^{\frac{1}{2}}_{i,j}$ is set to identity everywhere. On the other hand, it is possible to incorporate image-driven diffusion into the model by setting the matrix appropriately. We will later again use the

Nagel-Enkelmann operator 3.42 in our experiments.

The discrete version of (5.53) is now given by

$$\min_{v^\Delta \in \mathcal{C}^\Delta} \max_{\phi^\Delta \in \mathcal{K}^\Delta} \left\langle \nabla_\Sigma v^\Delta, \phi^\Delta \right\rangle \tag{5.68}$$

For optimization of the convex-concave saddle-point problem (5.68) we use the primal-dual algorithm [36]. The iterations of this algorithm are shown in Algorithm 5.1.

A crucial part of this algorithm are the pointwise projections $\mathbf{proj}_{\mathcal{K}^\Delta}(.)$ and $\mathbf{proj}_{\mathcal{C}^\Delta}(.)$ respectively. The projection of the primal variables is simple and can be carried out in closed-form:

$$(\mathbf{proj}_{\mathcal{C}^\Delta}(\hat{v}))_{i,j,k} = \begin{cases} \max\{0, \min\{1, \hat{v}_{i,j,k}\}\} & \text{if } k > 1 \\ 1 & \text{else.} \end{cases}$$

The projections for the dual variables $\mathbf{proj}_{\mathcal{K}^\Delta}(.)$, although also point-wise, are more complicated. The feasible set $\mathcal{K}^\Delta$ is defined point-wise via the intersection of two convex sets. We experimented with different variants to incorporate these constraints: Lagrange multipliers, solving the projection problem in each iteration of the primal-dual algorithm using FISTA [12] (including a preconditioned variant) and finally Dykstra‘s Projection algorithm [23]. Our experiments show that Dykstra‘s algorithm provides the best performance for this type of problem and is very light-weight, we therefore resort to this variant to incorporate the dual constraints. The iterations of Dykstra‘s algorithm are shown in Algorithm 2, where we set $\hat{n} = (w^n_{i,j,k}, -1)^T$ and $c = \lambda \rho_{i,j,k}$. In practice we run the algorithm until the distances to both convex sets ($|x_n - y_n|_2$ and $|y_n - x_{n+1}|_2$) are below a tolerance of 10^{-3} (which is typically achieved in under 10 iterations).

5.4.3 Solving the Convex Part

The subproblem $E_1(w|u)$ is a non-smooth convex optimization problem, which can be solved using standard techniques. We will show how to cast this problem in a saddle-point

```
1.  Initialize
    Set $v_0^\Delta \in \mathcal{C}^\Delta$, $\phi_0^\Delta \in \mathcal{K}^\Delta$, $\bar{v}_0 = v_0^\Delta$, $l = 0$
    Choose time-steps $\tau, \sigma > 0$, $\tau\sigma < \frac{1}{||\nabla_\Sigma||^2}$
2.  While not converged
    $\phi_{l+1}^\Delta = \mathbf{proj}_{\mathcal{K}^\Delta}(\phi_l^\Delta + \sigma \nabla_\Sigma \bar{v}_l)$
    $v_{l+1}^\Delta = \mathbf{proj}_{\mathcal{C}^\Delta}(v_l^\Delta - \tau \nabla_\Sigma^T \phi_{l+1}^\Delta)$
    $\bar{v}_{l+1} = 2v_{l+1}^\Delta - v_l^\Delta$
    $l = l + 1$
```

Algorithm 5.1: Primal-Dual algorithm for optimization of (5.68).

```
1. Initialize
   Set $l = 0$, $x_0 = \phi_{i,j,k}$, $p_0 = 0$, $q_0 = 0$
2. Iterate
   $y_l = \frac{(x_l + p_l)}{\max\{1, |x_l + p_l|_2\}}$
   $p_{l+1} = p_l + x_l - y_l$
   $x_{l+1} = \begin{cases} y_l + q_l & \text{if} \quad \langle y_l + q_l, \hat{n} \rangle \leq c \\ y_l + q_l - \frac{\langle y_l + q_l, \hat{n} \rangle - c}{\langle \hat{n}, \hat{n} \rangle} \hat{n} & \text{else} \end{cases}$
   $q_{l+1} = q_l + y_l - x_{l+1}$
   $l = l + 1$
```

Algorithm 5.2: Algorithm for projecting onto the set $\mathcal{K}$.

formulation and again apply the primal-dual algorithm [36].

The optimization problem reads

$$\min_{w} \int_{\Omega} |Du^{n+1} - w|_{\Sigma} + \alpha \int_{\Omega} |Dw|_{\Gamma}, \tag{5.69}$$

where u^{n+1} is given by the last solution of problem $E_2(u|w)$. This problem corresponds to denoising the gradients of u^{n+1}.

Using the definition $\operatorname{div}_M z = \operatorname{div}(M^{\frac{1}{2}} z)$, the equivalent saddle-point formulation is given by:

$$\min_{w} \sup_{\substack{||p||_{\infty} \leq 1 \\ ||q||_{\infty} \leq 1}} - \int_{\Omega} u^{n+1} \operatorname{div}_{\Sigma} p \, \mathrm{d}\mathbf{x} - \int_{\Omega} \left\langle w, \Sigma^{\frac{1}{2}} p + \alpha \operatorname{div}_{\Gamma} q \right\rangle \mathrm{d}\mathbf{x}. \tag{5.70}$$

Discretization of (5.70) follows analogously to the lifted problem: The two-dimensional grid is given by

$$\hat{G}^{\Delta} = \{(i\Delta x, j\Delta y) : (0,0) \leq (i,j) < (N_x, N_y)\}, \tag{5.71}$$

where the tuple (i,j) again denotes a location in the grid, which also coincides with the spatial coordinates of the lifted problem. The discrete saddle-point problem can be written as

$$\min_{w^{\Delta}} \max_{\substack{||p^{\Delta}||_{\infty} \leq 1 \\ ||q^{\Delta}||_{\infty} \leq 1}} \left\langle \nabla_{\Sigma} u^{n+1}, p^{\Delta} \right\rangle - \left\langle \Sigma^{\frac{1}{2}} w^{\Delta}, p^{\Delta} \right\rangle + \alpha \left\langle \mathcal{D}_{\Gamma} w^{\Delta}, q^{\Delta} \right\rangle, \tag{5.72}$$

where the discrete differential operators ∇_{Σ} and $\mathcal{D}_{\Gamma}$ are again based on forward differences with Neumann boundary conditions, *i.e.* we have

$$(\nabla_{\Sigma} u^{\Delta})_{i,j} = \Sigma^{\frac{1}{2}}_{i,j} \begin{pmatrix} (\delta_x u^{\Delta})_{i,j} \\ (\delta_y u^{\Delta})_{i,j} \end{pmatrix} \qquad (\mathcal{D}_{\Gamma} w^{\Delta})_{i,j} = \Gamma^{\frac{1}{2}}_{i,j} \begin{pmatrix} (\delta_x w^{\Delta}_1)_{i,j} & (\delta_y w^{\Delta}_2)_{i,j} \\ (\delta_y w^{\Delta}_1)_{i,j} & (\delta_x w^{\Delta}_2)_{i,j} \end{pmatrix}.$$

In practice direct usage of (5.69) for the estimation of the second-order part w may be problematic if the discretization step Δt for the solution of the lifted problem was chosen too coarsely. In this case discretization artifacts are propagated from the lifted problem to problem (5.69), which may deteriorate the estimation of the second-order part, since in the context of this subproblem such artifacts are merely additional edges.

To cope with this problem, we modify (5.72) to allow u^{n+1} to slightly vary in a neighborhood of half the discretization step Δt of the lifted problem:

$$\min_{w^\Delta, u^\Delta \in \mathcal{B}} \max_{\substack{||p^\Delta||_\infty \le 1 \\ ||q^\Delta||_\infty \le 1}} \left\langle \nabla_\Sigma u^\Delta - \Sigma^{\frac{1}{2}} w^\Delta, p^\Delta \right\rangle + \alpha \left\langle \mathcal{D}_\Gamma w^\Delta, q^\Delta \right\rangle , \tag{5.73}$$

where $\mathcal{B} = \left\{ u^\Delta \in \Re^{N_x N_y} : |(u^\Delta)_{i,j} - (u^{n+1})_{i,j}| \le \Delta t/2 \right\}$.

```
1. Initialize
   Set $u_0^\Delta = u^{n+1}, w_0^\Delta = \nabla_\Sigma u^{n+1}, \bar{u}_0 = u_0^\Delta, \bar{w}_0 = w_0^\Delta$
   Set $(p_0^\Delta)_{i,j} = (0,0)^T$, $(q_0^\Delta)_{i,j} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, $l = 0$
   Choose time-steps $\tau, \sigma > 0,\ \tau\sigma < \frac{1}{||A||^2}$, where $A = \begin{pmatrix} \nabla_\Sigma & -I \\ 0 & \mathcal{D}_\Gamma \end{pmatrix}$
2. Iterate
   $p_{l+1}^\Delta = \mathbf{proj}_{||p||_\infty \le 1}(p_l^\Delta + \sigma(\nabla_\Sigma \bar{u}_l - \Sigma^{\frac{1}{2}} \bar{w}_l))$
   $q_{l+1}^\Delta = \mathbf{proj}_{||q||_\infty \le 1}(q_l^\Delta + \sigma(\mathcal{D}_\Gamma \bar{w}_l))$
   $u_{l+1}^\Delta = \mathbf{proj}_{\mathcal{B}}(u_l^\Delta - \tau \nabla_\Sigma^T p_{l+1}^\Delta)$
   $w_{l+1}^\Delta = w_l^\Delta - \tau(\mathcal{D}_\Gamma^T q_{l+1}^\Delta - \Sigma^{\frac{1}{2}} p_{l+1}^\Delta)$
   $\bar{u}_{l+1} = 2u_{l+1}^\Delta - u_l^\Delta$
   $\bar{w}_{l+1} = 2w_{l+1}^\Delta - w_l^\Delta$
   $l = l + 1$
```

Algorithm 5.3: Primal-Dual algorithm for optimization of (5.73).

The iterations for optimizing (5.73) are shown in Algorithm 3. As before, we again have to perform projections onto convex sets in each iteration of the algorithm. The projections of the dual variables are given by $(\mathbf{proj}_{||r||_\infty \le 1}(r))_{i,j} = \frac{r_{i,j}}{\max\{1, |r_{i,j}|_2\}}$. For the primal variables u, the projection onto $\mathcal{B}$ can be computed by clamping $(u)_{i,j}$ to the interval

$$\left[(u^{n+1})_{i,j} - \frac{\Delta t}{2}, (u^{n+1})_{i,j} + \frac{\Delta t}{2} \right].$$

5.5 Solution via Coarse-To-Fine Warping

Another approach for optimizing variational models for stereo estimation is the solution via an optical flow-like coarse-to-fine warping approach, *i.e.* to treat the stereo model as a one-dimensional optical flow model. All models, which where described in the previous chapter can be applied, the only difference is that there is no vertical flow component. As an example we show a simple TGV-based model. Models with Image-Driven Total Generalized Variation (ITGV) or Non-Local Total Generalized Variation (NLTGV) regularization can be handled analogously:

$$\min_{u} TGV_{\alpha}^{2}(u) + \lambda \int_{\Omega} \hat{\rho}(\mathbf{x}, (u(\mathbf{x}), 0)^T, I_1, I_2)\,\mathrm{d}\mathbf{x}, \tag{5.74}$$

where the disparity is given by $u : \Omega \to \Re$ and the data term is given by the convex quadratic approximation 4.30. After standard discretization, the discrete model is given by

$$\min_{u,w} \alpha_1 \,||\nabla u - w||_{2,1} + \alpha_0 \,||\boldsymbol{\nabla} w||_{2,1} + \lambda \sum_{i,j} \hat{\rho}((i,j)^T, ((u)_{i,j}, 0)^T, I_1, I_2) \tag{5.75}$$

To optimize this model, a simplified version of Algorithm 5.3 can be used. The only difference is that variables corresponding to vertical flow component are always zero and thus can be neglected. The complete algorithm is specified in Algorithm 5.4. Moreover, the proximal mapping can be simplified since it is effectively a pointwise quadratic program in only one dimension, instead of the two-dimensional problem in the optical flow case. The proximal mapping is given by

$$u = \mathbf{prox}_{\tau g}(\hat{u}) \Leftrightarrow u_{i,j} = \arg\min_{u} \frac{1}{2}(u - \hat{u}_{i,j})^2 + \lambda\tau\hat{\rho}((i,j)^T, u, I_1, I_2), \tag{5.76}$$

where the quadratic model of the data term is given by

$$\begin{aligned} \hat{\rho}(\mathbf{x}, u(\mathbf{x})) \approx \rho(\mathbf{x}, u_0) &+ (u(\mathbf{x}) - u_0(\mathbf{x}))\nabla_x \rho(\mathbf{x}, u_0(\mathbf{x})) \\ &+ \frac{1}{2}(u(\mathbf{x}) - u_0(\mathbf{x}))^2 \rho_{xx}(\mathbf{x}, u_0(\mathbf{x})), \end{aligned} \tag{5.77}$$

and ρ_{xx} are second-derivatives in x-direction. The explicit solution of the proximal mapping is given by

$$u_{i,j} = \frac{\lambda\tau(\rho_{xx} \cdot (u_0)_{i,j} - \nabla_x \rho) + \hat{u}_{i,j}}{1 + \lambda\tau\rho_{xx}}, \tag{5.78}$$

were we dropped the explicit arguments for notational simplicity and ρ_{xx} and $\nabla_x \rho$ are evaluated at pixel (i, j).

Approaches based on coarse-to-fine warping suffer from the same drawbacks as in the optical flow task: Fine structures will not be well preserved and large baseline setups, which result in large disparity spaces, may not be handled well. On the positive side these approaches are fast and the complexity does not rise with the number of disparities.

```
1.  Set $u = \bar{u} \in \Re^M$, $w \in \Re^{2M}$
2.  Set $p \in P$, $q \in Q$, set $k = 0$
3.  While not converged
    $p_{k+1} = \mathbf{proj}_{p \in P}(p_k + \sigma(\nabla \bar{u}_k - \bar{w}_k))$
    $q_{k+1} = \mathbf{proj}_{q \in Q}(q_k + \sigma \boldsymbol{\nabla} \bar{w}_k)$
    $u_{k+1} = \mathbf{prox}_{\tau g}(u_k - \tau \nabla^* p_{k+1})$
    $w_{k+1} = w_k - \tau(\boldsymbol{\nabla}^* q_{k+1} - p_{k+1})$
    $\bar{u}_{k+1} = 2u_{k+1} - u_k$
    $\bar{w}_{k+1} = 2w_{k+1} - w_k$
    $k = k + 1$
```

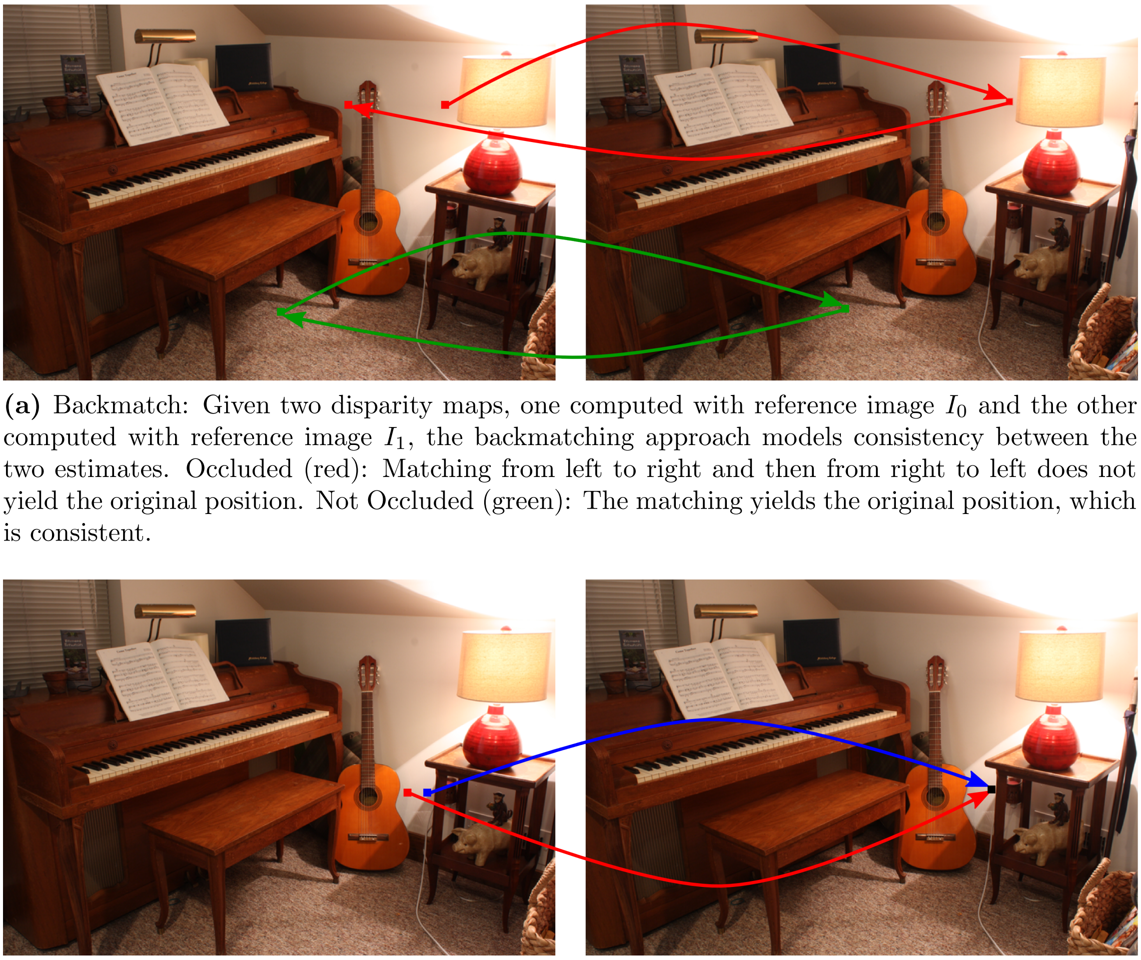
Algorithm 5.4: Primal-Dual algorithm for optimization of (5.75).

5.6 Post-Processing

Stereo matching pipelines typically include some inexpensive post-processing steps such as depth map filtering, occlusion estimation, rejection of low confidence matches and inpainting of missing values.

Occlusion Estimation Explicit handling of occluded areas that inevitable arise in stereo estimation is currently not supported in the proposed model. The output of the model is a dense disparity map, were a disparity value is assigned to each pixel. Occluded areas typically result in a smearing of the estimated depth maps along the edges of an object. For the estimation of occluded areas one can resort to one of two simple post-processing approaches: The first approach is known as the Mapping Uniqquness Criterion (MUC) [46]. In this approach candidate occlusions are found by searching for pixels that warp to the same pixel position. This non-unique mapping clearly violates the physical constraint that every 3D world point can only be projected to a single 2D point in the image plane by a projective transformation. To resolve the ambiguity that arises from such a non-unique mapping, one labels the pixel with the smallest depth value as occluding whereas the remaining pixels are labeled as occluded.

Another popular approach for occlusion handling is backmatching [57] (also known as left-right consistency checking). This approach requires the computation of two disparity maps: The initial computation of the disparity map is followed by a second disparity estimation, where the roles of the warped and the reference image are interchanged. Finally



(a) Backmatch: Given two disparity maps, one computed with reference image I_0 and the other computed with reference image I_1, the backmatching approach models consistency between the two estimates. Occluded (red): Matching from left to right and then from right to left does not yield the original position. Not Occluded (green): The matching yields the original position, which is consistent.

(b) Mapping Uniqueness Criterion: Given one disparity map, this measure models physical plausibility in the sense that an object can only occupy a single point in space. The example shows the configuration of a occlusion: The disparities of two-pixels (red and blue) from the left image map to the same position in the right image.

Figure 5.10: Illustration of occlusion estimation, both examples show disparities which would be classified as occlusion.

both maps are compared for consistency, i.e. pixels with a disparity difference over a (user-defined) threshold are marked as occluded. This approach typically yields better and less cluttered results than the *MUC* criterion. The computational cost of the second disparity estimation may however be simply too large for some applications. Direct incorporation of occlusions into the stereo estimation procedure would be clearly desirable, but the resulting energies typically become much more complicated to optimize [85, 148]. Occlusions are a minor problem in the long-range, small-baseline stereo estimation task, however.

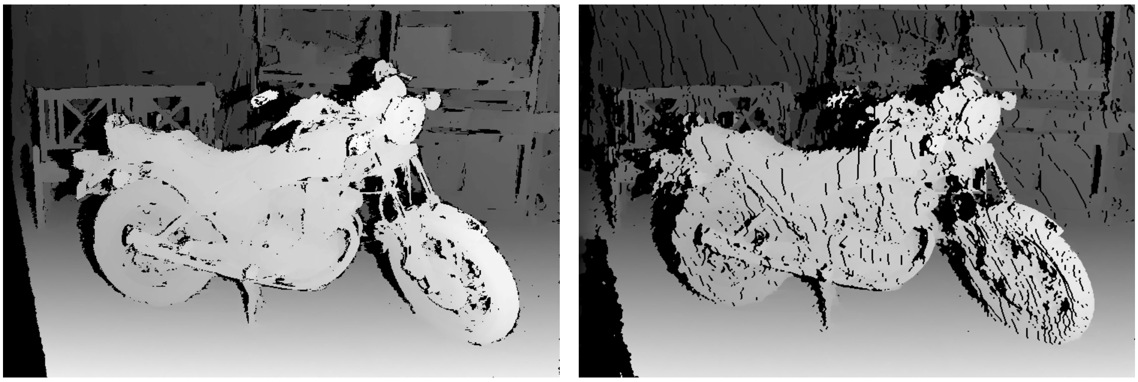
The differences between Backmatching and *MUC* can be seen in Figure 5.11. Backmatching produces occlusion maps of higher quality, at the cost of a significant higher computational effort. The occlusions maps that are generated using *MUC* show artifacts on slanted surfaces and are generally more cluttered.

Inpainting The occlusion estimation procedure provides a binary map of “invalid” pixels, which can be understood as occluded. This includes occlusions but often also image regions with gross errors which are due to untextured regions. In many practical applications a disparity map with hundred percent density is desirable. Thus invalid pixels need to be inpainted. A simple approach, which also exploits the specific geometry of the stereo setting, is to find the next valid pixels along the scanline (to the left and right) and assign the smaller disparity to the invalid pixel. This approach reflects the specific geometric relationship of occluder and oludee, e.g. areas that are occluded necessarily are part of the background. An example of this post-processing is shown in Figure 5.12. This approach is fast and simplistic, but may result in some artifacts. More sophisticated algorithms for occlusion inpainting have been proposed in the literature [161], but are beyond the scope of this work.

5.7 Experiments

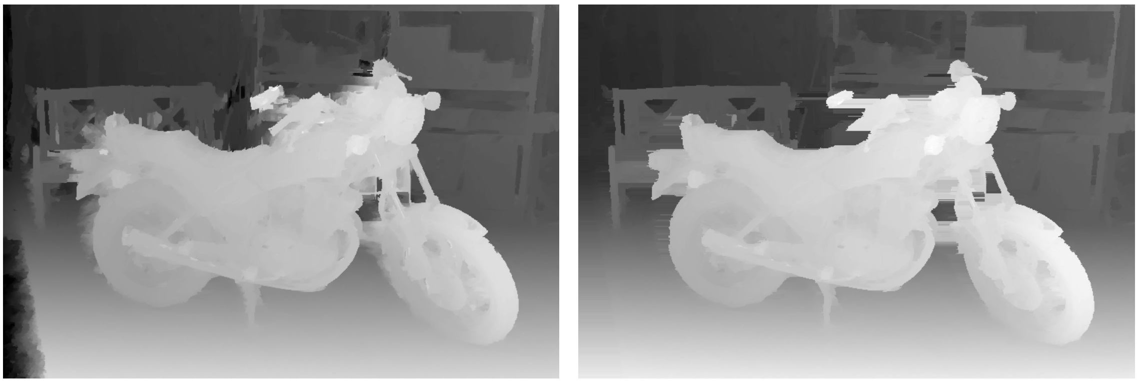
We evaluate the proposed algorithms on two different benchmarks, which have different scopes. Both benchmarks feature realistic scenarios and are well-suited to judge the practical applicability of the proposed approaches. We compare the alternating minimization approach to the coarse-to-fine warping approach with different TGV-based regularizers. We quantify the error using the average absolute distance from the groundtruth disparity and as the percentage of pixels with absolute error above a certain threshold.

We refer to the lifting-base higher-order model as $ATGV$. For all experiments the discretization of the disparity range was fixed to $\Delta t = 1px$ and Algorithm (A1) was run for 5 iterations. The optimization of each subproblem was run for 2000 iterations in the



(a) Backmatching **(b)** Mapping Uniqueness Criterion

Figure 5.11: Example of Backmatching and Mapping Uniqueness Criterion. Detected occlusions are shown in black.



(a) Before **(b)** After

Figure 5.12: Example of occlusion inpainting. The results are based on the *ATGV* algorithm, occlusions were computed using backmatching. Inpainting leads to sharper discontinuities, but due to the simple scanline approach streaking artifacts emerge in some areas.

first run and subsequently reduced by a factor of $\frac{1}{i+1}$, where i refers to the number of outer iterations. In order to allow for a fair comparison no post-processing was used, unless otherwise stated.

5.7.1 Middlebury v3 Benchmark

The Middlebury Benchmarks [79, 131–134] are a series of stereo and groundtruth images featuring semi-realistic data, which defined the standard stereo benchmarks in the last few years. Its newest incarnation, the Middlebury v3 Benchmark [131], features 30 stereo pairs. For 15 images groundtruth is provided, whereas for the remaining 15 images groundtruth is withheld and only used for an online evaluation. A unique feature of this benchmark is that it includes images of very high resolution (up to 6 megapixels) with disparity ranges of up to 800 pixels. Groundtruth was computed using a high-accuracy structured light approach. All experiments where performed on the quarter size images.

The first quantitative evaluation in Table 5.1 shows a comparison of different data terms when used in conjunction with the lifting-based *ATGV* model. For the patch-

	0.5px	1px	2px	4px	Avg-Err
SAG	24.51	14.60	9.91	6.70	1.30
MI	47.74	29.47	20.8	13.76	3.01
NCC	26.68	17.16	12.16	8.33	1.59
RT	24.72	15.55	11.01	7.63	1.38
CRT	24.69	15.12	10.53	7.10	1.34
CENSUS	**24.02**	**14.31**	**9.39**	**6.10**	**1.24**

Table 5.1: Evaluation of *ATGV* with different data terms.

based methods normalized-cross correlation (NCC), rank transform (RT), complete rank transform (CRT) and Census transform (CENSUS) we used a window size of 5×5. For the sum-of-absolute gradient differences (SAG) term we used central differences together with an aggregation window of 3×3, which again results in an effective support window size of 5×5. The mutual information term (MI) was not aggregated, making it weaker than the other terms. For the Census term we used the ternary Census transform with $\epsilon = 0.01$. This table shows some interesting results. On this data set the Census data term performs best, but the simple SAG term already performs second-best, much better than NCC and slightly better than CRT.

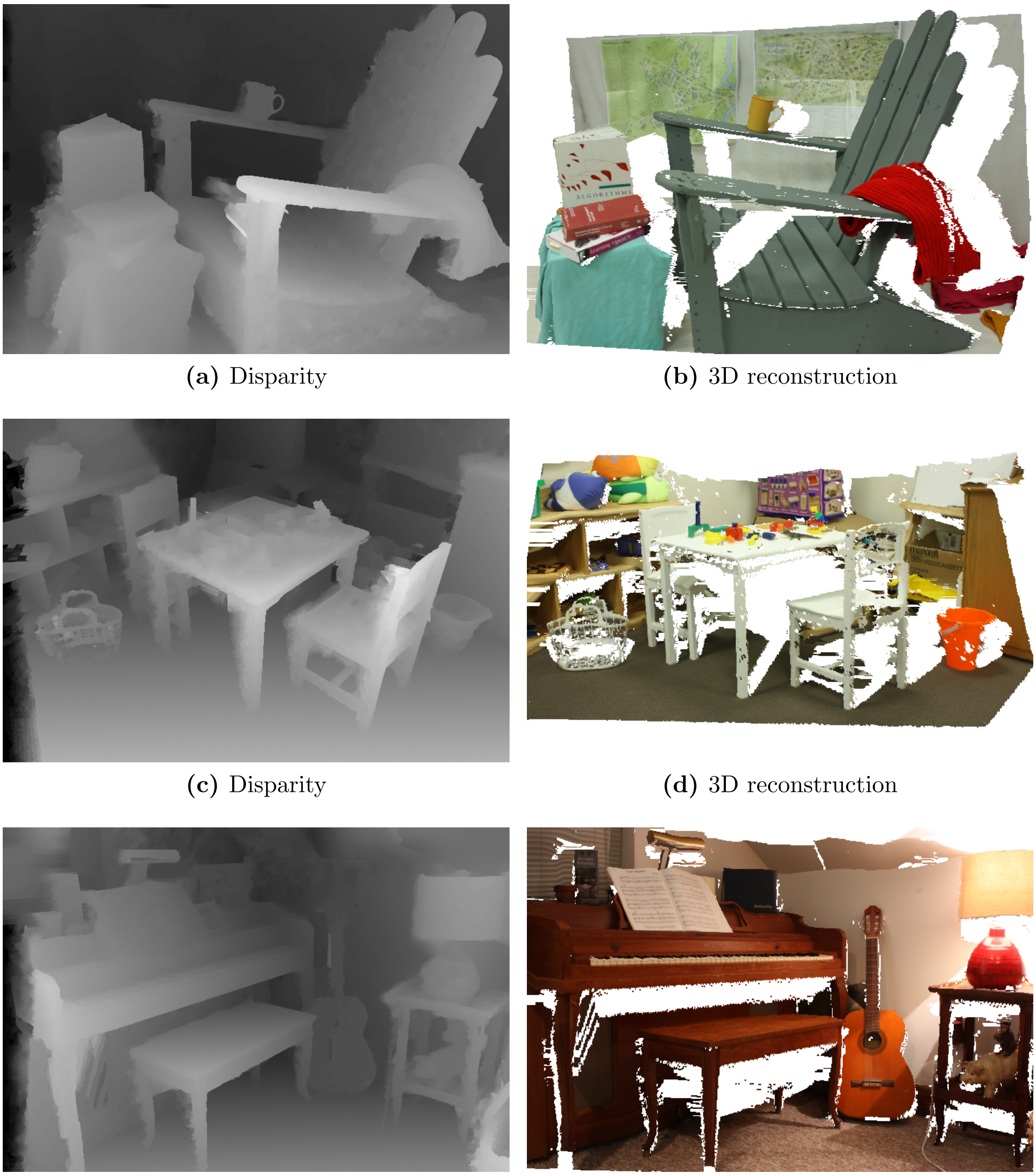
Table 5.2 shows a quantitative comparison between $ITGV$, $NLTGV$, both based on coarse-to-fine warping, with the lifting-based $ATGV$. For all evaluations the ternary Census data term of the previous evaluation was used. The settings for the coarse-to-fine warping are identical to the optical flow case. The remaining parameters were tuned for optimal performance using grid search. We report the percentage of pixels with absolute error below a threshold of 1 pixel as well as the average absolute error. The quantitative results show that the specialized $ATGV$ model clearly outperforms the other models on the stereo task. The difference is most pronounced for images, which show small structures, which tend to be missed by the coarse-to-fine approaches. The lifting approach on the other hand is better able to handle such cases, which results in a significantly smaller error. $NLTGV$ shows a similar improvement over $ITGV$ as for the optical flow task. In summary $NLTGV$ is more robust, but shows a similar accuracy as $ITGV$ on average. Figure 5.13 shows disparity maps, computed using $ATGV$, together with a 3D reconstruction. It can be seen that the proposed approach provides qualitatively good reconstructions, where also small structures can be resolved. Finally, Figure 5.14 compares $ATGV$ to failure cases of the coarse-to-fine approach. Small structures and dominant occlusions are the main sources of error. When compared to the state-of-the-art (Table 5.3), the $ATGV$ model compares favorably. The model provides high accuracy, but is slightly less robust than SGM.

	1px			Avg-Err		
	ITGV	NLTGV	ATGV	ITGV	NLTGV	ATGV
Adirondack	12.73^{2}	12.87^{3}	6.29^{1}	2.21^{2}	2.30^{3}	0.44^{1}
ArtL	24.25^{3}	22.63^{2}	12.14^{1}	2.36^{2}	2.42^{3}	1.18^{1}
Jadeplant	48.29^{3}	46.19^{2}	17.90^{1}	22.89^{3}	22.09^{2}	3.54^{1}
Motorcycle	10.24^{3}	9.78^{2}	7.04^{1}	1.31^{3}	1.28^{2}	0.71^{1}
MotorcycleE	9.43^{3}	9.35^{2}	6.43^{1}	1.26^{2}	1.27^{3}	0.68^{1}
Piano	12.90^{2}	12.82^{1}	13.73^{3}	0.64^{1}	0.64^{1}	0.69^{3}
PianoL	37.80^{3}	36.80^{2}	28.99^{1}	11.14^{2}	11.80^{3}	3.69^{1}
Pipes	21.74^{3}	20.66^{2}	10.96^{1}	3.90^{3}	3.75^{2}	1.43^{1}
Playroom	19.65^{3}	19.45^{2}	18.58^{1}	1.87^{3}	1.86^{2}	1.16^{1}
Playtable	13.46^{2}	13.16^{1}	16.67^{3}	1.01^{3}	1.00^{2}	0.98^{1}
PlaytableP	11.50^{3}	11.26^{2}	10.75^{1}	0.87^{2}	0.90^{3}	0.60^{1}
Recycle	10.89^{3}	10.47^{2}	9.50^{1}	0.83^{3}	0.72^{2}	0.63^{1}
Shelves	37.62^{2}	38.45^{3}	35.14^{1}	1.77^{1}	1.81^{2}	1.90^{3}
Teddy	6.39^{3}	4.92^{1}	5.27^{2}	0.42^{2}	0.38^{1}	0.46^{3}
Vintage	22.31^{2}	20.79^{1}	23.04^{3}	1.51^{2}	1.52^{3}	1.27^{1}
Average	$19.95^{2.67}$	$19.31^{1.87}$	$14.83^{1.47}$	$3.60^{2.27}$	$3.58^{2.27}$	$1.29^{1.29}$

Table 5.2: Results on the Middlebury training data set. We show the percentage of pixels with an error larger than 1 pixel (1px) and the average error (Avg-Err). A ranking among the methods is shown as superscript. *ATGV* outperforms the coarse-to-fine based approaches on most images. The difference is largest for images which show many thin structures (e.g. *Pipes*, *Jadeplant*).

	0.5px	1.0px	2.0	4.0px	Avg-Err	RMS
BSM [177]	82.2	61.9	37.1	23.4	13.40	35.8
SGM [77]	64.6	37.3	21.0	12.9	5.29	17.1
LCU	**67.3**	**38.2**	**17.3**	**9.63**	**3.63**	**11.9**
ATGV	68.8	45.3	23.8	13.3	4.85	14.9

Table 5.3: Comparison to the state-of-the-art on Middlebury benchmark. We show the average over different error measures. We compare to methods which have been computed on the quarter size images. Method LCU was unpublished at the time of writing.



(a) Disparity

(b) 3D reconstruction

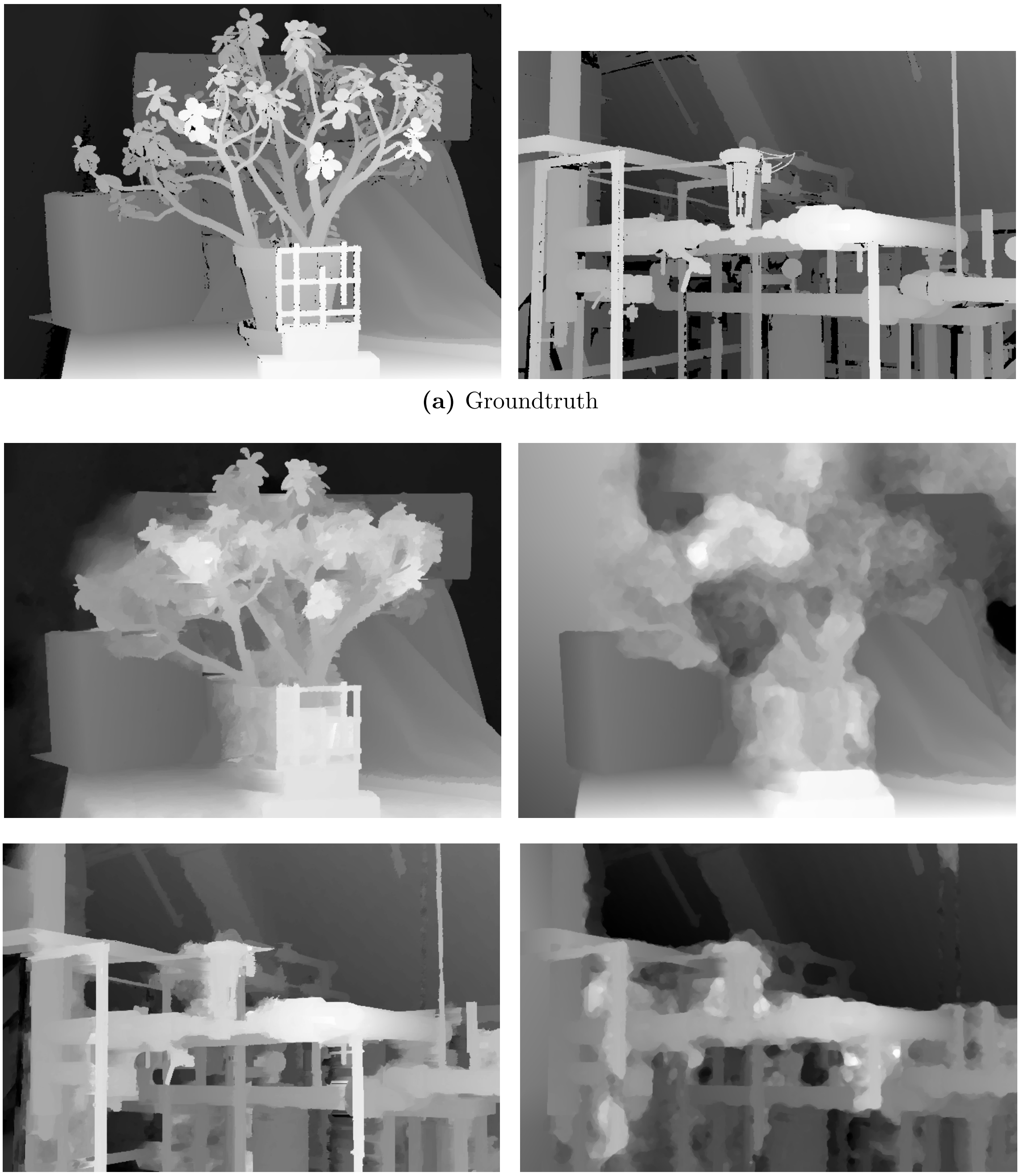
(c) Disparity

(d) 3D reconstruction

(e) Disparity

(f) 3D reconstruction

Figure 5.13: Qualitative results on the Middlebury benchmark. Left column: Disparity images computed using *ATGV* and Census transform. Right column: 3D reconstruction rendered from a novel viewpoint.



(a) Groundtruth

(c) *ATGV*

(d) *NLTGV*-CTF

Figure 5.14: Failure cases of coarse-to-fine warping and comparison to *ATGV*. Thin structures and dominant occlusions can significantly deteriorate the performance of the CTF approach. The lifting-based *ATGV* model is able to handle such situations more gracefully.

	2px	3px	4px	5px	Avg-Err
ITGV-CTF	9.62	7.03	5.78	5.00	1.36 px
NLTGV-CTF	9.05	6.67	5.49	4.70	1.29 px
ATGV	**7.56**	**5.24**	**4.12**	**3.44**	**1.10 px**

Table 5.4: Results on the KITTI training benchmark.

5.7.2 KITTI Benchmark

We compare the proposed approach to the baseline algorithm using the KITTI stereo benchmark [59]. This benchmark consists of 195 test images and 194 training images captured from an automotive platform. Groundtruth data is given in the form of semi-dense disparity maps that were captured using a Velodyne laser scanner.

Table 5.4 shows a comparison between $ATGV$ and coarse-to-fine-based methods with TGV regularizers. We used the Census data term for all evaluations. We observe a similar trend as before. $NLTGV$ performs slightly better than $ITGV$. $ATGV$ outperforms the coarse-to-fine-based methods by a comfortable margin. In general $ATGV$ again compares favorably with the state-of-the-art. This can be seen in Table 5.5. $ATGV$ shows a similar robustness as SGM, but a higher accuracy (Avg-Err). The currently leading methods MC-CNN [160], which is based on a learned matching function, and PCBP-SS [171], which refines estimates from SGM using a sophisticated model, are more robust than our approach, but again show similar accuracy. Figure 5.15 shows qualitative results for the $ATGV$ model on this benchmark. Finally, Figure 5.16 shows a direct comparison between $ATGV$ and $NLTGV$. It can be seen that $ATGV$ is better able to preserve small details like poles or small trees and also is more robust. This comes at the price of a significantly higher computational complexity. The KITTI benchmark includes disparity ranges of more than 200 pixels, the $ATGV$ model thus is very computationally demanding. With the presented settings computation takes roughly 3 minutes per image, whereas the coarse-to-fine approaches take between 1 (for TGV) and 15 seconds (for $NLTGV$) per image.

	2px	3px	4px	5px	Avg-Err
MC-CNN [160]	**4.34**	**2.61**	**2.04**	**1.75**	**1.0 px**
PCBP-SS [171]	5.19	3.40	2.62	2.18	**1.0 px**
wSGM [140]	7.27	4.97	3.88	3.25	1.6 px
SGM [77]	8.66	5.76	4.38	3.56	1.3 px
ATGV	7.08	5.02	3.99	3.33	**1.0 px**

Table 5.5: Results on the KITTI testing benchmark and comparison to the state-of-the-art.

5.8 Discussion

In this section we have introduced the problem of stereo estimation. We have described the fundamental geometric principles governing stereo camera setups and shown how higher-order variational models can be used to solve the stereo estimation problem. Specifically it was shown how the special one-dimensional structure of the matching problem can be leveraged in order to more directly solve a non-convex global variational model. We further adapted the coarse-to-fine warping approach to the stereo setting, which can be used to solve the stereo problem faster, although not as accurately.

In the case of the lifting-based model we have seen that TGV serves as an accurate prior for stereo estimation. The main challenge lies in robustness, which is mostly addressed by the data term. While the Census and related data terms are robust, they still fail under some conditions, like in the presence of heavy artifacts. Novel developments, like learned matching functions [160], could be directly incorporated into this model in order to address this issue. A last major challenge of the lifting approach is its dependence on the number of possible disparities. This currently prevents application of the model to larger input images, which typically also have a large range of possible disparities. One possible approach to address this issue is a technique known as narrow-banding [141], which could, depending on the number of disparities, possibly lead to a speed-up of one or two orders of magnitude, with only moderate loss in accuracy. Additionally, Problem (5.68) can be transformed to an equivalent ROF problem [33], which is strongly convex and thus could be optimized using accelerated algorithms like $FISTA$.

Last, it is possible to apply the proposed block-coordinate descent approach also to stereo models involving a $NLTGV$ prior, since the lifting approach is specified for quite general convex regularizers. This variant would very likely further enhance the accuracy and robustness.

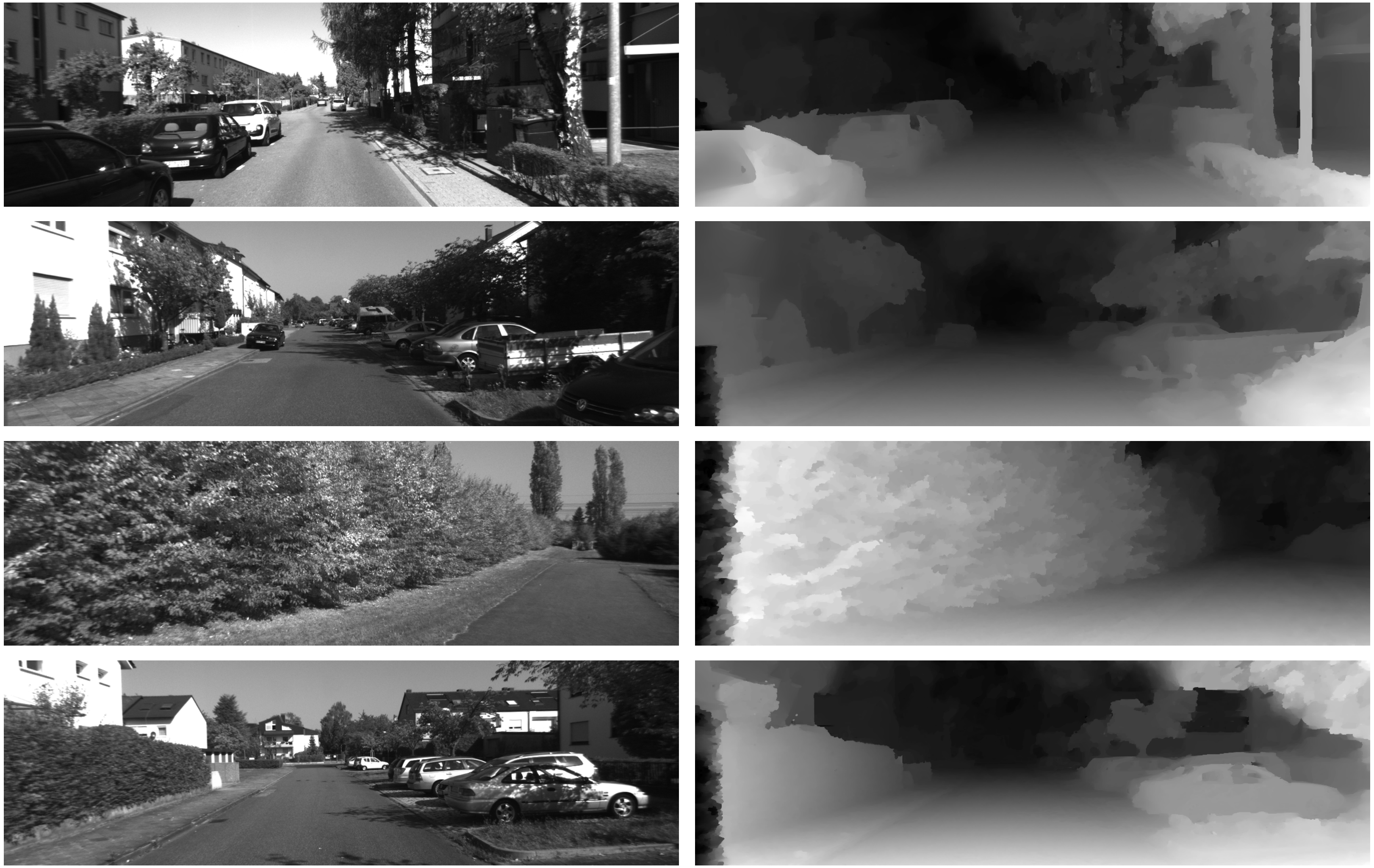
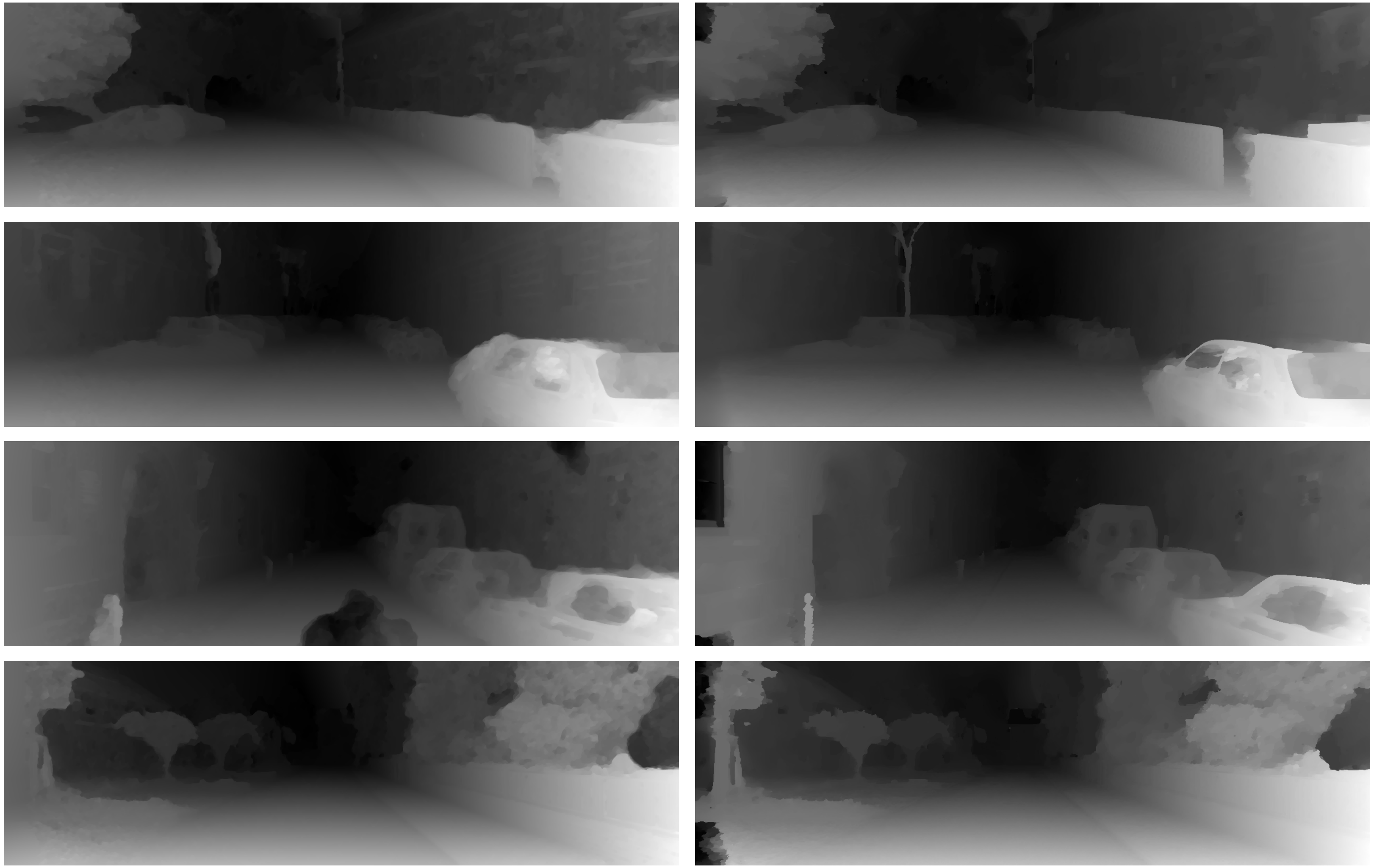


Figure 5.15: Qualitative results of *ATGV* on KITTI benchmark.



(g) *NLTGV*

(h) *ATGV*

Figure 5.16: Qualitative comparison between *NLTGV* and *ATGV* on KITTI benchmark. Rows 1 to 3: *ATGV* is better able to preserve fine details and is also more robust. Row 4: In the absence of small details both results are comparable.

6 Summary

In this thesis we have introduced higher-order variational methods for dense correspondence problems. We have shown that Total Generalized Variation (TGV) is a prior which is very well suited for the tasks of stereo and optical flow estimation. We have focused on TGV of order two, which models the assumption of piecewise affine solutions. This is in general an excellent assumption for both stereo and optical flow estimation, especially for scenes depicting man-made structures. We have proposed extensions to the TGV prior which are aimed at increasing robustness and accuracy. Image-Driven Total Generalized Variation (ITGV) can incorporate prior cues on likely locations of motion or depth discontinuities based on the reference image. This leads to sharp and well-localized discontinuities. Non-Local Total Generalized Variation (NLTGV) takes this idea one step further and allows to incorporate soft-segmentation cues, which model the assumption that connected pixels with similar color typically undergo the same motion or are part of the same surface. We have shown that this assumption can strongly increase the accuracy of optical flow and stereo models.

Variational models also consist of a data term, which evaluates the likelihood of the solution given the observed data. As our experiments have shown, careful design of this term is as important as the selection of a good prior term. We have shown that data terms based on the Census transform are well-suited for realistic scenarios, where challenging illumination conditions are the norm. For the optical flow task, we have introduced a scale-robust version of the Census data term, which accounts for scale changes between the observed images. Scale changes are most prominently encountered in the case of a forward or backward moving camera. This scenario is for example frequently given for cameras mounted on a car or moving robot. Our experiments have shown that accounting for scale changes in the data term can lead to increased accuracy and robustness. It is worth noting that the Census data term gains its robustness by discarding much of the information that is included in an image patch. It is thus important to couple this data term with a strong prior in order to get an overall accurate model.

We have shown that our proposed modifications lead to state-of-the-art optical flow models. For the case of optical flow estimation from a moving vehicle, the proposed *NLTGV* prior together with the scale-robust Census transform leads to very accurate results. At the time of writing this combination was among the top-ranking two-frame optical flow methods on the KITTI benchmark.

For stereo estimation we have shown how the specific properties of this problem can be exploited to build a more accurate optimization procedure. We applied this approach to a model based on the *ITGV* prior and the Census data term. Our experiments show that the resulting stereo estimation algorithm outperforms classical coarse-to-fine warping approaches by a large margin. This can be mainly attributed to increased robustness and the lack of problems introduced by the classical coarse-to-fine warping approach. The resulting stereo estimation algorithm compares favorably to state-of-the-art methods on common benchmarks, especially in terms of accuracy.

6.1 Limitations and Future Work

For optical flow estimation the coarse-to-fine warping is the major source of errors. Sequences, which feature very large motions are problematic in such approaches, regardless of the used priors and data terms. This problem could be tackled by augmenting the variational models with feature-based costs [24, 29, 164]. This, however, requires the availability of a sufficiently large number of high quality features, which may not be available for weakly textured scenes. Since replacing the coarse-to-fine approach by a more sophisticated method has clearly shown increased accuracy for the stereo problem, it would be worthwhile to further explore a similar approach for the optical flow problem. For first-order regularizers there exist tight convex relaxations for the optical flow problem, which are reminiscent of the lifting approach employed in our stereo model [64, 143]. Thus a similar strategy of non-smooth block descent is applicable. The main problem that needs to be overcome in such a formulation is the huge computational complexity that arises from the discretization of the label space. A different variant to get rid of the coarse-to-fine warping could be to use convex fusion moves, similar to the approach proposed in [153]. This approach in principle is also applicable to *TGV*-based energies, but has the drawback that an external process needs to be specified in order to generate high-quality candidate solutions.

Another source of errors in both stereo and optical flow estimation are occlusions, which are not directly handled in the proposed models since they are notoriously hard to model accurately in a convex optimization framework [3, 10]. Occlusion constraints can be modeled more directly in discrete frameworks [154, 167] which have larger computational complexity, however. How occlusions can be incorporated in a principled way in continuous models is an open question.

While models based on *TGV* and *ITGV* can achieve near-realtime performance when used in the coarse-to-fine framework, *NLTGV* incurs a significantly higher runtime. It might be possible to use filter-based optimization approaches [91] by slightly reformulating *NLTGV*, which would allow for faster optimization and also to incorporate larger neighborhoods into the prior.

Both *ITGV* and *NLTGV* could be enhanced by using more sophisticated edge detectors or segmentation procedures. Since they currently are directly conditioned on the reference image, oversegmentation artifacts may arise on strong image edges if the edge does not coincide with a true object boundary. Examples of this are often given by cast shadows or surfaces with strong texture. One approach to remedy this effect would be to derive the diffusion tensor of *ITGV* and the support weights of *NLTGV* from a more advanced edge detector [47].

The Census transform has shown to be a good generic data term and is extensively used nowadays. Recently it was shown that matching terms which were learned from groundtruth data can outperform generic matching terms by a large margin [160]. It would be interesting to incorporate such terms into variational models. Going one step further it could be possible to learn both the prior as well as the data term of a variational model for dense correspondence problems from training data [125]. This would also alleviate the need to set the parameters of the model by hand and thus would make the algorithms generally easier to use.

A

List of Acronyms

BTF	Brighness Transfer Function
EPE	End-Point Error
FISTA	Fast Iterative Shrinkage Thresholding Algorithm
ITGV	Image-Driven Total Generalized Variation
MRF	Markov Random Field
MUC	Mapping Uniquness Criterion
NLTGV	Non-Local Total Generalized Variation
NLTV	Non-Local Total Variation
OFC	Optical Flow Constraint
PDE	Partial Differential Equation
QPBO	Quadratic Pseudo-Boolean Optimization
SGM	Semi-Global Matching
TGV	Total Generalized Variation
TV	Total Variation

B

List of Publications

My work at the Institute for Computer Graphics and Vision led to the following peer-reviewed publications. They are listed in chronological order along with the respective abstracts.

B.1 2012

Pushing the Limits of Stereo Using Variational Stereo Estimation

Ranftl, R., Gehrig, S., Pock, T., and Bischof, H.
In: *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*
June 2012, Alcala de Henares, Madrid, Spain

Abstract: We examine high accuracy stereo estimation for binocular sequences that where obtained from a mobile platform. The ultimate goal is to improve the range of stereo systems without altering the setup. Based on a well-known variational optical flow model, we introduce a novel stereo model that features a second-order regularization, which both allows sub-pixel accurate solutions and piecewise planar disparity maps. The model incorporates a robust fidelity term to account for adverse illumination conditions that frequently arise in real-world scenes. Using several sequences that were taken from a mobile platform we show the robustness and accuracy of the proposed model.

Approximate Envelope Minimization for Curvature Regularity

Heber, S., Ranftl, R., and Pock, T.
In: *ECCV Workshop on Higher-Order Models and Global Constraints in Computer Vision*
October 2012, Firenze, Italy

Abstract: We propose a method for minimizing a non-convex function, which can be split up into a sum of simple functions. The key idea of the method is the approximation of the convex envelopes of the simple functions, which leads to a convex approximation of the original function. A solution is obtained by minimizing this convex approximation. Cost functions, which fulfill such a splitting property are ubiquitous in computer vision, therefore we explain the method based on such a problem, namely the non-convex problem of binary image segmentation based on Euler's Elastica.

B.2 2013

Multi-Modality Depth Map Fusion using Primal-Dual Optimization

Ferstl, D., Ranftl, R., Rüther, M., and Bischof, H.
In: *International Conference on Computational Photography (ICCP)*
April 2013, Cambridge, USA

Abstract: We present a novel fusion method that combines complementary 3D and 2D imaging techniques. Consider a Time-of-Flight sensor that acquires a dense depth map on a wide depth range but with a comparably small resolution. Complementary, a stereo sensor generates a disparity map in high resolution but with occlusions and outliers.In our method, we fuse depth data, and optionally also intensity data using a primal-dual optimization, with an energy functional that is designed to compensate for missing parts, filter strong outliers and reduce the acquisition noise. The numerical algorithm is efficiently implemented on a GPU to achieve a processing speed of 10 to 15 frames per second. Experiments on synthetic, real and benchmark datasets show that the results are superior compared to each sensor alone and to competing optimization techniques. In a practical example, we are able to fuse a Kinect triangulation sensor and a small size Time-of-Flight camera to create a gaming sensor with superior resolution, acquisition range and accuracy.

Minimizing TGV-based Variational Models with Non-Convex Data Terms

Ranftl, R., Pock, T., and Bischof, H.
In: *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*
June 2013, Schloss Seggau, Graz, Austria

Abstract: We introduce a method to approximately minimize variational models with Total Generalized Variation regularization (TGV) and non-convex data terms. Our approach is based on a decomposition of the functional into two subproblems, which can be both solved globally optimal. Based on this decomposition we derive an iterative algorithm for the approximate minimization of the original non-convex problem. We apply the proposed algorithm to a state-of-the-art stereo model that was previously solved using coarse-to-fine warping, where we are able to show significant improvements in terms of accuracy.

Variational Shape from Light Field

Heber, S., Ranftl, R., and Pock, T.
In: *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*
August 2013, Lund, Sweden

Abstract: In this paper we propose an efficient method to calculate a high-quality depth map from a single raw image captured by a light field or plenoptic camera. The proposed model combines the main idea of Active Wavefront Sampling (AWS) with the light field technique, i.e. we extract so-called sub-aperture images out of the raw image of a plenoptic camera, in such a way that the virtual view points are arranged on circles around a fixed center view. By tracking an imaged scene point over a sequence of sub-aperture images corresponding to a common circle, one can observe a virtual rotation of the scene point on the image plane. Our model is able to measure a dense field of these rotations, which are inversely related to the scene depth.

Revisiting loss-specific training of filter-based MRFs for image restoration

Chen, Y., Pock, T., Ranftl, R., and Bischof, H.
In: *German Conference on Pattern Recognition (GCPR)*
September 2013, Saarbrücken, Germany

Abstract: It is now well known that Markov random fields (MRFs) are particularly effective for modeling image priors in low-level vision. Recent years have seen the emergence of two main approaches for learning the parameters in MRFs: (1) probabilistic learning using sampling-based algorithms and (2) loss-specific training based on MAP estimate. After investigating existing training approaches, it turns out that the performance of the loss-specific training has been significantly underestimated in existing work. In this paper, we revisit this approach and use techniques from bi-level optimization to solve it. We show that we can get a substantial gain in the final performance by solving the lower-level problem in the bi-level framework with high accuracy using our newly proposed algorithm. As a result, our trained model is on par with highly specialized image denoising algorithms and clearly outperforms probabilistically trained MRF models. Our findings suggest that

for the loss-specific training scheme, solving the lower-level problem with higher accuracy is beneficial. Our trained model comes along with the additional advantage, that inference is extremely efficient. Our GPU-based implementation takes less than 1s to produce state-of-the-art performance.

Image Guided Depth Upsampling using Anisotropic Total Generalized Variation

Ferstl, F., Reinbacher, C., Ranftl, R., Rüther, M., and Bischof, H.
In: *International Conference on Computer Vision (ICCV)*
December 2013, Sidney, Australia

Abstract: In this work we present a novel method for the challenging problem of depth image upsampling. Modern depth cameras such as Kinect or Time of Flight cameras deliver dense, high quality depth measurements but are limited in their lateral resolution. To overcome this limitation we formulate a convex optimization problem using higher order regularization for depth image upsampling. In this optimization an anisotropic diffusion tensor, calculated from a high resolution intensity image, is used to guide the upsampling. We derive a numerical algorithm based on a primal-dual formulation that is efficiently parallelized and runs at multiple frames per second. We show that this novel upsampling clearly outperforms state of the art approaches in terms of speed and accuracy on the widely used Middlebury 2007 datasets. Furthermore, we introduce novel datasets with highly accurate groundtruth, which, for the first time, enable to benchmark depth upsampling methods using real sensor data.

B.3 2014

A bi-level view of inpainting-based image compression

Chen, Y., Ranftl, R., and Pock, T.
In: *Computer Vision Winter Workshop (CVWW)*
February 2014, Krtiny, Czech Republic
Best Paper Award

Abstract: Inpainting based image compression approaches, especially linear and non-linear diffusion models, are an active research topic for lossy image compression. The major challenge in these compression models is to find a small set of descriptive supporting points, which allow for an accurate reconstruction of the original image. It turns out in practice that this is a challenging problem even for the simplest Laplacian interpolation model. In this paper, we revisit the Laplacian interpolation compression model and introduce two fast algorithms, namely successive preconditioning primal dual algorithm and the recently proposed iPiano algorithm, to solve this problem efficiently. Furthermore, we extend the Laplacian interpolation based compression model to a more general form, which is based on principles from bi-level optimization. We investigate two different variants of the Laplacian model, namely biharmonic interpolation and smoothed Total Variation regularization. Our numerical results show that significant improvements can be obtained from the biharmonic interpolation model, and it can recover an image with very high quality from only 5% pixels.

Insights into analysis operator learning: From patch-based sparse models to higher-order MRFs

Chen, Y., Ranftl, R., and Pock, T.
In: *IEEE Transactions on Image Processing (TIPS)*
Vol. 23, No. 3, 2014

Abstract: This paper addresses a new learning algorithm for the recently introduced co-sparse analysis model. First, we give new insights into the co-sparse analysis model by establishing connections to filter-based MRF models, such as the Field of Experts (FoE) model of Roth and Black. For training, we introduce a technique called bi-level optimization to learn the analysis operators. Compared to existing analysis operator learning approaches, our training procedure has the advantage that it is unconstrained with respect to the analysis operator. We investigate the effect of different aspects of the co-sparse analysis model and show that the sparsity promoting function (also called penalty function) is the most important factor in the model. In order to demonstrate the effectiveness of our training approach, we apply our trained models to various classical image restoration problems. Numerical experiments show that our trained models clearly outperform existing analysis operator learning approaches and are on par with state-of-the-art image denoising algorithms. Our approach develops a framework that is intuitive to understand and easy to implement.

A higher-order MRF based variational model for multiplicative noise reduction

Chen, Y., Feng, W., Ranftl, R., Qiao, H., and Pock, T.
In: *IEEE Signal Processing Letters*
Vol. 21, No. 11, 2014

Abstract: The Fields of Experts (FoE) image prior model, a filter-based higher-order Markov Random Fields (MRF) model, has been shown to be effective for many image restoration problems. Motivated by the successes of FoE-based approaches, in this letter we propose a novel variational model for multiplicative noise reduction based on the FoE image prior model. The resulting model corresponds to a non-convex minimization

problem, which can be efficiently solved by a recently published non-convex optimization algorithm. Experimental results based on synthetic speckle noise and real synthetic aperture radar (SAR) images suggest that the performance of our proposed method is on par with the best published despeckling algorithm. Besides, our proposed model comes along with an additional advantage, that the inference is extremely efficient. Our GPU based implementation takes less than 1s to produce state-of-the-art despeckling performance.

A Deep Variational Model for Image Segmentation

Ranftl, R., and Pock, T.
In: *German Conference on Pattern Recognition (GCPR)*
September 2014, Münster, Germany
GCPR Best Paper Award - Honorable Mention

Abstract: In this paper we introduce a novel model that combines Deep Convolutional Neural Networks with a global inference model. Our model is derived from a convex variational relaxation of the minimum s-t cut problem on graphs, which is frequently used for the task of image segmentation. We treat the outputs of Convolutional Neural Networks as the unary and pairwise potentials of a graph and derive a smooth approximation to the minimum s-t cut problem. During training, this approximation facilitates the adaptation of the Convolutional Neural Network to the smoothing that is induced by the global model. The training algorithm can be understood as a modified backpropagation algorithm, that explicitly takes the global inference layer into account. We illustrate our approach on the task of supervised figure-ground segmentation. In contrast to competing approaches we train directly on the raw pixels of the input images and do not rely on hand-crafted features. Despite its generality, simplicity and complete lack of hand-crafted features, our approach is able to yield competitive performance on the Graz02 and Weizmann Horses datasets.

Non-Local Total Generalized Variation for Optical Flow Estimation

Ranftl, R., Bredies, K., and Pock, T.
In: *European Conference on Computer Vision (ECCV)*
September 2014, Zürich, Switzerland

Abstract: In this paper we introduce a novel higher-order regularization term. The proposed regularizer is a non-local extension of the popular second-order Total Generalized variation, which favors piecewise affine solutions and allows to incorporate soft-segmentation cues into the regularization term. These properties make this regularizer especially appealing for optical flow estimation, where it offers accurately localized motion boundaries and allows to resolve ambiguities in the matching term. We additionally propose a novel matching term which is robust to illumination and scale changes, two major sources of errors in optical flow estimation algorithms. We extensively evaluate the proposed regularizer and data term on two challenging benchmarks, where we are able to obtain state of the art results. Our method is currently ranked first among classical two-frame optical flow methods on the KITTI optical flow benchmark.

B.4 2015

Bilevel Optimization with Nonsmooth Lower Level Problems

Ochs, P., Ranftl, R., Brox, T., and Pock, T.
In: *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*
June 2015, Lége Cap Ferret, France

Abstract: We consider a bilevel optimization approach for parameter learning in nonsmooth variational models. Existing approaches solve this problem by applying implicit differentiation to a sufficiently smooth approximation of the nondifferentiable lower level problem. We propose an alternative method based on differentiating the iterations of a nonlinear primal-dual algorithm. Our method computes exact (sub)gradients and can be applied also in the nonsmooth setting. We show preliminary results for the case of multi-label image segmentation.

Bibliography

[1] Alvarez, L., Esclarin, J., Lefebure, M., and Sanchez, J. (1999). A PDE Model for Computing Optical flow. In *CEDYA XVI*, pages 1349–1356. (page 60)

[2] Arterberry, M. E. and Yonas, A. (2000). Perception of three-dimensional shape specified by optic flow by 8-week-old infants. *Perception & Psychophysics*, 62:550–556. (page 2)

[3] Ayvaci, A., Raptis, M., and Soatto, S. (2012). Sparse occlusion detection with optical flow. *International Journal of Computer Vision*, 97(3):322–338. (page 142)

[4] Baker, S., Gross, R., and Matthews, I. (2003a). Lucas-Kanade 20 Years On: A Unifying Framework: Part 3. Technical report, Robotics Institute, Carnegie Mellon University. (page 59)

[5] Baker, S., Gross, R., Matthews, I., and Ishikawa, T. (2003b). Lucas-Kanade 20 Years On : A Unifying Framework : Part 2. Technical report, Robotics Institute, Carnegie Mellon University. (page)

[6] Baker, S. and Matthews, I. (2002). Lucas-Kanade 20 Years On : A Unifying Framework : Part 1. Technical report, Robotics Institute, Carnegie Mellon University. (page 58)

[7] Baker, S., Matthews, I., and Gross, R. (2004a). Lucas-Kanade 20 Years On: A Unifying Framework: Part 4. Technical report, Robotics Institute, Carnegie Mellon University. (page)

[8] Baker, S., Patil, R., Cheung, G., and Matthews, I. (2004b). Lucas-Kanade 20 Years On: A Unifying Framework: Part 5. Technical Report 3, Robotics Institute, Carnegie Mellon University. (page 59)

[9] Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., and Szeliski, R. (2011). A Database and Evaluation Methodology for Optical Flow. *International Journal of Computer Vision*, 92(1):1–31. (page 84, 87)

[10] Ballester, C., Garrido, L., Lazcano, V., and Caselles, V. (2012). A TV-L1 optical flow method with occlusion detection. In *Lecture Notes in Computer Science (DAGM)*, volume 7476 LNCS, pages 31–40. (page 142)

[11] Barnes, C., Shechtman, E., Finkelstein, A., and Goldman, D. B. (2009). PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. *ACM Transactions on Graphics*, 28(3). (page 108)

[12] Beck, A. and Teboulle, M. (2009). A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202. (page 20, 124)

[13] Besse, F., Rother, C., Fitzgibbon, A., and Kautz, J. (2014). PMBP: PatchMatch Belief Propagation for Correspondence Field Estimation. *International Journal of Computer Vision*, 110(1):2–13. (page 108)

[14] Birchfield, S. and Tomasi, C. (1998). A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):401–406. (page 112, 113)

[15] Black, M. and Anandan, P. (1991). Robust dynamic motion estimation over time. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. (page 59, 60)

[16] Black, M. J. and Anandan, P. (1996). The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Computer Vision and Image Understanding (CVIU)*, 63:75–104. (page 59)

[17] Blake, A., Kohli, P., and Rother, C. (2011). *Markov Random Fields for Vision and Image Processing*. MIT Press. (page 117)

[18] Bleyer, M., Rhemann, C., and Rother, C. (2011). PatchMatch Stereo - Stereo Matching with Slanted Support Windows. In *British Machine Vision Conference (BMVC)*, pages 1–11. (page 108)

[19] Blomgren, P. and Chan, T. F. (1998). Color TV: Total variation methods for restoration of vector-valued images. *IEEE Transactions on Image Processing*, 7:304–309. (page 53)

[20] Boros, E. and Hammer, P. L. (2002). Pseudo-Boolean optimization. *Discrete Applied Mathematics*, 123(1-3):155–225. (page 61)

[21] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York, NY, USA. (page 9, 49)

[22] Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137. (page 118)

[23] Boyle, J. P. and Dykstra, R. L. (1986). A method for finding projections onto the intersection of convex sets in Hilbert spaces. *Lexture Notes in Statistics*, 37:28–47. (page 21, 124)

[24] Braux-Zin, J., Dupont, R., and Bartoli, A. (2013). A General Dense Image Matching Framework Combining Direct and Feature-based Costs. In *International Conference on Computer Vision (ICCV)*, pages 185–192. (page 60, 86, 93, 142)

[25] Bredies, K. (2014). Recovering piecewise smooth multichannel images by minimization of convex functionals with total generalized variation penalty. *Lecture Notes in Computer Science*, 8239:44–77. (page 53)

[26] Bredies, K., Kunisch, K., and Pock, T. (2010). Total Generalized Variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526. (page 6, 33, 34, 36, 38, 113)

[27] Bresson, X. and Chan, T. (2008). Fast dual minimization of the vectorial total variation norm and applications to color image processing. *Inverse Problems and Imaging*, 2(4):455–484. (page 53, 68)

[28] Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. (2004). High Accuracy Optical Flow Estimation Based on a Theory for Warping. In *European Conference on Computer Vision (ECCV)*, pages 25–36. (page 59, 60, 66, 72)

[29] Brox, T. and Malik, J. (2011). Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):500–513. (page 60, 142)

[30] Bruhn, A. and Weickert, J. (2005). Towards ultimate motion estimation: Combining highest accuracy with real-time performance. In *International Conference on Computer Vision (ICCV)*, volume I, pages 749–755. (page 59, 60)

[31] Bruhn, A., Weickert, J., and Schnorr, C. (2005). Lucas/Kanade meets Horn/Schunk: combining local and global optical flow methods. *International Journal of Computer Vision*, 61(3):211–231. (page 59)

[32] Butler, D. J., Wulff, J., Stanley, G. B., and Black, M. J. (2012). A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (ECCV)*, pages 611–625. (page 92)

[33] Chambolle, A. (2005). Total variation minimization and a class of binary MRF models. In *Lecture Notes in Computer Science (EMMCVPR)*, volume 3757 LNCS, pages 136–152. (page 137)

[34] Chambolle, A., Caselles, V., Novaga, M., Cremers, D., and Pock, T. (2009). An introduction to Total Variation for Image Analysis. Technical report, hal-00437581. (page 116)

[35] Chambolle, A. and Lions, P.-L. (1997). Image recovery via total variation minimization and related problems. *Numerische Mathematik*, 76(2):167–188. (page 55)

[36] Chambolle, A. and Pock, T. (2011). A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145. (page 21, 28, 71, 124, 125)

[37] Chambolle, A. and Pock, T. (2014). On the ergodic convergence rates of a first-order primal-dual algorithm. *Preprint.* (page 23)

[38] Chan, T. F., Esedoglu, S., and Nikolova, M. (2006). Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648. (page 117)

[39] Cohen, I. (1993). Nonlinear Variational Method for Optical Flow Computation. In *Scandinavian Conference on Image Analysis (SCIA)*, pages 523–530. (page 60)

[40] Combettes, P. and Pesquet, J. (2011). Proximal splitting methods in signal processing. *Fixed-point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. (page 21, 23)

[41] Craton, L. G. and Yonas, A. (1990). The Role of Motion in Infants' Perception of Occlusion. *Advances in Psychology*, 69(C):21–46. (page 2)

[42] Criminisi, A., Sharp, T., Rother, C., and Perez, P. (2010). Geodesic image and video editing. *ACM Transactions on Graphics*, 29:1–15. (page 83)

[43] Danielsson, P., Lin, Q., and Ye, Q. (2001). Efficient detection of second-degree variations in 2D and 3D images. *Journal of Visual Communication and Image and Image Representation*, 12(3):255–305. (page 54)

[44] Demetz, O., Hafner, D., and Weickert, J. (2013). The complete rank transform: A tool for accurate and morphologically invariant matching of structures. In *British Machine Vision Conference (BMVC).* (page 111)

[45] Demetz, O., Stoll, M., and Volz, S. (2014). Learning brightness transfer functions for the joint recovery of illumination changes and optical flow. In *European Conference on Computer Vision (ECCV)*, pages 455–471. (page 60, 71)

[46] Di Stefano, L., Marchionni, M., Mattoccia, S., and Neri, G. (2002). Dense stereo based on the uniqueness constraint. In *International Conference on Pattern Recognition (ICPR)*, volume 3, pages 657 – 661. (page 128)

[47] Dollar, P. and Zitnick, C. L. (2013). Structured Forests for Fast Edge Detection. In *International Conference on Computer Vision (ICCV)*, pages 1841–1848. (page 143)

[48] Dowson, N. and Bowden, R. (2008). Mutual Information for Lucas-Kanade Tracking (MILK): An inverse compositional formulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1):180–185. (page 59)

[49] Egnal, G. (2000). Mutual Information as a Stereo Correspondence Measure. Technical report, University of Pennsylvania. (page 110)

[50] Einecke, N. and Eggert, J. (2010). A Two-Stage Correlation Method for Stereoscopic Depth Estimation. In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 227–234. (page 107)

[51] Einecke, N. and Eggert, J. (2014). Block-matching stereo with relaxed fronto-parallel assumption. In *Intelligent Vehicles Symposium*, pages 700–705. (page 107)

[52] Esser, E., Zhang, X., and Chan, T. (2010). A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM Journal on Imaging Sciences*, 3(4):1015–1046. (page 21)

[53] Faugeras, O., Viéville, T., Theron, E., Vuillemin, J., Hotz, B., Zhang, Z., Moll, L., Bertin, P., Mathieu, H., Fua, P., Berry, G., and Proy, C. (1993). Real time correlation-based stereo: algorithm, implementations and applications. Technical report, Institut National de Recherche en Informatique et en Automatique. (page 109)

[54] Felzenszwalb, P. F. and Huttenlocher, D. P. (2006). Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54. (page 107)

[55] Fleming, W. and Rishel, R. (1960). An integral formula for total gradient variation. *Archiv der Mathematik*, 11(1):128–222. (page 116)

[56] Franke, U., Pfeiffer, D., Rabe, C., Knoeppel, C., Enzweiler, M., Stein, F., and Herrtwich, R. G. (2013). Making bertha see. In *International Conference on Computer Vision (ICCV)*, pages 214–221. (page 4, 108)

[57] Fua, P. (1993). A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49. (page 128)

[58] Fusiello, A., Trucco, E., and Verri, A. (2000). Compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22. (page 103, 105)

[59] Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. (page 72, 73, 84, 86, 136)

[60] Gibson, J. J. (1950). *The perception of the visual world*, volume 98. Houghton Mifflin, Boston. (page 1)

[61] Gilboa, G. and Osher, S. (2008). Nonlocal Operators with Applications to Image Processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028. (page 47)

[62] Goldluecke, B. and Cremers, D. (2010). Convex relaxation for multilabel problems with product label spaces. In *European Conference on Computer Vision (ECCV)*, volume 6315 LNCS, pages 225–238. (page 5, 60)

[63] Goldluecke, B., Strekalovskiy, E., and Cremers, D. (2012). The Natural Vectorial Total Variation Which Arises from Geometric Measure Theory. *SIAM Journal on Imaging Sciences*, 5(2):537–563. (page 53)

[64] Goldstein, T., Bresson, X., and Osher, S. (2012). Global minimization of markov random fields with applications to optical flow. *Inverse Problems and Imaging*, 6(4):623–644. (page 60, 142)

[65] Grant, M., Boyd, S., and Ye, Y. (2006). Disciplined convex programming. In *Global Optimization: From Theory to Implementation*, pages 155–210. Springer. (page 24)

[66] Gwosdek, P., Zimmer, H., Grewenig, S., Bruhn, A., and Weickert, J. (2010). A highly efficient GPU implementation for variational optic flow based on the Euler-Lagrange framework. In *ECCV Workshop for Computer Vision with GPUs (CVGPU)*, volume 6554 LNCS, pages 372–383. (page 65)

[67] Hafner, D., Demetz, O., and Weickert, J. (2013). Why Is the Census Transform Good for Robust Optic Flow Computation? In *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, volume 7893, pages 210–221. (page 74, 75)

[68] Hager, G. D. and Belhumeur, P. N. (1998). Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039. (page 58)

[69] Hartley, R. and Zisserman, A. (2004a). Epipolar Geometry and the Fundamental Maxtrix. In *Multiple View Geometry in Computer Vision*, pages 239–326. Cambridge University Press, second edition. (page 103)

[70] Hartley, R. and Zisserman, A. (2004b). *Multiple View Geometry in Computer Vision.* Cambridge University Press, second edition. (page 100)

[71] Hassner, T., Mayzels, V., and Zelnik-Manor, L. (2012). On SIFTs and their Scales. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1522–1528. (page 75)

[72] He, B. and Yuan, X. (2012). Convergence Analysis of Primal-Dual Algorithms for a Saddle-Point Problem: From Contraction Perspective. *SIAM Journal on Imaging Sciences*, 5(1):119–149. (page 21)

[73] Heinrich, M., Jenkinson, M., Brady, S. M., and Schnabel, J. (2010). Discontinuity preserving regularisation for variational optical-flow registration using the modified Lp norm. *MICCAI Workshop on Evaluation of Methods for Pulmonary Image Registration*, pages 185–194. (page 60)

[74] Heise, P., Klose, S., Jensen, B., and Knoll, A. (2013). PM-Huber: PatchMatch with huber regularization for stereo matching. In *International Conference on Computer Vision (ICCV)*, pages 2360–2367. (page 108)

[75] Hermann, S. and Klette, R. (2012). Iterative Semi-Global Matching for Robust Driver Assistance Systems. In *Asian Conference on Computer Vision (ACCV)*, pages 465–478. (page 108)

[76] Hestenes, M. R. and Stiefel, E. (1952). Methods of Conjugate Gradients for Solving Linear Systems. *Journal of Research of the National Bureau of Standards*, 49:409–436. (page 65)

[77] Hirschmueller, H. (2005). Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 807–814. (page 108, 111, 133, 136)

[78] Hirschmuller, H. and Gehrig, S. (2009). Stereo matching in the presence of sub-pixel calibration errors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 437–444. (page 108)

[79] Hirschmuller, H. and Scharstein, D. (2007). Evaluation of Cost Functions for Stereo Matching. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. (page 131)

[80] Horn, B. K. P. and Schunck, B. G. (1981). Determining Optical Flow. *Artificial Intelligence*, 17:185–203. (page 57, 58, 59)

[81] Hosni, A., Rhemann, C., Bleyer, M., Rother, C., and Gelautz, M. (2013). Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511. (page 107)

[82] Humenberger, M., Engelke, T., and Kubinger, W. (2010). A Census-based stereo vision algorithm using modified Semi-Global Matching and plane fitting to improve matching quality. In *Conference on Computer Vision and Pattern Recognition - Workshops (CVPRW)*, pages 77–84. (page 108)

[83] Ishikawa, H. (2003). Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336. (page 114, 117)

[84] Ishikawa, H. (2007). Total absolute Gaussian curvature for stereo prior. In *Asi*, pages 537–548. (page 55)

[85] Kang, S. B. K. S. B., Szeliski, R., and Chai, J. C. J. (2001). Handling occlusions in dense multi-view stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 103–110. (page 129)

[86] Kellman, P. J. and Spelke, E. S. (1983). Perception of partly occluded objects in infancy. *Cognitive Psychology*, 15(4):483–524. (page 2)

[87] Kim, J. K. J., Kolmogorov, V., and Zabih, R. (2003). Visual correspondence using energy minimization and mutual information. In *International Conference on Computer Vision (ICCV)*, pages 1033–1040. (page 110)

[88] Kolmogorov, V. and Zabih, R. (2001). Computing visual correspondence with occlusions using graph cuts. In *International Conference on Computer Vision (ICCV)*, pages 508–515. (page 107)

[89] Kolmogorov, V. and Zabih, R. (2004). What Energy Functions Can Be Minimized via Graph Cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159. (page 117)

[90] Köser, K., Zach, C., and Pollefeys, M. (2011). Dense 3D reconstruction of symmetric scenes from a single image. In *Lecture Notes in Computer Science (DAGM)*, volume 6835 LNCS, pages 266–275. (page 100)

[91] Krähenbühl, P. and Koltun, V. (2012). Efficient Nonlocal Regularization for Optical Flow. In *European Conference on Computer Vision (ECCV)*, pages 356–369. (page 47, 142)

[92] Kumar, A., Tannenbaum, A. R., and Balas, G. J. (1996). Optical flow: A curve evolution approach. *IEEE Transactions on Image Processing*, 5(4):598–610. (page 60)

[93] Kuschk, G. and Cremers, D. (2013). Fast and accurate large-scale stereo reconstruction using variational methods. *IEEE International Conference on Computer Vision - Workshops (ICCVW)*. (page 108)

[94] Lee, D. N. (1980). The optic flow field: the foundation of vision. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 290(1038):169–179. (page 2)

[95] Lellmann, J., Papafitsoros, K., Schoenlieb, C., and Spector, D. (2014). Analysis and Application of a non-local Hessian. *arXiv preprint arXiv:1410.8825*. (page 54)

[96] Lempitsky, V., Rother, C., Roth, S., and Blake, A. (2010). Fusion moves for markov random field optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1392–1405. (page 60, 107)

[97] Leordeanu, M., Zanfir, A., and Sminchisescu, C. (2013). Locally Affine Sparse-to-Dense Matching for Motion and Occlusion Estimation. In *International Conference on Computer Vision (ICCV)*, pages 1721–1728. (page 61, 92)

[98] Levine, M., O'Handley, D., and Yagi, G. (1973). Computer determination of depth maps. *Computer Graphics and Image Processing*, 2(2):131–150. (page 107)

[99] Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135. (page 103)

[100] Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 674–679. (page 58)

[101] Lysaker, M., Lundervold, A., and Tai, X.-C. (2003). Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *IEEE Transactions on Image Processing*, 12(12):1579–1590. (page 54)

[102] Meister, S. (2012). Outdoor stereo camera system for the generation of real-world benchmark data sets. *Optical Engineering*, 51(2). (page 4)

[103] Mileva, Y., Bruhn, A., and Weickert, J. (2007). Illumination-Robust Variational Optical Flow with Photometric Invariants. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)*, pages 152–162. (page 71)

[104] Mitiche, A. and Mansouri, A. R. (2004). On convergence of the Horn and Schunck optical-flow estimation method. *IEEE Transactions on Image Processing*, 13(6):848–852. (page 65)

[105] Miyata, T. (2013). L infinity total generalized variation for color image recovery. In *International Conference on Image Processing (ICIP)*, pages 449–4543. (page 53, 54)

[106] Müller, T., Rabe, C., Rannacher, J., Franke, U., and Mester, R. (2011). Illumination-robust dense optical flow using census signatures. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)*, pages 236–245. (page 73, 74, 111)

[107] Nagel, H.-H. and Enkelmann, W. (1986). An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593. (page 45, 60)

[108] Nakayama, K. (1985). Biological image motion processing: A review. *Vision Research*, 25(5):625–660. (page 2)

[109] Nesterov, Y. (1983). A method of solving a convex programming problem with convergence rate O (1/kˆ2). *Soviet Mathematics Doklady*, 27(2):372–376. (page 18)

[110] Nesterov, Y. (2003). *Introductory Lectures on Convex Optimization, 2004*. Springer US, first edition. (page 9, 18, 23)

[111] Nikolova, M. (2004). A Variational Approach to Remove Outliers and Impulse Noise. *Journal of Mathematical Imaging and Vision*, 20(1-2):99–120. (page 114)

[112] Ojala, T., Pietikäinen, M., and Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59. (page 76)

[113] Oram, D. (2001). Rectification for Any Epipolar Geometry. In *British Machine Vision Conference*, pages 653–662. (page 105)

[114] Ostrovsky, Y., Meyers, E., Ganesh, S., Mathur, U., and Sinha, P. (2009). Visual parsing after recovery from blindness. *Psychological Science*, 20(12):1484–1491. (page 2)

[115] Papafitsoros, K. and Schönlieb, C. (2014). A combined first and second order variational approach for image reconstruction. *Journal of Mathematical Imaging and Vision*, 48(2):308–338. (page 36, 54)

[116] Papenberg, N., Bruhn, A., Brox, T., Didas, S., and Weickert, J. (2006). Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision*, 67(2):141–158. (page 59, 72)

[117] Parikh, N. and Boyd, S. (2014). Proximal Algorithms. *Foundations and Trends in Optimization*, 1(3):123–231. (page 19)

[118] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann Publishers Inc. (page 107)

[119] Plyer, A., Besnerais, G. L., and Champagnat, F. (2014). Massively parallel Lucas Kanade optical flow for real-time video processing applications. *Journal of Real-Time Image Processing*. (page 59)

[120] Pock, T. and Chambolle, A. (2011). Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *International Conference on Computer Vision (ICCV)*, pages 1762–1769. (page 23)

[121] Pock, T., Cremers, D., Bischof, H., and Chambolle, A. (2010). Global Solutions of Variational Models with Convex Regularization. *SIAM Journal on Imaging Sciences*, 3(4):1122–1145. (page 114, 120, 121)

[122] Pock, T., Schoenemann, T., Graber, G., Bischof, H., and Cremers, D. (2008). A Convex Formulation of Continuous Multi-Label Problems. In *European Conference on Computer Vision (ECCV)*, pages 792–805. (page 115, 116, 117)

[123] Ranftl, R., Bredies, K., and Pock, T. (2014). Non-local total generalized variation for optical flow estimation. In *European Conference on Computer Vision (ECCV)*, pages 439–454. (page 8, 47)

[124] Ranftl, R., Gehrig, S., Pock, T., and Bischof, H. (2012). Pushing the Limits of Stereo Using Variational Stereo Estimation. In *Intelligent Vehicles Symposium*, pages 401–407. (page 8, 36, 43, 73, 74, 108, 123)

[125] Ranftl, R. and Pock, T. (2014). A Deep Variational Model for Image Segmentation. In *German Conference on Pattern Recognition (GCPR)*, pages 107–118. (page 143)

[126] Ranftl, R., Pock, T., and Bischof, H. (2013). Minimizing TGV-based variational models with non-convex data terms. In *International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, pages 282–293. (page 8, 113)

[127] Ren, X. (2008). Local grouping for optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. (page 61)

[128] Revaud, J., Weinzaepfel, P., Harchaoui, Z., and Schmid, C. (2015). EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. Technical report, INRIA, CNRS, Universite de Grenoble. (page 61, 86, 93)

[129] Rockafellar, R. T. (1997). *Convex analysis.* Princeton Landmarks in Mathematics. Princeton University Press, Princeton, NJ. (page 9, 35)

[130] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268. (page 24, 25, 26)

[131] Scharstein, D., Hirschmüller, H., Kitajima, Y., Nesic, N., Wang, X., and Westling, P. (2014). High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth. *German Conference on Pattern Recognition (GCPR)*, pages 31–42. (page 131)

[132] Scharstein, D. and Pal, C. (2007). Learning conditional random fields for stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. (page)

[133] Scharstein, D. and Szeliski, R. (2003). High-accuracy stereo depth maps using structured light. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 195–202. (page)

[134] Scharstein, D., Szeliski, R., and Zabih, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *International Journal of Computer Vision*, volume 47, pages 7–42. (page 131)

[135] Schnorr, C. (1994). Segmentation of visual motion by minimizing convex non-quadratic functionals. *Proceedings of 12th International Conference on Pattern Recognition*, 1. (page 60)

[136] Schraudolph, N. and Kamenetsky, D. (2008). Efficient exact inference in Planar Ising Models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1417–1424. (page 118)

[137] Shi, J. S. J. and Tomasi, C. (1994). Good features to track. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. (page 58)

[138] Shulman, D. and Herve, J.-Y. (1989). Regularization of discontinuous flow fields. In *Workshop on Visual Motion*, pages 81–86. (page 60)

[139] Shum, H. Y. and Szeliski, R. (2000). Systems and experiment paper: Construction of panoramic image mosaics with global and local alignment. *International Journal of Computer Vision*, 36(2):101–130. (page 58)

[140] Spangenberg, R., Langner, T., and Rojas, R. (2013). Weighted semi-global matching and center-symmetric census transform for robust driver assistance. In *International Conference on Computer Analysis of Images and Patterns (CAIP)*, pages 34–41. (page 108, 136)

[141] Stangl, F., Souiai, M., and Cremers, D. (2013). Performance Evaluation of Narrow Band Methods for Variational Stereo Reconstruction. In *German Conference on Pattern Recognition (GCPR)*, pages 194–204. (page 137)

[142] Steinbrücker, F., Pock, T., and Cremers, D. (2009). Advanced Data Terms for Variational Optic Flow Estimation. In *Vision, Modeling and Visualization (VMV)*, pages 155–164. (page 71)

[143] Strekalovskiy, E., Goldluecke, B., and Cremers, D. (2011). Tight convex relaxations for vector-valued labeling problems. In *International Conference on Computer Vision (ICCV)*, pages 2328–2335. (page 5, 60, 142)

[144] Sturm, J. (1999). Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 1:625–653. (page 24)

[145] Sun, D., Roth, S., and Black, M. J. (2010a). Secrets of optical flow estimation and their principles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439. (page 47)

[146] Sun, D., Sudderth, E., and Black, M. (2010b). Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2226–2234. (page 61)

[147] Sun, D., Sudderth, E. B., and Black, M. J. (2012). Layered segmentation and optical flow estimation over time. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1768–1775. (page 61)

[148] Sun, J., Li, Y., Kang, S. B., and Shum, H. Y. (2005). Symmetric stereo matching for occlusion handling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 399–406. (page 107, 129)

[149] Sun, J., Zheng, N.-N., and Shum, H.-Y. (2003). Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):787–800. (page 107)

[150] Sun, X., Mei, X., Jiao, S., Zhou, M., and Wang, H. (2011). Stereo matching with reliable disparity propagation. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pages 132–139. (page 113)

[151] Tao, M., Bai, J., Kohli, P., and Paris, S. (2012). SimpleFlow: A non-iterative, sublinear optical flow algorithm. In *Computer Graphics Forum (Eurographics)*, volume 31, pages 345–353. (page 59)

[152] Trobin, W., Pock, T., Cremers, D., and Bischof, H. (2008a). An Unbiased Second-Order Prior for High-Accuracy Motion Estimation. In *German Association for Pattern Recognition (DAGM)*, pages 396–405. (page 54, 60)

[153] Trobin, W., Pock, T., Cremers, D., and Bischof, H. (2008b). Continuous energy minimization via repeated binary fusion. In *European Conference on Computer Vision (ECCV)*, pages 677–690. (page 60, 142)

[154] Unger, M., Werlberger, M., Pock, T., and Bischof, H. (2012). Joint motion estimation and segmentation of complex scenes with label costs and occlusion modeling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1878–1885. (page 61, 142)

[155] Van De Weijer, J. and Gevers, T. (2004). Robust optical flow from photometric invariants. In *International Conference on Image Processing (ICIP)*, pages 1835–1838. (page 71)

[156] Vineet, V., Warrell, J., and Torr, P. H. S. (2012). Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. In *European Conference on Computer Vision (ECCV)*, pages 31–44. (page 60)

[157] Vogel, C., Roth, S., and Schindler, K. (2013a). An evaluation of data costs for optical flow. In *German Conference on Pattern Recognition (GCPR)*, pages 343–353. (page 73, 74, 86, 92, 93)

[158] Vogel, C., Roth, S., and Schindler, K. (2014). View-consistent 3d scene flow estimation over multiple frames. In *European Conference on Computer Vision (ECCV)*, pages 263–276. (page 108)

[159] Vogel, C., Schindler, K., and Roth, S. (2013b). Piecewise rigid scene flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1377–1384. (page 108)

[160] Žbontar, J. and LeCun, Y. (2014). Computing the stereo matching cost with a convolutional neural network. *arXiv preprint arXiv:1409.4326*. (page 136, 137, 143)

[161] Wang, L., Jin, H., Yang, R., and Gong, M. (2008). Stereoscopic inpainting: Joint color and depth completion from stereo images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. (page 130)

[162] Wedel, A., Pock, T., Zach, C., Bischof, H., and Cremers, D. (2009). An Improved Algorithm for TV-L1 Optical Flow. In *Lecture Notes in Computer Science Volume 5604*, pages 23–45. (page 60, 71)

[163] Wei, D., Liu, C., and Freeman, W. (2014). A data-driven regularization model for stereo and flow. *International Conference on 3D Vision (3DV)*, pages 277–284. (page 86, 108)

[164] Weinzaepfel, P., Revaud, J., Harchaoui, Z., and Schmid, C. (2013). DeepFlow: Large displacement optical flow with deep matching. In *International Conference on Computer Vision (ICCV)*, pages 1385–1392. (page 60, 61, 86, 92, 93, 142)

[165] Werlberger, M. (2012). *Convex Approaches for High Performance Video Processing.* PhD thesis, Institute for Computer Graphics and Vision, Graz University of Technology, Graz, Austria. (page 3, 73)

[166] Werlberger, M., Pock, T., and Bischof, H. (2010). Motion Estimation with Non-Local Total Variation Regularization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2464–2471. (page 46, 47, 50, 72)

[167] Woodford, O. J., Torr, P. H. S., Reid, I. D., and Fitzgibbon, A. W. (2008). Global stereo reconstruction under second order smoothness priors. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. (page 107, 142)

[168] Wright, M. H. (2005). The interior-point revolution in optimization: History, recent developments, and lasting consequences. *Bulletin of the American Mathematical Society*, 42:39–56. (page 23)

[169] Xu, L., Jia, J., and Matsushita, Y. (2012). Motion Detail Preserving Optical Flow Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9):1744–1757. (page 59, 60, 72)

[170] Yamaguchi, K., Hazan, T., McAllester, D., and Urtasun, R. (2012). Continuous Markov Random Fields for Robust Stereo Estimation. In *European Conference on Computer Vision (ECCV)*, pages 45–58. (page 108)

[171] Yamaguchi, K., McAllester, D., and Urtasun, R. (2013). Robust monocular epipolar flow estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1862–1869. (page 108, 136)

[172] Yamaguchi, K., McAllester, D., and Urtasun, R. (2014). Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *European Conference on Computer Vision (ECCV)*, pages 756–771. (page 108)

[173] Yang, Q., Wang, L., Yang, R., Stewenius, H., and Nister, D. (2009). Stereo Matching with Color-Weighted Correlation, Hierarchical Belief Propagation, and Occlusion Handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):492–504. (page 107)

[174] Yoon, K.-j. and Kweon, I.-s. (2005). Locally adaptive support-weight approach for visual correspondence search. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 924–931. (page 107)

[175] Zabih, R. and Ll, J. W. (1994). Non-parametric Local Transforms for Computing Visual Correspondence. In *European Conference on Computer Vision (ECCV)*, pages 151–158. (page 73, 111)

[176] Zach, C., Pock, T., and Bischof, H. (2007). A duality based approach for realtime tv-l1 optical flow. In *German Association for Pattern Recognition (DAGM)*, pages 214–223. (page 25, 60, 67, 69, 72)

[177] Zhang, K., Li, J., Li, Y., Hu, W., Sun, L., and Yang, S. (2012). Binary Stereo Matching. *International Conference on Pattern Recognition (ICPR)*, pages 356–359. (page 133)

[178] Zhang, K., Lu, J., and Lafruit, G. (2009). Cross-based local stereo matching using orthogonal integral images. *IEEE Transactions on Circuits and Systems for Video Technology*, 19:1073–1079. (page 107, 109)

[179] Ziegler, J., Bender, P., Schreiber, M., Lategahn, H., Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller, C. G., Kaus, E., Herrtwich, R. G., Rabe, C., Pfeiffer, D., Lindner, F., Stein, F., Erbs, F., Enzweiler, M., Knoppel, C., Hipp, J., Haueis, M., Trepte, M., Brenk, C., Tamke, A., Ghanaat, M., Braun, M., Joos, A., Fritz, H., Mock, H., Hein, M., and Zeeb, E. (2014). Making bertha drive-an autonomous journey on a historic route. *IEEE Intelligent Transportation Systems Magazine*, 6(2):8–20. (page 4)

[180] Zimmer, H., Bruhn, A., and Weickert, J. (2011). Optic flow in Harmony. *International Journal of Computer Vision*, 93:368–388. (page 65)