Data Challenges in Variational Optical Flow

Von der Fakultät Informatik, Elektrotechnik und
Informationstechnik der Universität Stuttgart
zur Erlangung der Würde eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Abhandlung

Vorgelegt von

Michael Stoll

aus St. Wendel

Hauptberichter:	Prof. Dr.-Ing. Andrés Bruhn
Mitberichter:	Prof. Luis Álvarez León, Ph.D.

Tag der mündlichen Prüfung: 16.03.2020

Institut für Visualisierung und Interaktive Systeme (VIS)
der Universität Stuttgart

2020

PRÜFUNGSAUSSCHUSS:

Vorsitzender:	Prof. Dr. rer. nat. Daniel Weiskopf, Universität Stuttgart
Hauptberichter:	Prof. Dr.-Ing. Andrés Bruhn, Universität Stuttgart
Mitberichter:	Prof. Luis Álvarez León, Ph.D., Universidad de Las Palmas de Gran Canaria
Mitprüfer:	Prof. Dr. sc. Michael Pradel, Universität Stuttgart
Mitprüfer:	Prof. Dr. rer. nat. habil. Miriam Mehl, Universität Stuttgart

Abstract

The estimation of the motion between consecutive images of a scene – the so-called optical flow – is a key problem in computer vision. Unfortunately, such consecutive images can expose severe data challenges for this estimation including large displacements due to temporal undersampling of the image sequence or illumination changes that are the outcome of changing lighting conditions.

In this thesis, we address these challenges by improving variational methods which have a successful history and allow for a transparent modeling: (i) We propose a robust integration of external feature matches into variational methods. While feature matching is inherently able to estimate large displacements, it is at the same time sensitive to false correspondences due to lacking regularization. (ii) As an alternative, we develop an extended variational method that is able to estimate large displacements with inherent regularization. This allows to handle many large displacement scenarios while not being sensitive to unconstrained false matches. The potential of such methods to handle these cases is widely underestimated in the literature. (iii) In the context of illumination changes, we learn parametrizations to capture the types of these changes and introduce a variational method that can jointly estimate their magnitudes along with the optical flow. This joint estimation provides robustness against such changes without discarding essential image information. (iv) We combine the most promising concepts of each of the prior methods, i.e. determining illumination changes, estimating regularized motion candidates for large displacements and integrating them robustly into the final optical flow estimation. This leads to a pipeline of variational methods that allows us to robustly handle large displacements even in the presence of illumination changes. (v) We embed all the involved data terms, which are responsible for handling any data within the process of variational motion estimation, into a common notational framework based on the well-known motion tensor notation. This notation not only allows for an easy integration of all of the presented concepts into variational frameworks, it also forms the basis for the integration of further recent concepts such as trajectorial regularization terms.

The results of all these improved variational methods demonstrate the benefits of the aforementioned strategies and show clear advances over prior works.

Kurzzusammenfassung

Die Schätzung der Bewegung innerhalb aufeinanderfolgender Bilder einer Szene – des sogenannten optischen Flusses – gehört zu den Kernproblemen im Bereich des maschinellen Sehens. Unglücklicherweise können derartige aufeinanderfolgende Bilder im Hinblick auf diese Schätzung bedeutende Herausforderungen aufweisen, wie etwa große Verschiebungen auf Grund einer zeitlichen Unterabtastung der Bildfolge oder auch Beleuchtungsänderungen, welche das Resultat sich verändernder Rahmenbedingungen hinsichtlich der Beleuchtung sind.

In dieser Arbeit nehmen wir diese Herausforderungen mit Variationsansätzen, welche eine erfolgreiche Historie vorweisen können und eine Modellierung in transparenter Weise erlauben, in Angriff und verbessern diese entsprechend: (i) Wir stellen eine robuste Integration extern gelieferter Verschiebungen zwischen übereinstimmenden Merkmalen in Variationsansätze vor. Während ein solcher Abgleich von Merkmalen inhärent fähig ist, große Verschiebungen zu schätzen, zeigt er auf Grund fehlender Regularisierung gleichermaßen die Tendenz, falsche Übereinstimmungen zu liefern. (ii) Als eine Alternative entwickeln wir einen erweiterten Variationsansatz, der in der Lage ist, große Verschiebungen unter Beibehaltung einer Regularisierung zu schätzen. Dies erlaubt es uns, viele Szenarien mit großen Verschiebungen abzudecken, ohne dabei unter den Einfluss unbeschränkter Falschverschiebungen zu geraten. Die Fähigkeit solcher Methoden, diese Szenarien abzudecken, wird in der Literatur stark unterschätzt. (iii) Im Umgang mit Beleuchtungsänderungen lernen wir Parametrisierungen, welche die Charakteristika solcher Änderungen erfassen, und bringen einen Variationsansatz ein, der unter Verwendung solcher Parametrisierungen die Ausprägungen dieser Änderungen gemeinsam mit dem optischen Fluss schätzen kann. Diese gemeinsame Schätzung liefert die nötige Robustheit gegen solche Änderungen ohne dabei essentielle Bildinformation zu verwerfen. (iv) Wir kombinieren die vielversprechendsten Konzepte der bislang vorgestellten Ansätze, d.h. die Bestimmung der Beleuchtungsänderungen, die Schätzung von Kandidaten für große Verschiebungen mit inhärenter Regularisierung und die robuste Integration solcher Kandidaten in die finale Schätzung des optischen Flusses. Dies führt zu einer Pipeline von Variationsansätzen, die es uns ermöglicht, großen Verschiebungen selbst in der Gegenwart von Beleuchtungsänderungen Herr zu werden. (v) Schlussendlich betten wir alle involvierten Datenterme, die dafür verantwortlich sind, innerhalb von Variationsansätzen Daten zu verarbeiten, in einen gemeinsamen Notationsrahmen ein, welcher auf der wohlbekannten Bewegungstensor-Notation aufbaut. Diese Notation ermöglicht uns nicht nur eine einfach zu handhabende Integration aller vorgestellten Konzepte in variationelle Rahmenwerke, sie bildet darüberhinaus eine Basis für die Integration weiterer moderner Konzepte wie etwa trajektorialer Regularisierungsterme.

Die Ergebnisse für alle diese verbesserten Variationsansätze demonstrieren die Vorzüge der vorgenannten Strategien und legen klare Fortschritte gegenüber vorherigen Arbeiten dar.

Resumen

La estimación del movimiento entre imágenes consecutivas de una escena, llamado el flujo óptico, es un problema clave en la visión artificial. Desafortunadamente, dichas imágenes consecutivas presuponen importantes desafíos con respecto a los datos para esta estimación. Entre estos desafíos se incluyen grandes desplazamientos debido a un submuestreo temporal de la secuencia de imágenes o cambios de iluminación que son el resultado de condiciones de iluminación variables.

En esta tesis, abordamos estos desafíos mejorando los métodos variacionales que tienen una historia exitosa y permiten un modelado transparente: (i) Proponemos una integración robusta de los partidos de características externos en los métodos variacionales. Si bien la comparación de características es inherentemente capaz de estimar grandes desplazamientos, al mismo tiempo es sensible a las falsas correspondencias debido a la falta de regularización. (ii) Como alternativa, desarrollamos un método variacional extendido que es capaz de estimar grandes desplazamientos con la regularización inherente. Esto permite manejar muchos escenarios de grande desplazamiento sin ser sensible a los falsos partidos ilimitados. El potencial de estos métodos para manejar estos casos está ampliamente subestimado en la literatura. (iii) En el contexto de los cambios de iluminación, aprendemos parametrizaciones para capturar los tipos de estos cambios e introducir un método variacional que pueda estimar conjuntamente sus magnitudes junto con el flujo óptico. Esta estimación conjunta proporciona robustez contra tales cambios sin descartar información esencial de la imagen. (iv) Combinamos los conceptos más prometedores de cada uno de los métodos anteriores, es decir, determinando los cambios de iluminación, estimando candidatos de movimiento regularizados para grandes desplazamientos e integrándolos sólidamente en la estimación del flujo óptico final. Esto nos lleva a una serie de métodos variacionales que nos permite manejar de manera robusta grandes desplazamientos incluso en presencia de cambios de iluminación. (v) Finalmente, incorporamos todos los términos de ligadura involucrados, los cuales son responsables de manejar cualquier dato dentro del proceso de estimación de movimiento variacional, en un marco de notación común además de la conocida notación del tensor de movimiento. Esta notación no solo permite una fácil integración de todos los conceptos presentados en marcos variacionales, sino que que también forma la base para la integración de otros conceptos recientes, como los términos de regularización trajectorial.

Los resultados de todos estos métodos variacionales mejorados demuestran los beneficios de las estrategias antes mencionadas y muestran claros avances sobre trabajos anteriores.

Acknowledgments

This thesis would not have been possible without the support of many people. First of all, let me give many thanks to Andrés Bruhn who gave me the opportunity to work in the Computer Vision and Intelligent Systems Group (CVIS). Amongst many other things, he provided an excellent working atmosphere, the freedom that I could choose the topics and tools I find helpful for my research, great support when writing papers, all the hardware I needed and jobs for many students that assisted me in the context of teaching. Second, my thanks go to Luis Álvarez who agreed to review this thesis and to take the long way from Las Palmas (Spain) to Stuttgart for my defense.

Each position benefits from a solid funding. Hence, I would like to thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for support within Project B04 "Adaptive Algorithms for Motion Estimation" of SFB/Transregio 161 "Quantitative Methods for Visual Computing" (Project Number 251654672).

And each position also benefits from pleasant and fruitful collaborations, which greatly allows to combine success and fun. In this context, I foremost express my gratitude to my (former and current) colleagues Sebastian Volz, Yong-Chul Ju, Daniel Maurer, Simon Rühle and Azin Jahedi. Among these, Sebastian Volz and Daniel Maurer have also been great co-authors who have their valuable share in making the underlying papers of this thesis successful publications. However, conducting joint work and sharing co-authorship does not restrict to members of our chair but tears down borders between chairs or even universities. Hence, I also thank Robert Krüger, Oliver Demetz, Patrick Gairing and Kuno Kurzhals for outstanding collaborations that have found recognition in renowned conferences. Moreover, special thanks go to my colleagues from the institute for the varying leisure activities that we enjoyed together.

Evidently, our scientific work also depends to a great extent on people whose names usually do not appear on any publications. This on the one hand includes our secretaries Margot Roubicek, Christine Schütz and Sophie Schroth who are always the helping hands that manage to get any organizational issues done. On the other hand, this comprises our administrators Anton Malina and Martin Schmid that keep those systems running that enable us to conduct our research. I thank you all for providing me an environment that makes working fun and free of any obstacles.

Finally, there are those people whose support reaches far beyond this work affecting all my life and that have their valuable share on making me the person I am. Hence, I foremost express my deepest gratitude to my family, in particular my parents Judith and Thomas, my sister Lisa, my grandmother Gerlinde and my grandmother Franziska ("Susi") who sadly is no longer with us. You have always unconditionally supported me. Furthermore, I thank my friends for providing me with a lot of fun and diversion.

Table of Contents

Chapter

1 Introduction

In the first chapter, we will introduce the optical flow as a crucial research task in the field of computer vision. This includes its definition, desirable properties that influence the design of corresponding estimation algorithms as well as many data challenges that make its estimation difficult. After these important aspects, we will review basic categories of estimation algorithms and discuss how their performance can be assessed. Finally, we state the contributions of this thesis and give an overview of its organization.

1.1 Optical Flow

The research field of computer vision aims at allowing machines to understand their environment by means of visual data. In this context, the human visual system with its way of processing signals and its abilities to solve tasks such as motion estimation, depth retrieval, scene segmentation and object recognition serves as the role model. In this system, the majority of information origins from the perceptions of the human eyes that each receive 2-D projections of the 3-D world at subsequent time steps. Consequently, most research in computer vision also focuses on the processing of 2-D image data from one or more views of a 3-D scene at one or more time steps. This research is based on methods from the domain of digital image processing but exceeds it in the level of abstraction w.r.t. the information that is extracted from the images, since it recovers scene information like depth or motion in contrast to low level information like e.g. colors or regions.

Among the variety of computer vision problems, motion estimation belongs to the very active research areas with tremendous advances in the past years. When we try to capture the motion that is present within a sequence of images, we estimate the so-called optical flow. It is given by a *dense* two dimensional vector field of displacements that establishes the correspondences of all pixels in the reference frame, i.e. the frame

Figure 1.1: Exemplary illustration of the optical flow. **Top and Bottom:** Two adjacent frames of Sequence 15 of the KITTI 2012 training data set [52]. **Center:** Vector plot of the 2-D displacement field.

where we want to capture the motion, to locations in the successive frame. These frames are usually acquired with a monocular camera at subsequent time steps. Please note that the optical flow is a 2-D projection of the 3-D displacements of the scene onto the image plane. However, in this thesis we are only interested in the 2-D motion and do not estimate or rely on any kind of 3-D information.

1.1.1 Mathematical Definition

Given two consecutive input images I_t and I_{t+1} at time steps t and $t+1$, the optical flow $\mathbf{w}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}), 1)^\top$ is the displacement field that connects a location $\mathbf{x} = (x, y, t)^\top$ in the first frame to a location $\mathbf{x} + \mathbf{w}(\mathbf{x}) = (x + u(\mathbf{x}), y + v(\mathbf{x}), t + 1)^\top$ in the second frame. Hereby, u and v are the horizontal and vertical components, respectively, and the last, temporal component indicates the temporal distance between the time steps t and $t+1$.

1.2 Desirable Properties of the Optical Flow

In order to formulate desirable properties of the optical flow which we want to estimate for a scene, we take a look at the properties that the actual optical flow of the same scene must have. The source of each optical flow is the motion within the 3-D scene, which can be caused both by moving objects that are visible in the image plane and by the motion of the camera itself. For the sake of simplicity, we will refer to the latter case as if the camera stood still and all objects of the scene moved in a way that leads to the same apparent motion in the image plane. Hence, we only focus on changes in the appearance of objects in the scene between image frames. In the prevalent case of rigid objects, i.e. that objects can only undergo translational and rotational motion, we can deduce the following properties of the actual optical flow:

Continuity of the Motion. All continuously connected visible parts of a moving object O undergo an intrinsically continuous optical flow.

Independence of Motions. Objects that move independently from their background introduce edges in the optical flow, so-called *motion discontinuities*. These coincide with structural edges of the respective object. Textural edges do not affect the optical flow.

Fronto-Parallel Translations. Objects that do neither change their apparent size nor their orientation between frames undergo pure translational motion in the image plane, i.e. the optical flow is constant within the object.

Fronto-Parallel Rotations. Objects that change their orientation but keep their apparent size undergo rotational motion within the image plane, i.e. the optical flow changes its direction but not its magnitude along the direction of motion within the object.

Rotations in z-direction. Objects with parts becoming larger and other parts becoming smaller undergo rotational motion with a share of rotation in z-direction, i.e. the optical flow changes its magnitude but may keep its direction (in case of a pure out-of-plane rotation).

Translations in z-direction. Objects that change their size but keep their orientation undergo translational motion with a share of motion in z-direction, which is orthogonal to the image plane, i.e. motion towards the camera or away from it. Their optical flow is divergent (in case of motion towards the camera) or convergent (in case of motion away from the camera) w.r.t. the point at infinity of the 2-D projections of the respective beam of the 3-D parallel motion lines. Along these projected lines, the optical flow keeps its direction but may change its magnitude. Orthogonal to these projected lines, it changes its direction.

As we will see later on, the given image data is qualitatively not perfect and, moreover, it is locally not always expressive enough to estimate the correct optical flow. The

identification of desirable properties of the optical hence can help both designing appropriate algorithms for optical flow estimation and judging the quality of their respective results.

1.3 Data Challenges

The source of any optical flow estimation is the image data that is acquired by cameras. Besides knowing important properties of the structure of a flow field, it is hence also important to know difficulties that come along with the data in order to implement a proper handling for them. Such image data is restricted w.r.t. both quality and quantity where both aspects influence the quality of the optical flow that we can estimate. Due to its discrete nature, the visual information of the captured scene is sampled in both the spatial and the time domain, restricting the quantity of image data. The sensors, that transform intensities into pixel values, are of varying manufacturing quality which directly influences the quality of the pixel values that are acquired. In the following, we describe the five most important data challenges.

Noise. During the physical process of image acquisition a sensor is exposed to the light that is reflected by the scene in front of the camera and the captured information is transformed into digital image data. However, the information that is digitalized is not a clean 2-D representation of the scene. External factors like e.g. dust on the sensor or cosmic rays traversing the camera as well as internal processes like dark current affect the acquisition process. The introduced degradations influence the pixel values, where the difference between the actual and the ideal (non-degraded) pixel values is referred to as *noise*. Moreover, digital image processing, like e.g. lossy compression, can also lead to artifacts that degrade the quality of the image data.

Blur. There are often some objects in a scene that are not represented sharply in the acquired image. This is due to different reasons that can lead to blurry depictions: First of all, the camera lens is focused such that a certain level of depth from the camera's view is depicted sharply. All (parts of) objects that have the appropriate distance to be in this level of depth show sharp edges and textures. The remaining parts of the scene that are closer or farther away undergo the so-called *defocus blur*. Second, the exposure time of the sensor plays a role. If an object undergoes significant motion, i.e. motions larger than one pixel, within the exposure time, the reflected light that is captured by a pixel of the sensor origins from different parts of the object. Hence, the object's depiction is a mixture of different parts of that object or a mixture of the object and its background at the object's boundaries in motion direction. This type of blur is called *motion blur*.

Illumination Changes. The light intensities that are transformed into pixel values depend on different factors, including the incoming light, the reflectance properties of the objects within a scene and the camera settings, in particular its aperture settings and the exposure time. While the reflectance properties of the objects in a scene usually do not change over time, the properties of the incoming light as well as the settings of the camera can do so. While the incoming light changes e.g. due to moving hard shadows of objects, soft shadows of clouds, sunrise/sunset or a changing position of the camera, also the settings of the camera are adapted (either automatically or manually). Since they have a limited dynamic range of light intensities that can be captured, it is useful to automatically adapt the camera settings such that the present intensities are shifted into the dynamic range of the camera. Popular adaptations include a narrowing of the aperture to decrease the incoming light intensities in bright environments and a prolongation of exposure times in dark environments in order to collect more photons over time if the rate of incoming photons is rather low. Moreover, the adaptations of the camera settings usually do not perfectly compensate for the changed properties of the incoming light. Hence, all of these extrinsic illumination changes and intrinsic adaptations likely lead to a variation of the pixel values of an object over time, including both local and global variations. In the remainder of this thesis, we will summarize these intensity-induced variations of pixel values under the term *illumination changes*.

Large Displacements. The actual displacement of an object depends on two quantities: its speed and the time between two depictions of its location, i.e. the acquisition of two image frames, which we refer to as *frame rate*. While the object's speed is an external property of the scene that we capture, the frame rate is a property of the recording system. It generally makes sense to categorize different displacement sizes: small displacements and large displacements. Small displacements are achieved if the frame rate is sufficient for the present velocities within the captured scene. In the sense of human perception, they are visually smooth and the respective moving objects are not considered to be jumping. In the sense of algorithmic motion estimation, small displacements in general constitute the better solvable problem as they are a local problem. Depending on the estimation algorithm, simple assumptions are possible: e.g. in discrete methods an exhaustive search for correspondences can be restricted to a local window both reducing runtime and the probability of ambiguous matches (due to a less amount of potential match partners), while in continuous methods with complex terms locally valid linearizations are possible.

Nevertheless, even large displacements of *large objects* can be treated this way as they coincide with small displacements on a coarser resolution. Downsampling the image decreases both, the size of the objects and the size of the displacement. As long as the object is still visible on a coarser resolution, the displacement size that is to be estimated can be appropriately decreased. Having this in mind, one can solve the problem hierarchically starting on a coarse resolution, estimate displacements there,

upsample the estimated motion, compensate the data for it and iteratively go on in a coarse-to-fine manner up to the original resolution. Hence, different displacement scales are estimated on different resolution levels.

However, if a *small object* undergoes a large displacement, the following problem arises: We cannot treat this problem on the appropriate coarser resolution since the corresponding object is not visible there. Instead, the estimation becomes more complex and the ability of estimating larger displacements comes along with more potential match candidates, thus leading to a higher probability of mismatches. Nevertheless, if the motion of a small object can be explained as the sum of a large displacement of a large background and a small incremental displacement, only the easy to handle small increment remains after the image data has been compensated for the background motion on a coarser level. Thus, the problematic case is a *relative* large displacement, i.e. both relative to the scale of the object *and* relative to the background motion.

Although absolute large displacements – which arise due to an insufficient frame rate of the collected data – may be visually unpleasant for any object size, the real algorithmic challenge is given by *relative* large displacements.

Occlusions. As all image data only covers 2-D projections of 3-D scenes, only those (solid) objects are recorded that are closest to the image plane. Objects in the background may be occluded by those in the foreground. Whenever an object moves within a scene in a different way than its background, there will be parts of the background that are not visible in all frames. Thus, a correspondence for those parts cannot be established. Parts of the background that become hidden over time are called *occlusions* whereas parts that become visible over time are called *disocclusions*. Since the optical flow by definition is only dense in the reference frame, disocclusions are not a problem. These regions appear at later time steps and there is no intrinsic requirement to establish a correspondence for them. Occlusions, however, affect pixels in the reference frame where we want to compute a displacement without having a visual correspondence in the successive frame. The more an object's motion coincides with the background motion, the less it creates an occlusion. After a potential background motion has been compensated for, the amount of occlusion, which an object creates, depends on the overlap of its old and its new position. The overlap depends on both the scale of its relative displacement and the object size: it increases with the scale of the displacement but can never exceed the size of the moving object. As the occlusion is maximal for a zero overlap, we have a natural relation between occlusions and relative large displacements.

In this thesis, we will concentrate on the handling of relative large displacements and illumination changes.

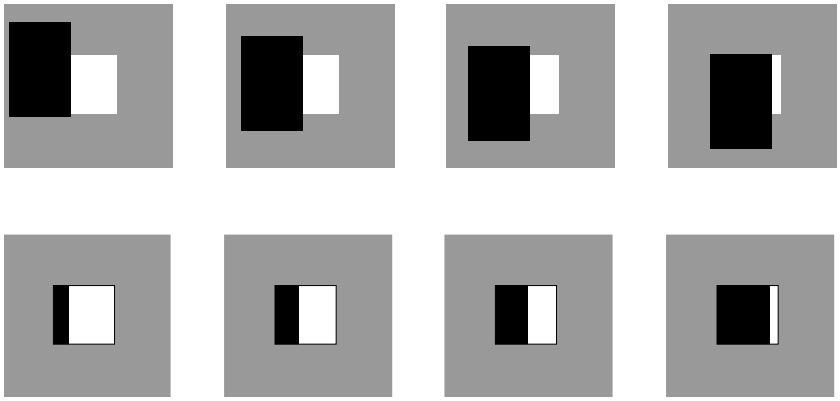


Figure 1.2: Illustration of the aperture problem. **Top row:** Real motion of the black rectangle. **Bottom row:** Visible motion of the black rectangle within the image plane (white). Only the horizontal part of the motion is visible.

1.4 Optical Flow Algorithms

After we have discussed desirable properties of the optical flow as well as important challenges that origin from the image data that is the basis of the optical flow, let us now discuss how these aspects come into play in the design of optical flow algorithms.

The estimation of the optical flow between two or more frames of an image sequence belongs to the class of *inverse problems*. They are defined as problems where we have observations and want to estimate those factors that are the cause of these observations. In the context of optical flow, these observations are given by at least two images and we want to find out which displacement field moves the pixels from the locations in the reference frame to those of the successive frame.

Not only is optical flow estimation an inverse problem, it also highly ill-posed. This can be seen at hand of the *aperture problem* [16]: (i) For pixels in a homogeneous area, it is not clear which of the equal appearing pixels belong to each other. (ii) For pixels on a line, only the motion perpendicular to the line can be estimated, but not the motion along it (see Figure 1.2). Hence, the solution is not unique. Moreover, in occluded areas, a solution does not even exist as there are no visual correspondences. For transparent objects, there may even be multiple valid displacements for a pixel. Please note, that although the optical flow problem itself is ill-posed, this does not necessarily hold for optical flow algorithms using specific modeling assumptions as we will see later on.

All these inherent difficulties in the problem of optical flow estimation as well as the challenges that origin from the data make the usage of appropriate assumptions on both the data (like temporal constancies of certain image features) and on the solution (like smoothness constraints) inevitable. The formulation of these assumptions is *the*

important degree of freedom in designing optical flow algorithms, which has led to a broad variety of algorithms within the past decades. Such algorithms are formulated in different mathematical calculi like the calculus of variations or the calculus of probability; amongst other differences they contain different data constancy assumptions, they have different priors on the solution or they are solved in a different manner.

The set of optical flow algorithms can be categorized w.r.t. different aspects. As this thesis focuses on data and information, we separate the following two classes of algorithms: local ones and global ones.

1.4.1 Local Approaches

Local approaches for optical flow estimation compute flow vectors for all pixels *separately*. There, the sought flow vector minimizes some matching cost that is usually based on a constancy assumption between corresponding features in successive image frames.

Block Matching

One of the simplest local methods is called *block matching*; see e.g. [96]. In this discrete method, a local neighborhood of a certain size around a pixel in the reference frame is compared to local neighborhoods of the same size in the successive frame. When comparing these local neighborhoods, there are different possibilities w.r.t. the chosen distance metric like the sum of squared distances (SSD) or the more robust sum of absolute distances (SAD). The optical flow is then calculated by an exhaustive search that finds the local neighborhood in the successive frame that has the smallest distance, i.e. that is most similar. Although this method alleviates the aperture problem by using neighborhood information, it does not completely solve it. Particularly for pixels whose neighborhood does not cover corners, the solution is still ambigue. Moreover, this method leads to noisy results due to these ambiguities as well as due to noise in the data and it produces block-artifacts in the solution, since discriminative pixels, that dominate the computation of the distance, are present in the neighborhoods of several pixels and thus can lead to the same displacement vectors at these locations. An alternative that does not build on exhaustive but on randomized search is given by patch matching [10].

Feature Matching

If we treat a block as a kind of feature of a pixel that includes neighborhood information, we can embed block matching into the more general concept of feature matching. In contrast to conventional block matching, however, feature matching is usually applied only to discriminative points like corners, so-called key points in the reference frame, with no restriction in the search space. The result is further sparsified by forward-backward consistency checks in order to remove inconsistent matches. There are a lot of requirements on features for the usage in feature matching, including discriminativeness

in order to avoid ambigue matches, illumination invariance to be appropriate under illumination changes or geometric invariances like scale or rotation invariance to allow for feature matching even in the presence of complex motion patterns that change the way a 3-D object is projected on the image plane. There is a whole research area on features, popular examples are given by Histogram of Oriented Gradients (HOG) [36], Geometric Blur (GB) [14], Scale-Invariant Feature Transform (SIFT) [80] or Speeded Up Robust Features (SURF) [12]. Besides these hand-crafted features there are also methods that learn such features for matching [48, 50, 119, 141, 102]. Since the search space is not restricted in feature matching, this type of matching is often used in the context of relative large displacements where other methods with restricted search spaces fail. As there is only a small number of key points that are supposed to be highly discriminative, we achieve sparse results. There are also methods like SIFT Flow [79] or other feature-based methods [40, 109, 145] that do not restrict to these key points. However, they require additional smoothness constraints and can hence not be considered local methods any longer.

Local Differential Methods

Two major drawbacks of the techniques mentioned above are caused by the matching via an exhaustive search: (i) it is time-consuming, since the computational cost per pixel depends on the image size and (ii) without further post-processing, it only provides integer-precise results. An alternative was proposed by Lucas and Kanade [81]. Their method is based on the assumptions that corresponding pixels share a similar brightness value and that the flow is constant within a local neighborhood. Both assumptions can be expressed in terms of a *local energy* that quadratically penalizes deviations from the brightness constancy assumption in the local neighborhood. The basic assumptions are similar to those in block matching approaches. Nevertheless, in contrast to the block matching, the local energy uses a linearized version of the brightness constancy assumption which makes an explicit computation of the solution $(u, v)^\top$ possible and avoids exhaustive search. This explicit computation only requires a more or less constant amount of time per pixel which comes down to linear complexity in the number of pixels. Moreover, local differential methods intrinsically provide sub-pixel accurate results. Their performance improves over pure block matching, but also suffers from the problem of non-dense flow fields as there is not always a unique solution. The used neighborhoods introduce the same inaccuracies as for block matching, i.e. they are not invariant under non-translational motion and motion discontinuities are not preserved. More robustness can be achieved e.g. by assuming an affine flow instead of a constant flow within the local neighborhood as proposed by Shi and Tomasi [121] or by a spatio-temporal extension as proposed by Bigün *et al.* [17] where additionally a temporal component of the flow is estimated. A method that combines these concepts was proposed by Farnebäck [46].

1.4.2 Global Approaches

All of the aforementioned local approaches share the drawback of typically providing sparse results since they cannot overcome the problem of non-discriminative locations. Thus, a separate flow estimation for each pixel is not the appropriate way to achieve dense, high-accuracy results. A contrary approach is hence given by *global methods* which – in complete contrast to local methods – estimate the flow vectors for all pixels in the image *jointly*. With appropriate assumptions on the solution, i.e. relations between the flow vectors of neighboring pixels, a global information flow from discriminative pixels to non-discriminative pixels is possible in order to disambiguate their solutions and thus, to overcome the aperture problem. In contrast to local methods, this information flow is not restricted to a certain distance, i.e. to a fixed neighborhood.

Variational Methods

Usually, these global approaches comprise a data term and a smoothness term. The purpose of the data term is to enforce that source and target pixels of flow vectors share common image properties which are encoded in image features, i.e. to enforce feature constancies. This starts with the simple brightness constancy assumption [68], includes simple illumination-invariant features like the gradient constancy assumption [26], constancy assumptions on more illumination-robust features like the Census transform [124] or the Complete Rank Transform [40] and ranges up to constancy assumptions based on complex features like Histogram of Oriented Gradients (HOG) [36, 109] or Geometric Blur (GB) [14] that are also used in feature matching approaches. The smoothness term, in contrast, imposes regularity assumptions on the solution within the neighborhood. This can be a piecewise constant flow [103, 163], a piecewise affine flow [24] or constraints enforcing other preferred motion patterns. Moreover, modern smoothness terms respect motion discontinuities by not enforcing these patterns across object boundaries [164, 105, 61, 89]. The design of such terms highly reflects the importances of the desired properties of the optical flow as discussed in Sect. 1.2.

Global methods can be categorized w.r.t. the continuity of their domain and co-domain. Methods that are continuous in both domain and co-domain often are given in terms of variational methods [68] whereas discrete methods can be sub-divided into two categories: (i) the domain of the energy is discrete (i.e. defined in terms of pixel coordinates) [126, 79] and (ii) the co-domain is discrete which constitutes a labeling problem [22, 78]. All categories of methods have had their impact in the past where especially in the last decade pure continuous approaches were favorable [26, 29, 150, 149] while today's top-performing methods often are based on discrete approaches [93, 58, 158, 118] or at least include these as a step in a pipeline [111, 112]. Nonetheless, many of the recent top-performing methods also comprise steps using continuous approaches which, in particular, are applied in order to refine flow fields obtained from prior discrete steps [158, 84, 118, 85].

Convolutional Neural Networks

Recently, also learning-based approaches that make use of convolutional neural networks (CNN) have become popular. The global communication within these methods is enforced by the convolutions in these networks. Simple networks like the *FlowNetSimple* in [43] are generic and learn to estimate the optical flow via end-to-end learning using only the input images and the available real solution (also called *ground truth).* More advanced networks even contain a layer that learns and matches features [43, 158, 73, 134]. However, there are also many networks that share similar cost functions as conventional global methods but are parametrized in terms of hierarchical convolutions within different layers [90, 72]. Others integrate individual concepts from conventional global methods into CNNs such as coarse-to-fine warping [134, 107] or the integration of more than two frames [110]. Moreover, there are methods that use CNNs to implement an individual part of a pipeline approach. This includes a CNN-based inpainting step [166] or the prediction of forward displacements with a CNN that uses a backward flow to augment the matching stage [84].

Focus of this Thesis

Among all these methods, variational approaches have a long and successful history and they are still part of many top-performing methods. Their accuracy in combination with their transparent modeling make them a very interesting research target that leaves a lot of room for improvements particularly w.r.t. relative large displacements and illumination changes. Hence, in this thesis we will focus on variational methods although in the meantime other types of approaches also provide very good results.

1.5 Performance Evaluation

The performance of optical flow algorithms can be judged in different ways: qualitatively by using an appropriate visualization technique and judging visually if flow directions, magnitudes and structures fit the subjective expectation which the viewer develops from the image sequence, or quantitatively by assigning numbers to the estimated flow fields and comparing these numbers to flow estimations using other methods and/or settings for the same set of image sequences.

For the qualitative judgment, different visualization techniques have been developed including sparse arrow plots for a coarse impression and dense color representations, whereby the latter accounts for the abilities of the human visual system to precisely distinguish even slight variations among colors.

In order to judge the quality of optical flow algorithms quantitatively, performance measures are necessary. Such a measure can be of different type, but in the optimal

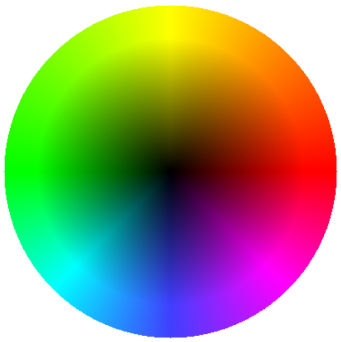


Figure 1.3: This color code [28] will be used for flow visualization where brighter colors indicate longer flow vectors. The color type indicates the direction, e.g. red represents a flow in right-direction, blue represents a flow in down-direction.

case it is (based on) a distance w.r.t. the true optical flow, the so-called *ground truth*. As they formulate deviations from the ground truth, we call these performance measures *error measures* and their resulting values *error values*. The ground truth, however, is not available in most cases, since we usually only have image data in the real world. Performance evaluation thus is performed on so-called benchmarks where both image data and ground truth are available. The ground truth is obtained in different ways, e.g. by creating synthetic scenes where both the flow and the images are created simultaneously (and the frames are given as projections of a scene with given 3-D motion, i.e. by solving a forward problem) or by using information from external sensors like RADAR or LIDAR that have been developed for the detection of distances.

1.5.1 Flow Visualization

As mentioned above, different visualization techniques for the optical flow are available. Since the focus of this thesis lies in high accuracy methods, we are interested in details of the estimated optical flow. In addition to the individual properties of the different flow vectors, i.e. the direction and the magnitude, this includes also structural properties of the flow field, which covers the discontinuities at object boundaries or the presence or absence of staircasing artifacts or other oversegmentation artifacts (due to image textures). Such details can hardly be taken into account by using sparse plots. Hence, we will only use color visualizations of the optical flow in this thesis. Moreover, we will restrict to a specific color code as given in Fig. 1.3 for consistency reasons, although some more variants can be found in the literature.

In order to use the wide color spectrum in a meaningful way, we assign specific properties of flow vectors, i.e. the direction and the magnitude, to specific properties of colors, i.e. the type of the color and the brightness. This can be achieved by having appropriate

representations of both, the flow vectors and the colors. Direction and magnitude of a flow vector can be obtained by transforming it into polar coordinates, i.e. given a flow vector $\mathbf{w} = (u, v, 1)^\top$ we compute the angle $\Phi = \text{atan2}(u, v)$ and the magnitude $r = \sqrt{u^2 + v^2}$. The type of a color and its brightness can then be explicitly stated in the HSV color space. To this end, we assign the angle Φ to the hue component and the magnitude r to the value component of the corresponding HSV color vector. In order to change and/or widen the range of visualizable flow magnitudes, we further rescale r and/or apply a logarithmic transform to it. The backtransformation of the HSV color vectors into the RGB space finally gives us the color visualization of a flow field.

1.5.2 Error Measures

In current important benchmarks [9, 52, 31, 92], three different error measures have become popular: the *average angular error* (AAE), the *average endpoint error* (AEE) and the *bad pixel measure* (BPT) where T is a threshold. In the following $\mathbf{w}^{\text{gt}} = (u^{\text{gt}}, v^{\text{gt}}, 1)^\top$ denotes the ground truth and $\mathbf{w} = (u, v, 1)^\top$ denotes the estimated flow.

Let us start by defining the local quantities that give us the distances between flow vectors. There are two different ones. First, we provide the angular error which focuses on the directions of the flow vectors rather than on their magnitude, as it measures the angular deviation between both vectors in the spatio-temporal domain where flow vectors are considered to be velocities. It is given by

$$\begin{aligned} \text{AE}(\mathbf{w}, \mathbf{w}^{\text{gt}}) &= \arccos\left(\frac{\mathbf{w}^{\text{gt}\top}\mathbf{w}}{||\mathbf{w}^{\text{gt}}||_2 \cdot ||\mathbf{w}||_2}\right) \\ &= \arccos\left(\frac{u^{\text{gt}}u + v^{\text{gt}}v + 1}{\sqrt{u^{\text{gt}2} + v^{\text{gt}2} + 1^2}\sqrt{u^2 + v^2 + 1^2}}\right). \end{aligned} \tag{1.1}$$

Second, there is the endpoint error that measures the spatial Euclidean distance between both flow vectors. It takes into account both the direction and the magnitude of the flow vectors and is given by

$$\text{EE}(\mathbf{w}, \mathbf{w}^{\text{gt}}) = ||\mathbf{w}^{\text{gt}} - \mathbf{w}||_2 = \sqrt{\left(u^{\text{gt}} - u\right)^2 + \left(v^{\text{gt}} - v\right)^2}. \tag{1.2}$$

Using these two local distance measures, we create different error measures for the whole flow field. To this end, we consider a discrete image domain with N pixels in horizontal direction and M pixels in vertical direction, respectively.

Average Angular Error

The average angular error is given by the average over all angular errors in the image domain. This error measure has been introduced by Barron *et al.* [11] in the context of

optical flow performance measurement. It reads

$$\mathrm{AAE}(\mathbf{w}, \mathbf{w}^{\mathrm{gt}}) = \frac{1}{NM} \sum_{i,j=1}^{N,M} \mathrm{AE}(\mathbf{w}_{i,j}, \mathbf{w}^{\mathrm{gt}}_{i,j}) \,. \tag{1.3}$$

Due to the limited range of the AE, this measure can not arbitrarily deteriorate in the presence of occasional faulty flow vectors.

Average Endpoint Error

A quite natural distance measure between vectors is the Euclidean distance. Averaging it over the image domain provides the average endpoint error which reads

$$\mathrm{AEE}(\mathbf{w}, \mathbf{w}^{\mathrm{gt}}) = \frac{1}{NM} \sum_{i,j=1}^{N,M} \mathrm{EE}(\mathbf{w}_{i,j}, \mathbf{w}^{\mathrm{gt}}_{i,j}) \,. \tag{1.4}$$

However, even a single faulty flow vector can lead to arbitrarily large results which obscures the overall quality of the flow field.

Bad Pixel Measure

Another approach that depends on the endpoint error (EE) is the bad pixel measure. It counts the appearances of flow vectors whose endpoint deviates by at least T pixels from the corresponding ground truth vector. Originally coming from the stereo vision [115] it reads

$$\mathrm{BP}T(\mathbf{w}, \mathbf{w}^{\mathrm{gt}}) = 100 \cdot \frac{1}{NM} \sum_{i,j=1}^{N,M} \chi[\mathrm{EE}(\mathbf{w}_{i,j}, \mathbf{w}^{\mathrm{gt}}_{i,j}) < T] \,, \tag{1.5}$$

where $\chi[\text{condition}]$ is 1 if the given condition is fulfilled and 0 else.

As this error measure does not continuously depend on the EE, it is robust in the presence of small imprecisions in the ground truth data. Due to the same reason, however, small variations in the flow field (due to small variations in the corresponding optical flow algorithm) may lead to big variations in the resulting error value if the endpoint errors of a significant amount of flow vectors lie around the threshold T.

Visualization. A visualization of this bad pixel measure is given in Fig. 1.4. This type of visualization has been co-developed by us for the *Special Session on Robust Optical Flow*[1,2] within the *German Conference on Pattern Recognition 2013*. Particularly since 2015, when the KITTI 2015 benchmark [92] was published, it has become increasingly popular and has been used in many recent works such as [146, 58, 158, 86, 89, 13].

[1] https://www.dagm.de/symposien/special-sessions/

[2] https://resources.mpi-inf.mpg.de/conference/dagm/2013/SpecialSession.html

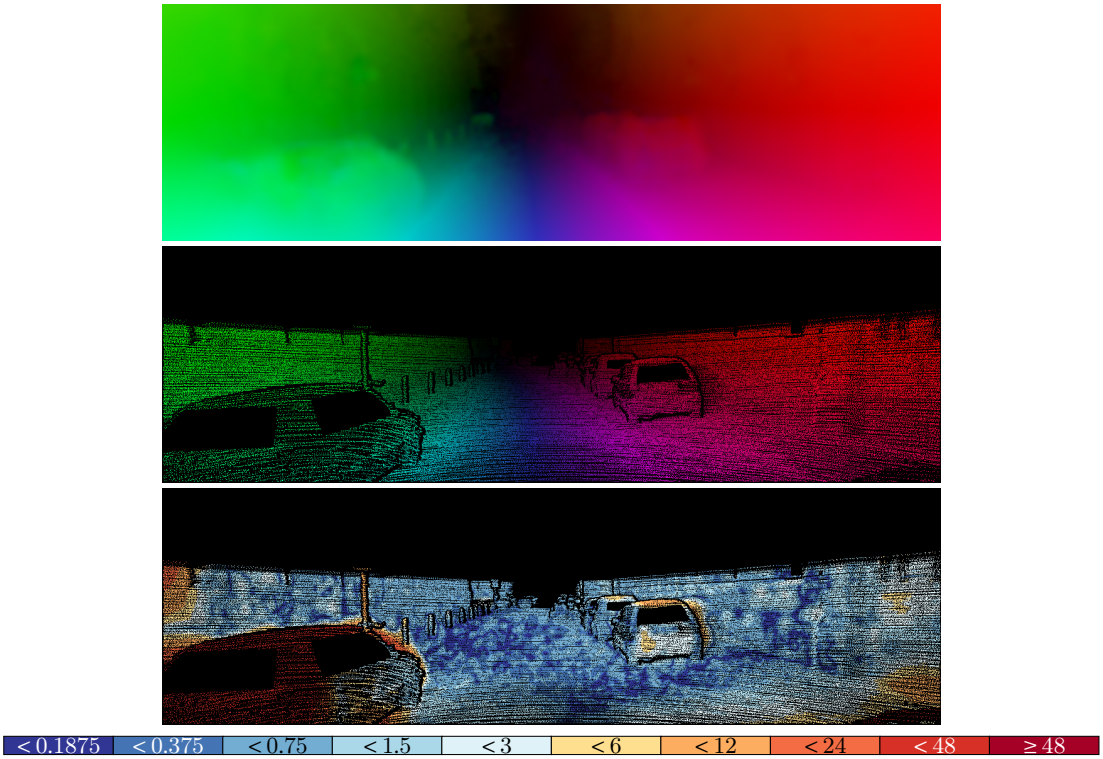


Figure 1.4: Exemplary illustration of the bad pixel visualization. **Top:** Estimated flow for Sequence 15 of the KITTI 2012 training data set. **Center:** Ground truth flow. **Bottom:** Bad pixel visualization for a threshold $T = 3$. Blue regions indicate an endpoint error below the threshold T. In white regions it is close to but still smaller than T and in brown regions it is above T.

1.5.3 Benchmarks

In the early days of quantitative performance evaluation, many works tested their methods on a single artificial image sequence, the *Yosemite* sequence [11], and provided the respective average angular error w.r.t. its ground truth. Fortunately, a lot of benchmarks have been presented since then. The ones which are still important today are the following four.

Middlebury Benchmark

In 2007, the Middlebury benchmark was published providing not only 8 training sequences with given ground truth, but also 8 testing data sets where the ground truth was retained [8, 9]. This allowed for a more systematic and restrictive evaluation: In a first step, method parameters can be trained on the training sequences and in the

second step, results on the testing sequences are estimated using the learned settings. The final evaluation is done after these results have been uploaded to the corresponding website of the benchmark. Using the non-public ground truth, rankings are created among all methods for which results have been uploaded. The two most popular ones are based on the average angular error (AAE) and on the average endpoint error (AEE), respectively. Compared to the single Yosemite sequence, this procedure allows more general conclusions on the methods as it reduces an overfitting in method design and parameters to too few data in many ways: (i) overall it provides more data for evaluation, (ii) it splits the evaluation in a training and a testing stage, which avoids an overfitting of method settings to the testing data, and (iii) it ranks w.r.t. more than one error measure which can shed light on different aspects of the presented methods.

Nevertheless, the low amount of data and their mostly artificial nature cannot provide all the challenges that optical flow methods can face within the wide range of potential applications. In 2012, two more benchmarks were published, the KITTI Vision Benchmark (KITTI 2012) [52] and the MPI Sintel benchmark [31].

KITTI Vision Benchmark (2012)

The KITTI Vision Benchmark (KITTI 2012) [52] provides an increased amount of data with 194 training and 195 testing image sequences of static scenes which have been created in a mostly urban environment from a camera setup that has been mounted on a driving car. The ground truth data have been obtained by LIDAR measurements and hence are sparse in contrast to prior benchmarks. Due to the imprecisions in ground truth data generation, the BP3 error measure is used for the ranking of the published optical flow methods. It advances the Middlebury benchmark w.r.t. the amount of data; and due to their real-world nature new challenges are provided, particularly comprising illumination changes, disturbances in data such as lens flares, under- and oversaturations, and noise, as well as considerable out-of-plane motions.

MPI Sintel Benchmark

The MPI Sintel Benchmark [31] obtains image sequences and ground truths from artificial data, but provides tough challenges by creating specular reflections, motion blur, defocus blur and atmospheric effects. It contains a huge amount of data created out of different scenes from an adapted version of the animated short film Sintel by Roosendaal and the Blender Foundation[3,4]. The training data comprise 23 multi-frame scenes in three different rendering settings (called *albedo*, *clean* and *final*) providing a total of 1064 images per setting and 1041 ground truths. The testing data comprise 12 multi-frame scenes in two different rendering settings (clean and final) providing

[3] https://www.blender.org/foundation/
[4] https://durian.blender.org/about/

Table 1.1: Definition of our MPI Sintel data subset called *Sintel (sub.)*.

Scene	Reference frames
ambush_2	5, 10, 15
ambush_4	11, 16, 21
ambush_6	5, 10, 15
market_6	15, 20, 25
all others	20, 25, 30

a total of 564 images per setting, from which 552 flow fields shall be computed per setting for evaluation. The different settings contain different complexities regarding the rendering of surfaces, shadows, camera and motion blur, atmospheric effects etc., increasing from albedo over clean to final. For both rendering settings of the testing data, i.e. clean and final, a ranking w.r.t. the average endpoint error (AEE) is created out of the uploaded methods' results.

Subset of the Training Data. Since even the training data of the MPI Sintel benchmark contain more than 1000 separate image sequences per pass (clean, final, albedo) which come from 13 different scenes, processes like optimizing parameters – that involve many evaluations on a data set – can become really cumbersome. Hence, we decided to define an additional, reduced data set which consists of the following three image pairs of the clean pass of each scene: the pair in the middle, the pair five frames before and the pair five frames after it, respectively. An overview of the respective reference frames is given in Tab. 1.1. Usually any component analysis using MPI Sintel training data will be conducted on the defined subset. Whenever we need to distinguish between results from different benchmarks, the respective results will be labeled as *Sintel (sub.)*.

KITTI Vision Benchmark (2015)

In 2015, another edition of the KITTI Vision Benchmark Suite, KITTI 2015 [92], has been published. It advances over the KITTI 2012 benchmark by relying on color images and providing dynamic scenes with individually moving objects like cars in the scene. The ground truth of the static parts of the scenes is generated from LIDAR scans as in KITTI 2012 whereas the ground truth of dynamically moving cars is generated by masking the cars, fitting 3-D CAD models of cars to these cars in all frames of the sequence and estimating densely their 3-D scale, pose and rigid body motion within the scene. Although this procedure also introduces a source of inaccuracy, visual inspections with manual exclusion of critical parts let the producers conclude that the ground truth is at least 3 pixels accurate at most parts. Hence, they provide a mixed sparse-dense ground

truth where again the BP3 error measure, which is robust to small imprecisions, can be used to evaluate the performances of the optical flow methods.

Impact on Optical Flow Research

Due to the different characteristics that the different benchmarks have w.r.t. data challenges or apparent motion patterns, methods that have been published in two or more of the respective rankings can have a substantially different order. A method that is accurate in a specific setting may be less accurate in another. Hence, this variety of benchmarks helps finding appropriate models depending on the application scenario.

1.6 Contributions

Among the presented data challenges, particularly two of them are addressed within the scope of this thesis: large displacements and illumination changes. Within a general framework for variational motion estimation, we will provide advanced concepts for the treatment of these challenges. First of all, for both of them we will compare extrinsic concepts for addressing them with an intrinsic variational estimation. In the case of large displacements, the extrinsic concept is the inclusion of separately estimated feature matches that are supposed to contain the large displacements while the intrinsic concept is the adaptation of a variational baseline method such that it can estimate the large displacements itself. In the case of illumination changes, the extrinsic concept is the usage of illumination-invariant features in the data term that simply ignore the type of information that contains certain types of illumination changes while the intrinsic approach is the modification of a variational baseline method such that it can estimate and respect the illumination changes instead of discarding them. And on top of this, we will show, how all the concepts can be combined for the estimation of large displacements in the presence of illumination changes in a purely variational setting. In the following, let us comment on the contributions of this thesis in detail.

1.6.1 Large Displacement Optical Flow

The successful era of variational methods started with the seminal approach of Horn and Schunck [68] in the early 80s of the last century. Its linearized data term, however, is a limiting factor w.r.t. the displacement sizes as the linearization is usually only valid for small displacements. A concept to postpone the linearization to the numerical step has been proposed by Brox *et al.* [26]. This allows for the estimation of absolute large displacements. Relative large displacements, however, are still a considerable challenge in optical flow. In order to overcome this limitation, Brox *et al.* integrated the local

feature matching approach into the variational framework [25, 27]. Nevertheless, their straightforward integration lead to deteriorations in the small displacement setting.

Before we will discuss our contributions to improve the handling of relative large displacements, we will start by further sub-categorizing these displacements into moderately and arbitrarily large displacements. These are the result of a deeper analysis of the deficiencies of the conventional coarse-to-fine warping scheme which also gives hints on how to handle both cases. While the literature does not follow this sub-categorization and directly targets arbitrarily large displacements, moderately large displacements offer a broader spectrum of approaches for their estimation.

In this thesis, we provide two major contributions in the context of moderately large displacements and of arbitrarily large displacements. Both of them have been published at conference venues [129, 127].

▶ Our first approach follows the ideas of Brox *et al.* [25, 27] targeting the estimation of arbitrarily large displacements but addresses this problem by restricting the integration of feature matches to only those locations where additional guidance is considered to be helpful [129]. At locations where the optical flow that is computed with the variational baseline method is already appropriate, we avoid the integration of feature matches. The optical flow at these locations could hardly be improved by good feature matches but it could be severely deteriorated by false matches. We will present a scheme to determine such locations and, moreover, we will also present an additional confidence measure that rates the improvement of a feature match over the baseline flow vector. Both concepts are based on the evaluation of the data term which serves as an indicator for the quality of feature matches and optical flow vectors. Overall, the restricted integration of feature matches and the improved confidence measure lead to strongly improved results compared to prior works in the context of relative large displacements.

Although this strategy does reduce the number of false matches at locations where conventional optical flow already is appropriate, it cannot prevent prevailing false matches from deteriorating the result at locations where additional guidance is necessary. In this case, different strategies are possible that address the problem from different viewpoints and may complement each other: the integration of more discriminative features as e.g. proposed by Weinzaepfel *et al.* [154], or a post-regularization step of a given set of features as proposed by Drayer and Brox [44].

▶ Our second approach focuses on a different strategy which is targeted at handling moderately large displacements: We integrate matches with inherent regularization [127] where we apply a de-regularization strategy within a variational approach. To this end, we follow our analysis of the deficiencies of conventional coarse-to-fine warping schemes which reveals a balancing problem between the data term and the smoothness term on coarse levels in case of moderately large displacements. The key idea of our approach is to maintain different smoothness weights at the same time in order to

obtain results for different balances between both terms. Within appropriate balances, the resilient brightness constancy assumption (BCA) is in many cases able to establish a large displacement correspondence even for small objects. The estimation of these results is done jointly in a combined variational framework which, moreover, also includes a fusion term that adaptively combines the differently smooth results into a final flow field.

1.6.2 Illumination Changes

The basic brightness constancy assumption that has been used in the early works [68, 81] of optical flow estimation is a very intuitive assumption to use. In the presence of illumination changes, however, it becomes invalid as the brightness values of corresponding pixels do not coincide anymore. As a consequence, constancy assumptions on features with advanced illumination invariance have been developed. This includes gradient-based features of different orders to cope with additive illumination changes as used in [26, 94, 155, 79, 109] or relative-order based features to cope with any type of monotonic illumination changes as used in [162, 124, 106, 23, 40]. Nevertheless, this invariance is bought by a loss of information. Especially in homogeneous regions, constancy assumptions based on these invariant features cannot help steering the estimation of the optical flow.

▶ In this thesis, we address this problem by proposing a very general variational framework that is parametrized in terms of exchangeable basis functions and thus is able to estimate different types of illumination changes jointly with the optical flow. The corresponding approach has been published at a conference [41]. Our joint approach has two advantages compared to approaches based on invariances: (i) It does not discard important illumination information, and (ii) It allows to learn these basis functions from training data.

The coefficients that determine the influence of each type of illumination change (where the type is parametrized by a basis function) enter the variational approach as additional functions that have to be estimated. In order to distinguish motion-induced changes of pixel values from illumination-induced changes, a well-balanced regularization strategy is necessary. Our research focuses on the embedding of these strategies into the variational framework, particularly by extending the motion tensor notation by entries for the illumination coefficients and balancing appropriate regularizers for the optical flow and for the illumination coefficients. In contrast to the PhD thesis of Demetz [39], who is co-author of the corresponding paper [41], we rather focus on the modeling than on the learning part.

1.6.3 Large Displacement Optical Flow in the Context of Illumination Changes

Finally, it would be desirable to extend our approach that handles moderately large displacements such that it can handle these large displacements in the presence of illumination changes. The pure de-regularization (as realized in [127]) has deficiencies in the context of illumination changes, since the brightness constancy assumption (BCA) does not hold. Simply replacing it by invariant constancy assumptions such as the gradient constancy assumption (GCA) to handle these illumination changes, however, does not work, since this constancy assumption provides sparse information and hence is not resilient enough to find a balance with the smoothness term that allows for the estimation of large displacements. Unfortunately, the de-regularization strategy strongly depends on resilient data terms, such as the BCA.

▶ Hence, as a further contribution, we combine our approach from [127] with the approach from [41], that estimates illumination changes, in order to be able to keep the BCA in the data term. The corresponding approach has been published at a conference [128]. Unfortunately, a straightforward combination is not possible, since the joint estimation of optical flow and illumination changes from [41] requires a significant and well-balanced regularization strategy for all involved unknowns which, however, is not possible with all the low smoothness weights that appear during the de-regularization in [127]. Hence, we disassemble the joint variational model into a pipeline approach including a distinguished step for the estimation of illumination changes.

Similar to our first work on large displacements [129], we start by computing the baseline flow – where the variational model includes invariant data constancy assumptions such as the GCA. This flow field serves both as a basis to subsequently estimate the illumination changes and as a basis to determine locations where additional matches can be helpful. In contrast to [129], however, these matches are flow candidates obtained from *de-regularized variational approaches* instead of *feature matches*. The illumination changes are determined similar to [41], this time, however, with an initial flow field provided by the baseline. Using the estimated illumination changes, we can then compensate the first frame for these changes, which allows us to apply the BCA as a resilient data term within a de-regularization similar to [127]. This way, we obtain flow candidates that consist of relative large displacements in the context of illumination changes. A fusion based on data reliability and candidate reliability measures similar to [127] provides the final field of flow candidates. Finally, we integrate these candidates only at promising locations where further guidance by additional candidates is necessary, similar to [129].

1.6.4 Tensors for Point Constraints

The handling of data challenges requires appropriate data terms that are able to integrate corresponding data into a variational optical flow estimation. In this context, linear data terms or linearized versions of data terms play an important role, since typical modern variational optimization frameworks eventually end up in solving series of linear equation systems. Such linear(ized) data terms allow for a formulation in terms of a motion tensor notation [28, 47, 30] which allows to embed different linear data terms into a variational framework by only providing the corresponding motion tensor.

▶ Hence, on the one hand, we provide motion tensor notations for all data terms that are used within this thesis. On the other hand, we derive motion tensors for important concepts from the literature and, inspired by these concepts, develop novel directional similarity and directional regularization [88] constraints that allow for a corresponding tensor formulation.

1.7 Organization

The organization of this thesis roughly breaks down into three main parts: preliminaries on variational optical flow methods, our research on data challenges and the presentation of a suitable notation in terms of motion tensors for point constraints.

Preliminaries. In Chapter 2, we present the foundations on variational motion estimation. This includes a definition as well as the minimization of such models. Moreover, we present several methods from the literature that constitute a golden thread from the very beginning of this type of approaches up to the immediate baseline methods and techniques which are the starting point of our research.

Research on Data Challenges. The next block of chapters, which comprises the Chapters 3 to 6 presents our research on relative large displacements introducing two novel methods (Chapter 3 and Chapter 4), our research on the handling of illumination changes (Chapter 5) and as a comprising culmination our research on estimating relative large displacements in the context of illumination changes (Chapter 6).

Chapter 3. The presentation of our method on arbitrarily large displacements in Chapter 3 introduces a novel variational method with a similarity term, demonstrates the adaptive integration of feature matches and completes with a thorough evaluation.

Chapter 4. We go on with our method on moderately large displacements in Chapter 4 where we begin with an introduction into the notion of *moderately* large displacements. In the following, we present a novel variational model for their estimation, which is complemented by an adaptive weighting scheme, and demonstrate its performance in terms of an extensive evaluation.

Chapter 5. The part on handling illumination changes in Chapter 5 starts by introducing the concept of estimating illumination changes in terms of brightness transfer functions (BTFs) followed by the presentation of a very flexible parametrization framework to express them in terms of basis functions and coefficients . The core of the joint estimation of illumination changes and optical flow is given by our novel variational model which estimates these illumination coefficients along with the optical flow. We go on further by demonstrating how the basis functions for the parametrization of BTFs can be learned from training data. Finally, we present several experiments in order to evaluate our method.

Chapter 6. Our final method on estimating relative large displacements in the context of illumination changes is presented in Chapter 6 and combines concepts from the previous chapters. Since the joint estimation requires a well-balanced regularization strategy for all unknowns, which would be perturbed by a de-regularization scheme on the flow, a completely joint model for estimating illumination changes and relative large displacements is out of scope. We thus start with the presentation of a partially decoupled method where a sequence of optical flow estimation and illumination change estimation is followed by a step that compensates the image data for the illumination changes. Thus, the variational approach that incorporates a de-regularization strategy can be applied on photometrically-compensated image data to account for the illumination changes. Afterwards, we present the results of the evaluation. In order to obtain improving results, we further decouple the method where we present different confidence measures as indicators of the flow quality, select candidate regions to restrict the integration of flow candidates to promising locations, introduce variational models which generate candidate flows and present a novel scheme to select the most promising flow candidates at each location of the candidate regions. Finally, we present several experiments that examine the different components of our method and the overall performance.

Tensor Notation. The last main part in Chapter 7 gives an overview on important tensors for point constraints (including the well-known family of motion tensors), both embedding tensors for the data constraints of the previous chapters into a general framework and deriving novel tensors for concepts from the literature within the same framework.

Finally, Chapter 8 concludes this thesis. The appendix consists of three parts: In Appendix A, we state the final parameters that we have obtained through the experiments on our different methods for the different benchmarks. As a supplement to Chapter 5, Appendix B provides the results of an extensive experiment on handling color channels when estimating illumination changes. Appendix C provides an overview of all peer-reviewed publications of the author of this thesis.

Chapter

2 Preliminaries on Variational Optical Flow

Variational approaches have a long and successful tradition in the context of optical flow estimation. Over the last four decades, many of the leading methods of their time have belonged to the class of variational methods, such as [68, 26, 165, 155, 159, 164, 148, 129, 154, 108, 41, 105, 87, 89], or use variational methods as an important refinement step within a pipeline approach, such as [111, 6, 58, 7, 157, 69, 86]. These allow for a transparent modeling where different minimization methods such as the Euler-Lagrange framework, used e.g. in [68, 97, 117, 5, 26, 148, 41, 89], or primal-dual approaches, used e.g. in [163, 150, 156, 159, 105], can be used to find a solution.

In the calculus of variations [45], mathematical problems are formulated in terms of a global energy which is to be minimized. This global energy itself usually is composed as a weighted sum of energy expressions that encode assumptions on the minimizing solution, the so-called *variational model*. Typically, these assumptions favor constancies in some aspects and thus measure deviations, s.t. any kind of deviation from the assumption will contribute to the global energy with a positive value. The weights that are associated with the assumptions express their importance within the model. Typically not all assumptions can be fulfilled at the same time, such that this balancing steers the characteristics of the solution w.r.t. the different assumptions in the variational model. Hence, the solution is the optimal compromise between such assumptions.

2.1 Variational Optical Flow Estimation

A typical variational model for computing the optical flow contains at least two different terms, a data term and a smoothness term (accompanied by some balancing weight). For the optical flow $\mathbf{w}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}), 1)^\top$ over the rectangular image domain $\Omega \subset \mathbb{R}^2$

as a minimizer of such a model, the structure of such a global energy is given by

$$E(\mathbf{w}) = \int_\Omega \underbrace{D(\mathbf{w})}_{\text{Data Term}} + \underbrace{\alpha}_{\text{Weight}} \cdot \underbrace{S(\mathbf{w})}_{\text{Smoothness Term}} \, d\tilde{\mathbf{x}}\,, \tag{2.1}$$

where $\mathbf{x} = (x, y, t)^\top$ is a coordinate in the spatio-temporal domain $\Omega \times \mathbb{R}$ with the spatial counterpart $\tilde{\mathbf{x}} = (x, y)^\top \in \Omega$. For the sake of readability, we will omit the coordinate $\mathbf{x}$ in the estimated functions as long as it can be derived from the context.

Data Term. The data term $D(\mathbf{w})$ connects the given (image) data and the solution $\mathbf{w}$. The literature has proposed a variety of constancy assumptions on these data that can be used within the data term. However, most of them rely on data from a given set of images (at least two). Such an image sequence I is given by

$$I : \Omega \times \mathbb{R} \to \mathscr{R}^{N_\mathrm{c}}$$

where $\mathscr{R} \subset \mathbb{R}$ is the range for each of the N_c image channels. The original image input usually is given either by a grey value image sequence ($N_\mathrm{c} = 1$) or by an RGB color image sequence ($N_\mathrm{c} = 3$). Nevertheless, images can additionally undergo some kind of image transformation (like e.g. the Census Transform [124] or the Complete Rank Transform [40]) which results in an image sequence of higher dimensionality, i.e. $N_\mathrm{c} > 3$.

Smoothness Term. The smoothness term $S(\mathbf{w})$, in contrast, formulates assumptions on the structure of the solution, i.e. how the solution may vary within its neighborhood. Although such a smoothness term can be guided by image data, it only encodes relations between neighboring displacements. Small (higher-order) derivatives of u and v prevent arbitrary fluctuations in the neighborhood of the flow which in general is a desirable assumption. Moreover, it guides the estimation to a solution at those locations where the motion cannot be uniquely determined by the data term – which has been introduced as the aperture problem in Chapter 1, Sect. 1.4 (filling-in effect).

2.2 The Euler-Lagrange Framework

For variational models of this and other types, we want to find a minimizer that fulfills the underlying assumptions as effectively as possible. There are different types of approaches to calculate them. On the one hand, this includes primal-dual approaches [163] which can cope with non-differentiable expressions in the model. On the other hand, if all expressions are differentiable, solving the Euler-Lagrange equations is the widely-used and straightforward approach [68].

Given a general variational model of N functions $\mathbf{u} = (u_1, \ldots, u_N)^\top$ on a two-dimensional domain Ω containing derivatives of order two or less which reads

$$E(\mathbf{u}) = \int_\Omega \mathscr{F}(\mathbf{x}; u_1, \ldots, u_N; \nabla u_1, \ldots, \nabla u_N; \mathscr{H} u_1, \ldots, \mathscr{H} u_N)\, d\tilde{\mathbf{x}}\,, \tag{2.2}$$

where the integrand $\mathscr{F}$ is the Lagrange-Function, $\nabla = \nabla_2 := (\partial_x, \partial_y)^\top$ is the standard gradient operator and $\mathscr{H}$ is the Hessian matrix, a minimizing vector-valued function $\mathbf{u}(\tilde{\mathbf{x}}) = (u_1(\tilde{\mathbf{x}}), ..., u_N(\tilde{\mathbf{x}}))^\top$ must fulfill the Euler-Lagrange equations

$$\begin{aligned}
\mathscr{F}_{u_1} - \frac{\partial}{\partial x}\mathscr{F}_{u_{1x}} - \frac{\partial}{\partial y}\mathscr{F}_{u_{1y}} + \frac{\partial^2}{\partial x \partial x}\mathscr{F}_{u_{1xx}} + \frac{\partial^2}{\partial x \partial y}\mathscr{F}_{u_{1xy}} + \frac{\partial^2}{\partial y \partial x}\mathscr{F}_{u_{1yx}} + \frac{\partial^2}{\partial y \partial y}\mathscr{F}_{u_{1yy}} &= 0\,, \\
\mathscr{F}_{u_2} - \frac{\partial}{\partial x}\mathscr{F}_{u_{2x}} - \frac{\partial}{\partial y}\mathscr{F}_{u_{2y}} + \frac{\partial^2}{\partial x \partial x}\mathscr{F}_{u_{2xx}} + \frac{\partial^2}{\partial x \partial y}\mathscr{F}_{u_{2xy}} + \frac{\partial^2}{\partial y \partial x}\mathscr{F}_{u_{2yx}} + \frac{\partial^2}{\partial y \partial y}\mathscr{F}_{u_{2yy}} &= 0\,, \\
\vdots \qquad \vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad\qquad\qquad \vdots & \\
\mathscr{F}_{u_N} - \frac{\partial}{\partial x}\mathscr{F}_{u_{Nx}} - \frac{\partial}{\partial y}\mathscr{F}_{u_{Ny}} + \frac{\partial^2}{\partial x \partial x}\mathscr{F}_{u_{Nxx}} + \frac{\partial^2}{\partial x \partial y}\mathscr{F}_{u_{Nxy}} + \frac{\partial^2}{\partial y \partial x}\mathscr{F}_{u_{Nyx}} + \frac{\partial^2}{\partial y \partial y}\mathscr{F}_{u_{Nyy}} &= 0\,,
\end{aligned}$$

with boundary conditions

$$\mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_{ix}} - \partial_x \mathscr{F}_{u_{ixx}} - \partial_y \mathscr{F}_{u_{ixy}} \\ \mathscr{F}_{u_{iy}} - \partial_x \mathscr{F}_{u_{iyx}} - \partial_y \mathscr{F}_{u_{iyy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_{ixx}} \\ \mathscr{F}_{u_{ixy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_{iyx}} \\ \mathscr{F}_{u_{iyy}} \end{pmatrix} = 0,$$

for all equations for the sought functions u_i ($i = \{1, ..., N\}$), where $\mathbf{n}$ is an outer normal vector pointing across the image boundary. A derivation of these boundary conditions can be found in [83]. In case that $E(\mathbf{u})$ is strictly convex, there is a unique solution of the equation system and thus a unique minimizer of the functional.

Variational Optical Flow Model. For the general variational model for optical flow in Eq. 2.1 that depends on two functions u and v contained in $\mathbf{w}$, we obtain

$$\begin{aligned}
\mathscr{F}_u - \frac{\partial}{\partial x}\mathscr{F}_{u_x} - \frac{\partial}{\partial y}\mathscr{F}_{u_y} + \frac{\partial^2}{\partial x \partial x}\mathscr{F}_{u_{xx}} + \frac{\partial^2}{\partial x \partial y}\mathscr{F}_{u_{xy}} + \frac{\partial^2}{\partial y \partial x}\mathscr{F}_{u_{yx}} + \frac{\partial^2}{\partial y \partial y}\mathscr{F}_{u_{yy}} &= 0 \\
\mathscr{F}_v - \frac{\partial}{\partial x}\mathscr{F}_{v_x} - \frac{\partial}{\partial y}\mathscr{F}_{v_y} + \frac{\partial^2}{\partial x \partial x}\mathscr{F}_{v_{xx}} + \frac{\partial^2}{\partial x \partial y}\mathscr{F}_{v_{xy}} + \frac{\partial^2}{\partial y \partial x}\mathscr{F}_{v_{yx}} + \frac{\partial^2}{\partial y \partial y}\mathscr{F}_{v_{yy}} &= 0
\end{aligned}$$

with boundary conditions

$$\mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_x} - \partial_x \mathscr{F}_{u_{xx}} - \partial_y \mathscr{F}_{u_{xy}} \\ \mathscr{F}_{u_y} - \partial_x \mathscr{F}_{u_{yx}} - \partial_y \mathscr{F}_{u_{yy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_{xx}} \\ \mathscr{F}_{u_{xy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{u_{yx}} \\ \mathscr{F}_{u_{yy}} \end{pmatrix} = 0,$$

and

$$\mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{v_x} - \partial_x \mathscr{F}_{v_{xx}} - \partial_y \mathscr{F}_{v_{xy}} \\ \mathscr{F}_{v_y} - \partial_x \mathscr{F}_{v_{yx}} - \partial_y \mathscr{F}_{v_{yy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{v_{xx}} \\ \mathscr{F}_{v_{xy}} \end{pmatrix} = 0, \quad \mathbf{n}^\top \begin{pmatrix} \mathscr{F}_{v_{yx}} \\ \mathscr{F}_{v_{yy}} \end{pmatrix} = 0.$$

2.3 Minimization

Although the modeling is elegant in the continuous domain, we first have to discretize the Euler-Lagrange equations in order to have conditions for each pixel of the minimizer when using discrete data.

Discretizations. Based on a grid of N pixels in x-direction and M pixels in y-direction with pixel sizes h_x and h_y, we sample all zeroth order expressions v of data and solution vector entries as

$$v_{i,j} \quad = \quad v(i \cdot h_x, j \cdot h_y)\,.$$

Derivatives are approximated with finite differences. Hereby, spatial derivatives are discretized using central differences and temporal derivatives are discretized using forward differences. The stencils are as follows: spatial derivatives of the images are approximated using the stencil $\frac{1}{12h_{x,y}}(1,-8,0,8,-1)$ (consistency order 4) while temporal differences of the images are approximated using the stencil $(-1,1)$ (consistency order 1), spatial first-order derivatives of the unknown flow are discretized using the stencil $\frac{1}{2h_{x,y}}(-1,0,1)$ (consistency order 2) and second-order derivatives of the flow are discretized using the stencil $\frac{1}{h_{x,y}^2}(1,-2,1)$ (consistency order 1).

Discrete Equation System. After all quantities have been discretized appropriately, we obtain an equation system with conditions for each pixel of each component of the solution vector $\mathbf{u}$. This indeed means, that we have $\dim(\mathbf{u}) \cdot (N \cdot M)$ equations. Even in the case of a linear equation system, a direct solution using e.g. Gauss-Elimination is intractable. These problem sizes usually require iterative solvers such as the Jacobi-method or the Gauss-Seidel-method, potentially embedded in over- or underrelaxation strategies like the successive overrelaxation method (SOR) [161] or like the Fast-Jacobi method (FJ) [151].

2.3.1 Lagged Nonlinearity Method

In the prevalent case of nonlinear equation systems, there are nonlinear functions of the unknowns. This makes the direct application of iterative linear solvers impossible. Nonlinearities usually appear in two variants: (i) the equations are not explicit in the unknowns and (ii) the unknowns are explicit but weighted by functions that depend on the unknowns themselves.

We will later detail on the first case. For the second case, we can apply the so-called *lagged nonlinearity method*, also known as the Kačanov-Galerkin method [49]. That means, we introduce a fixed-point iteration that transforms a non-linear equation system into a series of linear equation systems.

Introducing Different Time Steps. For all expressions of an unknown u of the type $0 = \ldots + g(u) \cdot u$, where g is a non-constant function, we compute u at a new time step $k+1$ while evaluating g on an old time step k and keeping $g(u)$ fixed. In this case, we obtain $0 = \ldots + g(u^k) \cdot u^{k+1}$ which is linear in u^{k+1}. After we have solved the equation system for u^{k+1}, we move on to the next time steps $k+2$, $k+3$, … and obtain a new linear equation system in each time step where all appearances g are treated as fixed weights. This can be related to an iterative reweighted least squares (IRLS) approach [67, 71], where the reweighting is identified with the evaluations of the function g.

2.4 A Golden Thread of Variational Motion Estimation

After we have discussed the mathematical foundations of variational motion estimation, we are now in the position to build a golden thread in variational motion estimation that lead to the development of the immediate baselines of our research. To this end, let us give an overview of these methods and which of their basic concepts influenced our work (see Fig. 2.1). Please note that this is *not* intended to be a complete review of the history of variational motion estimation but an excerpt that guided the development of our baseline methods. We start with the method of Horn and Schunck [68] as *the* pioneering variational method that is the basis for most successive works in this field. It combined the brightness constancy assumption (BCA) (in a linearized version) with a global smoothness assumption to obtain dense optical flow fields. On top of this, the method of Brox *et al.* [26] combined several concepts that include a sub-quadratic penalization [18] for both terms to make the estimation more robust against outliers, the consideration of the gradient constancy assumption (GCA) [142] to gain invariance under illumination changes and the use of a coarse-to-fine warping strategy [15] to overcome the drawbacks of the so far linearized data term and to allow for the estimation of large displacements. A slight variation of the penalization strategy has been introduced by Bruhn and Weickert [29], who proposed a separate penalization of the BCA and GCA within the data term, since outliers of one of these assumptions are not necessarily outliers of the other. Two further works improved over the latter method in different aspects: the method of Zimmer *et al.* [165, 164] and the method of Brox *et al.* [25, 27]. On the one hand, the method of Zimmer *et al.* reduces an implicit weighting in the data constancy assumptions – that provides more influence to these constraints in regions of high contrast compared to low contrast regions – in terms of a constraint normalization [77], it employs a data term that handles color images [101] and it proposes an anisotropic smoothness term that works complementary to the data term. On the other hand, Brox *et al.* [25] supplement the variational motion estimation with a feature matching step to allow for an estimation of relative large displacements.

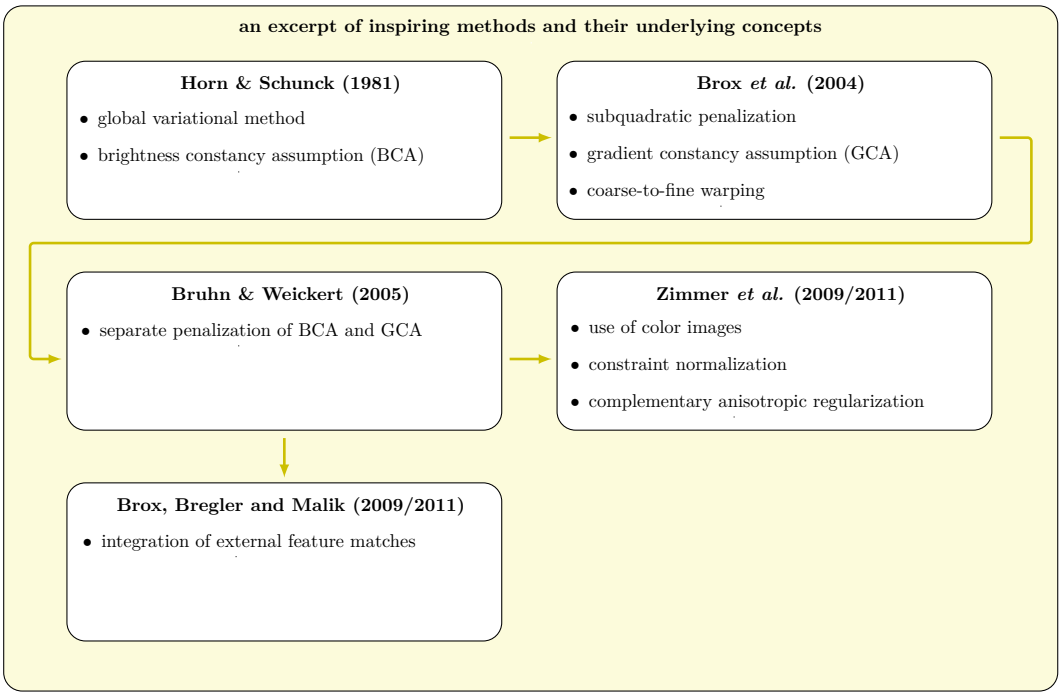


Figure 2.1: An overview of the variational methods that are important in this thesis. For each method, we state the underlying concepts that are important for our research.

2.5 The Method of Horn and Schunck

Let us now discuss the starting point of our golden thread: the method of Horn and Schunck [68]. Its data term formulates the most intuitive and basic constancy assumption for corresponding pixels of the reference frame and the successive frame: the *brightness constancy assumption (BCA)*. Given two successive (grey value) image frames of an image sequence I at the time steps t and $t+1$, the grey values of pixel $\mathbf{x} = (x, y, t)^\top$ and its corresponding pixel $\mathbf{x}+\mathbf{w} = (x+u, y+v, t+1)^\top$ shall coincide, i.e.

$$I(\mathbf{x}) \quad = \quad I(\mathbf{x}+\mathbf{w})\,. \tag{2.3}$$

In order to formulate this assumption in terms of an energy expression, we put all terms on one side of the equation in order to derive an expression that is zero in the ideal case:

$$0 \quad = \quad I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})\,. \tag{2.4}$$

The smoothness term expresses the very intuitive assumption that neighboring pixels shall undergo the same motion, which can e.g. be seen at the surfaces of objects where

all pixels move consistently. In that ideal case, the gradient of the optical flow vanishes and we obtain

$$\nabla u = \mathbf{0}, \tag{2.5}$$
$$\nabla v = \mathbf{0}. \tag{2.6}$$

Basic Energy Functional. In order to plug all this into a global variational energy, we have to formulate the derived expressions in a way such that deviations in both directions symmetrically lead to positive energies. Due to the derivatives that come into play when deriving the Euler-Lagrange equations, this formulation must furthermore be differentiable. A suitable formulation is to square the expressions for the brightness constancy assumption and the smoothness assumption. This way, all expressions are positive, differentiable and on top of this, the derived Euler-Lagrange equations do not contain any additional nonlinear expressions. By summing up both terms, we ensure that both assumptions influence the final solution weighted by a parameter α. The integration over the image domain Ω ensures a dense result while the smoothness term enforces communication between pixels making this method global. Hence, the simplest global variational model for estimating a dense optical flow $\mathbf{w}$ reads

$$E(\mathbf{w}) = \int_\Omega (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x}))^2 + \alpha \left(|\nabla u|^2 + |\nabla v|^2\right) d\tilde{\mathbf{x}}. \tag{2.7}$$

Linearization. When we now determine the Euler-Lagrange equations of this model, there is a problem: the flow vector $\mathbf{w}$ is only implicit in the data term. This is an obstacle to solving the equation system. However, if I is sufficiently smooth, we can linearize this constancy assumption around $\mathbf{x}$ which reads

$$\begin{aligned} 0 &= \underbrace{I + I_x u + I_y v + I_t \cdot 1}_{\text{Linearization of } I(\mathbf{x}+\mathbf{w})} - I \\ \Leftrightarrow\ 0 &= I_x u + I_y v + I_t = \nabla_3 I^\top \mathbf{w}. \end{aligned} \tag{2.8}$$

This constraint is also known as the *optical flow constraint (OFC)*.

2.5.1 Final Model

If we now plug the expression in Eq. 2.8 into the model in Eq. 2.7, we obtain the final variational model of Horn and Schunck which reads

$$E_{\mathrm{HS}}(\mathbf{w}) = \int_\Omega \left(I_x u + I_y v + I_t\right)^2 + \alpha \left(|\nabla u|^2 + |\nabla v|^2\right) d\tilde{\mathbf{x}}. \tag{2.9}$$

This leads to linear Euler-Lagrange equations for the method. Nevertheless, the linearization is typically only valid for small displacements and the quadratic penalizer functions make this method vulnerable to both outliers in the data constancy assumption as well as outliers in the smoothness assumption.

2.5.2 Differentiability Aspects

Particularly the OFC makes use of derivatives of image data while other conceivable data terms might even involve higher-order derivatives. Since image data has external sources, differentiability of that data cannot be guaranteed a priori. A widely used remedy to this problem is the application of a Gaussian smoothing to the given data. It ensures that the filtered data is infinitely many times continuously differentiable and thus stabilizes the numerical evaluation of its derivatives [59]. In the remainder of this thesis, we will hence assume all images to be filtered with a Gaussian with standard deviation σ.

2.5.3 Euler-Lagrange Equations

Let us state the Euler-Lagrange equations explicitly for this pioneering work. The integrand $\mathscr{F}(x, y, u, v, u_x, u_y, v_x, v_y)$ is given by

$$\begin{aligned} \mathscr{F} &= \left(I_x u + I_y v + I_t\right)^2 + \alpha\left(|\nabla u|^2 + |\nabla v|^2\right) \\ &= \left(I_x u + I_y v + I_t\right)^2 + \alpha\left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right). \end{aligned} \tag{2.10}$$

Since we have two unknowns, the general equation system associated to a 2-D functional consists of two equations, one for u and one for v, and is given by

$$0 = \mathscr{F}_u - \frac{\partial}{\partial x}\mathscr{F}_{u_x} - \frac{\partial}{\partial y}\mathscr{F}_{u_y}, \tag{2.11}$$

$$0 = \mathscr{F}_v - \frac{\partial}{\partial x}\mathscr{F}_{v_x} - \frac{\partial}{\partial y}\mathscr{F}_{v_y}. \tag{2.12}$$

The partial derivatives can be computed as

$$\begin{aligned} \mathscr{F}_u &= 2\cdot I_x\cdot\left(I_x u + I_y v + I_t\right), & \tfrac{\partial}{\partial x}\mathscr{F}_{u_x} &= \alpha 2u_{xx}, & \tfrac{\partial}{\partial y}\mathscr{F}_{u_y} &= \alpha 2u_{yy}, \\ \mathscr{F}_v &= 2\cdot I_y\cdot\left(I_x u + I_y v + I_t\right), & \tfrac{\partial}{\partial x}\mathscr{F}_{v_x} &= \alpha 2v_{xx}, & \tfrac{\partial}{\partial y}\mathscr{F}_{v_y} &= \alpha 2v_{yy}. \end{aligned}$$

Plugging in these expressions and dividing by 2, the final system of equations reads

$$0 = I_x\cdot\left(I_x u + I_y v + I_t\right) - \alpha\Delta u, \tag{2.13}$$

$$0 = I_y\cdot\left(I_x u + I_y v + I_t\right) - \alpha\Delta v, \tag{2.14}$$

with reflecting Neumann boundary conditions $\mathbf{n}^\top\nabla u = 0$ and $\mathbf{n}^\top\nabla v = 0$ (where $\mathbf{n}$ is an outer normal vector pointing across the image boundary) and $\Delta u = u_{xx} + u_{yy}$ being the standard Laplace-operator.

2.5.4 Motion Tensor Notation

Any kind of quadratic data term based on linear or linearized constancy assumptions for the optical flow $\mathbf{w}$ can be written as $(\mathbf{w}^\top \mathbf{p})^2$ where $\mathbf{p}$ defines the constraint on $\mathbf{w}$. We can derive the so-called motion tensor J [30, 47] from this formulation via:

$$\begin{aligned} & (\mathbf{w}^\top \mathbf{p})^2 && = 0 \\ \Leftrightarrow \quad & (\mathbf{w}^\top \mathbf{p})(\mathbf{w}^\top \mathbf{p})^\top && = 0 \\ \Leftrightarrow \quad & \mathbf{w}^\top \underbrace{\mathbf{p}\,\mathbf{p}^\top}_{=:J} \mathbf{w} && = 0 \\ \Leftrightarrow \quad & \mathbf{w}^\top J\, \mathbf{w} && = 0\,. \end{aligned} \tag{2.15}$$

For the example of the linearized BCA, we obtain $\mathbf{p}_{\mathrm{BCA}} = (I_x, I_y, I_t)^\top$ as the generating or constraint vector (see Eq. 2.8) such that the motion tensor J_{BCA} reads:

$$\begin{aligned} J_{\mathrm{BCA}} &= \mathbf{p}_{\mathrm{BCA}}\, \mathbf{p}_{\mathrm{BCA}}^\top \\ &= (I_x, I_y, I_t)^\top (I_x, I_y, I_t) \\ &= \begin{pmatrix} I_x I_x & I_x I_y & I_x I_t \\ I_y I_x & I_y I_y & I_y I_t \\ I_t I_x & I_t I_y & I_t I_t \end{pmatrix}. \end{aligned} \tag{2.16}$$

Rewriting Equations. Using this notation, we can rewrite the partial derivatives $\mathscr{F}_u$ and $\mathscr{F}_v$ of the Euler-Lagrange equations. To generalize things, we identify $w_1 := u$ and $w_2 := v$ and denote the j-th row of the motion tensor J as $J_{(j)}$. The respective partial derivative $\mathscr{F}_{w_j}$, where j is the index of an unknown of the variational model, is then given by

$$\begin{aligned} \mathscr{F}_{w_j} &= 2 \cdot J_{(j)}\, \mathbf{w} \\ &= 2 \cdot p_j \cdot \mathbf{p}^\top \mathbf{w} \end{aligned} \tag{2.17}$$

with $\mathbf{p} = (p_1, p_2, p_3)^\top$.

Basis of a General Framework. This notation allows for a general framework that is able to express a lot of pointwise constraints comprising constancy assumptions based on higher order features (see Sect. 2.6.2), similarity terms (see Sect. 2.10.1) or trajectorial regularizers (see Chapter 7, Sect. 7.5).

2.6 The Method of Brox et al.

The method of Horn and Schunck has been a pioneering work in variational optical flow estimation. However, it has some drawbacks: It is vulnerable to outliers in the data and

the solution due to the quadratic terms, it is only able to estimate small displacements due to the linearized data term, and it is not robust against illumination changes between both frames since the BCA cannot match correspondences with different brightness levels. The method of Brox *et al.* [26] addresses these issues and introduces concepts to overcome these drawbacks: It uses sub-quadratic penalizer functions to add robustness against outliers, it adds some degree of illumination invariance due to using an advanced data constancy assumption, and it allows to estimate large displacements due to the usage of a data term without linearization. Based on the method of Horn and Schunck, we will discuss these concepts in the following independently from each other.

2.6.1 Sub-Quadratic Penalization

In a global variational optical flow method, there are at least two terms, i.e. the data term and the smoothness term, that steer the final solution.

Outliers. In the image domain there are always locations where the correct solution cannot fulfill the assumptions behind both terms. This includes intentional motion discontinuities that contradict the smoothness assumption as well as noisy data or occlusions where the correct solution contradicts the data constancy assumption(s). By using quadratic terms, the estimated solution tends to be a trade-off and thus violating both assumptions to some comparably small extent instead of putting most trust in the locally more appropriate term and violating the other one to a potentially greater extent.

Introducing Penalizer Functions. Let us consider the variational model of Horn and Schunck to be equipped with a quadratic penalizer function $\Psi(s^2) = s^2$ around both terms, then we can rewrite Eq. 2.9 as

$$E(\mathbf{w}) \quad = \quad \int_\Omega \Psi\left(\left(I_x u + I_y v + I_t\right)^2\right) + \alpha \Psi\left(|\nabla u|^2 + |\nabla v|^2\right) \, d\tilde{\mathbf{x}} \,. \tag{2.18}$$

Now, we are able to replace this penalizer function Ψ by a sub-quadratic differentiable counterpart that locally allows for higher deviations without affecting the overall energy too much. In the minimization using the Euler-Lagrange equations, the expressions that are associated to each of the terms are multiplied with the outer derivative $\Psi' = \frac{\partial}{\partial s^2}\Psi(s^2)$ which is a decreasing function if Ψ is sub-quadratic. For local deviations in one of the assumptions, which lead to large arguments of both Ψ and Ψ', this comes down to a local downweighting of the respective assumption. Still, such a sub-quadratic penalizer function should have some particular properties: it should be positive, increasing in the argument s^2 and strictly convex to allow for a unique solution [152]. However, we will later refrain to some extent from the last requirement when introducing advanced smoothness terms.

Subquadratic Penalizers. A typical choice for such a sub-quadratic penalizer function is the (regularized) absolute value function $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ [26], where $\epsilon > 0$ ensures differentiability at $s^2 = 0$, or its weighted equivalent, the Charbonnier regularizer $\Psi(s^2) = 2\epsilon^2\sqrt{1 + s^2/\epsilon^2}$ [33]. When such a penalizer function is applied to the data term, it adds robustness against noise and occlusions [18, 19], whereas in the case of the smoothness term, such a function allows for motion discontinuities, i.e. it makes the smoothness term *discontinuity-preserving* and the results become piecewise smooth [97, 117, 42, 2, 152].

2.6.2 Robustness against Illumination Changes

The drawback of the brightness constancy assumption (BCA) is the fact that it is not robust against illumination changes. A constancy assumption can be made robust against some type of illumination changes if the information that encodes this type of changes can be discarded. We can consider the brightness just as a feature of a pixel in the image and the BCA as an instance of a feature constancy assumption. A first step towards illumination invariance is achieved by using gradient information as a feature in the constancy assumption [137, 142, 136, 117]. This way the data term becomes robust against additive illumination changes. In contrast to the brightness, which is a scalar feature, the gradient is vector-valued feature consisting of the x-derivative and the y-derivative. Hence, the *gradient constancy assumption (GCA)* provides two constraints:

$$I_x(\mathbf{x}) - I_x(\mathbf{x} + \mathbf{w}) = 0\,, \tag{2.19}$$
$$I_y(\mathbf{x}) - I_y(\mathbf{x} + \mathbf{w}) = 0\,. \tag{2.20}$$

Similar to the BCA, we can also linearize these constraints which reads

$$I_{xx}u + I_{xy}v + I_{xt} = 0\,, \tag{2.21}$$
$$I_{yx}u + I_{yy}v + I_{yt} = 0\,. \tag{2.22}$$

Motion Tensor Notation

From this linearization, we obtain the generating vectors of the motion tensor for each constraint as $\mathbf{p}_{\mathrm{GCA},x} = (I_{xx}, I_{xy}, I_{xt})^\top$ and $\mathbf{p}_{\mathrm{GCA},y} = (I_{yx}, I_{yy}, I_{yt})^\top$, respectively. The motion tensor of a vector-valued constraint is given by the sum of the motion tensors generated by the vectors $\mathbf{p}$ of each constraint. In this case, this reads

$$\begin{aligned} J_{\mathrm{GCA}} &= J_{\mathrm{GCA},x} + J_{\mathrm{GCA},y} \\ &= \mathbf{p}_{\mathrm{GCA},x}\,\mathbf{p}_{\mathrm{GCA},x}^\top + \mathbf{p}_{\mathrm{GCA},y}\,\mathbf{p}_{\mathrm{GCA},y}^\top \end{aligned}$$

$$
\begin{aligned}
&= \begin{pmatrix} I_{xx}I_{xx} & I_{xx}I_{xy} & I_{xx}I_{xt} \\ I_{xy}I_{xx} & I_{xy}I_{xy} & I_{xy}I_{xt} \\ I_{xt}I_{xx} & I_{xt}I_{xy} & I_{xt}I_{xt} \end{pmatrix} + \begin{pmatrix} I_{yx}I_{yx} & I_{yx}I_{yy} & I_{yx}I_{yt} \\ I_{yy}I_{yx} & I_{yy}I_{yy} & I_{yy}I_{yt} \\ I_{yt}I_{yx} & I_{yt}I_{yy} & I_{yt}I_{yt} \end{pmatrix} \\
&= \begin{pmatrix} I_{xx}I_{xx} + I_{yx}I_{yx} & I_{xx}I_{xy} + I_{yx}I_{yy} & I_{xx}I_{xt} + I_{yx}I_{yt} \\ I_{xy}I_{xx} + I_{yy}I_{yx} & I_{xy}I_{xy} + I_{yy}I_{yy} & I_{xy}I_{xt} + I_{yy}I_{yt} \\ I_{xt}I_{xx} + I_{yt}I_{yx} & I_{xt}I_{xy} + I_{yt}I_{yy} & I_{xt}I_{xt} + I_{yt}I_{yt} \end{pmatrix}. \qquad (2.23)
\end{aligned}
$$

Both tensors J_{BCA} and J_{GCA} are then combined in a weighted sum, such that the linearized data term reads

$$
E_{\mathrm{Data}}(\mathbf{w}) = \int_{\Omega} \mathbf{w}^{\top} \left(J_{\mathrm{BCA}} + \gamma J_{\mathrm{GCA}} \right) \mathbf{w} \, d\tilde{\mathbf{x}} \,. \qquad (2.24)
$$

This data term keeps the full information of the given data via the BCA and adds robustness against illumination changes via the GCA. Depending on the context, the weight γ can be adjusted to determine the balance between both aspects.

2.6.3 Large Displacements

So far, we have only considered data constraints in their linearized version, since the linearization makes the unknowns explicit. The linearization, however, is only valid for small displacements, since it is a local approximation of the original problem which usually is not linear. An alternative approach is to keep the original assumptions without linearizations [97, 74, 3] and to postpone the linearization step to the numerical scheme. In contrast to the functional with linearized data constancy assumptions, which is convex, the original one is non-convex and has multiple local minima instead of a global one.

Coarse-to-fine Schemes. In order to find a global or at least a good local minimum of a minimization problem, coarse-to-fine schemes have become popular [19, 91]. They iteratively solve downsampled versions of the problem starting from a coarse resolution up to the original resolution and use solutions from coarser resolutions as initializations for the solutions on the finer levels. In the context of motion estimation, one can observe that not only the sizes of objects become smaller on coarser resolutions but also their displacements. Hence, for each displacement scale, there will eventually be a resolution level where the displacement is shrinked to an order of magnitude where the linearization is valid. Having this in mind, it is obvious that large displacements can be estimated by a linearization on the respective coarser resolution level where the corresponding displacement becomes small. Doing this iteratively on multiple resolution levels allows to estimate displacements of arbitrary scales.

Coarse-to-fine Warping. Hence, we can apply a coarse-to-fine warping strategy [15] which is a particular instance of an incremental coarse-to-fine fixed point iteration:

Starting with a small displacement estimation on the coarsest possible resolution with a linearized version of the functional, we upsample the current optical flow to the next finer level and compensate the second frame for it, i.e. we eliminate the large displacements that have been estimated on the coarser level (the warping step). This way, only small displacements remain that can again be estimated with a linearized version of the functional in a fixed point iteration. These small displacements are then an incremental solution to the overall problem and added to the upsampled initial solution (i.e. we conduct an incremental computation). The summed up solution is then transferred to the next level where again the incremental problem is to be solved.

Mathematical Derivation

Let us now introduce the mathematical concepts behind this strategy. Given the following non-linear functional

$$E(\mathbf{w}) \quad = \quad \int_{\Omega} (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x}))^2 + \alpha \left(|\nabla u|^2 + |\nabla v|^2 \right) \, d\tilde{\mathbf{x}} \,, \tag{2.25}$$

the corresponding Euler-Lagrange equations read

$$I_x(\mathbf{x}+\mathbf{w}) \, (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})) + \alpha \Delta u \quad = \quad 0 \,, \tag{2.26}$$

$$I_y(\mathbf{x}+\mathbf{w}) \, (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})) + \alpha \Delta v \quad = \quad 0 \,, \tag{2.27}$$

with boundary conditions $\mathbf{n}^\top \nabla u = 0$ and $\mathbf{n}^\top \nabla v = 0$.

Fixed Point Iteration. We can now introduce a fixed point iteration by considering $\mathbf{w}$ at different time steps k and $k+1$ and modifying the equations accordingly:

$$I_x(\mathbf{x}+\mathbf{w}^k) \left(I(\mathbf{x}+\mathbf{w}^{k+1}) - I(\mathbf{x}) \right) - \alpha \Delta u^{k+1} \quad = \quad 0 \,, \tag{2.28}$$

$$I_y(\mathbf{x}+\mathbf{w}^k) \left(I(\mathbf{x}+\mathbf{w}^{k+1}) - I(\mathbf{x}) \right) - \alpha \Delta v^{k+1} \quad = \quad 0 \,. \tag{2.29}$$

Incremental Formulation. Here, we can embed the incremental computation by splitting the unknown $\mathbf{w}^{k+1}$ into a known part $\mathbf{w}^k$ – which is given by upsampling the solution from a coarser level or by the initialization $\mathbf{w}^0 = (0,0,1)^\top$ on the coarsest level – and an unknown increment $\mathbf{dw}^k = (du^k, dv^k, 0)^\top$ from the new time step via

$$\underbrace{\mathbf{w}^{k+1}}_{\text{final solution}} \quad = \quad \underbrace{\mathbf{w}^k}_{\text{upsampled solution}} \quad + \quad \underbrace{\mathbf{dw}^k}_{\text{unknown increment}} \,. \tag{2.30}$$

Hence, we rewrite Eqs. 2.28 and 2.29 as

$$I_x(\mathbf{x}+\mathbf{w}^k) \left(I(\mathbf{x}+\mathbf{w}^k+\mathbf{dw}^k) - I(\mathbf{x}) \right) - \alpha \Delta u^{k+1} \quad = \quad 0 \,, \tag{2.31}$$

$$I_y(\mathbf{x}+\mathbf{w}^k) \left(I(\mathbf{x}+\mathbf{w}^k+\mathbf{dw}^k) - I(\mathbf{x}) \right) - \alpha \Delta v^{k+1} \quad = \quad 0 \,. \tag{2.32}$$

Postponed Linearization. Now, we can linearize the data term with respect to $\mathbf{dw}^k$ leaving the known large displacement part $\mathbf{w}^k$ non-linearized and making the small displacement increment $\mathbf{dw}^k$ explicit. The linearized version of the equations reads

$$0 = I_x(\mathbf{x}+\mathbf{w}^k)\left(\left(\nabla_3 I(\mathbf{x}+\mathbf{w}^k)\right)^\top \mathbf{dw}^k + \underbrace{I(\mathbf{x}+\mathbf{w}^k)-I(\mathbf{x})}_{\approx f_t}\right) - \alpha\Delta u^{k+1}\,, \quad (2.33)$$

$$0 = I_y(\mathbf{x}+\mathbf{w}^k)\left(\left(\nabla_3 I(\mathbf{x}+\mathbf{w}^k)\right)^\top \mathbf{dw}^k + \underbrace{I(\mathbf{x}+\mathbf{w}^k)-I(\mathbf{x})}_{\approx f_t}\right) - \alpha\Delta v^{k+1}\,. \quad (2.34)$$

where the spatio-temporal gradient is defined as $\nabla_3 := (\partial_x, \partial_y, \partial_t)^\top$.

Motion Tensor Notation. Also in this postponed linearization – which for a coarse-to-fine warping scheme with only one resolution level comes down to the equations for the linearized model – the motion tensor notation is applicable. However, we have a different motion tensor J^k for each resolution level. For convenience reasons, let us re-define $\mathbf{dw}^k = (du^k, dv^k, \underline{1})^\top$. The re-formulated equations are then given by

$$0 = J^k_{(1)}\mathbf{dw}^k - \alpha\Delta u^{k+1}\,, \quad (2.35)$$

$$0 = J^k_{(2)}\mathbf{dw}^k - \alpha\Delta v^{k+1}\,, \quad (2.36)$$

where $J_{(i)}$ denotes the i-th row of the motion tensor J. The motion tensor J^k can be obtained from any linearized constancy assumption or be constituted as the sum of multiple motion tensors for different linearized constancy assumptions.

Overall Minimization Strategy

Similar to the case of the linearized version of the energy functional, the linearized sub-problem within the coarse-to-fine warping scheme is convex and thus has a unique minimizer. Hence, the non-convex minimization problem is approximated by a series of convex sub-problems.

If the variational model contains sub-quadratic penalizer functions Ψ_D in the data term and Ψ_S the smoothness term, the equations on some coarse-to-fine level k read

$$\begin{aligned} 0 = {} & \Psi'_D(\mathbf{dw}^{k\top} J^k \mathbf{dw}^k)\cdot(J^k_{(1)}\mathbf{dw}^k) \\ & -\alpha\,\mathrm{div}\left(\Psi'_S(|\nabla(u^k+du^k)|^2+|\nabla(v^k+dv^k)|^2)(u^k+du^k)\right)\,, \quad (2.37) \\ 0 = {} & \Psi'_D(\mathbf{dw}^{k\top} J^k \mathbf{dw}^k)\cdot(J^k_{(2)}\mathbf{dw}^k) \\ & -\alpha\,\mathrm{div}\left(\Psi'_S(|\nabla(u^k+du^k)|^2+|\nabla(v^k+dv^k)|^2)(v^k+dv^k)\right)\,. \quad (2.38) \end{aligned}$$

Here, there are two fixed point iterations which are applied in a nested way:

1. In order to get rid of the non-convexity of the original functional, the coarse-to-fine warping strategy is applied, where the non-convex problem is solved as a series of convex sub-problems.

2. In order to get rid of the potentially non-linear terms in the resulting convex sub-problems, the lagged nonlinearity method is applied, where nonlinear sub-problems are solved as series of linear sub-problems.

Finally, there are series of linear sub-problems which can be solved using iterative solvers for linear equation systems.

Please note that coarse-to-fine warping fails in cases where a small object is not distinctive on the coarse-to-fine level that is appropriate to determine its displacement. This case covers relative large displacements as defined in 1.3.

2.6.4 Final Model

By integrating all the improvements discussed so far, we obtain the following final variational model for the method of Brox *et al.*:

$$\begin{aligned} E_{\text{Brox}}(\mathbf{w}) \quad = \quad & \int_{\Omega} \Psi_D\left(|I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})|^2 + \gamma|\nabla I(\mathbf{x}+\mathbf{w}) - \nabla I(\mathbf{x})|^2\right) \\ & + \alpha\, \Psi_S\left(|\nabla_2 u|^2 + |\nabla_2 v|^2\right)\, d\tilde{\mathbf{x}}\,, \end{aligned} \tag{2.39}$$

where $\Psi_D(s^2) = \Psi_S(s^2) = \sqrt{s^2+\epsilon^2}$ is the (regularized) absolute value function and γ and α are balancing weights. This type of sub-quadratic smoothness term is also called *Total Variation (TV)* [163]. Please note that we omit the spatio-temporal smoothness assumption that has also been proposed in [26], since it may only improve results for rotational or divergent motions (when using more than two frames). In other cases, however, it is not appropriate. Moreover, since the two-frame case is prevalent in this thesis, we resort to a purely spatial regularization strategy.

2.7 The Method of Bruhn and Weickert

The variational model of the method of Bruhn and Weickert [29] is based on that of Brox *et al.* [26]. However, there is a difference when it comes to the penalization strategy of the involved data term. As we have seen so far, it is beneficial to separately apply penalization functions on both the data term and the smoothness term as there are locations where one of the terms is fulfilled while the other is not. This way the term that is fulfilled is still actively steering the estimation while the influence of the other is reduced during the minimization. The same argumentation, however, also is valid for a data term that contains more than one constancy assumption.

Outliers in the Data Constraints. Let us explain this at hand of the brightness constancy assumption (BCA) and the gradient constancy assumption (GCA). For each of both assumptions, there are cases where only the respective data term is valid. In the case of additive illumination changes, the GCA is fulfilled while the BCA obviously is not valid. On the other hand, if an object moves in front of a changing background, the corresponding gradients between the object's boundaries and the background are not consistent. Hence, the GCA is not valid while the BCA may be fulfilled.

2.7.1 Final Model

The variational model of Bruhn and Weickert, hence, is a slight variation of the model of Brox *et al.* where the sub-quadratic penalizer function Ψ is applied separately to each data constraint. It is given by

$$\begin{aligned} E_{\mathrm{BW}}(\mathbf{w}) \quad = \quad & \int_{\Omega} \Psi_D\left(|I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})|^2\right) + \gamma \Psi_D\left(|\nabla I(\mathbf{x}+\mathbf{w}) - \nabla I(\mathbf{x})|^2\right) \\ & + \alpha\, \Psi_S\left(|\nabla u|^2 + |\nabla v|^2\right)\, d\tilde{\mathbf{x}}\,, \end{aligned} \tag{2.40}$$

where $\Psi_D(s^2) = \Psi_S(s^2) = \sqrt{s^2+\epsilon^2}$ is the (regularized) absolute value function, and γ and α are balancing weights.

2.8 The Method of Zimmer et al.

While the variational methods of Brox *et al.* [26] and Bruhn and Weickert [29] so far introduced concepts that lead to considerable improvements over the pioneering work of Horn and Schunck, still there are some important aspects which have been neglected so far. This starts with the obvious fact that the presented methods have been tailored to use grey value images, disregarding any type of color information. Moreover, the data term, which eventually is linearized within the minimization, furthermore contains implicit weightings of the data constraints which depend on the image contrast. Hence, at objects of high contrast the data term has more influence than at objects of low contrast. Finally, the smoothness term provides a constraint on the amount of smoothing but not on its direction. It is, however, desirable to only reduce local smoothing across motion discontinuities, which are a subset of the image edges, but not along them. Including these aspects into the modeling allows for a far more adaptive variational optical flow model, as proposed by Zimmer *et al.* [165, 164].

2.8.1 Constraint Normalization

Let us start by discussing a strategy to overcome the implicit weightings in the data constraints, which is called constraint normalization. The motivation behind constraint

normalization origins from a weighting deficiency in the optical flow constraint (OFC) which as a reminder reads

$$0 \quad = \quad I_x u + I_y v + I_t \,. \tag{2.41}$$

This single constraint is not sufficient to compute both components u and v of the optical flow, but it provides a line constraint on the optical flow. Given that $|\nabla I| > 0$, the OFC allows to compute the share of the flow that is orthogonal to the local image edges which is known as the *normal flow* $\mathbf{w}_n = (\tilde{\mathbf{w}}_n^\top, 1)^\top = (u_n, v_n, 1)^\top$ [16]. It is the solution with the smallest L_2-norm to the OFC and reads

$$\tilde{\mathbf{w}}_n \quad := \quad \frac{-\nabla I\, I_t}{|\nabla I|^2} \,. \tag{2.42}$$

Let $\tilde{\mathbf{w}} = (u, v)^\top$ be the spatial share of the optical flow, then we can reformulate the squared right side of the OFC as

$$\begin{aligned}
\left(I_x u + I_y v + I_t\right)^2 &= (\nabla I^\top \tilde{\mathbf{w}} + I_t)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top \tilde{\mathbf{w}} + I_t}{|\nabla I|}\right)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top \tilde{\mathbf{w}}}{|\nabla I|} + \frac{I_t}{|\nabla I|}\right)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top \tilde{\mathbf{w}}}{|\nabla I|} + \frac{\nabla I^\top \nabla I}{|\nabla I|^2}\frac{I_t}{|\nabla I|}\right)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top}{|\nabla I|}\tilde{\mathbf{w}} + \frac{\nabla I^\top}{|\nabla I|}\frac{\nabla I\, I_t}{|\nabla I|^2}\right)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top}{|\nabla I|}\left(\tilde{\mathbf{w}} + \frac{\nabla I\, I_t}{|\nabla I|^2}\right)\right)^2 \\
&= |\nabla I|^2 \left(\frac{\nabla I^\top}{|\nabla I|}\left(\tilde{\mathbf{w}} - \frac{-\nabla I\, I_t}{|\nabla I|^2}\right)\right)^2 \\
&= |\nabla I|^2 \left(\underbrace{\frac{\nabla I^\top}{|\nabla I|}\left(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}_n\right)}_{=:d}\right)^2 \,.
\end{aligned} \tag{2.43}$$

The expression that is denoted by d is a normal form of the line l that is given by the OFC (with normal $\frac{\nabla I^\top}{|\nabla I|}$ and position vector $\tilde{\mathbf{w}}_n$). For any $\tilde{\mathbf{w}}$, the absolute value of the projection of its difference to a point on the line (here given by $\tilde{\mathbf{w}}_n$) onto the normal of

l provides the distance of $\tilde{\mathbf{w}}$ and l. Hence, $|d|$ is a distance measure. The OFC hence is a distance term d^2 weighted by $|\nabla I|^2$. This means that the data term is amplified in high-gradient regions and suppressed in low-gradient regions. This is not intended, since, on the one hand, there may be low-contrast boundaries of moving objects which are supposed to introduce a motion discontinuity and, on the other hand, there are large gradients induced by noise or in occluded regions where a data term should not be amplified.

Hence, Lai and Vemuri [77] proposed a constraint normalization, i.e. to divide the constraint by the unwanted weight, which introduces a normalization weight

$$\theta_{\mathrm{BCA}} \quad := \quad \frac{1}{|\nabla I|^2 + \epsilon_{\mathrm{cNorm}}^2}\,, \tag{2.44}$$

where $\epsilon_{\mathrm{cNorm}} > 0$ avoids divisions by zero and prevents small gradients from being too influential. Such gradients may e.g. come from noise in flat regions, where the data term should not be amplified. In the motion tensor notation, the normalization reads

$$\bar{J}_{\mathrm{BCA}} \quad := \quad \theta_{\mathrm{BCA}} \cdot J_{\mathrm{BCA}}\,. \tag{2.45}$$

Normalization of General Data Constraints

In the literature, there are also similar estimation problems with linear constraints of higher dimensionality. These comprise the estimation of scene flow [143], the estimation of multiframe optical flow [148, 130], the joint estimation of optical flow and illumination changes [41] as well as the simultaneous estimation of multiple optical flows [127]. Hence, let us consider a general linear constraint with N-dimensional generating vector $\mathbf{p} = (p_i)_{1\le i\le N}$. Then we set $\tilde{\mathbf{p}} = (p_i)_{1\le i\le N-1}$ as the vector with the first $N-1$ components of $\mathbf{p}$ and a flow $\tilde{\mathbf{w}}$ with the corresponding $N-1$ sought functions. The linear constraint then can be written as

$$0 \quad = \quad \tilde{\mathbf{p}}^\top \tilde{\mathbf{w}} + p_N\,. \tag{2.46}$$

The squared right side can then analogously be re-formulated as

$$\begin{aligned}
(\tilde{\mathbf{p}}^\top \tilde{\mathbf{w}} + p_N)^2 &= |\tilde{\mathbf{p}}|^2 \left(\frac{\tilde{\mathbf{p}}^\top \tilde{\mathbf{w}} + p_N}{|\tilde{\mathbf{p}}|} \right)^2 \\
&= |\tilde{\mathbf{p}}|^2 \left(\frac{\tilde{\mathbf{p}}^\top \tilde{\mathbf{w}}}{|\tilde{\mathbf{p}}|} + \frac{p_N}{|\tilde{\mathbf{p}}|} \right)^2 \\
&= |\tilde{\mathbf{p}}|^2 \left(\frac{\tilde{\mathbf{p}}^\top \tilde{\mathbf{w}}}{|\tilde{\mathbf{p}}|} + \frac{\tilde{\mathbf{p}}^\top \tilde{\mathbf{p}}}{|\tilde{\mathbf{p}}|^2} \frac{p_N}{|\tilde{\mathbf{p}}|} \right)^2 \\
&= |\tilde{\mathbf{p}}|^2 \left(\frac{\tilde{\mathbf{p}}^\top \tilde{\mathbf{w}}}{|\tilde{\mathbf{p}}|} + \frac{\tilde{\mathbf{p}}^\top}{|\tilde{\mathbf{p}}|} \frac{\tilde{\mathbf{p}}}{|\tilde{\mathbf{p}}|^2} p_N \right)^2
\end{aligned}$$

$$
\begin{aligned}
&= |\tilde{\mathbf{p}}|^2 \left(\frac{\tilde{\mathbf{p}}^\top}{|\tilde{\mathbf{p}}|} \left(\tilde{\mathbf{w}} + \frac{\tilde{\mathbf{p}}\, p_N}{|\tilde{\mathbf{p}}|^2} \right) \right)^2 \\
&= |\tilde{\mathbf{p}}|^2 \left(\underbrace{\frac{\tilde{\mathbf{p}}^\top}{|\tilde{\mathbf{p}}|} (\tilde{\mathbf{w}} - \tilde{\mathbf{w}}_n)}_{=:d} \right)^2 , \qquad (2.47)
\end{aligned}
$$

where $\tilde{\mathbf{w}}_n := \frac{-\tilde{\mathbf{p}}\, p_N}{|\tilde{\mathbf{p}}|^2}$ is a point on the hyperplane defined by Eq. 2.46. Here, d is the normal form of the corresponding hyperplane which analogously to the case of the OFC provides a distance measure for a point $\tilde{\mathbf{w}}$ to the hyperplane. This holds for any linearized constraint with generating vector $\mathbf{p}$ of any dimensionality. The corresponding normalization factor is given by

$$
\theta := \frac{1}{|\tilde{\mathbf{p}}|^2 + \epsilon_{\text{cNorm}}^2} , \qquad (2.48)
$$

where $\epsilon_{\text{cNorm}} > 0$. Please note that the normalization factor is applied for each generating vector $\mathbf{p}$ separately. Hence, if a normalized motion tensor is assembled of the N_p normalized constraints, it is given by

$$
\bar{J} := \sum_{i=1}^{N_\text{p}} \theta_i \cdot J_i = \sum_{i=1}^{N_\text{p}} \frac{1}{\text{tr}((J_i)_N) + \epsilon_{\text{cNorm}}^2} \cdot J_i , \qquad (2.49)
$$

where $(J_i)_N$ is the subtensor excluding the N-th row and the N-th column, i.e. consisting of the first $N-1$ columns and $N-1$ rows of the motion tensor J_i associated to the generating vector $\mathbf{p}_i$, and tr(A) denotes the trace of a matrix A.

Since $\mathbf{p}$ can define any linear constraint, this general motivation in particular also holds for the two constraints $\mathbf{p}_{\text{GCA,x}}$ and $\mathbf{p}_{\text{GCA},y}$ that are obtained from the linearized gradient constancy assumption.

2.8.2 Color Image Sequences

Using color images instead of grey value images allows us to consider more constraints that connect the image data and the optical flow [101]. Each image channel I^c out of the N_c color channels provides one constraint $\mathbf{p}^c$ for each type of feature constraint, i.e. we have three constraints for each pixel in the BCA and six constraints for each pixel in the GCA.

Penalization Strategy. Considering the penalization strategy on these constraints, there are different possibilities which may in particular depend on the color space that is considered. In the case of an RGB color space, usually a joint penalization of the

color channels is used, since these channels are of the same type. A different example is the HSV color space where the channels have different properties w.r.t. the degree of illumination invariance [54, 94]. In this case, a separate robustification makes sense, since in the context of illumination changes the respective constancy assumption may be fulfilled for a more invariant channel but not for a less invariant channel.

Final Data Term. While the original work of Zimmer *et al.* [164] focuses on the HSV color space, we will focus on the RGB color space and thus conduct a joint penalization of the color channels. Hence, the overall data term is assembled by summing up the data terms for each image channel within a joint penalization and it reads

$$\begin{aligned} E_{\text{Data}} \quad = \quad & \int_{\Omega} \quad \Psi\left(\sum_{c=1}^{N_{\text{c}}} \left(I^c(\mathbf{x}+\mathbf{w}) - I^c(\mathbf{x})\right)^2\right) \\ & + \gamma\, \Psi\left(\sum_{c=1}^{N_{\text{c}}} |\nabla I^c(\mathbf{x}+\mathbf{w}) - \nabla I^c(\mathbf{x})|^2\right) d\tilde{\mathbf{x}}\,. \end{aligned} \tag{2.50}$$

Similarly, the associated motion tensor including the normalization factors is given by

$$\bar{J} \quad := \quad \sum_{i=1}^{N_{\text{p}}} \sum_{c=1}^{N_{\text{c}}} \frac{1}{|\tilde{\mathbf{p}}_i^c|^2 + \epsilon_{\text{cNorm}}^2} \cdot J_i^c \quad = \quad \sum_{i=1}^{N_{\text{p}}} \sum_{c=1}^{N_{\text{c}}} \frac{1}{\operatorname{tr}((J_i^c)_N) + \epsilon_{\text{cNorm}}^2} \cdot J_i^c\,. \tag{2.51}$$

2.8.3 Anisotropic Smoothness Term

So far, the smoothness term lead to an equal smoothing in all directions, i.e. the flow is also smoothed across edges in the solution which leads to unsharp flow fields and inaccuracies at motion discontinuities. It is, however, desirable to reduce smoothing across edges while keeping it along edges and, thus, to consider directions in the smoothness terms.

In order to develop a regularization strategy that respects motion discontinuities, let us start by reviewing the smoothness term of the method of Brox *et al.* [26] which reads

$$E_{\text{Smooth}}(\mathbf{w}) \quad = \quad \int_{\Omega} \Psi\left(|\nabla u|^2 + |\nabla v|^2\right) d\tilde{\mathbf{x}}\,, \tag{2.52}$$

where $\Psi(s^2)$ is a subquadratic (and convex) penalizer function in s.

The respective Euler-Lagrange equations, where we omit the explicit statement of a particular data term for the sake of simplicity, are given by

$$0 \quad = \quad \mathscr{F}_u - \alpha \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right) \nabla u\right)\,, \tag{2.53}$$

$$0 \quad = \quad \mathscr{F}_v - \alpha \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right) \nabla v\right)\,, \tag{2.54}$$

and can be considered the steady state of the following diffusion-reaction system [152] which reads

$$\partial_t u = \mathscr{F}_u - \alpha \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right)\nabla u\right), \tag{2.55}$$
$$\partial_t v = \mathscr{F}_v - \alpha \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right)\nabla v\right). \tag{2.56}$$

Underlying Diffusion Process. The smoothing within the variational method is done in the diffusion part of the above system which reads

$$\partial_t u = \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right)\nabla u\right), \tag{2.57}$$
$$\partial_t v = \operatorname{div}\left(\Psi'\left(|\nabla u|^2 + |\nabla v|^2\right)\nabla v\right). \tag{2.58}$$

In the non-robust case of the model of Horn and Schunck (where $\Psi'(s^2) = 1$), the corresponding diffusion process falls down to simple homogeneous linear diffusion:

$$\partial_t u = \operatorname{div}(1 \cdot \nabla u) = \Delta u, \tag{2.59}$$
$$\partial_t v = \operatorname{div}(1 \cdot \nabla v) = \Delta v. \tag{2.60}$$

In the general case, it makes sense to classify smoothness terms at hand of the corresponding diffusion processes. Isotropic variants of diffusion processes for some sought function $w_i \in \{u, v\}$ are given by

$$\partial_t w_i = \operatorname{div}\left(g \cdot \nabla w_i\right), \tag{2.61}$$

with g being the scalar diffusivity. There are different possibilities for the choice of g.

Homogeneous Diffusion

The simplest choice $g := 1$ leads to homogeneous (i.e. space-independent) and isotropic (i.e. direction-independent) diffusion [68]. Since this diffusion does not respect any edges, neither from image data nor from the estimated optical flow, edges are oversmoothed and results are overall rather blurry.

Image-Driven Isotropic Diffusion

Since we know that the image edges, which consist of structural edges and textural edges, are a superset of the motion discontinuities, one strategy to avoid an oversmoothing is to reduce smoothing at image edges, i.e. to make the diffusivity depend on the magnitude of the image gradient $|\nabla I|$. One possible choice for the scalar diffusivity is $g := g(|\nabla I|^2)$ with $g(s^2)$ being a positive and decreasing function [2]. Such functions are usually given as the derivatives of a strictly convex function $\Psi(s^2)$. These types

of diffusivities are called *image-driven isotropic*. They are inhomogeneous, since the diffusion is space-dependent. Corresponding smoothness terms are simply multiplied by $g(|\nabla I|^2)$. Since g does not depend on $\mathbf{w}$, it is not affected by the derivations in the Euler-Lagrange equations. Since, however, $g := g(|\nabla I|^2)$ not only reduces smoothing at structural edges but also at textural edges, the resulting flow field is likely to be oversegmented in highly-textured regions.

Flow-Driven Isotropic Diffusion

In order to avoid oversegmentation artifacts, g can be chosen as $g := \Psi'(\sum_{j=1}^{2} |\nabla w_j|^2)$, with Ψ' being positive and decreasing, which is the natural result when applying a sub-quadratic penalizer function Ψ to the otherwise quadratic smoothness term [122, 117, 152] (as already presented for the method of Brox *et al.* [26]). Again, this diffusion is inhomogeneous since it is reduced at flow edges. In contrast to image-driven diffusion, this type of diffusion is nonlinear, since g depends on the sought functions $\mathbf{w}$. The corresponding diffusivities are called *flow-driven isotropic*. They do not show over-segmentation artifacts like in the image-driven case, but they are still isotropic and thus lead to the same diffusion across edges and along edges which still is not optimal.

Anisotropic Variants

It is typically desirable to strengthen edges by having a considerable smoothing along them but a reduced one across them. This requires the use of direction-dependent, i.e. non-scalar, diffusivities in the diffusion process. Hence, we replace the scalar diffusivity g by a diffusion tensor D [116, 139], such that the general diffusion process reads

$$\partial_t w_i \quad = \quad \operatorname{div}(D \cdot \nabla w_i) \tag{2.62}$$

where the eigenvectors of D state the main directions of the diffusion and the corresponding eigenvalues state the magnitude of the diffusion.

All of the aforementioned isotropic diffusivities can obviously be embedded into this formulation as $D := g \cdot \mathrm{Id}_{2\times 2}$, where $\mathrm{Id}_{2\times 2}$ is the identity matrix of size 2×2. In this case, the eigenvalues are the same for both directions. Let us now introduce the anisotropic counterparts to the aforementioned inhomogeneous, isotropic diffusivities by assembling diffusion tensors with in general different eigenvalues for both directions.

Image-Driven Anisotropic Diffusion

For the *image-driven anisotropic* case, the strategy is to project the gradients of the sought functions onto the direction of the edge (that is orthogonal to the gradient), which is represented by $\mathbf{r} = \frac{1}{|\nabla I^{\perp}|} I^{\perp}$. A simple intuitive approach would be to consider the projection matrix $D := \mathbf{r}\mathbf{r}^{\top}$ as diffusion tensor. However, this would avoid diffusion

across edges or in homogeneous regions completely (since one or both eigenvalues would be zero) which is usually not intended due to stability reasons. Thus, the diffusion tensor is regularized to allow for a small amount of smoothing across image edges and reads

$$D \;:=\; \frac{1}{|\nabla I|^2 + 2\epsilon^2}\left(\nabla I^{\perp}\nabla I^{\perp\top} + \epsilon^2_{\text{diffReg}}\text{Id}_{2\times 2}\right) \tag{2.63}$$

with $\epsilon_{\text{diffReg}} > 0$ and knowing that $|\nabla I^{\perp}| = |\nabla I|$ holds. The corresponding smoothness term [97] is given by

$$E_{\text{Smooth}}(\mathbf{w}) \;=\; \int_{\Omega} \sum_{j=1}^{2} \nabla w_j^{\top} D \nabla w_j \, d\tilde{\mathbf{x}}\,. \tag{2.64}$$

Still, the anisotropic case leads to oversegmentation artifacts and the image gradient is sensitive to noise.

Flow-Driven Anisotropic Diffusion

In the *flow-driven anisotropic* case, the idea is also based on the structure tensor. But here, it is a multi-channel variant of the structure tensor $S := \sum_{j=1}^{2} \nabla w_j \nabla w_j^{\top}$ on the flow functions and without rotation of the gradients [152]. Its eigenvalues indicate the change rate along and across the flow edge, similar to structure tensors for image structures. Since the eigenvalue across edges is high and we want to reduce the diffusion across them, a positive, decreasing function Ψ' is applied to S in order to form the diffusion tensor which reads

$$D \;:=\; \Psi'(S) \;=\; \Psi'\left(\sum_{j=1}^{2} \nabla w_j \nabla w_j^{\top}\right), \tag{2.65}$$

where the application of the scalar function $g := \Psi'$ to a symmetric square matrix A of size $N \times N$ is defined as

$$g(A) \;=\; (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N)\ \text{diag}\big(g(\sigma_1), g(\sigma_2), \ldots, g(\sigma_N)\big)\ (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N)^{\top}$$

with $\mathbf{v}_i$ being the eigenvectors of A and σ_i being the corresponding eigenvalues. This means that $g(A)$ has the same eigenvectors but modified eigenvalues compared to A. Weickert and Schnörr have shown in [152] that the corresponding smoothness term is given by

$$E_{\text{Smooth}}(\mathbf{w}) \;=\; \int_{\Omega} \text{tr}\Psi\left(\sum_{j=1}^{2} \nabla w_j \nabla w_j^{\top}\right) d\tilde{\mathbf{x}}\,. \tag{2.66}$$

The advantage of this smoothness term is that it avoids oversegmentation artifacts. However, the edges may not be well localized and unsharp, since they are derived from a structure with an evolving nature.

Joint Image- and Flow-Driven Anisotropic Diffusion

There are several variations that combine image- and flow-driven diffusion in order to overcome the limitations from the above presented approaches. In general, anisotropic approaches are preferable, since they allow for a direction-dependent smoothing that respects motion discontinuities. Hence, we will focus on a combined anisotropic approach. Since we are interested in well-localized and sharp edges like in the image-driven case, it makes sense to consider the structure from the images. However, not all image edges coincide with motion discontinuities. Thus, it makes sense to steer the amount of regularization using the flow contrast instead of the image contrast [133].

Diffusion Tensor. Given two orthogonal directions $\mathbf{r}_1$ and $\mathbf{r}_2 = \mathbf{r}_1^\perp$, such an anisotropic joint image- and flow-driven diffusion tensor [165, 164] reads

$$D \;:=\; \Psi'_{S1}\left(\sum_{j=1}^{2}(\mathbf{r}_1^\top \nabla w_j)^2\right)\cdot \mathbf{r}_1\mathbf{r}_1^\top + \Psi'_{S2}\left(\sum_{j=1}^{2}(\mathbf{r}_2^\top \nabla w_j)^2\right)\cdot \mathbf{r}_2\mathbf{r}_2^\top . \tag{2.67}$$

It is invariant under rotations due to the joint penalization of all sought functions w_j and anisotropic since it has different eigenvalues for different directions.

Regularization Tensor. In contrast to other approaches like [133], the directions $\mathbf{r}_i$ are not directly derived from the structure tensor of the image, but from the motion tensors of the data constraints. The addition of the normalized upper-left 2×2 matrices of the motion tensors provides the so-called *regularization tensor* $\mathscr{R}_\rho$ which is given by

$$\mathscr{R}_\rho \;:=\; \sum_{c=1}^{3} K_\rho * \left[\sum_{i=1}^{N_\mathrm{p}} \gamma_i \theta_i \left(\tilde{\mathbf{p}}_i^c (\tilde{\mathbf{p}}_i^c)^\top\right)\right], \tag{2.68}$$

where N_p is the total number of data constraints, $\tilde{\mathbf{p}}_i = (p_{i,1}, p_{i,2})^\top$ contains the first two components of the generating vector $\mathbf{p}_i$ for some data constraint, c denotes the image channel, $K_\rho *$ denotes the convolution with a Gaussian of standard deviation ρ, γ_i are the weights of the individual data terms and θ_i are the corresponding normalization factors. The two eigenvectors $\mathbf{r}_i$ of the regularization tensor contain the *constraint edges* as a generalization of the image edges. Since, hence, this type of regularization works complementary to the data term, it is called *complementary regularizer*. Please note, that when neglecting the normalization factor and restricting to the brightness constancy assumption, the regularization tensor coincides with the structure tensor.

Smoothness Term. The corresponding smoothness term is given by

$$\begin{aligned} E_{\mathrm{Smooth}}(\mathbf{w}) &= \int_\Omega \Psi_{S1}\left(\sum_{j=1}^{2}\left(\mathbf{r}_1^\top \nabla w_j\right)^2\right) + \Psi_{S2}\left(\sum_{j=1}^{2}\left(\mathbf{r}_2^\top \nabla w_j\right)^2\right) d\tilde{\mathbf{x}} \\ &= \int_\Omega \sum_{i=1}^{2} \Psi_{Si}\left(\sum_{j=1}^{2}\left(\mathbf{r}_i^\top \nabla w_j\right)^2\right) d\tilde{\mathbf{x}} . \end{aligned} \tag{2.69}$$

2.8.4 Final Model

The final model of Zimmer *et al.* [165, 164] is obtained by modifying the model of Bruhn and Weickert [29]. To this end, we apply the data constraints on multiple color channels and replace the isotropic smoothness term by the complementary regularizer. As mentioned before, we make use of the RGB color space. Finally, the model reads

$$\begin{aligned} E_{\text{Zimmer}}(\mathbf{w}) \quad = \quad & \int_{\Omega} \quad \delta \Psi_D \left(\sum_{c=1}^{3} |I^c(\mathbf{x}+\mathbf{w}) - I^c(\mathbf{x})|^2 \right) \\ & + \gamma \Psi_D \left(\sum_{c=1}^{3} |\nabla I^c(\mathbf{x}+\mathbf{w}) - \nabla I^c(\mathbf{x})|^2 \right) \\ & + \alpha \sum_{i=1}^{2} \Psi_{Si} \left(\sum_{j=1}^{2} \left(\mathbf{r}_i^\top \nabla w_j \right)^2 \right) d\tilde{\mathbf{x}} \,. \end{aligned} \tag{2.70}$$

where α, δ and γ are global weights and the direction vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ are derived as mentioned before. Our penalization strategy does not directly follow [165, 164]. Instead, we replace the regularized absolute value function $\Psi_D(s^2) = \sqrt{s^2 + \epsilon_D^2}$ by the similar Charbonnier penalizer $\Psi_D(s^2) = 2\epsilon_D^2 \sqrt{1 + s^2/\epsilon_D^2}$ which is a weighted equivalent. Moreover, regarding the smoothness term we follow Volz *et al.* [148] and choose the edge-enhancing and non-convex Perona-Malik penalizer $\Psi_{S1}(s^2) = \epsilon_{S1}^2 \log(1 + s^2/\epsilon_{S1}^2)$ [104] when smoothing across the edge and the edge-preserving Charbonnier penalizer $\Psi_{S2}(s^2) = 2\epsilon_{S2}^2 \sqrt{1 + s^2/\epsilon_{S2}^2}$ when smoothing along the edge. Please note that the constraint normalization is not explicitly stated in the model but applied in the numerics, since it is motivated by the linearized data constraints.

2.9 A Variant for Affine Flow Fields

The regularizers that are crucial for the already presented methods favor piecewise constant flow fields, since they aim at keeping the gradients low within the flow field. This is beneficial in case of dominant fronto-parallel motion. In the presence of considerable ego-motion in forward or backward direction, however, piecewise constant flow fields are not appropriate. Here, the dominant motion follows the z-axis (depth direction) in 3-D such that the apparent flow field in 2-D is typically convergent or divergent. Examples of such scenes are provided by the KITTI 2012 and 2015 benchmarks [52, 92]. In this case, regularizers that penalize deviations on the gradients of the flow components – so-called first-order regularizers – do not perform well.

Second-Order Regularization. In case of motions that are not piecewise constant, it is useful to replace such first-order regularizers by second-order regularizers that instead penalize deviations in the second derivatives of the flow. Such regularizers allow for piecewise affine flow fields instead of piecewise constant flow fields and are thus better suited for the estimation of non-fronto-parallel flow fields [138, 106, 23, 145]. A very intuitive isotropic example of such a regularizer that has already been used in the context of image denoising [82] and shape-from-shading [147] is based on the Hessian matrix and reads

$$E_{\text{Smooth}}(\mathbf{w}) = \int_{\Omega} \Psi_S\left(\sum_{j=1}^{2} ||\mathscr{H} w_j||_F^2\right), \tag{2.71}$$

where $||\mathscr{H}\cdot||_F$ is the Frobenius norm of the Hessian and $\Psi_S(s^2)$ denotes the Charbonnier penalizer that encourages piecewise affine solutions if applied to the Hessian. Recently, also more advanced second-order regularization strategies have been developed [24, 105, 61, 87, 89]. A further analysis of these, however, is out of the scope of this thesis.

2.9.1 Final Model

Regarding the data constraint, we keep the one of the method of Zimmer *et al.* including the constraint normalization in the numerics. Together with the second-order smoothness term, the final model reads

$$\begin{aligned} E_{\text{Zimmer-AFF}}(\mathbf{w}) = \int_{\Omega} & \delta\Psi_D\left(\sum_{c=1}^{3} |I^c(\mathbf{x}+\mathbf{w}) - I^c(\mathbf{x})|^2\right) \\ & +\gamma\Psi_D\left(\sum_{c=1}^{3} |\nabla I^c(\mathbf{x}+\mathbf{w}) - \nabla I^c(\mathbf{x})|^2\right) \\ & +\alpha\Psi_S\left(\sum_{j=1}^{2} ||\mathscr{H} w_j||_F^2\right) d\tilde{\mathbf{x}}\,. \end{aligned} \tag{2.72}$$

where α, δ and γ are global weights and the penalizer function Ψ_S is the Charbonnier penalizer as defined before. If not explicitly stated otherwise, this model will be the baseline for the KITTI benchmarks while the method of Zimmer *et al.* will be the baseline for any other benchmark or data set.

2.10 The Method of Brox and Malik

So far, a lot of advances regarding both the modeling part and the numerical part have been discussed. This includes the handling of large displacements in general. Relative large displacements of small objects, however, remain a severe problem. If the

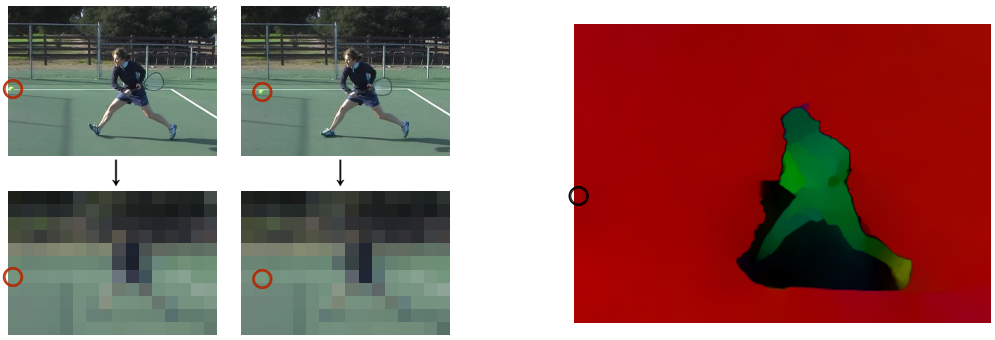


Figure 2.2: Illustration of the problematic case of a relative large displacement at hand of the Tennis sequence (Frames 496 and 497) from [27]. **Left:** When the image is downsampled to a resolution where the displacement of the small tennis ball is in the order of one pixel, the ball is hardly visible anymore. **Right:** The baseline method is not able to capture its displacement correctly.

displacement of an object relative to the background motion is larger than its size, it is in general not distinctive (compared to its background) on that level in the coarse-to-fine scheme which is appropriate to handle its displacement, i.e. where the corresponding displacement is small (see Fig. 2.2 for an example). This is a conceptual problem of coarse-to-fine warping schemes.

Small Objects vs. Noise. In a different sense, this can also be considered as a regularity-enforcing behavior of the coarse-to-fine warping scheme. On that mentioned level, such an object is either not visible at all or it is so small that it is indistinguishable from noise. Not adapting to noise on any level, however, is a key consequence of the combination of sub-quadratic data terms and regularity-enforcing smoothness terms. Among all possible motion candidates, the motion of the object's background is the most regular and hence also preferred as a candidate for the motion of the small object.

Integration of Feature Matches. In this context, Brox *et al.* [25] and Brox and Malik [27] came up with the idea to combine the standard variational method with the complementary feature matching approach. The latter completely comes without regularity and determines displacements by a brute force nearest neighbor search. Nevertheless, a unique solution is desired also in this approach. While in variational approaches a unique solution is the result of having enough constraints coming from both the data term and the smoothness term, the missing regularity constraint in the feature matching approach requires a different strategy to provide the procedure with an improved uniqueness.

Enforcing Uniqueness. The uniqueness enforcing strategy in feature matching covers several concepts: (i) restricting the matching process to locations where there is enough structure to provide unique features (so-called key-points), (ii) using features that assemble the local information in a discriminative way while preserving desired geometric and/or photometric invariances, and (iii) performing forward-backward consistency checks of the matches in order to remove ambigue – and thus inconsistent – matches.

If one assumes that those small objects that undergo a less regular motion (due to their relative large displacements) at least have a unique appearance, their motion can effectively be determined by feature matching and complement a variational optical flow model that is able to estimate the more regular parts of the apparent motion.

2.10.1 General Variational Model

In order to allow a set of pre-computed feature matches of type $\mathbf{w}_\mathrm{P} = (u_\mathrm{P}, v_\mathrm{P}, 1)^\top$ to guide the motion estimation, Brox and Malik proposed to add a similarity term to a variational baseline model which guides the estimation of the optical flow $\mathbf{w}$ by the matches $\mathbf{w}_\mathrm{P}$ at those locations where $\mathbf{w}_\mathrm{P}$ can be provided. The general model then reads

$$E_\mathrm{BM}(\mathbf{w}) \quad = \quad E_\mathrm{base}(\mathbf{w}) + \beta E_\mathrm{sim}(\mathbf{w}, \mathbf{w}_\mathrm{P}) \tag{2.73}$$

where $E_\mathrm{base}(\mathbf{w})$ is the model of Bruhn and Weickert [29] (see Sect. 2.7) as the baseline, β is a global weight and the additional similarity term E_sim is given by

$$E_\mathrm{sim}(\mathbf{w}, \mathbf{w}_\mathrm{P}) \quad = \quad \int_\Omega \chi_\mathrm{P}(\mathbf{x}) \rho_\mathrm{P}(\mathbf{x}) \Psi\left(|\mathbf{w} - \mathbf{w}_\mathrm{P}|^2\right) d\tilde{\mathbf{x}} \,. \tag{2.74}$$

This term includes an activation flag $\chi_\mathrm{P}(\mathbf{x})$, which is 1 if a feature match is given at $\mathbf{x}$ and 0 otherwise, a local confidence function $\rho_\mathrm{P}(\mathbf{x})$ which rates the reliability of the match and the similarity constraint with a sub-quadratic penalizer function $\Psi(s^2)$, which adds robustness against outliers.

2.10.2 Discussion

The question arises why variational approaches are *combined with* instead of being *replaced by* feature matching approaches. First of all, let us start with a property of feature matches that is a direct consequence of the concepts (i) and (iii): the matches are sparse whereas the optical flow has the desirable property of being dense.

Guidance by Features vs. Inpainting of Features. Densifying the set of feature matches could also be done with simple inpainting. However, there are other undesired properties of feature matches that remain present with simple inpainting. This includes

the fact that feature matches are typically only pixel-accurate and that the spatial extent of discriminative features introduces inaccuracies at motion discontinuities since a single feature may cover parts from different objects. Both issues are not resolved via inpainting. Variational optical flow approaches, in contrast, are sub-pixel accurate and can better adapt to local motion patterns due to their more local data terms. These beneficial properties are preserved when using feature matches as guidance during the coarse-to-fine optimization.

Discrete Matches in a Continuous Optimization. Within the continuous optimization of the variational optical flow approach, the influence of the discrete matches varies among the coarse-to-fine levels. There is a fixed number of matches but an increasing number of pixels from coarse levels to fine levels. Hence, the optimization is rather dominated by the feature matches at coarse levels while the influence decreases at finer levels. In any case, the standard data and smoothness terms of the variational model still apply to each pixel. This has multiple benefits: (i) The feature matches steer the estimation of the optical flow at coarse levels where a good initialization otherwise is missing due to an inappropriate upsampled flow. (ii) At finer levels, some of the remaining false positive matches are removed, since the conventional parts of the variational approach take over. In a pure inpainting approach, where all information only comes from the feature matches, they would not be removed . (iii) Finally, the variational approach also provides an optical flow for regions that do not carry enough information to assemble descriptive features, such as homogeneous regions.

2.10.3 Features

In order to apply feature matching, it is first necessary to compute features at each pixel of each frame. A lot of research has been done in order to find descriptive features – also called descriptors – with desired properties such as invariances w.r.t. geometry, illumination or scale. Brox *et al.* [25] and Brox and Malik [27] especially used three different features in their work: segmentation-based Region Matching Descriptors [25, 4], Histogram of Oriented Gradients (HOG) [36] and Geometric Blur (GB) [14]. Let us now describe the versions of the descriptors that have been used in [27].

Region Matching Descriptors

The idea behind the first type of feature is based on a segmentation method as proposed by [4]. The resulting segments form the regions that are matched between frames. Since the underlying boundary detection does not only simply consider edges but also the overall texture, it avoids e.g. to detect boundaries within repetitive textures.

Hierarchy of Regions. The method delivers a hierarchy of regions that result from this robust boundary detection step. In this hierarchy, regions with strong edges persist

through many levels while regions with weak edges are quickly merged into larger regions. Out of the reference frame, only the most stable regions are considered. According to [25] this excludes regions that are too small or that are present in less than five levels of the hierarchy. The authors finally fit an ellipse to each of the remaining regions and normalize the area around the centroid to a 32×32 patch, on which descriptors are computed.

Descriptors. For each region, two descriptors S and C are computed. The descriptor S shall account for the shape of a region by considering 16 orientation histograms with 8 bins, as inspired by the SIFT- [80] and HOG-descriptors [36]. The descriptor C contains the mean color of the same 16 parts as the descriptor S. For C, however, these are restricted to those parts that belong to the region.

Descriptor Distance. The centroids of the regions serve as the locations of the descriptors. The distance between two different regions is based on separate Euclidean distances of the associated descriptors S and C. In a first step, for both types of descriptors, normalized squared Euclidean distances are computed for pairs of regions (i, j) which read

$$d^2(S_i, S_j) = \frac{||S_i - S_j||^2}{\frac{1}{N}\sum_{k,l} ||S_k - S_l||^2}, \tag{2.75}$$

$$d^2(C_i, C_j) = \frac{||C_i - C_j||^2}{\frac{1}{N}\sum_{k,l} ||C_k - C_l||^2}, \tag{2.76}$$

where N denotes the number of all combinations i, j. The final squared distance of a pair of regions is the the average of the normalized squared distances which reads

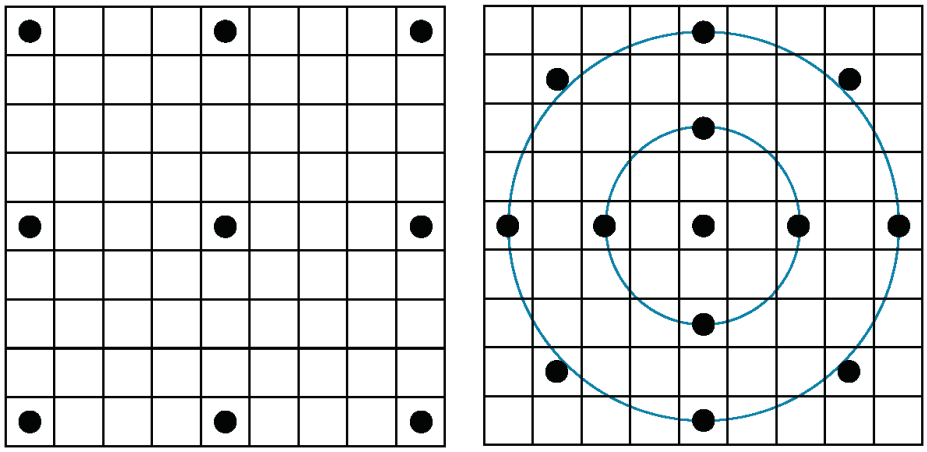
$$d^2(i, j) = \frac{1}{2}\left(d^2(S_i, S_j) + d^2(C_i, C_j)\right). \tag{2.77}$$

Further details and additional filtering steps can be found in [25].

Histogram of Oriented Gradients

The Histogram of Oriented Gradients (HOG) descriptor has been introduced in [36] in the context of human detection. In the context of motion estimation, it has been used both in the context of feature matching [27] and as a feature constancy based data term of an optical flow method [109].

Gathering Shape Information. The descriptor encodes shape information by considering gradient orientations, which at the same time provides invariance to additive illumination changes. Its computation comprises the computation of gradients, the binning of



HOG descriptor [27] GB descriptor [27]

Figure 2.3: The blocks that are used in the HOG and the GB descriptor.

gradient orientations into histograms, a sub-division of the local neighborhood (which is also called a block) into smaller cells, where the cell-wise histograms are normalized w.r.t. contrast, and finally the assembling of the descriptor.

Descriptor. There are a lot of parameters such as the number of bins in the histograms, the decision if the sign of the gradient is considered (providing orientations in the range $[0° - 360°]$) or neglected (providing orientations in the range $[0° - 180°]$), the size of the local blocks and cells, the choice of the norm that is used in the normalization step and a lot more. Further parameters can be found in [36]. In [27], the following procedure is applied: The descriptor considers a block (neighborhood) of 15×15 pixels around the central pixel where the central pixel as well as its neighboring pixels with distances of four pixels are the centers of cells of size 7×7 in which the histograms are computed (see Fig. 2.3). These histograms consist of 15 different orientations in the range $[0° - 360°]$, i.e. the signs of the gradient orientations are considered.

Geometric Blur

Another example for a descriptor is given by the Geometric Blur (GB) descriptor which has been introduced in [14] in the context of template matching.

Geometric Distortions. In contrast to the HOG descriptor, the GB descriptor explicitly addresses the problem of geometric distortions between corresponding objects that can origin from a change in the relative viewpoint. Since descriptors usually consider large neighborhoods in order to be descriptive, conventional descriptors are highly sensitive

to geometric distortions. The larger the considered neighborhood is, the larger are the differences between a descriptor in one image and a corresponding descriptor in another image.

Positional Uncertainty. When regarding two corresponding neighborhoods of similar objects that undergo a relative geometric distortion, it becomes apparent that there is a positional uncertainty between corresponding pixels in the peripheral regions around the central pixel. The higher the distance to the center, the higher is the positional uncertainty.

Descriptor. A descriptor can be made robust against positional uncertainties by considering a blurred version of the image where information from different positions is smeared. The core idea behind the Geometric Blur descriptor is to consider multiple differently smoothed versions of the underlying image in order to account for the varying positional uncertainty within the considered neighborhood. The descriptor is assembled by considering a less smoothed version of the image at the central pixel of the neighborhood and more strongly smoothed versions at the peripheral regions. Usually, the images are pre-processed by computing the gradient orientations since, on the one hand, the method works best on sparse images – gradients are typically sparse – and, on the other hand, gradient orientations introduce some degree of illumination invariance. Further details can be found in [14].

In the variant of [27], again histograms of gradient orientations are used that cover 15 bins. Instead of a fixed 7×7 window, however, three different Gaussian windows with $\sigma_0 = 0$, $\sigma_1 = 1$ and $\sigma_2 = 2$ are considered. As can be seen from Fig. 2.3, the descriptor contains one entry from the histograms for σ_0 (at the center), four entries from the histograms for σ_1 (four inner neighbors) and eight entries from the histograms for σ_2.

2.10.4 Details on Feature Matching

The basis of the feature matching approach is a nearest neighbor search, i.e. for each descriptor in the reference frame (source descriptor) one looks for that descriptor in the subsequent frame (target descriptor) that has the smallest (Euclidean) distance (SSD). The difference between the locations of the target and of the source descriptor describes the displacement between both features, the so-called feature match.

Robust Matching. In order to add robustness, two further techniques are applied. First, a forward-backward consistency check is performed: A match is only kept if the source descriptor also is the best match in an opposite nearest neighbor search that is conducted starting with the target descriptor. This excludes ambigue matches. Second, a local confidence is assigned to the feature match. Given the distance d_1 to the best

match and the distance d_2 to the second best match, one computes

$$\rho(\mathbf{x}) \quad = \quad \frac{d_2 - d_1}{d_1} \,. \tag{2.78}$$

This confidence measure amplifies the weight of a feature match for very similar descriptors (i.e. d_1 close to zero) but dampens its influence if the best match is not notably more similar to the source descriptor than the second best match, i.e. if the source descriptor is not distinctly discriminative.

2.10.5 Final Method

The authors compared and evaluated the different descriptors for feature matching and concluded that the HOG descriptor [36] provides sufficient correct matches while producing the least false matches. Hence, their final method uses HOG descriptors for feature matching. In both the conference version [25] and the journal version [27] of their work the authors consistently made use of the model of Bruhn and Weickert [29] as their baseline, which was state-of-the-art at the time of their conference publication [25]. The final model then reads

$$\begin{aligned} E_{\mathrm{BM}}(\mathbf{w}) \quad = \quad & \int_\Omega \Psi\left(|I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})|^2\right) + \gamma \Psi\left(|\nabla I(\mathbf{x}+\mathbf{w}) - \nabla I(\mathbf{x})|^2\right) \\ & + \beta \chi_{\mathrm{P}}(\mathbf{x}) \rho_{\mathrm{P}}(\mathbf{x}) \Psi\left(|\mathbf{w} - \mathbf{w}_{\mathrm{P}}|^2\right) \\ & + \alpha \Psi\left(|\nabla u|^2 + |\nabla v|^2\right) \, d\tilde{\mathbf{x}} \,, \end{aligned} \tag{2.79}$$

where $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ is the (regularized) absolute value function, and γ, β and α are balancing weights.

Chapter

3 Large Displacement Optical Flow

In this chapter, we will further analyze the deficiencies of the coarse-to-fine warping scheme when relative large displacements are apparent, i.e. when there are small objects that undergo a large motion relative to their background (as described in Chapter 1, Sect. 1.3). This allows us to sub-divide the set of relative large displacements into two categories: moderately large displacements and arbitrarily large displacements. For each of these categories, we will present a novel method that implements a strategy for estimating the respective type of large displacements correctly; while in this chapter we present a method for handling arbitrarily large displacements, in the next chapter we will present a method for handling moderately large displacements. These are based on two papers [129, 127].

3.1 Deficiencies of Coarse-to-fine Warping

The incremental coarse-to-fine warping strategy [15, 26, 103] has become the de-facto standard as the optimization strategy for variational optical flow methods. It covers the estimation of *small* displacements in general and, moreover, it covers the estimation of *large* displacements of *large* objects. In each of these cases, the optical flow is sufficiently regular, such that on the respective resolution level that is appropriate to estimate a particular displacement these results are far away from noisy results caused by outliers.

Downsampling Objects. Let us now have a look how an object behaves through successive downsampling within the coarse-to-fine scheme. When downsampling an image, there may be different states for an object on different levels w.r.t. its distinguishability from the background. We will consider three cases: (i) In the first, finer group of resolution levels, it is clearly distinguishable from its background, i.e. there is at least one pixel which only covers the object but not its background. (ii) In the subsequent second group, there is a state of smearing where a coarse pixel does not

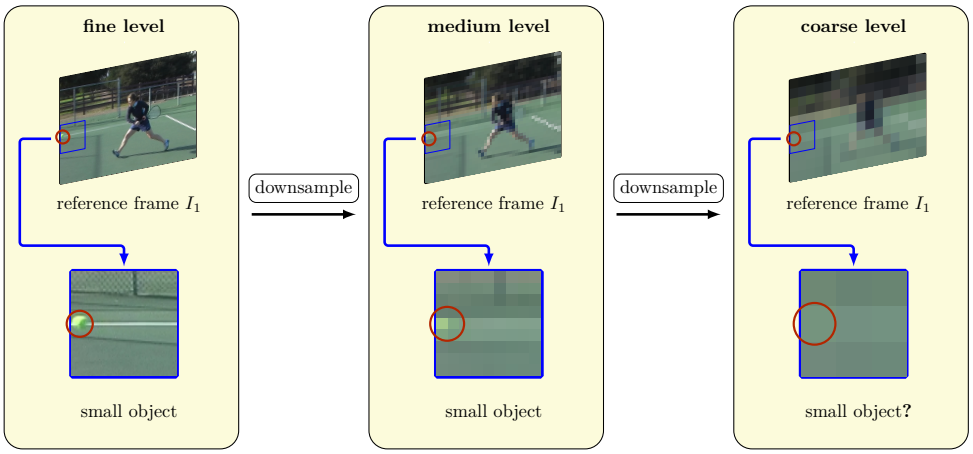


Figure 3.1: Evolution of a small object during the process of downsampling at hand of the tennis ball in the Tennis sequence [27]. While the tennis ball is completely and distinctively visible on the fine level, on the medium level there is only one pixel that can clearly be associated with it. On the coarse level, the marked pixel, which covers the ball, is mostly influenced by the background.

only cover the small object but also parts of the background, but this pixel is still clearly distinguishable from the surrounding pure background pixels due to a beneficial mixture of color values. (iii) In the third and last group of resolution levels, the background dominates and the share of color at a pixel that origins from the small object is too small to be distinguishable from the remaining background. An illustrative example of this problem is shown in Fig. 3.1.

Estimating Different Scales of Displacements. For an object whose displacement scale can be estimated at a resolution level of the first group, the conventional coarse-to-fine warping scheme is sufficient to estimate its motion. This covers small displacements and absolute (non-relative) large displacements (simple Case 1). If the displacements can be estimated at a level from the second group, we face moderately large displacements. Here, adaptations to the variational approach may help (Case 2). In contrast, for an object whose displacement scale can only be estimated at a level of the last group, the coarse-to-fine scheme is clearly not able to estimate its motion. This covers the remaining arbitrarily large displacements (Case 3).

Please note that the transitions between these groups are smooth and they also depend on the local contrast between foreground and background, since this property essentially influences the distinguishability between foreground and background.

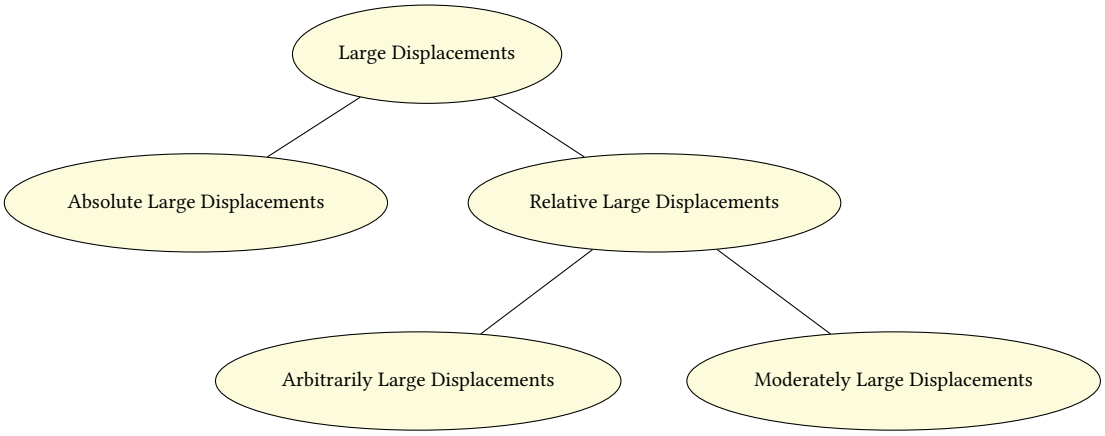


Figure 3.2: Hierarchy of terms that describe different categories of large displacements.

While the first case yet can easily be handled by coarse-to-fine schemes, the cases two and three remain a challenging problem. In the literature, there is no distinction between these cases so far; they are simply recognized as *large displacements*. Typical solutions are inspired by the method of Brox and Malik [27], which we have discussed in Chapter 2 (Sect. 2.10). Hence, we will – for now – concentrate on case three – handling arbitrarily large displacements – and present a novel method for handling this type of displacements.

3.2 Terminology

So far, we have introduced several sub-categories of large displacements. Let us briefly clarify their terminology as depicted in Fig. 3.2. In order to distinguish those large displacements that need particular care from those that can be handled by conventional coarse-to-fine warping schemes, we introduced the term *relative large displacements*. Moreover, we introduced the terms *moderately large displacements* and *arbitrarily large displacements* as sub-categories of relative large displacements. In order to avoid overly complex linguistic expressions, the word "relative" is left out in these cases. In any case, we may restrict to the simple expression *large displacements* in the following, if the particular category can be derived from the context.

3.3 Estimation of Arbitrarily Large Displacements

For the scenario of arbitrarily large displacements, different approaches have been proposed. This starts with very basic block matching approaches [96] which yield almost dense flow fields, but are known to provide poor results due to the ambiguity of the search problem. In contrast, characteristic image features such as SIFT [80] can be used within an exhaustive search. These features typically provide accurate but sparse results. In order to improve results to be dense and accurate at the same time, some popular strategies based on the combination of variational approaches with exhaustive search have been introduced. One implementation of such an exhaustive search is the feature matching-step in approaches like the work of Brox and Malik [27] (see Chapter 2, Sect. 2.10). Another implementation is given by the method of Steinbrücker *et al.* [126]. It abandons the coarse-to-fine warping strategy and applies quadratic relaxation with alternating global optimizations on a data term without linearization and on a smoothness term. This allows for an exhaustive search when optimizing the data term, which is not limited to a specific displacement scale. The evaluation on small displacement data, however, is limited, and the given results of this scenario do not demonstrate state-of-the-art performance. Another different alternative that provides accurate and dense results is given by Rhemann *et al.* [113] which is based on adaptive window matching via a nonlinear filtering step.

Feature Matching. In the literature, from all those methods approaches based on the integration of discriminative feature matches have been favored, since the exhaustive search in the matching step allows for arbitrary displacements and proper inclusion of feature matches provides superior results. However, the following problem has to be solved: If a feature is not unique enough, the matching step will likely result in a false match (also known as false positive). A forward-backward consistency check can alleviate but not completely avoid this problem. Moreover, the direct integration of the obtained matches into a variational approach via an additional similarity term as proposed in [25, 27] does not only adapt the estimation to the real large displacements of small objects. It also adapts to the false matches that are transferred from the feature matching step. Hence, it might be promising to make the integration of matches *adaptive*. A corresponding strategy shall integrate useful matches but discard false matches.

3.4 Related Work

An approach to avoid the inclusion of false positives into the estimation was proposed by Xu *et al.* [159, 160]. In this method, the integration of feature matches is post-poned to the minimization. At each coarse-to-fine level, multiple sets of flow candidates are obtained. These come from the upsampled optical flow of the previous coarser level,

from a feature matching step using SIFT features [80] and from a patch matching step [10]. It has to be noted that the number of candidates from the additional matching steps is reduced to a small number of promising candidates, which, however, are flow candidates for all pixels of the image. The upsampled result as well as the set of additional flow candidates are then fused into a single flow field via discrete optimization. Finally, this single flow field then serves as initialization for the current coarse-to-fine level. Although this method achieves good results, it is rather slow due to the discrete fusion steps during the optimization of the variational approach. Thus, it would be desirable to have an approach that integrates feature matches without complicating and slowing down the optimization of the variational method.

3.5 Contributions

In order to address this problem, our novel method keeps the integration model-based and does not affect the variational optimization but adapts the matching process. Based on our corresponding paper [129], it builds upon the method of Brox and Malik [27] which integrates pre-computed feature matches into a variational optical flow framework via a similarity term in the model. Our method improves over this inspiring work in three ways: (i) We build upon a more advanced baseline optical flow method, the method of Zimmer *et al.* [164], in order to improve small displacement results in general. (ii) We restrict the integration of feature matches to those locations where an additional guidance by such matches is considered to be helpful in order to avoid false matches to deteriorate the result at those locations where the baseline already provides an accurate result. (iii) We extend the uniqueness confidence measure for feature matches as known from [27] in order to additionally consider the expected improvement of a feature match over the corresponding baseline flow vector.

We will demonstrate that the abandonment of matches at locations where no improvement is expected does not only improve the flow quality but also decreases the workload. Please note that our method [129] was state-of-the-art in 2012. In the meantime since its publication, a lot of progress has been done in the literature that has been partially inspired by our work. This progress includes but is not limited to the usage of improved features that lead to less false positives such as [154, 58], the setup of extended pipelines for the integration of feature matches [111, 69] as well as the integration of improved feature matching methods such as [70]. A further review of these methods, however, is out of the scope of this thesis.

3.6 ALD-Flow

Our first method, that aims at the estimation of arbitrarily large displacements, is called *Adaptive Large Displacement Optical Flow (ALD-Flow)* and has been proposed in [129].

In this approach, we apply a novel three-step strategy in order to determine where additional feature matches are actually helpful. First of all, we compute flow fields forward and backward between the reference frame and its successor using our baseline approach. In a second step, based on these flow fields, we decide at which locations the estimation could actually benefit from supplementary feature matches. In this context, we also identify and remove unreliable locations where feature matches would potentially lead to outliers. Finally, we compute feature matches only at those carefully selected positions and adaptively integrate them into the estimation. By restricting ourselves only to those locations where feature matches are *really needed*, we improve both the quality and the speed of the estimation.

3.7 Variational Model

As our baseline method, we use the method of Zimmer *et al.* [165, 164] as presented in Chapter 2 (Sect. 2.8) whose variational model reads

$$\begin{aligned} E_{\text{base}}(\mathbf{w}) \quad = \quad \int_{\Omega} \quad & \delta \Psi_D \left(\sum_{c=1}^{3} |I^c(\mathbf{x}+\mathbf{w}) - I^c(\mathbf{x})|^2 \right) \\ & + \gamma \Psi_D \left(\sum_{c=1}^{3} |\nabla I^c(\mathbf{x}+\mathbf{w}) - \nabla I^c(\mathbf{x})|^2 \right) \\ & + \alpha \sum_{i=1}^{2} \Psi_{Si} \left(\sum_{j=1}^{2} \left(\mathbf{r}_i^\top \nabla w_j \right)^2 \right) d\tilde{\mathbf{x}} \,. \end{aligned} \tag{3.1}$$

In the following, the corresponding baseline flow will be denoted as $\mathbf{w}_{\text{base}}$.

Given a set of feature matches $\mathbf{w}_{\text{P}}$, we will integrate them into our approach by equipping the model with a similarity term E_{sim} as presented in Chapter 2 (Sect. 2.10.1):

$$E(\mathbf{w}) \quad = \quad E_{\text{base}}(\mathbf{w}) + E_{\text{sim}}(\mathbf{w}, \mathbf{w}_{\text{P}}) \,. \tag{3.2}$$

This additional term is given by

$$E_{\text{sim}}(\mathbf{w}, \mathbf{w}_{\text{P}}) \quad = \quad \beta \int_{\Omega} \chi_{\text{P}}(\mathbf{x}) \rho_{\text{P}}(\mathbf{x}) \Psi_{\text{P}} \left(|\mathbf{w} - \mathbf{w}_{\text{P}}|^2 \right) d\tilde{\mathbf{x}} \,, \tag{3.3}$$

where β is a balancing weight and Ψ_{P} is the Charbonnier penalizer [33]. The activation flag $\chi_{\text{P}}(\mathbf{x})$ and the confidence function $\rho_{\text{P}}(\mathbf{x})$ will be described later in this thesis, since they are a by-product of the process of feature matching.

3.8 Adaptive Integration of Feature Matches

The process of simple feature matching is hardly robust against outliers. It highly depends on the uniqueness of the underlying features. Although forward-backward checks are an improvement in this context, there can still be considerably many false positive matches that survive this check. Hence, in order to make the integration of feature matches more robust, some additional steps are necessary. To this end, we start by reviewing the ideas behind the selection strategy of Brox and Malik [27]. Afterwards, we explain our novel strategy to integrate large displacement flow vectors into the variational baseline method without severe deterioration of the accuracy.

3.8.1 The Selection Strategy of Brox and Malik

In [27], the authors propose to reduce the computational effort by integrating feature matches only at every *fourth* pixel in x- and y-direction. This is a valid reduction of effort since the descriptors at these locations still cover information from the neighborhood due to their patch-based nature. Furthermore, to improve the quality, it is helpful to analyze the structure in the image. Having the aperture problem in mind, it is obvious that not all locations in the image are equally suitable to be matched without any regularization. Hence, the authors propose to avoid ambiguities by only using feature matches at locations with sufficient structure. To this end, they determine the structuredness of each pixel by computing the smaller eigenvalue $\lambda(\mathbf{x})$ of the structure tensor [64], and use features for matching only at those locations where $\lambda(\mathbf{x}) \geq \frac{1}{8}\bar{\lambda}$ holds, with $\bar{\lambda}$ being the average structuredness within the image.

3.8.2 Adaptive Sparsification Strategy

The structuredness-based selection strategy of Brox and Malik answers the following question: *At which locations do we have enough image information to assemble a discriminative descriptor?* (see Fig. 3.3, top). While this is undoubtedly an important question, we can go one step beyond and also try to answer the following important question: *At which locations is the integration of feature matches particularly useful to improve the final result?* (see Fig. 3.3, bottom). Assuming that we can answer this question correctly, it allows us to discard those locations for the integration of feature matches where the baseline flow is already sufficiently accurate. Overall, this improves the quality of the flow estimation, since the feature matches at these locations cannot improve the estimation (due to an already accurate baseline flow) but only deteriorate it (due to the chance of being a false positive). Moreover, it reduces the computational effort, such that we can even consider every *second pixel* in x- and y-direction in the sparsification process and thus double the sampling rate while still having a much lower workload compared to [27]. In the end, this means that we have more matches at locations of

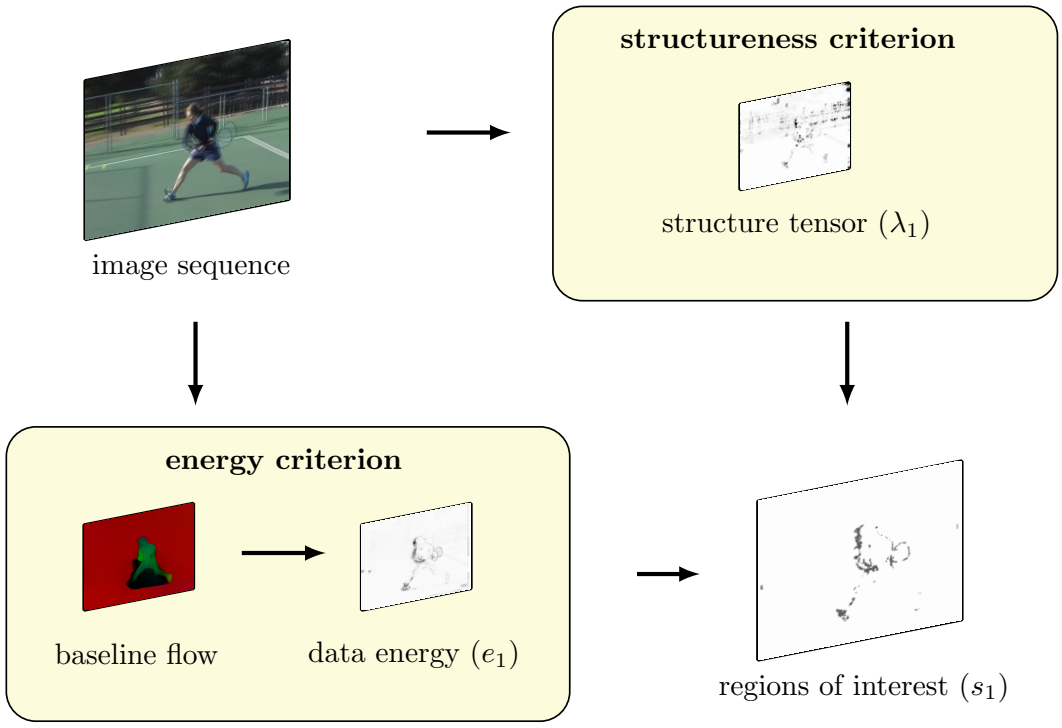


Figure 3.3: Illustration of the selection criteria. Thresholding the smaller eigenvalue λ_1 of the structure tensor of the first frame indicates sufficient structuredness for a feature descriptor while thresholding the data energy $e_1(\mathbf{x})$ of the baseline indicates the need for additional guidance by feature matches. The combination of both gives us the regions of interest for feature matching. Here, dark colors indicate high values.

small objects with large displacements in order to support the correct estimation of their displacement while strongly reducing the potential to harm the estimation at other locations. Let us now provide the details on how to select the corresponding locations (regions of interest, see Fig. 3.3, right).

Selection Criteria

In order to decide if a feature match may improve the estimation or not, we first compute an initial flow field using the baseline method. Using this flow, we evaluate the local energy $e_1(\mathbf{x})$ of the data term, which can be seen as a generalized registration error, respecting the data constancy assumptions. More precisely, we consider it to be a reciprocal confidence measure of the baseline flow where a higher flow quality leads to lower values. In general, the energy of the data term is high at those locations where the motion is not estimated well and thus the estimation needs additional guidance by a feature match. In our sparsification strategy, we call this the *energy criterion* (see

Fig. 3.3, bottom). Using this criterion, unnecessary matches are avoided and, moreover, the number of outliers is reduced.

Please note that a simple forward-backward check using baseline flows in forward and backward directions will not indicate the relevant locations, since relative large displacements are typically not captured by both flow fields in either direction. We additionally make use of the *structuredness criterion* from [27] in order to incorporate only matches of discriminative features.

Adaptive Forward-Backward Sparsification

Following [27], we will not only conduct a matching in forward direction from the first frame to the second frame but also a consistency check in backward direction (from the second frame to the first frame). In this context, there are three different sets of pixels to consider: the origins of the matches in the first frame (matching candidate set), the set of potential matching targets in the second frame, and the set of potential targets in the first frame for the consistency check in the backward direction. While the forward matching establishes correspondences between the candidate set and the target set (see Fig. 3.4, top), the consistency check establishes correspondences between the target set and the consistency set (see Fig. 3.4, bottom).

Applying the Criteria. In [27], only the matching candidate set has been sparsified and this has been done using only the structuredness criterion. In contrast, we furthermore apply the energy criterion to this set. Such a sparsification is already beneficial, since a proper sparsification of the candidate set has two positive effects: it improves the quality of the candidate set and it increases the speed of the matching process. Nevertheless, also a sparsification of the two target sets has a positive effect: it further decreases the runtime of the matching process by reducing the search space. Such a reduction makes sense, since e.g. a discriminative feature of a highly structured region will hardly be matched with a feature of a rather homogeneous region. Moreover, as another example, a feature of a mismatched small object is also not supposed to match with a feature of an object whose motion can already be correctly determined by the baseline flow. Hence, we will apply our sparsification strategy to *all* sets.

Additional Criteria in the Second Frame. When sparsifying all sets, which includes the matching target set in the second frame, we also need a structuredness criterion in this frame and an energy criterion in backward direction. To this end, we make use of the smaller eigenvalue $\lambda_2(\mathbf{x})$ of the structure tensor of the second frame for the structuredness criterion and of the data energy $e_2(\mathbf{x})$ of a backward flow of the baseline method for the energy criterion. In this context, we need the *backward* flow, since only the data energy of a flow that *origins* in the second frame can indicate mismatches at the

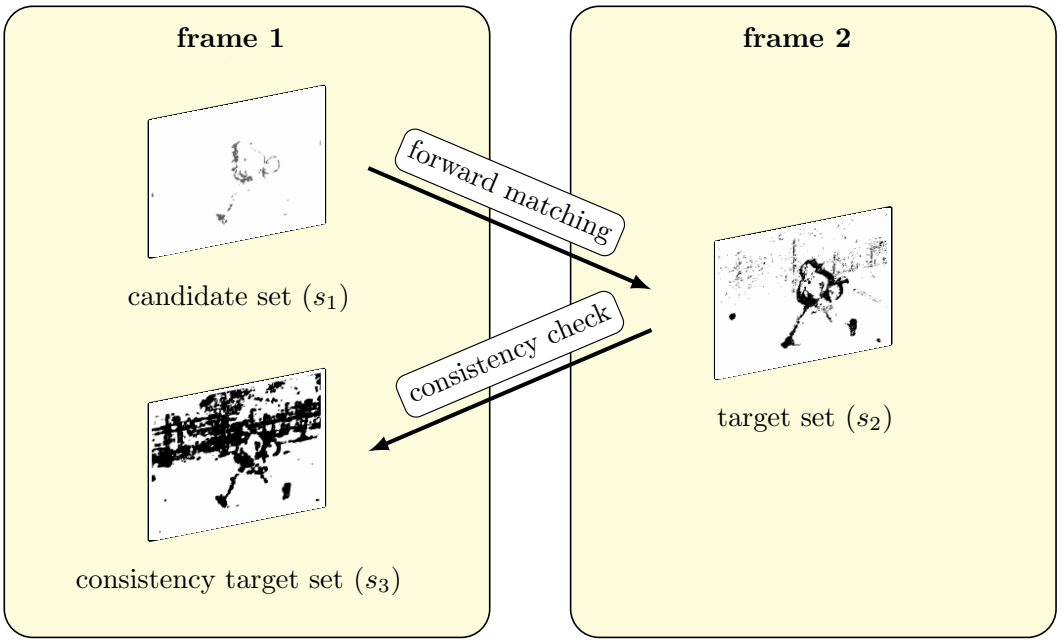


Figure 3.4: Illustration of the subsequent sets of locations during the matching process with a forward-backward consistency check (considered pixels are black). The sparsification reduces the runtime while the increasing size of the sets supports the consistency check to filter out unreliable matches.

correct location in this frame and the backward flow is assumed to show inaccuracies for the same objects for which the forward flow shows inaccuracies.

Sizes of the Sets. Nevertheless, we make sure that for both matching directions the target sets are bigger than the respective sets of origins in order to keep multiple potential targets per match and thus to keep some potential ambiguity. This is necessary for the consistency check to filter out unreliable matches that are not unique enough. Hence, the sets of locations that are considered subsequently during the matching process will have an increasing size (see Fig. 3.4).

Definition of the Sets of Locations

In order to define these sets, we will make use of the smaller eigenvalue $\lambda_1(\mathbf{x})$ of the structure tensor in the first frame, its equivalent $\lambda_2(\mathbf{x})$ in the second frame, the energy $e_1(\mathbf{x})$ of the data term for the baseline flow in forward direction and its equivalent $e_2(\mathbf{x})$ for the backward direction. In any case, the respective sets of locations are determined by thresholding these structuredness and (reciprocal) confidence measures from below, i.e. at these locations the respective values exceed the given thresholds.

Set 1: Candidates for Matching (Regions of Interest). Since we start the whole matching process from the first frame, the candidate set is based on the maps $\lambda_1(\mathbf{x})$ and $e_1(\mathbf{x})$. The corresponding thresholds are given by $\theta_{\lambda 1}$ and θ_{e1}. The first set s_1, which contains the locations of the candidates for matching, is hence defined as

$$\mathbf{x} \in s_1 \quad \Leftrightarrow \quad \lambda_1(\mathbf{x}) > \theta_{\lambda 1} \wedge e_1(\mathbf{x}) > \theta_{e1} \,. \tag{3.4}$$

Set 2: Targets for Matching. As already discussed before, we also apply the sparsification strategy to the target sets. For the matching targets in the second frame, we consider the maps $\lambda_2(\mathbf{x})$ and $e_2(\mathbf{x})$ with corresponding thresholds $\theta_{\lambda 2}$ and θ_{e2}. The second set s_2 is then analogously defined as

$$\mathbf{x} \in s_2 \quad \Leftrightarrow \quad \lambda_2(\mathbf{x}) > \theta_{\lambda 2} \wedge e_2(\mathbf{x}) > \theta_{e2} \,. \tag{3.5}$$

In order to achieve the intended increase in size compared to the last set, these thresholds are usually chosen lower than their counterparts in Set 1.

Set 3: Targets for Consistency Check. Whenever a match has been established, also a backward matching is established starting from the target position and feature of the match in order to check consistency (i.e. if the backward match hits the original point). To this end, we need a set of target locations in Frame 1 that is a superset of s_1 including the original points. At the same time it should be bigger than s_1 to be a real challenge for the consistency check, i.e. it is a strict superset of s_1. The third set s_3 is defined by

$$\mathbf{x} \in s_3 \quad \Leftrightarrow \quad \lambda_1(\mathbf{x}) > \theta_{\lambda 3} \,. \tag{3.6}$$

where $\theta_{\lambda 3}$ is another structuredness threshold. At this stage, we omit the use of the data energy, since we only want to check if the features themselves, and therefore the matches, are sufficiently unique – which is not related to the baseline flow in any way.

3.8.3 Features

In the literature, various descriptors have been proposed that can be used for feature matching. Following [27], we first consider HOG descriptors [36], since these descriptors lead to the best compromise of true positive and false positive matches. Hence, in a first step, we compute HOG descriptors in s_3 (which covers s_1) and s_2 and apply the process of feature matching including the forward-backward consistency check on this type of features. However, since Brox and Malik also recognized a higher discriminativeness of the GB descriptor [14] and our sparsification strategy explicitly addresses the problem of false positives (which is the emphasized weakness of GB descriptors), we also take GB descriptors into account. To this end, we consider those locations from the set of

Table 3.1: The quantiles that have been used in order to determine the thresholds within our adaptive sparsification strategy. Values in brackets indicate the strict threshold within a double thresholding strategy.

Set	quant_λ (structuredness)	quant_e (data energy)
s_1	80%	98.5% (99.7%)
s_2	60%	92%
s_3	70%	–

matching candidates s_1 again where the feature matching step with HOG descriptors did not result in a match (i.e. $\chi_P(\mathbf{x}) = 0$, although $\mathbf{x} \in s_1$), and apply the feature matching step together with the forward-backward consistency check also to the GB descriptors. We, hence, enrich a set of HOG feature matches with additional GB feature matches.

3.8.4 Thresholds

In order to determine the thresholds of the adaptive sparsification strategy, we have to take into account the descriptiveness of the given features and thus the tendency to produce false matches. The work of Brox and Malik [27] has shown that despite of the forward-backward consistency check there are still considerably many false positives remaining. The most important aspect of our work is the reduction of false positives such that small displacement scenarios are not deteriorated by the integration of feature matches. To this end, we strongly limit the amount of feature matches to the most promising locations and only keep a very low amount of them for the integration of feature matches. We achieve this using a quantile-based approach which gives us a relatively extensive control over this amount – which is also necessary to arrange the increasing sizes of the sets of locations (i.e. $|s_1| < |s_2|$ and $|s_1| < |s_3|$). To this end, we choose quantiles $quant_\lambda$ and $quant_e$ for all sets s_1 to s_3 on the structureness and energy maps of the respective frames that indicate which share of pixels in these maps shall not pass the corresponding thresholds θ_λ and θ_e that are computed from these quantiles.

Tab. 3.1 gives an overview of the used quantiles that we have chosen for an optimal performance on the Middlebury training sequences [9] while still being able to capture the large displacements of some particular sequences that have been used in [27]. Please note that for quality reasons, we conduct a double thresholding scheme [32] on the data energy when determining the candidate set s_1 using a strict threshold in order to determine seeds and a soft threshold that is applied in the neighborhood of those seeds. The strict threshold is given in brackets.

3.8.5 Activation Flag $\chi_{\mathrm{P}}(\mathbf{x})$

During the final estimation of the optical flow, the activation flag $\chi_{\mathrm{P}}(\mathbf{x})$ is supposed to indicate where we have a reliable feature match that can be integrated. So, as an initial definition, its values depend on whether the location of a feature match belongs to the candidate set s_1 or not:

$$\chi_{\mathrm{P}}(\mathbf{x}) \quad = \quad \begin{cases} 1, & \text{if } \mathbf{x} \in s_1 \\ 0, & \text{otherwise}\,. \end{cases} \tag{3.7}$$

This activation flag reflects all sparsification steps that have been done *a priori* to the feature matching. In the following refinement stages that are conducted *a posteriori* to the feature matching step, we will further remove outliers from the integration into the final flow estimation.

Refinement 1: Forward-Backward Consistency Check

The forward-backward consistency check subsequently sets $\chi_{\mathrm{P}}(\mathbf{x}) = 0$ at each location where consistency is not achieved, i.e. the backward match does not target to the original point $\mathbf{x}$.

Refinement 2: Data Energy of the Feature Match

Our initial goal is to integrate only feature matches that are presumed to improve the final result. In order to further enforce this goal, we compute the data energy $e_p(\mathbf{x})$ of each match and compare it to the data energy $e_1(\mathbf{x})$ of the baseline flow vector at the same location. Only if $e_p(\mathbf{x})$ is smaller than $e_1(\mathbf{x})$, i.e. the costs of the data term have improved using the feature match, we keep the corresponding feature match. Otherwise, we discard it and set $\chi_{\mathrm{P}}(\mathbf{x}) = 0$ at that location.

3.8.6 Confidence Measure $\rho_{\mathrm{P}}(\mathbf{x})$

Apart from the activation flag, we also have to define a confidence measure that rates the reliability of a match. Our measure consists of two components $\rho_{P1}(\mathbf{x})$ and $\rho_{P2}(\mathbf{x})$ which constitute the final confidence measure via

$$\rho_{\mathrm{P}}(\mathbf{x}) \quad = \quad \rho_{P1}(\mathbf{x}) \cdot \rho_{P2}(\mathbf{x})\,. \tag{3.8}$$

Component 1: Uniqueness of the Match

For the first component, we follow [27] and compute the confidence function

$$\rho_{P1}(\mathbf{x}) \quad = \quad \frac{d_2(\mathbf{x}) - d_1(\mathbf{x})}{d_1(\mathbf{x})}\,, \tag{3.9}$$

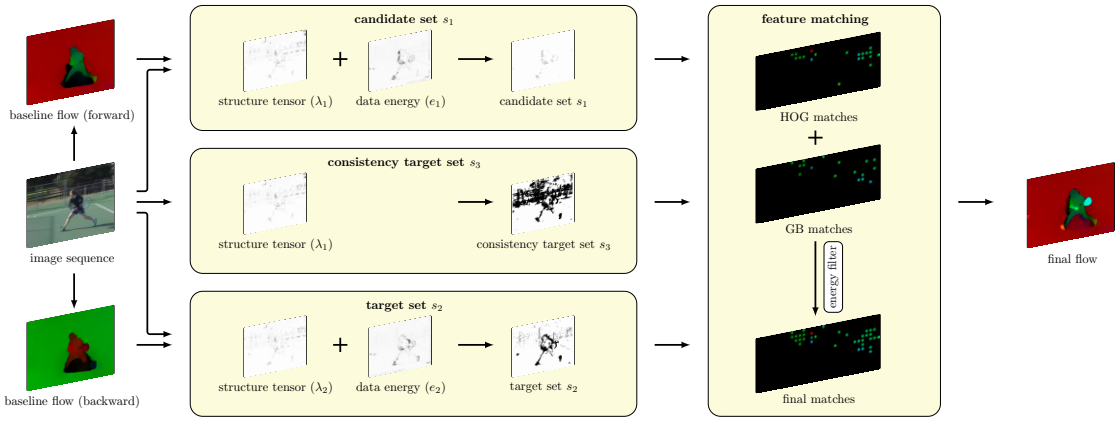


Figure 3.5: Detailed illustration of the flow estimation for Frame 496 of the Tennis sequence.

where $d_1(\mathbf{x})$ and $d_2(\mathbf{x})$ denote the matching costs of the best and the second best match, respectively (see also Chapter 2, Sect. 2.10.4). This function rates the uniqueness of the match.

Component 2: Improvement of the Data Energy

Additionally, we define a second confidence function

$$\rho_{P2}(\mathbf{x}) \quad = \quad \left(\frac{e_1(\mathbf{x})}{e_p(\mathbf{x})}\right)^2, \tag{3.10}$$

which is based on the data energies (reciprocal confidences) $e_1(\mathbf{x})$ and $e_p(\mathbf{x})$ of the baseline flow and of the feature match, respectively. This confidence measure assigns a higher weight to those feature matches that exhibit a stronger relative decrease of the local data energy, i.e. to those matches that are supposed to be more beneficial for the estimation. Please note that $\rho_{P2}(\mathbf{x}) > 1$ for all matches that have not been discarded.

Similar to our sparsification strategy where the structuredness measure targets the discriminativeness of the features and the data energy addresses the potential for an improvement in flow quality, our confidence measure $\rho_{\mathrm{P}}(\mathbf{x})$ respects both aspects, the feature uniqueness via $\rho_{P1}(\mathbf{x})$ and the potential to improve the flow quality via $\rho_{P2}(\mathbf{x})$.

3.8.7 Overview of the Method

Let us briefly recapitulate the three main steps of our method. In a first step, we compute dense flow fields forward and backward between both frames using our baseline method. Then, in a second step, we carefully select positions for feature matching based on

the analysis of the image structure and the baseline flow fields, and compute feature matches at those locations. In a final step, these matches are incorporated into the estimation using our complete model with the similarity term. In this context, the activation flag and the confidence measure are used as weights. A detailed illustration of the steps is depicted in Fig. 3.5.

3.9 Aspects of the Minimization

Basically, we minimize the nonconvex and nonlinear functional using concepts from Chapter 2. This is based on the coarse-to-fine warping strategy as described in Sect. 2.6.3 along with the lagged nonlinearity method as described in Sect. 2.3.1. After discretization, the resulting sequence of linear equation systems is solved with a successive overrelaxation scheme (SOR) as hinted in Sect. 2.3. This is similar to the minimization strategy of several other variational approaches from the literature [26, 29, 164, 27]. Moreover, we apply constraint normalization as described in Sect. 2.8.1.

3.9.1 Scale-Wise Weighting Scheme

An important novel aspect used in our minimization scheme is the scale-wise weight β^k of the similarity term that depends on the current level k of the coarse-to-fine warping scheme which is defined as

$$\beta^k \;:=\; \beta \cdot \left(\frac{k}{k_{\max}}\right)^{1.8}, \tag{3.11}$$

where $k_{\max}$ is the index of the coarsest level and $k = 0$ denotes the finest level. It provides the highest weight β at the coarsest level and evaluates to zero at the finest level. This weighting scheme makes the estimation more robust against persisting outliers in the matches, since both the data and the smoothness term will have a high local energy for such an outlier, guiding the estimation away from it. It follows the idea of Brox and Malik [27] who run one last iteration on the finest level using $\beta = 0$.

3.10 Evaluation

Let us now evaluate the performance of our *Adaptive Large Displacement Optical Flow (ALD-Flow)* method. To this end, we consider both synthetic and real-world sequences for which we performed a variety of experiments with small and large displacements. Since there is no ground truth for the sequences that contain relative large displacements, as they come from real-world data, the evaluation of our method is done visually. For the sequences of benchmarks, we conduct quantitative analyses by minimizing error measures on the given data and comparing the resulting error values. Details on the parameters and their retrieval can be found in Appendix A.5.

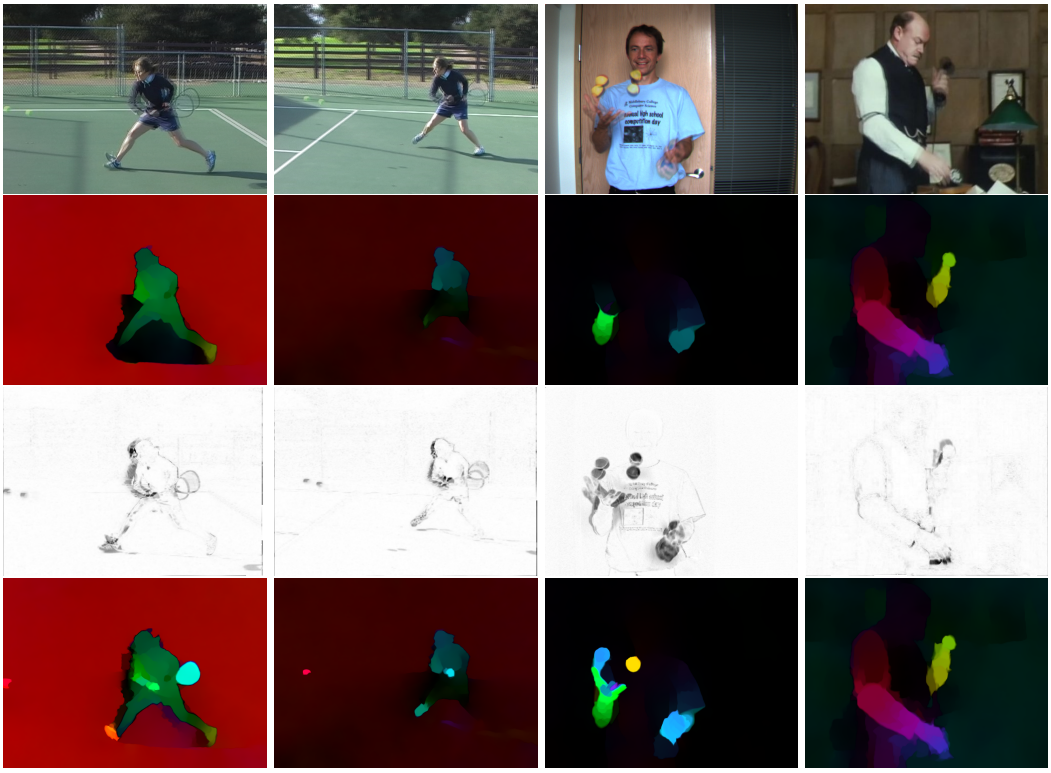


Figure 3.6: Condensed overview of the flow estimation for different large displacement sequences. **From Left to Right:** *Tennis* sequence (Frames 496 and 577), *Beanbags* sequence (Frame 10), *Miss Marple* sequence (Frame 52). **From Top to Bottom:** (a) Overlayed input images. (b) Initial flow field without feature matches. (c) Local energy of the data term (registration error). (d) Final result of our method.

3.10.1 Relative Large Displacements

In our first experiment, we focus on real-world sequences with relative large displacements. Therefore, we have applied our method to four popular image pairs from the literature: two from the *Tennis* sequence [27], one from the *Miss Marple* episode *"A pocket full of rye"* [27] and one from the *Beanbags* sequence known from the Middlebury benchmark [9]. The corresponding results obtained by our approach as well as the local energy of the data term are depicted in Fig. 3.6. As one can see, the energy of the data term is well suited to identify regions where supplementary feature matches can improve the estimation. Moreover, in contrast to the initial baseline method, our final approach is clearly able to handle large displacements correctly. This becomes explicit at various locations, e.g. at the tennis ball, the tennis racket, the right arm of the tennis player, the beanbags, both hands of the man throwing the beanbags and the thumb of the man picking up the phone.

Table 3.2: Comparison of the average angular error (AAE) for the *ALD-Flow*, the LDOF method [27] as well as the corresponding baseline methods. Results are given in degrees. Underlined fonts indicate the best results among a method and its baseline, while bold fonts indicate the overall best result.

Method	Avg.	Rub.	Hyd.	Gro2	Gro3	Urb2	Urb3	Dim.	Ven.
LDOF (base)	3.38	3.77	2.32	2.09	5.59	2.28	3.99	**1.82**	5.19
LDOF	3.93	3.94	2.44	2.68	6.38	2.64	5.07	1.85	6.45
Our baseline	2.62	**2.23**	**1.68**	**1.75**	5.05	2.11	2.83	1.83	3.50
ALD-Flow	**2.57**	**2.23**	**1.68**	**1.75**	**4.94**	**1.88**	**2.74**	1.83	**3.48**

Table 3.3: The average endpoint error (AEE) of *ALD-Flow* and its baseline method.

Method	Avg.	Rub.	Hyd.	Gro2	Gro3	Urb2	Urb3	Dim.	Ven.
Baseline	0.218	**0.068**	**0.135**	**0.118**	0.521	0.214	0.336	**0.096**	0.256
ALD-Flow	**0.212**	0.069	**0.135**	**0.118**	**0.513**	**0.202**	**0.309**	**0.096**	**0.255**

3.10.2 Small Displacements

In our second experiment, we investigate the impact of incorporating feature matches for sequences that *do not* contain large displacements. To this end, we evaluated our method on the Middlebury training data set. As one can see in Tab. 3.2 (AAE) and Tab. 3.3 (AEE), the results do not deteriorate when incorporating feature matches in our baseline method. In contrast, for some of the sequences we even observe notable improvements although the displacements are relatively small (Urban 2, Urban 3, Grove 3). These findings differ significantly from the ones reported in [27] for the LDOF method where outliers lead to severe degradations of the results (see also Tab. 3.2). We attribute this behavior to our adaptive sparsification strategy that carefully selects only those locations where supplementary feature matches are actually needed and that discards matches that might deteriorate the results.

The observed improvement in terms of quality and robustness compared to the LDOF method becomes even more obvious in Fig. 3.7, where we visually compare the corresponding results for some sequences of the Middlebury evaluation data set: We observe an enormous decrease in the number of artifacts caused by wrong feature matches while still being able to handle large displacements, as can be seen by the example of the *Backyard* sequence.

Besides the improvements in terms of quality and robustness, our adaptive approach offers another advantage: a highly reduced matching workload. This becomes evident in Tab. 3.4 that compares the amount of matching operations required for the LDOF method

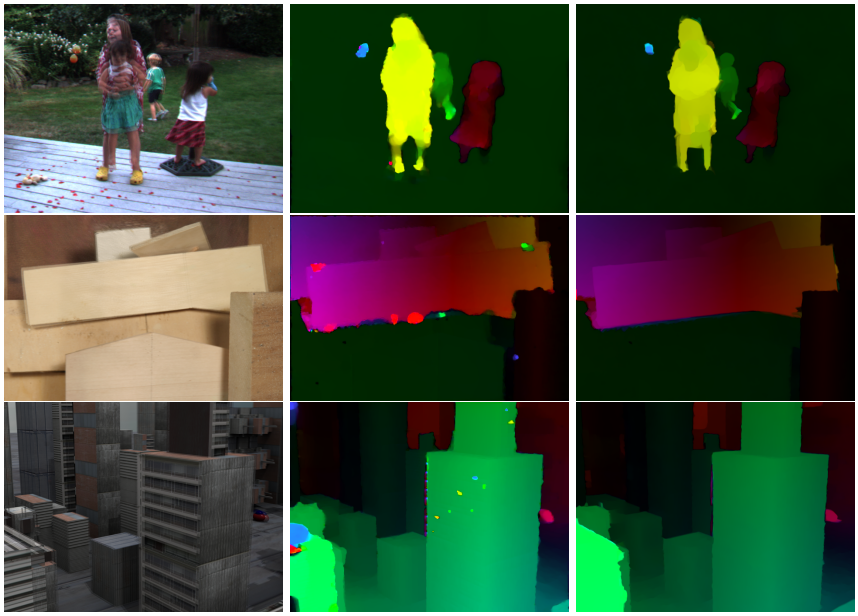


Figure 3.7: Comparison between LDOF and *ALD-Flow* with respect to outliers for some sequences from the Middlebury benchmark. **From top to down:** Backyard, Wooden and Urban. **From left to right:** Overlayed frames, LDOF, *ALD-Flow*.

Table 3.4: Comparison of feature statistics between LDOF and *ALD-Flow*. Listed are the number of extracted features for both frames, the number of computed feature matches and the required number of feature comparisons to compute these matches.

Method		Beanbags		Backyard		Basketball	
LDOF	#Features F1	287K		287K		287K	
	#Features F2	287K		287K		287K	
	#Matches	12K		13K		8K	
	#Comparisons	6,753,628K		7,746,183K		4,537,080K	
ALD-Flow	#Features F1	89K	(31.0%)	88K	(30.7%)	90K	(31.4%)
	#Features F2	24K	(7.3%)	19K	(6.6%)	24K	(8.4%)
	#Matches	1,402	(11.7%)	1,393	(10.7%)	1,641	(20.5%)
	#Comparisons	158,602K	(2.3%)	149,484K	(1.9%)	186,972K	(4.1%)

and *ALD-Flow*. As one can see, our approach reduces the number of comparisons to determine the matches by up to two orders of magnitude compared to the sparsification and structuredness criteria from [27].

Table 3.5: Evaluation of the impact of different combinations of selection criteria and confidence strategies on the final result. The reported average angular error (AAE) has been computed for the Middlebury training data set.

Selection	Confidence	Avg.	Rub.	Hyd.	Gro2	Gro3	Urb2	Urb3	Dim.	Ven.
No Selection	Uniform	5.04	**2.23**	**1.68**	**1.75**	5.25	3.34	2.78	2.27	21.04
Struct. (λ_1, λ_2)	Uniqueness	3.19	2.44	2.12	1.77	5.03	3.42	4.13	**1.83**	4.78
Energy (e_1, e_2)	Energy	2.68	2.25	**1.68**	1.76	5.01	2.74	**2.68**	**1.83**	3.51
Both	Both	**2.57**	**2.23**	**1.68**	**1.75**	**4.94**	**1.88**	2.74	**1.83**	**3.48**

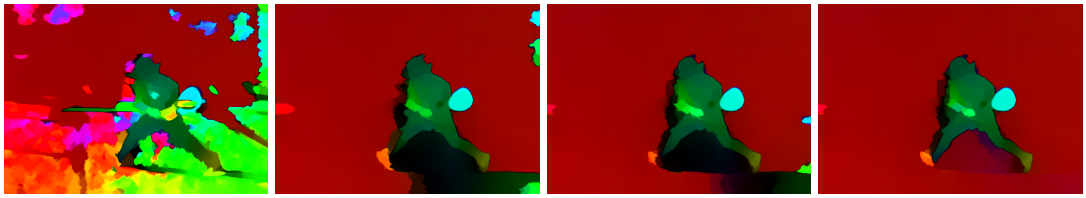


Figure 3.8: Results for the *Tennis* sequence (Frame 496) for different combinations of selection criteria and confidence strategies. **From Left to Right:** (a) No selection + uniform (46336 matches). (b) Structure + uniqueness (9573 matches). (c) Energy + energy (1220 matches). (d) Both + both (only 734 matches).

3.10.3 Component Analysis

In our third experiment, we analyze the impact of different combinations of selection criteria and confidence strategies on the final result. To this end, we used different variants of our *ALD-Flow* method and computed the results for the Middlebury training data set. As one can see from the result in Tab. 3.5, already the novel energy-based selection-criterion in combination with the energy-based confidence measure yields good results. However, combining structure- and energy-based selection criteria and confidence strategies clearly yields the best results. This also confirmed by Fig. 3.8 that depicts results for the *Tennis* sequence (Frame 496) based on the same combinations.

3.10.4 Sensitivity to Variations of the Thresholds

In our fourth experiment, we investigate the sensitivity of the thresholds for feature selection, which control the amount of feature matches that are integrated into the variational optical flow estimation. To this end, we used our approach with optimal settings and varied *all* thresholds from 40 to 160 percent of their original value. The results in Fig. 3.9 show only slight deteriorations for moderate parameter variations (80%, 120%) and more pronounced degradations for strong parameter changes (40%, 160%). This demonstrates that the approach offers a certain stability w.r.t. moderate parameter variations.

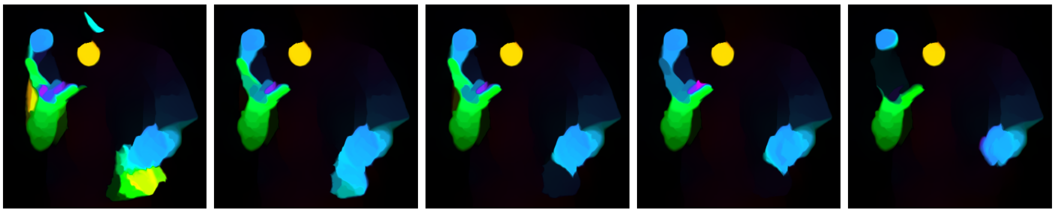


Figure 3.9: Evaluation of the sensitivity of the threshold parameters for the *Beanbags* sequence (zoom-ins). **From Left to Right:** All thresholds changed to (a) 40 percent, (b) 80 percent, (c) 100 percent, (d) 120 percent, (e) 160 percent of the optimal value.

Average angle error	avg. rank	Army (Hidden texture) GT im0 im1			Mequon (Hidden texture) GT im0 im1			Schefflera (Hidden texture) GT im0 im1			Wooden (Hidden texture) GT im0 im1			Grove (Synthetic) GT im0 im1			Urban (Synthetic) GT im0 im1			Yosemite (Synthetic) GT im0 im1			Teddy (Stereo) GT im0 im1		
		all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
nLayers [61]	6.5	**2.80** 1	**7.42** 1	2.20 2	2.71 14	7.24 2	2.55 30	2.61 2	6.24 2	2.45 28	2.30 2	12.7 5	1.16 2	**2.30** 1	**3.02** 1	**1.70** 1	2.62 2	**6.95** 1	2.09 2	2.29 18	3.46 13	1.89 16	1.38 3	3.06 4	1.29 3
ADF [72]	8.7	2.98 4	8.32 6	2.28 3	2.27 3	8.35 9	1.81 4	3.55 15	9.74 16	2.17 15	3.15 23	16.8 24	1.29 5	2.64 4	3.55 5	1.81 2	3.02 4	9.08 3	2.38 4	2.29 18	3.48 15	2.07 21	1.34 2	3.03 2	1.11 2
Layers++ [38]	10.9	3.11 6	8.22 4	2.79 18	2.43 7	**7.02** 1	2.24 16	**2.43** 1	**5.77** 1	2.18 17	**2.13** 1	**9.71** 1	**1.15** 1	2.35 2	**3.02** 1	1.96 4	3.81 22	11.4 18	3.22 26	2.74 34	4.01 37	2.35 27	1.45 4	3.05 3	1.79 10
IROF++ [62]	11.8	3.17 10	8.69 9	2.61 8	2.79 15	9.61 15	2.33 19	3.43 9	8.86 11	2.38 22	2.87 13	14.8 13	1.52 17	2.74 7	3.57 6	2.19 8	3.20 10	9.70 10	2.71 11	1.96 8	3.45 12	1.22 5	1.80 11	4.06 13	2.50 20
ALD-Flow [73]	12.0	2.82 2	7.86 2	**2.16** 1	2.84 18	10.1 19	1.86 8	3.73 17	10.4 19	1.67 3	3.10 19	16.8 24	1.28 4	2.69 6	3.60 7	1.85 3	2.79 3	11.3 17	2.32 3	2.07 10	3.25 5	3.10 48	2.03 18	5.11 20	1.94 12
MDP-Flow2 [40]	12.2	3.32 16	8.76 12	2.85 20	**2.18** 1	7.47 4	1.85 7	2.77 4	6.95 4	2.06 12	3.25 25	17.3 30	1.59 23	2.87 13	3.73 11	2.32 11	3.15 8	11.1 16	2.65 7	2.04 9	3.64 21	1.60 9	1.88 12	4.49 15	1.49 4
Efficient-NL [65]	13.0	3.01 5	8.29 5	2.30 5	3.12 29	10.3 21	2.40 20	3.83 20	9.97 18	2.08 13	2.76 11	14.4 11	1.45 9	2.64 4	3.51 4	2.07 7	3.06 5	8.23 2	2.49 6	2.53 27	3.73 23	2.46 29	1.91 13	3.32 7	2.40 19
Sparse-NonSparse [59]	13.1	3.14 8	8.75 11	2.76 15	3.02 26	10.6 23	2.43 24	3.45 12	8.96 12	2.36 20	2.66 6	13.7 7	1.42 6	2.85 12	3.75 13	2.33 12	3.28 11	9.40 7	2.73 12	2.42 23	3.31 6	2.69 32	1.47 5	3.07 5	1.66 7
LSM [41]	14.2	3.12 7	8.62 8	2.75 14	3.00 25	10.5 22	2.44 25	3.43 9	8.85 10	2.35 19	2.66 6	13.6 6	1.44 7	2.82 9	3.68 8	2.36 14	3.38 14	9.41 8	2.81 16	2.69 32	3.52 18	2.84 36	1.59 7	3.38 9	1.80 11
⋮	⋮		⋮			⋮									⋮									⋮	
Complementary OF [21]	26.2	4.44 41	11.2 37	4.04 44	2.51 11	9.77 17	1.74 3	3.93 22	10.6 22	2.04 10	3.87 36	18.8 34	2.19 36	3.17 20	4.00 19	2.92 35	4.64 39	13.8 28	3.64 34	2.17 13	3.36 9	2.51 30	3.08 27	7.04 26	3.65 37
⋮	⋮		⋮			⋮									⋮									⋮	
Brox et al. [5]	37.4	4.44 41	12.4 41	4.22 48	3.72 39	13.5 41	3.06 40	4.97 29	13.3 32	3.11 37	4.58 43	22.0 46	2.37 40	3.79 49	4.60 43	4.33 61	3.91 26	17.0 49	3.45 29	2.22 17	3.79 26	1.19 4	4.62 43	10.0 42	3.38 32
⋮	⋮		⋮			⋮									⋮									⋮	
LDOF [28]	44.7	4.60 45	13.0 47	3.77 41	4.67 47	15.5 53	3.67 44	5.63 35	14.0 33	4.21 49	5.80 49	27.1 63	3.43 48	3.52 41	4.50 38	3.46 45	4.84 44	17.8 52	4.04 38	2.46 25	4.14 40	3.25 53	4.85 47	12.0 55	3.78 41

Figure 3.10: Rank of our *ALD-Flow* in the Middlebury benchmark w.r.t. the average angular error (AAE) (time of submission: June 28th, 2012).

3.10.5 Comparison to the Literature

In our fifth experiment, we compare the performance of our method to that of other approaches from the literature. This is done by means of the Middlebury evaluation data set (some results have already been shown in Fig. 3.7). The corresponding table for the average angular error (AAE) is depicted in Fig. 3.10 and the table for the average endpoint error (AEE) is depicted in Fig. 3.11. The tables show that our method achieved ranks five and seven out of 73 methods at the time of submission (June 28th, 2012). Moreover, it significantly outperforms the Complementary Optical Flow method [165] (rank 26), the LDOF-method [27] (rank 47) and the LDOF-baseline [26] (rank 36).

3.11 Additional Evaluation

In meantime since the development of our method, a lot of progress in the field has been done. Amongst others, this comprises the development of improved features such as Deep Matches [154, 112] and the publication of a variety of new benchmarks, i.e. the KITTI 2012 [52] benchmark, the MPI Sintel [31] benchmark and the KITTI

Average endpoint error	avg. rank	Army (Hidden texture) GT im0 im1			Mequon (Hidden texture) GT im0 im1			Schefflera (Hidden texture) GT im0 im1			Wooden (Hidden texture) GT im0 im1		
		all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
ADF [72]	8.6	0.08 4	0.22 5	**0.06** 1	0.18 3	0.62 8	0.14 8	0.29 14	0.71 17	0.17 7	0.16 15	0.91 25	**0.07** 1
IROF++ [62]	9.0	0.08 4	0.23 7	0.07 8	0.21 17	0.68 15	0.17 18	0.28 10	0.63 8	0.19 18	0.15 8	0.73 10	0.09 12
Layers++ [38]	9.2	0.08 4	0.21 2	0.07 8	0.19 7	0.56 3	0.17 18	**0.20** 1	**0.40** 1	0.18 12	**0.13** 1	0.58 2	**0.07** 1
MDP-Flow2 [40]	9.4	0.09 15	0.23 7	0.07 8	0.16 2	**0.52** 1	0.13 3	0.22 2	0.46 3	0.17 7	0.17 21	0.93 31	0.09 12
nLayers [61]	9.8	**0.07** 1	**0.19** 1	**0.06** 1	0.22 20	0.59 4	0.19 29	0.25 6	0.54 5	0.20 27	0.15 8	0.84 18	0.08 6
Sparse-NonSparse [59]	13.4	0.08 4	0.23 7	0.07 8	0.22 20	0.73 22	0.18 21	0.28 10	0.64 9	0.19 18	0.14 4	0.71 7	0.08 6
ALD-Flow [73]	13.5	**0.07** 1	0.21 2	**0.06** 1	0.19 7	0.64 13	0.13 3	0.30 17	0.73 18	0.15 2	0.17 21	0.92 29	**0.07** 1
COFM [63]	13.6	0.08 4	0.26 24	**0.06** 1	0.18 3	0.62 8	0.14 8	0.30 17	0.74 19	0.19 18	0.15 8	0.86 20	**0.07** 1
Efficient-NL [65]	13.9	0.08 4	0.22 5	**0.06** 1	0.23 30	0.73 22	0.18 21	0.32 22	0.75 20	0.18 12	0.14 4	0.72 9	0.08 6
⋮	⋮		⋮			⋮							
Complementary OF [21]	29.3	0.11 32	0.28 31	0.10 40	0.18 3	0.63 10	0.12 2	0.31 20	0.75 20	0.18 12	0.19 30	0.97 33	0.12 32
⋮	⋮		⋮			⋮							
Brox et al. [5]	38.5	0.11 32	0.32 40	0.11 46	0.27 40	0.93 41	0.22 42	0.39 29	0.94 28	0.24 38	0.24 40	1.25 45	0.13 35
⋮	⋮		⋮			⋮							
LDOF [28]	45.2	0.12 42	0.35 46	0.10 40	0.32 45	1.06 51	0.24 44	0.43 30	0.98 32	0.30 49	0.45 55	2.48 71	0.26 51

Average endpoint error	Grove (Synthetic) GT im0 im1			Urban (Synthetic) GT im0 im1			Yosemite (Synthetic) GT im0 im1			Teddy (Stereo) GT im0 im1		
	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
ADF [72]	0.69 12	1.03 12	0.47 8	0.43 11	0.91 3	0.28 6	0.12 15	0.12 6	0.20 18	**0.43** 1	**0.88** 1	0.63 5
IROF++ [62]	0.60 3	0.89 4	0.42 5	0.43 11	1.08 10	0.31 12	0.10 4	0.12 6	0.12 4	0.47 5	0.98 7	0.68 10
Layers++ [38]	**0.48** 1	**0.70** 1	0.33 2	0.47 19	1.01 6	0.33 18	0.15 37	0.14 30	0.24 31	0.46 3	**0.88** 1	0.72 13
MDP-Flow2 [40]	0.65 7	0.98 9	0.43 6	**0.29** 1	0.91 3	0.26 4	0.11 9	0.13 18	0.17 11	0.51 15	1.11 18	0.72 13
nLayers [61]	0.53 2	0.78 2	0.34 3	0.44 14	**0.84** 1	0.30 10	0.13 23	0.13 18	0.20 18	0.47 5	0.97 6	0.67 8
Sparse-NonSparse [59]	0.67 11	0.99 11	0.48 11	0.49 23	1.06 8	0.32 15	0.14 29	**0.11** 1	0.28 44	0.49 9	0.98 7	0.73 17
ALD-Flow [73]	0.78 21	1.14 21	0.59 21	0.33 3	1.30 22	**0.21** 1	0.12 15	0.12 6	0.28 44	0.54 18	1.19 21	0.73 17
COFM [63]	0.79 22	1.14 21	0.74 36	0.35 5	0.87 2	0.28 6	0.14 29	0.12 6	0.28 44	0.49 9	0.94 4	0.71 12
Efficient-NL [65]	0.60 3	0.88 3	0.43 6	0.57 32	1.11 14	0.35 24	0.14 29	0.13 18	0.25 34	0.48 7	0.90 3	0.63 5
⋮		⋮									⋮	
Complementary OF [21]	0.97 44	1.31 40	1.00 49	1.78 65	1.73 45	0.87 58	0.11 9	0.12 6	0.22 26	0.68 29	1.48 30	0.95 37
⋮		⋮									⋮	
Brox et al. [5]	1.10 56	1.39 51	1.43 63	0.89 44	1.77 49	0.55 41	0.10 4	0.13 18	0.11 2	0.91 44	1.83 48	1.13 47
⋮		⋮									⋮	
LDOF [28]	1.01 48	1.37 46	1.05 50	1.10 47	2.08 58	0.67 48	0.12 15	0.15 40	0.24 31	0.94 47	2.05 56	1.10 44

Figure 3.11: Rank of our *ALD-Flow* in the Middlebury benchmark w.r.t. the average endpoint error (AEE) (time of submission: June 28th, 2012).

2015 [92] benchmark that can be publicly used for performance evaluation. Hence, we additionally demonstrate the results of our method on those benchmarks and, moreover, investigate how our adaptive sparsification strategy performs when applied to more modern features by conducting an experiment that incorporates Deep Matches instead of HOG [36] and GB [14] feature matches.

3.11.1 Variational Framework Implementation

For the additional experiments, we have embedded our method into a more sophisticated coding framework that easily allows us to modify both individual components of our strategy as well as terms in the variational model. The most important difference is that it offers an implementation for the second-order smoothness term as described in Chapter 2 (Sect. 2.9) that allows us to conduct meaningful experiments on the KITTI benchmarks. Since, moreover, it also offers faster numerical solvers and other minor numerical improvements, results are a bit different and we, hence, start by investigating the results for the Middlebury benchmark that are obtained using the new implementation.

3.11.2 Comparison among Both Implementations

Tab. 3.6 and Tab. 3.7 depict the results of the old and the new implementation of *ALD-Flow*, each including the corresponding baseline method. When comparing both implementations, the new implementation overall obtains slightly better results (an AAE of 2.55 vs. 2.57 and an AEE of 0.212 vs. 0.212 for *ALD-Flow* as well as an AAE of 2.59 vs. 2.62 and an AEE of 0.215 vs. 0.218 for the baseline method). Moreover, *ALD-Flow* is superior to its baseline method with an AAE of 2.55 vs. 2.59 and an AEE of 0.212 vs. 0.215.

Table 3.6: Comparison of the average angular error (AAE) for both implementations of *ALD-Flow* as well as the corresponding baseline methods. Results are given in degrees. Underlined fonts indicate the best results among a given implementation while bold fonts indicate the overall best result.

Method	Avg.	Rub.	Hyd.	Gro2	Gro3	Urb2	Urb3	Dim.	Ven.
Baseline (old)	2.62	**2.23**	**1.68**	**1.75**	5.05	2.11	2.83	1.83	3.50
ALD-Flow (old)	2.57	**2.23**	**1.68**	**1.75**	4.94	**1.88**	2.74	1.83	**3.48**
Baseline (new)	2.59	2.43	1.71	1.76	5.07	2.01	2.56	**1.63**	3.52
ALD-Flow (new)	**2.55**	2.42	1.71	1.77	**4.92**	2.00	**2.41**	**1.63**	3.54

Table 3.7: Comparison of the average endpoint error (AEE) for both implementations of *ALD-Flow* as well as the corresponding baseline methods. Underlined fonts indicate the best results among a given implementation while bold fonts indicate the overall best result.

Method	Avg.	Rub.	Hyd.	Gro2	Gro3	Urb2	Urb3	Dim.	Ven.
Baseline (old)	0.218	**0.068**	**0.135**	0.118	0.521	0.214	0.336	0.096	0.256
ALD-Flow (old)	**0.212**	0.069	**0.135**	0.118	0.513	**0.202**	0.309	0.096	**0.255**
Baseline (new)	0.215	0.074	0.138	**0.117**	0.520	0.238	0.293	**0.085**	**0.255**
ALD-Flow (new)	**0.212**	0.073	0.138	0.118	**0.512**	0.248	**0.265**	**0.085**	0.256

3.11.3 Performance on Major Benchmarks

In order to see how well our strategy generalizes, we conduct further experiments on all major benchmarks. Please note that for the KITTI benchmarks we resort to second-order regularization, as already stated in Chapter 2 (Sect. 2.9.1). Tab. 3.8 displays the corresponding results w.r.t. the associated error measure for both the baseline method and our *ALD-Flow*. From this overview, it becomes obvious that our *ALD-Flow* method slightly improves results in four out of five cases. Only for the KITTI 2012 benchmark, the results are worse. However, this is the only benchmark that contains only ego motion which is completely regular. The unregularized matching step using HOG- and GB-features can harm the estimation while the potential to improve results (compared to a regularized variational baseline) in a setting with a completely regular motion is lower compared to the other benchmarks that contain more dynamic motions. But even in this case, there is only a slight overall deterioration.

Table 3.8: Results of *ALD-Flow* and its baseline on training data from different benchmarks.

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline	2.59	0.215	7.055	4.722	**10.39%**	24.56%
ALD-Flow	**2.55**	**0.212**	**7.039**	**4.684**	10.68%	**24.48%**

3.11.4 Integration of Improved Matches

A further interesting question is whether our adaptive sparsification strategy is also useful if we use more advanced features that produce less false positives. To answer this question, we make use of Deep Matches [154, 112] that have been proposed after the publication of our method and have been widely used in the literature, e.g. in [154, 111, 44, 86], since the original work demonstrated clear improvements over HOG- and SIFT-features in the context of optical flow. Hence, we replaced the HOG and GB matches by Deep Matches and integrated them once on the original provided grid of locations and once on a grid of locations that is obtained after applying our adaptive sparsification strategy. Please note that we have used the original implementation of the authors to conduct the matching and applied our strategy *after* we obtained the set of matches. We, hence, only applied the sparsification to the (already sparse) set s_1 of matching locations but not to any target sets, i.e. not to the sets s_2 or s_3, since the matching process is not affected by our strategy in this case.

Small Displacements

Let us now compare the performance of our adaptive sparsification strategy applied on Deep Matches to the performance of the baseline and to the performance of the baseline with a direct integration of all matches. To this end, we evaluated our strategy once with a common set of thresholds for all benchmarks – indicated as *ALD+Deep* – and once with thresholds optimized separately for each benchmark – indicated as *ALD+Deep (opt. θ)*. The results are given in Tab. 3.9.

There are manifold observations to be discussed. We can see that a direct integration of Deep Matches on average improves results compared to the baseline by 9.8% – except for the Middlebury benchmark where results drop by 4 to 6 percent. Using a common set of thresholds for all benchmarks, our sparsification strategy further improves results by 2% percent on average compared to a direct integration, but leads to slight deteriorations of less than 1 percent for the KITTI benchmarks. Compared to the baseline, results are still superior in almost all cases (except for the AAE error on the Middlebury data) with average improvements of 12%. This already is a convincing result, since in any

Table 3.9: Comparison of the baseline and a direct integration as well as adaptive integrations of Deep Matches into the baseline (*ALD*+Deep and *ALD*+Deep (opt. θ)).

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline	2.59	0.215	7.055	4.722	10.39%	24.56%
ALD-Flow	**2.55**	0.212	7.039	4.684	10.68%	24.48%
ALD+Deep (direct)	2.69	0.227	5.529	**2.952**	10.13%	22.90%
ALD+Deep	2.60	**0.209**	**5.205**	3.055	10.20%	23.07%
ALD+Deep (opt. θ)	2.56	0.213	**5.205**	3.055	**9.87%**	**22.60%**

comparison the improvements of our adaptive sparsification strategy on the one hand supersede the slight deteriorations on the other hand. If, however, we decide to refrain from using a common set of thresholds for the adaptive integration but optimize them individually for each benchmark, we obtain a consistent and significant improvement compared to both the baseline (by 12.8%) and to a direct integration of Deep Matches (by 2.9%). Compared to a direct integration, only on the complete Sintel data set there is a slight drop in performance. This, however, may be explained by the fact that we have not conducted separate cumbersome optimizations on the complete data set but used the parameters that have been obtained from the optimizations on the subset also on the complete data set. When comparing conventional *ALD-Flow* with HOG- and GB-matches to its counterpart with Deep Matches, the errors on average drop by 9.6% for a direct integration of Deep Matches, by 11.8% for an adaptive integration with common thresholds and by 12.6% with individual sets of thresholds for each benchmark.

Discriminativeness of Deep Features. An observation that confirms the higher discriminativeness of Deep Features compared to HOG or GB features is given by the optimized common thresholds for our sparsification strategy. While the high values for these thresholds for the HOG and GB features (> 80%, see Tab. 3.1) are necessary in all benchmarks in order to obtain accurate results, the common thresholds used for the superior Deep Features all lie in a range between 50 and 60 percent for the respective quantiles. This indicates that compared to the other features more feature matches are helpful during the estimation of the optical flow when using Deep Features. Considering the thresholds that were optimized individually for the Middlebury benchmark, they are relatively high (64% for the energy threshold θ_e, 99% for the structuredness threshold θ_λ). This is no surprise, since these matches deteriorated results within a direct integration and hence have to be integrated with care.

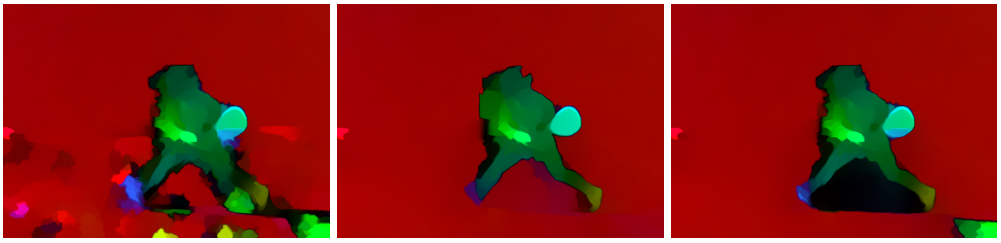


Figure 3.12: Comparison of a non-adaptive and two adaptive integrations of Deep Matches into our baseline. **From left to right:** (a) Non-adaptive integration. (b) Adaptive integration with the same thresholds as for the HOG- and GB-features. (c) Adaptive integration with lower thresholds between 50 and 60 percent.

Large Displacements

Similar findings can be drawn from a visual comparison on the Tennis sequence in Fig. 3.12. We observe that a direct, non-adaptive integration of all Deep Matches leads to considerable artifacts in the optical flow. In contrast, an adaptive integration avoids such artifacts – whereby we additionally compare different thresholds. While the original thresholds, as used in the case of HOG- and GB-features, throw away too many matches, thresholds between 50 and 60 percent (as learned for Deep Matches on the Sintel benchmark) lead to convincing results, since the flow is more accurate at the arm, the tennis racket and the right foot of the player while not showing severe artifacts.

Influence of the Selection Criteria

While the overall improvements of applying our strategy to Deep Matches are already significant, we are furthermore interested in the influence of the different parts of the sparsification strategy, i.e. the structuredness-based part and the energy-based part. Please note that in contrast to the corresponding experiment on HOG- and GB-matches in Sect. 3.10.3, the sparsification is only applied to the candidate set s_1 and we do not vary the confidence measure. Here, we rely on the publicly available implementation of Deep Matching without any modifications where there is no single confidence measure that we can dedicate to the structure-based part of our sparsification strategy. In contrast, it provides an autocorrelation-based confidence value for each match which we combine with our energy-based confidence measure in all cases (*mixed*).

The corresponding results can be found in Tab. 3.10, where compared to Tab. 3.9 we additionally state results on the Middlebury and Sintel benchmarks, where only one part of our strategy is active. Also in this case, we can observe that each part alone already improves results while the combination of both gives by far the best results. Evidently, the results demonstrate that our adaptive sparsification strategy provides good performance both for different types of features and on different benchmarks.

Table 3.10: Comparison of the influence of the structuredness and the energy criteria in the adaptive sparsification strategy.

Selection	Confidence	Middlebury (AAE)	Middlebury (AEE)	Sintel (sub.) (AEE)
No Selection	mixed	2.69	0.227	5.529
Structure (λ_1)	mixed	**2.56**	0.214	5.373
Energy (e_1)	mixed	2.59	0.215	5.425
Both	mixed	**2.56**	**0.213**	**5.205**

3.12 Summary

In this chapter, we have addressed the problem of robustly integrating large displacement feature matches into variational optical flow methods in order to achieve accurate results containing arbitrarily large displacements. To this end, we built upon the method of Brox and Malik [27], which integrates pre-computed feature matches into a variational optical flow estimation via a similarity term in the variational model. In the feature matching step, this method inspects the image structure when selecting feature locations in order to match only discriminative features.

In addition, we also considered the matching energy of the baseline method as an indicator of the flow quality for this task. This means that we did not only consider where there is enough structure to assemble a meaningful descriptor but we also determined if the baseline flow locally leaves room for improvements (due to being inaccurate). By respecting both aspects we avoided the integration of feature matches at locations where the only possible effect is a deterioration of the results. Moreover, the matching energy enabled us to sort out unreliable matches before the integration and it helped us to determine the reliability of the remaining matches.

Our experiments have demonstrated that our adaptive sparsification strategy based on structure *and* energy is very useful: When integrating the corresponding feature matches into our baseline method, we succeeded in handling large displacements while maintaining or even improving the accuracy of small displacements at the same time. We could not only observe such improvements for matches using traditional features like the Histogram of Oriented Gradients (HOG) or the Geometric Blur (GB) features but we have also seen that even the newer and more sophisticated Deep Matches, whose development was dedicated to the area of motion estimation, can be filtered effectively to obtain improved results. Thus we demonstrated that it is possible to combine feature-based and variational methods without compromising their advantages.

Chapter

4 Moderately Large Displacement Optical Flow

A borderline case w.r.t. relative large displacements is the scenario of moderately large displacements. Here, an object is smeared with its background but remains still distinguishable on that resolution level of the coarse-to-fine scheme that is appropriate to estimate its motion.

Basically, this case could also be covered by methods that estimate arbitrarily large displacements. As we have seen in the previous chapter, such methods typically obtain their large displacement abilities by integrating matching steps that are not inherently regularized. The main problem is: Without regularization, this may lead to a deterioration of the result by *arbitrarily* large false matches. While making such methods more adaptive addresses the problem a-posteriori, it is also worthwhile to consider avoiding these false matches a-priori by using regularized methods. This includes variational approaches with coarse-to-fine schemes for the estimation of such potential motion candidates. Since the respective object has not completely disappeared on the appropriate level in the coarse-to-fine scheme, let us now analyze how such methods can be adapted in order to make the correct estimation possible.

4.1 A Balancing Problem

If a pixel does not uniquely belong to a single object, there are – depending on the perspective – either multiple correct flow vectors or none. Since we aim at computing dense flow fields, we discard the latter perspective and thus have to consider the case of multiple possible flow candidates. Hence, these candidates comprise the flow that corresponds to the background motion and the flow that corresponds to the motion of the small object. The important question now is: *Which flow will come out on top within a standard variational framework?*

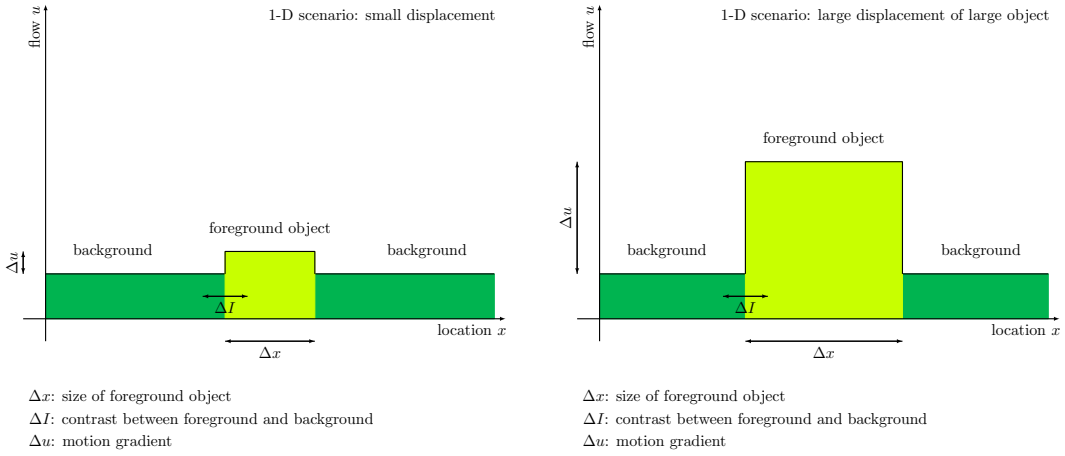
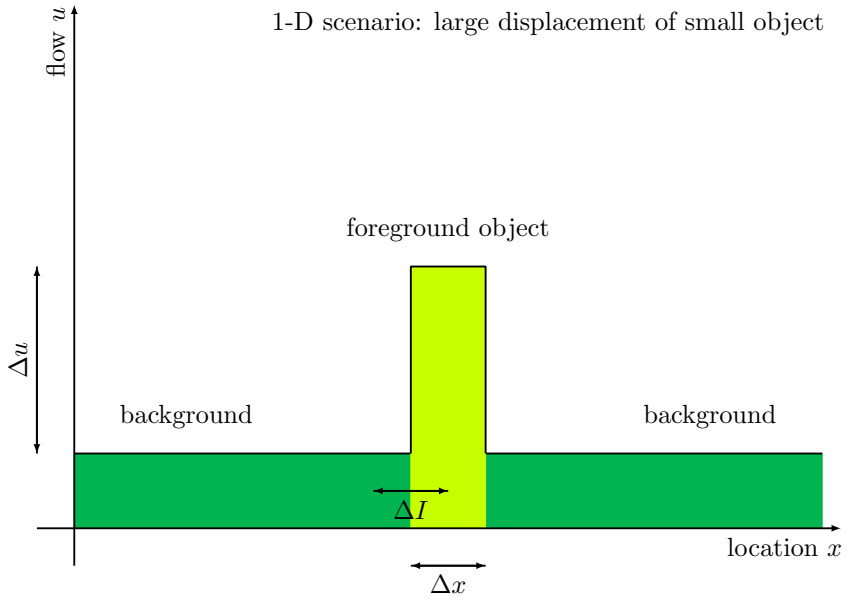


Figure 4.1: 1-D Illustrations of different displacement scales that can be handled by coarse-to-fine schemes. **From left to right:** Small displacement, large displacement of a large object.

Data Term and Smoothness Term. To this end, let us have a look at the two terms that constitute the variational model: the data term and the smoothness term. First of all, we can state that, in any case, the smoothness term favors the background flow: It does not introduce any motion gradient, while a relative large displacement introduces large motion gradients being a clear violation of the smoothness assumption. In contrast, the data term favors the flow candidate of that object/background that dominates in the mixture of colors. In case of a small object that dominates this mixture, the multi-objective variational optimization is trapped within a balancing problem.

4.1.1 A Simple Example in 1-D

Let us illustrate a simplified version of this problem in a 1-D scenario (see Fig. 4.1). To this end, we make the following simplifying assumptions: (i) There is one unicolored foreground object in front of some unicolored background. (ii) Our data term is given by the brightness constancy assumption with an (almost) linear penalizer. (iii) The estimated flow of the foreground object either completely maps the object to itself or completely maps it to the background. (iv) There is a smoothness term with an (almost) linear penalizer. With these assumptions, the data costs of the foreground object are given by $I_u \cdot \Delta x$, where Δx is the size of the object and I_u is constant within the object – either given as $I_u = 0$ for the correct match or $I_u = \Delta I$ being the difference ΔI between the colors of foreground and background for the incorrect match. Moreover, the smoothness costs for the foreground object linearly depend on the motion gradient u_x – which either reads $u_x = \Delta u$ being the difference between the foreground and the background motion for the correct match or it reads $u_x = 0$ for the incorrect match



Δx: size of foreground object
ΔI: contrast between foreground and background
Δu: motion gradient

Figure 4.2: 1-D Illustration of a relative large displacement

(due to estimating the background motion everywhere). In our 1-D example, the x-axis covers the spatial dimension x of the scenario, the y-axis depicts the flow u and the colored area shows the colors of the objects.

Costs of Different Scales of Displacements. We will now compare, how different scales of displacements and objects influence potential violations of both terms. In Fig. 4.1, we depict the standard scenarios that can be handled by conventional coarse-to-fine schemes: a small displacement scenario (on the left) and a large displacement scenario with a large object. In the small displacement scenario, the motion gradient u_x is small in any case, such that the solution mainly depends on the data costs ($I_u \cdot \Delta x$). Since these are typically only small for the correct solution ($I_u = 0$) given a sufficient size Δx of the object, the motion estimate of the foreground object will be the correct flow. In the case of a large displacement of a large object, both Δu, which influences the smoothness costs, and Δx, which influences the data costs, increase, such that the ratios between both types of costs are similar. In Fig. 4.2, we depict the scenario of a relative large displacement. In this case, the correct motion gradient $u_x = \Delta u$ highly violates the smoothness term. Moreover, the small object size Δx likely leads to small data costs $I_u \cdot \Delta x$ for any solution. Hence, the solution is mainly steered by the smoothness term which favors an estimate that corresponds to the background flow vector, since in this

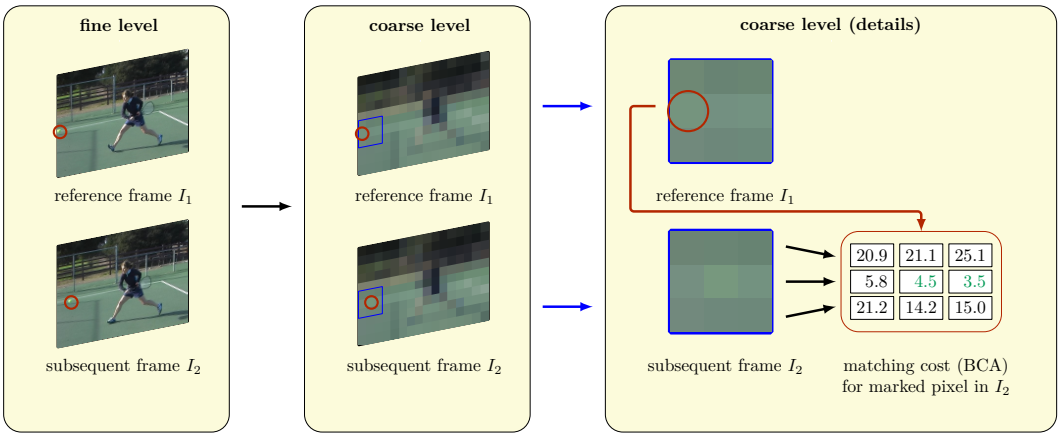


Figure 4.3: Illustration of the data costs at hand of the tennis ball of the Tennis sequence [27]. At the coarse level, the tennis ball is hardly distinguishable from the background but the displacement between the correct pixels still gives the least data costs.

case $u_x = 0$ holds. Here, we have an inconvenient balance between the size of the object Δx, which here downweights the data term, and the motion gradient Δu, which here makes the correct estimate too expensive to be chosen.

4.1.2 Balances within the Coarse-to-Fine Scheme

Let us now have a look at the balance between these terms within the coarse-to-fine scheme. Usual smoothness weights are appropriate to estimate small displacements and large displacements of large objects in a piecewise smooth flow field that does not contain noise. This is possible, since on the respective level where the displacements can be estimated, the objects are big enough to make a violation of the data term too expensive and/or the motion gradients are small enough to prevent the smoothness term from producing high energy costs. For a relative large displacement, the appropriate resolution level of this displacement becomes coarser, the object is smaller and thus the support of the data term, that is in favor of the correct flow candidate, shrinks. Moreover, due to the inevitable mixing of colors, which is the result of aliasing-free resampling, the contrast of the object compared to its environment also decreases. On that level, this object is hardly distinguishable from noise, such that modern variational models, that are robust against noise, will ignore it. Moreover, the large motion gradient of the correct flow candidate produces high smoothness costs. Hence, the larger the relative displacement is, the more the balance of power shifts towards the smoothness term that favors the background motion candidate. This wrong decision cannot be corrected on later stages of the coarse-to-fine approach.

An Exemplary Relative Large Displacement. An illustration of this problem can be seen in Fig. 4.3 at hand of the tennis ball of the Tennis sequence [27]: On that coarse level from the coarse-to-fine pyramid where the displacement between the ball in both frames is only one pixel, it is still distinguishable from the background but not very clearly. When having a look at the data costs of potential displacements between the tennis ball in the reference frame I_1 and the corresponding neighborhood pixels in the subsequent frame I_2, we observe that they are slightly smaller for the correct large displacement vector (somewhere between one and two pixels to the right) than for the background flow vector (zero displacement at this level). But the exemplary advantage of 3.5/4.5 over 5.8 (using the BCA) is very small. When adding a smoothness term, there are considerable costs for a potential motion gradient to all neighboring pixels. Such a motion gradient very likely is a lot more expensive with a conventional smoothness weight than having a slightly higher data cost at only one pixel by propagating the background flow as the final solution for the object. In this case, there is a clear tendency for the background motion to be the cheapest solution for the object.

4.2 Related Work

A strategy to overcome the balancing problem is to locally choose an appropriate balance between both the data term and the smoothness term. To this end, we have to consider multiple global smoothness weights at once. The basic idea is to keep more than one flow candidate by estimating flow fields for each of the global smoothness weights and combine them into a final flow field. It has already been considered in the approaches of Lempitsky *et al.* [78] and Tu *et al.* [140] but their effectiveness for large displacement optical flow has not been elaborated, yet. While [78] did not focus on large displacements at all, the approach of [140] uses flow candidates generated by PatchMatch [10] to address large displacements. It remains unclear, how a regularized variational approach can perform in the context of moderately large displacements.

Moderately Large Displacements. Although so far the case of moderately large displacements has not been recognized in the literature explicitly, there are hints in the literature that it is worth considering it as a separate case: (i) Concerning the general problem that the estimation of arbitrarily large displacements yields the chance to include arbitrarily large false matches, the work in [70] directly embeds feature matching into a coarse-to-fine scheme where on each level the search space is *restricted* in contrast to the otherwise unrestricted exhaustive search. Evidently, this suppresses noisy results. (ii) Regarding the question, how useful even blurry information still can be, the Geometric Blur feature [14] sticks out. It is recognized as a very discriminative feature in the work of Brox and Malik [27]. Here, the blur is considered as a feature that introduces some positional uncertainty which helps matching parts of objects

that are not perfectly aligned. To some extent, this is comparable to our case where corresponding objects within two frames are not close enough on that coarse-to-fine level which is appropriate to estimate the correct displacement. The smearing connects the objects and makes them virtually closer [2]. (iii) Having a look at the aspect of regularization, the method of Drayer and Brox [44] becomes apparent. It shows that feature matches can be robustified by regularizing them in a post-processing step. (iv) Concerning the balancing problem, Brox and Malik [27] made the observation that fast motion of high-contrast objects is more likely to be accurately estimated than the motion of low-contrast objects. This is related to the fact that there is an implicit weighting of the constancy assumptions with the corresponding image gradient as observed in [165]. In view of the data costs, mismatches of high-contrast objects are thus more expensive than those of low-contrast objects. Overall, this observation hints that locally re-balancing the weights of the data term and the smoothness term may improve the estimation of relative large displacements. Hence, given the four observations from above, it seems desirable to develop a regularized variational method that tackles the balancing problem in order to allow for the robust estimation of moderately large displacements without introducing arbitrarily large false matches in the flow field.

4.3 Contributions

We address this issue by proposing a novel method based on our paper [127]. Our regularized variational model jointly estimates multiple flow candidates using varying smoothness weights and fuses these candidates into a single final flow field. The fusion generally favors the smoothest flow field to obtain an overall noise-free flow field but it allows to locally integrate flow candidates that origin from a less smooth field to also allow for less regular motions. This is done in three ways: (i) We consider multiple instances of the underlying baseline variational model with varying smoothness weights for the estimation of multiple candidate flows that respect the variety of motion patterns that can be present within a single optical flow field and that differ regarding their scale. Moreover, a further instance is part of the estimation of the final flow. (ii) We design and integrate a fusion term that intrinsically fuses flow candidates from the different instances of the baseline model into the final flow field. (iii) We apply a weighting scheme between the instances of the baseline for the candidate flows, the instance for the final flow and the fusion term between all flows in order to make a joint estimation of all components within a single, purely variational optimization possible.

In this way, we demonstrate that the limitations of variational approaches w.r.t. relative large displacements can be shifted, if we refrain from requiring the estimation of *arbitrarily* large displacements in order to avoid arbitrarily large false matches.

4.4 ContFusion-Flow

Let us now discuss the design of our novel variational method that is called *Continuous Fusion Flow (ContFusion-Flow)*. It builds upon the method of Zimmer *et al.* [165, 164] which comprises a variational model that very well adapts to the image data while generating noise-free results. Our method pushes its limits w.r.t. the estimation of relative large displacements a bit further without incorporating any external matching algorithms.

Unlike many other methods, we do not need a pipeline of different algorithms consisting of multiple independent steps to estimate large displacements. Although such pipeline methods in the meantime provide excellent results and can cope with a lot of large displacement cases, there are still cases where the large displacement problem of small objects is intrinsically unsolvable – e.g. in the presence of multiple non-unique instances with arbitrarily large displacements. On the other hand, however, a surprisingly large share of large displacement cases that are solvable can actually be solved *with* a-priori regularization, i.e. using a variational method. This is a very interesting observation, particularly in comparison to the *seminal* work of Brox and Malik [27] in the context of large displacement optical flow.

In contrast to our work on arbitrarily large displacements, our *ContFusion-Flow* method does not need to handle false positives from unregularized matching steps and, moreover, it is intrinsically able to cope with non-unique objects due to the regularized estimations.

4.5 Variational Model

As our baseline method, we make also use of the method of Zimmer *et al.* [165, 164] as presented in Chapter 2 (Sect. 2.8). In the following, it will be denoted by E_{base}. Based on this functional, we are in the position to describe our joint estimation and fusion model. In the style to methods from the literature and our previous method that include descriptor matches [27, 129, 154, 112], we also combine a baseline method with some kind of similarity term that integrates matches into the optical flow estimation. In our case this term E_{cpl} is called coupling term and feeds N_{cand} candidate flows $\mathbf{w}_{\text{P}} = \{\mathbf{w}_{\text{P}1}, \dots, \mathbf{w}_{\text{P}N_{\text{cand}}}\}$ from the candidate model E_{cand} into the solution. To this end, we propose the joint variational model

$$E(\mathbf{w}_{\text{P}}, \mathbf{w}_{\text{f}}) = \underbrace{E_{\text{base}}(\mathbf{w}_{\text{f}})_{\alpha_{\text{f}}}}_{\text{Baseline Model}} + \underbrace{E_{\text{cpl}}(\mathbf{w}_{\text{P}}, \mathbf{w}_{\text{f}})}_{\text{Coupling Term}} + \underbrace{E_{\text{cand}}(\mathbf{w}_{\text{P}})}_{\text{Candidate Model}} , \tag{4.1}$$

that consists of three terms. Apart from those terms which we will describe in the following, the joint model comprises one instance $E_{\text{base}}(\mathbf{w}_{\text{f}})_{\alpha_{\text{f}}}$ of the baseline model for estimating the final flow $\mathbf{w}_{\text{f}}$ with smoothness weight α_{f}.

4.5.1 The Candidate Model

Before we detail on the fusion of the flow candidates, let us discuss how these candidates are obtained. To this end, we consider multiple instances of the baseline model $E_{\text{base}}(\mathbf{w})_\alpha$ with different smoothness weights α_i that estimate the corresponding candidate optical flows $\mathbf{w}_{\text{P}i}$. The candidate model is thus given by

$$E_{\text{cand}}(\mathbf{w}_{\text{P}}) \quad = \quad \lambda_{\text{cand}} \cdot \sum_{i=1}^{N_{\text{cand}}} E_{\text{base}}(\mathbf{w}_{\text{P}i})_{\alpha_i} \,. \tag{4.2}$$

where the single instances can capture different levels of motion details, i.e. displacement scales, due to the different smoothness weights. The weight λ_{cand} balances $E_{\text{cand}}(\mathbf{w}_{\text{P}})$ and $E_{\text{base}}(\mathbf{w}_{\text{f}})_{\alpha_{\text{f}}}$ by steering the direction of information flow between the candidate flows and the final flow. The higher it is, the more the estimation of the candidates $\mathbf{w}_{\text{P}}$ remains unaffected by the coupling term such that the information only flows from $\mathbf{w}_{\text{P}}$ to $\mathbf{w}_{\text{f}}$ via E_{cpl} while backward information flow is suppressed.

4.5.2 The Coupling Term

Finally, in order to couple the candidate flows $\mathbf{w}_{\text{P}i}$ and the final optical flow $\mathbf{w}_{\text{f}}$, we introduce a coupling term E_{C} for each of these instances weighted by an individual parameter β_i and a global parameter λ_{cpl}. The combined coupling term reads

$$E_{\text{cpl}}(\mathbf{w}_{\text{P}}, \mathbf{w}_{\text{f}}) \quad = \quad \lambda_{\text{cpl}} \cdot \sum_{i=1}^{N_{\text{cand}}} \beta_i E_{\text{C}}(\mathbf{w}_{\text{P}}, \mathbf{w}_{\text{f}})_i \,, \tag{4.3}$$

where the distinct coupling terms are defined as

$$E_{\text{C}}(\mathbf{w}_{\text{P}}, \mathbf{w}_{\text{f}})_i \quad = \quad \int_\Omega c_i(\mathbf{x}, \mathbf{w}_{\text{P}}) \cdot \Psi_C\left(|\mathbf{w}_{\text{P}i} - \mathbf{w}_{\text{f}}|^2\right) d\tilde{\mathbf{x}} \,. \tag{4.4}$$

Here, c_i is a local confidence function for the candidate flow $\mathbf{w}_{\text{P}i}$ and Ψ_C is the Charbonnier penalizer [33] that makes the estimation more robust against outliers in the candidate flows. In Sect. 4.6, we will define appropriate confidence functions c_i that steer the local influence of each instance flow $\mathbf{w}_{\text{P}i}$ on the final flow $\mathbf{w}_{\text{f}}$.

4.6 Smoothness Weights and Confidence Functions

Since we desire candidate flows at different smoothness scales, the questions arise how to choose the global smoothness weights of these flows and how to locally decide which flow candidate is the most appropriate. Let us discuss these two issues in the following sections.

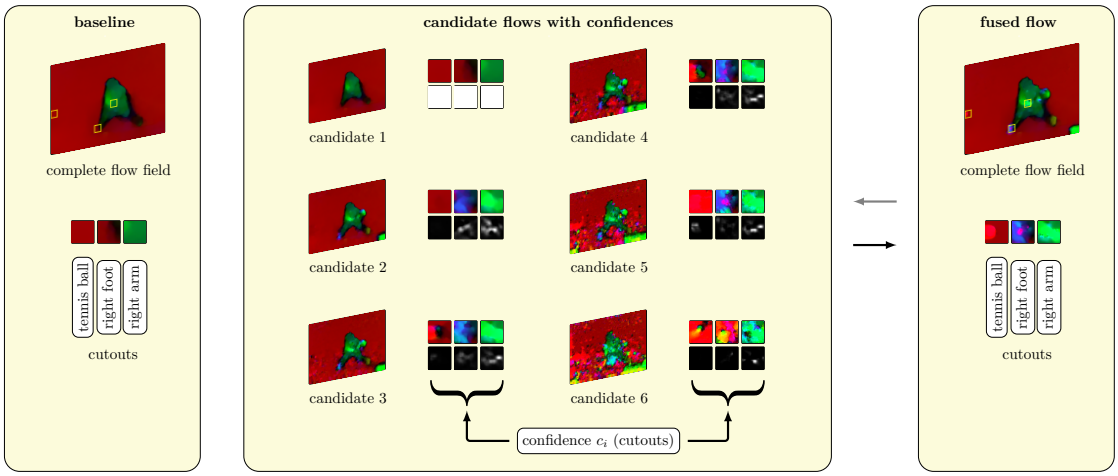


Figure 4.4: Illustration of the results for a decreasing smoothness weight with confidence functions and the fused result. Brighter areas in the confidence visualizations denote higher values.

4.6.1 Smoothness Weights

First of all, we define a maximum smoothness weight α_1 which is intended to be appropriate at most locations. On top of this, we consider smoothness weights that are significantly smaller in order to be able to capture relative large displacement motions. Our choice for the smoothness weights α_i of the flow candidates $\mathbf{w}_{\mathrm{P}i}$ is an exponential decrease w.r.t. α_1 given by

$$\alpha_i \;:=\; \frac{\alpha_1}{2^{i-1}}\,. \tag{4.5}$$

With this choice, we can cover a wide range of different smoothness scales with only a low number of candidate flows. By the example of the Tennis sequence [27] depicted in Fig. 4.4, one can see at which smoothness scale the different motion patterns appear. While the first, smoothest flow covers the background motion and the overall motion of the Tennis player smoothly, the second flow covers the motion of the racket and the arm well, the third flow covers the motion of the hand and the right foot while the fifth flow covers the motion of the ball. Please note that we intentionally used isotropic regularization in this depiction in order to make results visually comparable to LDOF [27] which uses the same baseline with isotropic regularization.

4.6.2 Assumptions on Local Confidences

Now that we have determined global smoothness weights for the candidate model that are appropriate to obtain a wide range of helpful candidate flows, we need a local measure for the quality of each candidate in order to let the most appropriate one

dominate the overall estimation of the optical flow. Given a set of candidate flows $\mathbf{w}_{\mathrm{P}i}$ with different smoothness scales, we take into account the considerations from Sect. 4.1 to state the local assumptions on how to integrate these flows in the estimation of the final flow $\mathbf{w}_{\mathrm{f}}$:

1. A less smooth flow is likely to fulfill the data term better than a smoother flow, independently from being reliable or unreliable. Hence, a less smooth flow shall only have influence if it provides *significantly* less data costs than both the next smoother flow candidate and the smoothest flow candidate.
2. The less smooth a flow is, the more texture is necessary in order to achieve meaningful flow vectors (similar to [27]). Otherwise, we might likely get trapped into the aperture problem.
3. A less smooth flow should not be considered if the data is unreliable (i.e. in over- or undersaturated regions).

In order to integrate those assumptions in our local confidence functions c_i, we make use of the same measures for the data cost and for the local structure as in our *ALD-Flow* approach (see Chapter 3, Sects. 3.8.1 and 3.8.2), i.e. we evaluate the data term in order to compute the data costs and we compute the structure tensor [64] to measure structuredness. For an increased robustness, we evaluate both of them on local patches.

4.6.3 Composition of the Local Confidence Function c_i

Following the assumptions from the last section, we model the local confidence function c_i as used in the coupling term (see Eq. 4.4 where i is the index of the candidate flow) as the product of three weights which will be defined in the following.

Cost Reduction Weight

Let e be the data costs and let $\rho_{L\times L}(g,\mathbf{x})$ be a functional that averages the function g in a $L\times L$ neighborhood around $\mathbf{x}$. As required by Assumption 1 in Sect. 4.6.2, the following two functions describe the patch-wise energy improvement of the flow $\mathbf{w}_{\mathrm{P}i}$ compared to the previous, smoother flow $\mathbf{w}_{\mathrm{P}i-1}$ and the first and smoothest flow $\mathbf{w}_{\mathrm{P}1}$, respectively:

$$\delta_{\mathrm{prev},L}(\mathbf{x},\mathbf{w}_{\mathrm{P}},i) = \rho_{L\times L}(e(\mathbf{w}_{\mathrm{P}i-1}),\mathbf{x}) - \rho_{L\times L}(e(\mathbf{w}_{\mathrm{P}i}),\mathbf{x})\,, \tag{4.6}$$

$$\delta_{\mathrm{first},L}(\mathbf{x},\mathbf{w}_{\mathrm{P}},i) = \rho_{L\times L}(e(\mathbf{w}_{\mathrm{P}1}\quad),\mathbf{x}) - \rho_{L\times L}(e(\mathbf{w}_{\mathrm{P}i}),\mathbf{x})\,. \tag{4.7}$$

The cost reduction weight is then defined as

$$w_i^d(\mathbf{x},\mathbf{w}_{\mathrm{P}}) = \log\left(1+\mathrm{e}^{\kappa_d(\delta_{\mathrm{prev},L}(\mathbf{x},\mathbf{w}_{\mathrm{P}},i)+\delta_{\mathrm{first},L}(\mathbf{x},\mathbf{w}_{\mathrm{P}},i))}\right), \tag{4.8}$$

where κ_d is a free parameter. This function resembles a linear one for large arguments of the exponential while it approaches zero for decreasing (negative) arguments.

Structuredness Weight

Let $s(\mathbf{x})$ be the smaller eigenvalue of the structure tensor (integrated over a 7×7 neighborhood) of the reference frame I_1, let $\bar{s}$ be its average value over the whole image and let $r_i = \frac{\alpha_1}{\alpha_i}$. The structuredness weight is then defined as

$$w_i^s(\mathbf{x}) \quad = \quad \left(\frac{s(\mathbf{x})}{\bar{s}} \right)^{\kappa_s \cdot \log(r_i)} , \tag{4.9}$$

where κ_s is a free parameter. This weight is more pronounced for less smooth candidate flows (i.e. if r_i is bigger) as required by Assumption 2 in Sect. 4.6.2.

Data Reliability Weight

We define $\chi_I(\mathbf{x})$ as an indicator function that excludes under- or undersaturated regions. It reads

$$\chi_I(\mathbf{x}) \quad = \quad \begin{cases} 1, & I_1^c(\mathbf{x}) > \tau \text{ and } I_1^c(\mathbf{x}) < 255 - \tau \quad \forall c \in \{1,2,3\} \\ 0, & \text{otherwise} . \end{cases} \tag{4.10}$$

where $\tau = 1$ is a robustness threshold. This weight implements Assumption 3 from Sect. 4.6.2.

Overall Confidence Function

The overall confidence functions $c_1, \ldots, c_{N_{\text{cand}}}$ are then defined as follows

$$\hat{c}_i(\mathbf{x}, \mathbf{w}_{\text{P}}) \quad = \quad w_i^d(\mathbf{x}, \mathbf{w}_{\text{P}}) \cdot w_i^s(\mathbf{x}) \cdot \chi_I(\mathbf{x}) \qquad\qquad (i > 1) . \tag{4.11}$$

In order to be numerically robust, they are bounded from above via

$$c_i(\mathbf{x}, \mathbf{w}_{\text{P}}) \quad = \quad \min\left(\hat{c}_i\left(\mathbf{x}, \mathbf{w}_{\text{P}}\right), 1000\right) . \tag{4.12}$$

Since the smoothest flow $\mathbf{w}_{\text{P}1}$ serves as reference, it should be used everywhere except for those locations where a less smooth flow could improve the result. Hence, we define the confidence c_1 of the smoothest flow as

$$c_1(\mathbf{x}, \mathbf{w}_{\text{P}}) \quad = \quad 1 , \tag{4.13}$$

which corresponds to the confidence of the other flows at averagely structured areas with only a small energy reduction.

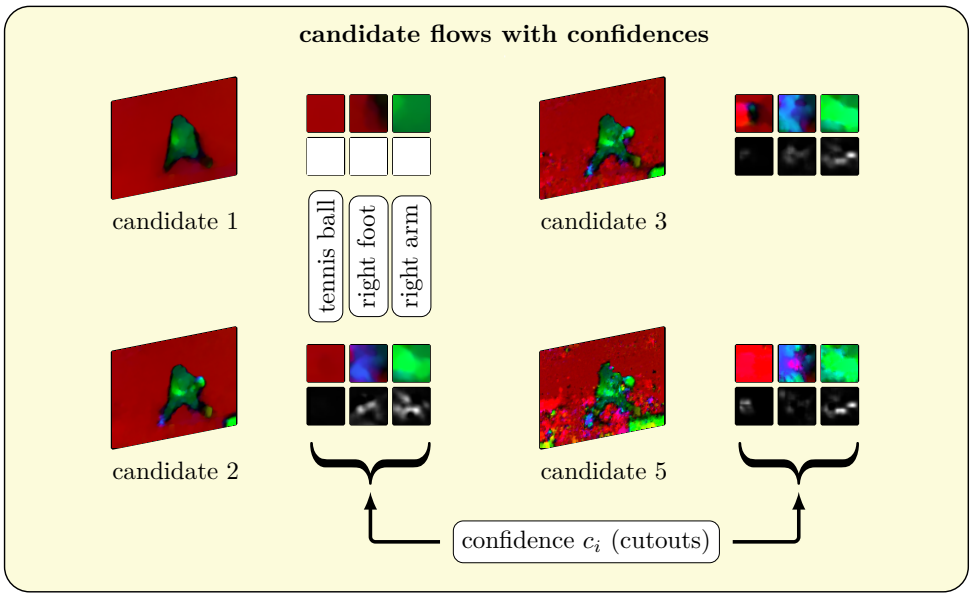


Figure 4.5: Illustration of exemplary candidate flows with confidence functions (excerpt of Fig. 4.4). Brighter areas in the confidence visualizations denote higher values.

Exemplary visualizations of these local confidence confidence functions c_i for the Tennis sequence are shown in Fig. 4.4 (bottom row) where brighter values indicate higher confidence. A zoom of the most important candidate flows can be found in Fig. 4.5. As one can see, for each relative large displacement, we have a high confidence in the smoothest candidate flow that is able to capture it.

4.7 Distinguishing Small Objects from Noise

As we have seen at the beginning of this chapter, a small object undergoing a relative large displacement is typically hardly distinguishable from noise on that (coarse) level that is appropriate to estimate its motion. Let us illustrate at hand of the Tennis sequence why our model is still able to estimate such motion without adapting to noise. To this end, we consider the motion of the tennis ball in Fig. 4.6 whose correct estimation appears in terms of a bright red spot on the left side of the color coded flow. As we can see, the displacement is estimated within a relatively unsmooth candidate flow field that adapts to *both* small objects *and* noise. In contrast to the false motions that are the results of real noise, however, the motion candidate of the tennis ball is fused at later levels of the coarse-to-fine pyramid where the ball is distinguishable from noise. Hence, our method benefits from its ability to estimate a motion candidate at one coarse-to-fine level (where the large displacement can be estimated) and choose it to be the most appropriate one on *another* level (where the object is distinguishable from noise).

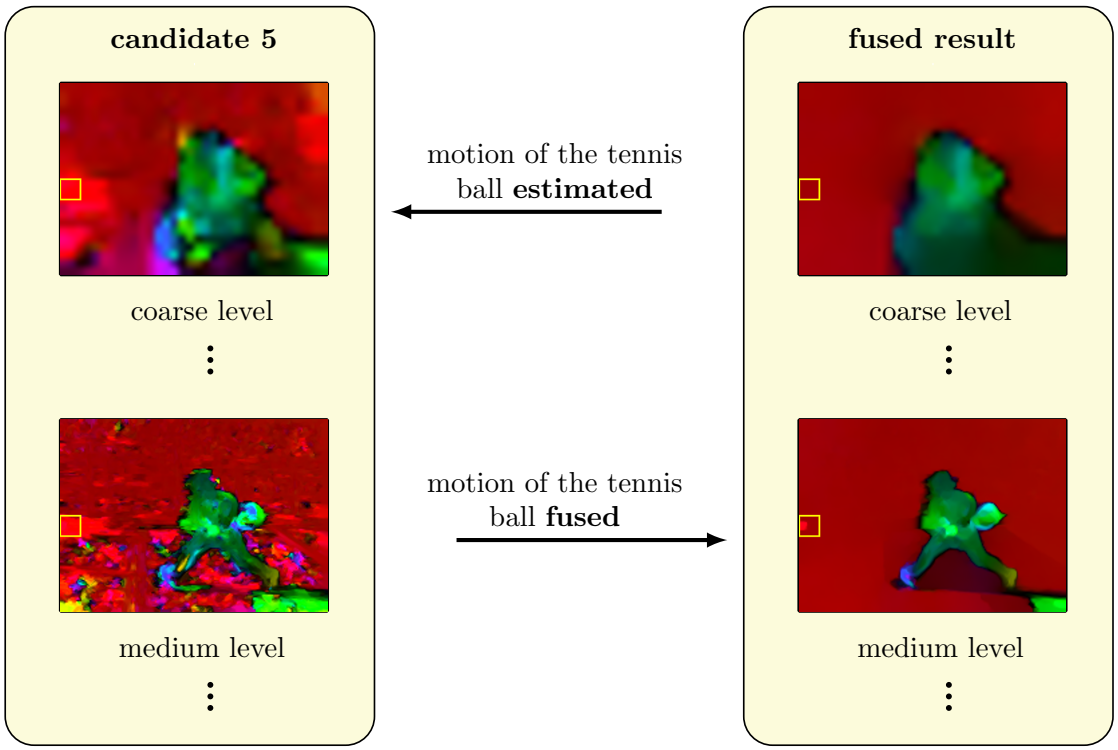


Figure 4.6: Estimation and fusion of the large displacement of the tennis ball from the Tennis sequence (Frame 496) [27]. While the corresponding candidate motion is estimated at a coarse level, the fusion into the final result is done at a medium level.

4.8 Aspects of the Minimization

Similar to *ALD-Flow*, we basically minimize the nonconvex and nonlinear functional using concepts from Chapter 2, including the coarse-to-fine warping strategy as described in Sect. 2.6.3 along with the lagged nonlinearity method as described in Sect. 2.3.1. After discretization, the resulting sequence of linear equation systems is solved with a successive overrelaxation scheme (SOR) as hinted in Sect. 2.3, this time, however, using a multicolor variant [1] that can be parallelized and SIMD vectorized. Moreover, we apply constraint normalization as described in Sect. 2.8.1.

Please note that in Eq. 4.4 the candidates $\mathbf{w}_P$ are apparent in both the confidence functions and the coupling term. In order to avoid multiplications of unknowns during the minimization, in each coarse-to-fine level we compute the confidence functions based on the flow from the previous level. This can also be seen as a lagged nonlinearity method regarding the computation of the confidences.

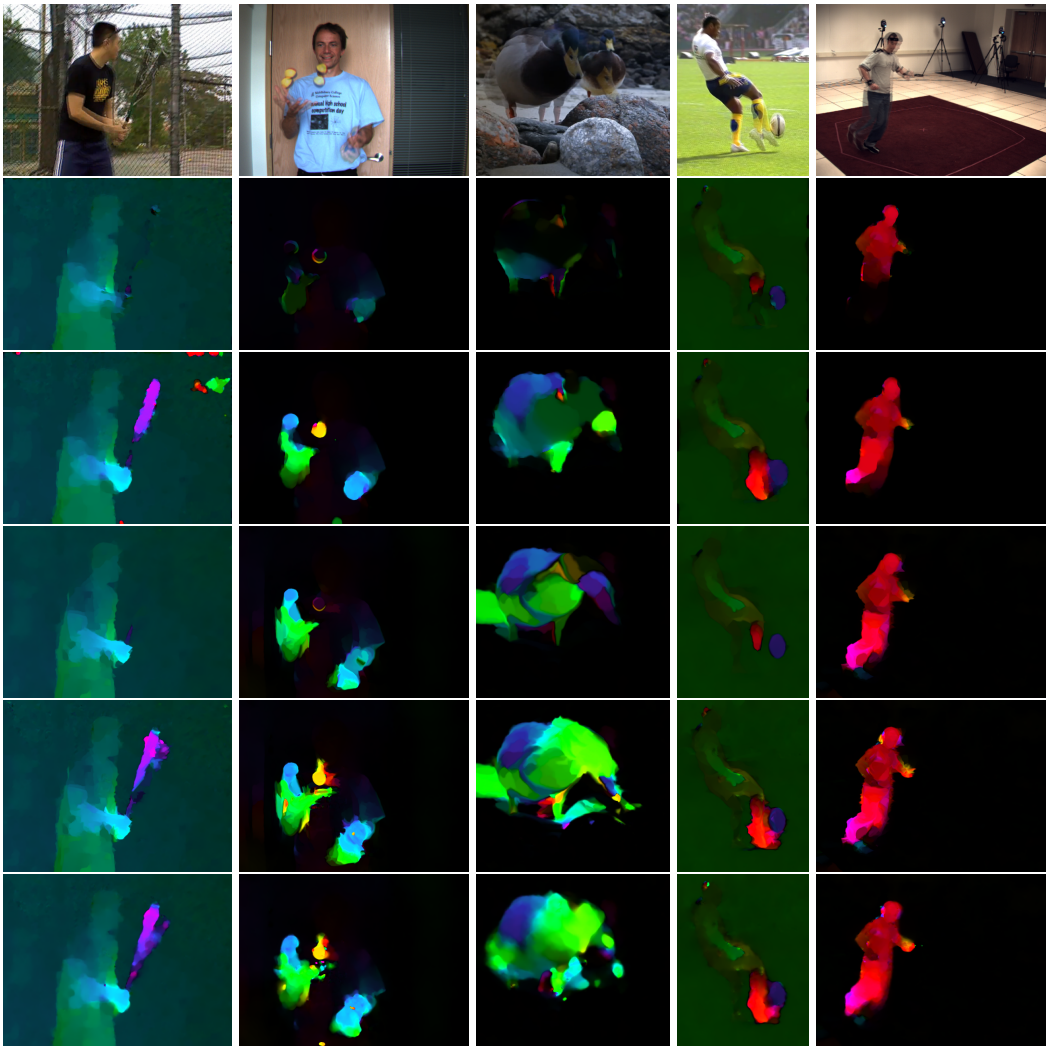


Figure 4.7: Comparison of LDOF and our method with the corresponding baseline results. **Left to right:** Baseball sequence [160], Beanbags sequence [9], Bird sequence, Football sequence [160], Human Eva sequence [123]. **Top to down:** Overlayed frames, LDOF baseline, LDOF, our baseline, our result, our result (LDOF regularizer).

4.9 Evaluation

In order to evaluate the performance of our method, we conduct several experiments. These include a qualitative comparison against LDOF [27] that investigates the large displacement capabilities of our method, an experiment that analyzes the effect of constraint normalization in this context, an experiment that evaluates the effect of different types of data costs and a quantitative experiment on all major benchmarks. Details on the parameters and their retrieval can be found in Appendix A.6.

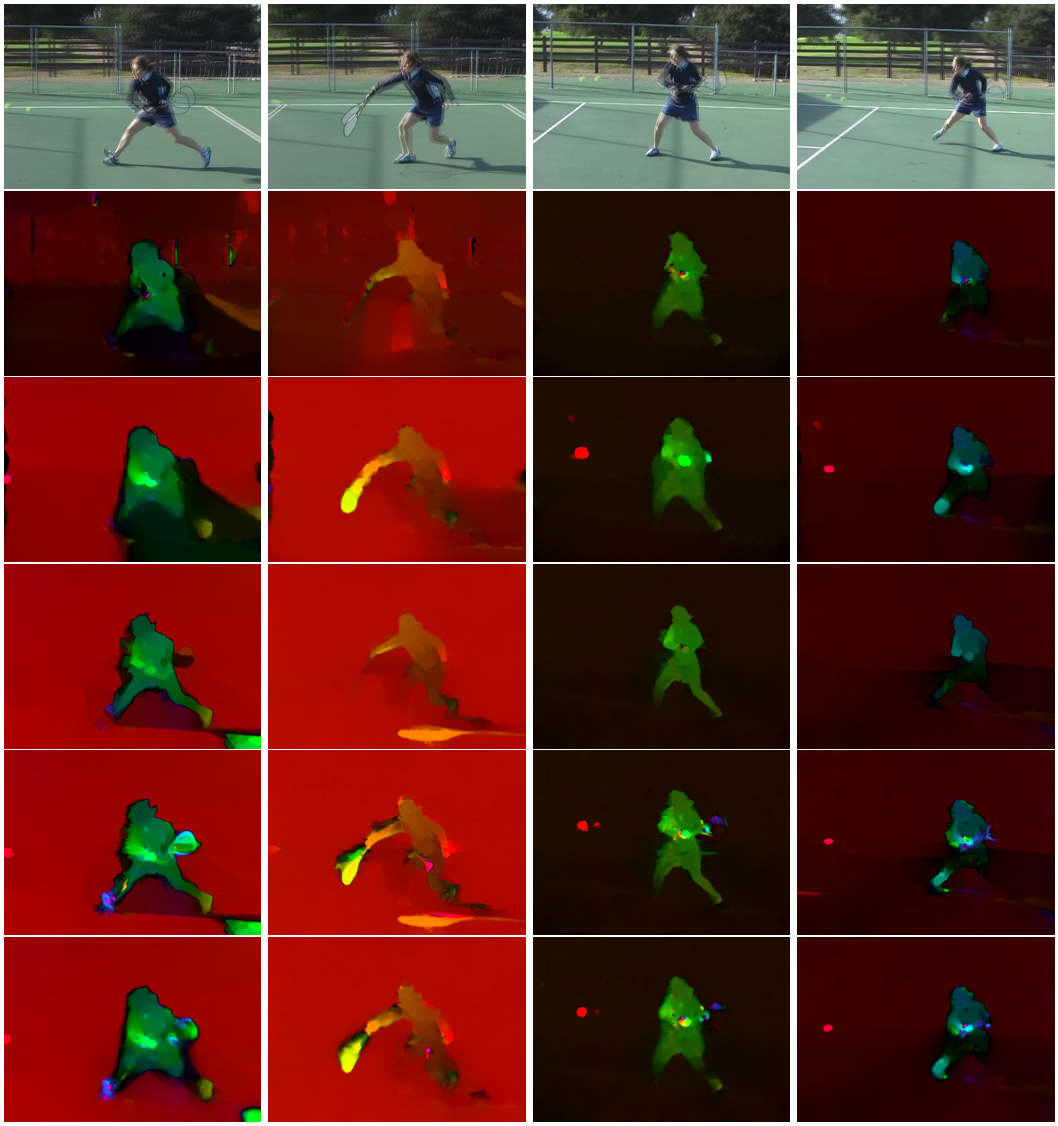


Figure 4.8: Comparison of LDOF and our method with the corresponding baseline results. **Left to right:** Tennis sequences 496, 502, 538, 577 [27]. **Top to down:** Overlayed frames, LDOF baseline, LDOF, our baseline, our result, our result (LDOF regularizer).

4.9.1 Large Displacement Sequences

In our first experiment, we evaluate the performance of our method in the context of large displacements. To this end, we consider various challenging large displacement sequences from the literature and compare our results to those of the method of Brox and Malik (LDOF) [27] which has introduced descriptor matching in variational methods for large displacement optical flow.

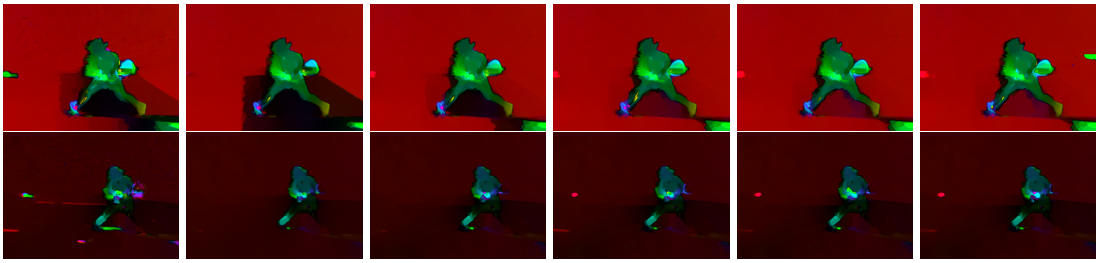


Figure 4.9: Effect of constraint normalization on the estimation of relative large displacements. **From left to right:** No constraint normalization, $\zeta = 1$, $\zeta = 0.1$, $\zeta = 0.01$, $\zeta = 0.001$, $\zeta = 0.00001$. **From top to bottom:** Tennis sequences 496 and 577.

In Figs. 4.7 and 4.8 we show the results of both the publicly available implementation of LDOF and our novel variational method for large displacement optical flow. As one can see, our method correctly estimates the large displacements that LDOF is able to estimate – and even some more (see e.g. Tennis sequence 496 in Fig. 4.7). This particularly includes the displacements of the tennis balls that evidently exceed their sizes. The extremely challenging Bird sequence [160] in Fig. 4.7 shows the limitations of both methods as none of them could capture the motion of the bird's head. In order to demonstrate that the correct estimation of large displacements does not depend on the anisotropic regularizer, we also added results for our method with an isotropic smoothness term (which is also used in LDOF).

While we have chosen the number of candidate flows fixed for all sequences, one may actually improve the results further by choosing this number according to the extent of large displacements. For instance, for the Beanbags sequence, already a value of $N_{\text{cand}} = 3$ is sufficient to estimate the large displacements, while we need a value of $N_{\text{cand}} = 7$ in order to capture the motion of the tennis ball in Tennis sequence 577.

4.9.2 Constraint Normalization

In our second experiment, we show that constraint normalization [165] is helpful in the context of large displacements (see also Chapter 2, Sect. 2.8.1). To this end, we estimated flow fields without normalization and with normalization for different values of the normalization parameter ϵ_{cNorm}. While the general benefits of the constraint normalization have already been shown in [165], Fig. 4.9 shows the results on two large displacement sequences. As one can see particularly at hand of the tennis balls, both the deactivation of the constraint normalization and a too high value of ϵ_{cNorm} inhibit the estimation of large displacements. A too low value for ϵ_{cNorm}, in contrast, leads to noisier results. Using constraint normalization with a value between 0.001 and 0.01 (our standard value) for ϵ_{cNorm} provides the best results.

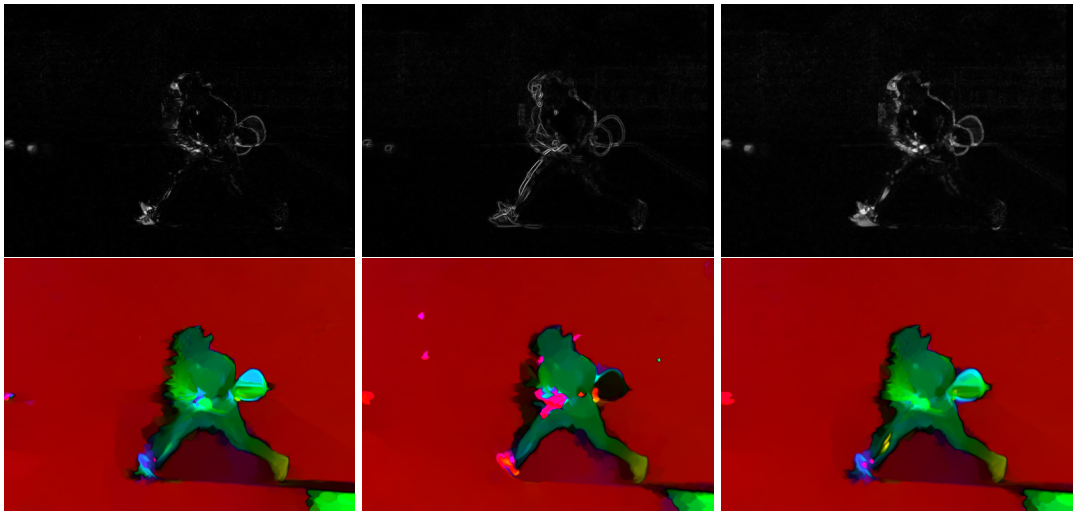


Figure 4.10: Effect of different data constancy assumptions on the final result. **From left to right:** Brightness constancy assumption (BCA), Gradient constancy assumption (GCA) and both combined. **From top to bottom:** Data costs of the baseline flow (brighter grey values indicate larger energies), final result.

This demonstrates that the weight balancing effect of the constraint normalization also helps in the context of large displacements, since we have to find weight balances between data term and smoothness term that fit the present motion patterns.

4.9.3 Influence of the Data Constancy Assumptions

In our third experiment, we analyze the two types of data terms we used in our model w.r.t. their data costs and their influence on the fusion scheme. While the brightness constancy assumption (BCA) can produce high costs at any part of a mismatched object, the gradient constancy assumption (GCA) can only produce high data costs where edges are involved. It is hence a lot sparser (see Fig. 4.10, top row). As can be seen from the bottom row of Fig. 4.10, the fusion using only the GCA data term is by far inferior to the results of using BCA or combining both data terms. The data costs of a pure GCA data term for *incorrect* matches are too low and hence it cannot compete with the smoothness term which prevents the motion discontinuity of a relative large displacement. In contrast, when including the BCA, the denser data costs make the misestimation of relative large displacements more expensive and thus increase the probability to estimate such displacements correctly. This shows that data costs with dense coverage for mismatched objects are important for our fusion scheme.

Table 4.1: Results of Continuous Fusion Flow and its baseline on training data from different benchmarks.

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline	2.73	**0.229**	6.375	4.084	10.68%	24.25%
$N_{\text{cand}} = 1$	2.73	0.231	6.365	4.082	10.60%	**24.18%**
$N_{\text{cand}} = 2$	**2.72**	0.231	6.028	4.081	10.55%	24.25%
$N_{\text{cand}} = 3$	2.75	0.232	**5.808**	**3.974**	10.53%	24.39%
$N_{\text{cand}} = 4$	2.77	0.232	5.832	3.986	10.48%	24.45%
$N_{\text{cand}} = 5$	2.80	0.235	5.967	4.029	**10.47%**	24.49%
$N_{\text{cand}} = 6$	2.84	0.245	6.119	4.102	**10.47%**	24.72%
$N_{\text{cand}} = 7$	2.99	0.274	6.189	4.128	10.48%	24.51%

4.9.4 Major Benchmarks

In order to see how this method performs in a quantitative sense, we conduct a fourth experiment on all major benchmarks. As before, we use first-order regularization for the Middlebury and MPI Sintel benchmarks and second-order regularization for the KITTI benchmarks. Tab. 4.1 shows the corresponding results for the training data. The most significant changes can be seen in for the MPI Sintel benchmark [31]. Compared to the baseline method, the average endpoint error (AEE) decreases from 4.084 down to 3.974 (by 2.7%) and from 6.375 down to 5.808 (by 8.9%) on the respective subset of sequences that we used for parameter optimization. This behavior is partially confirmed by the results for the evaluation data sets that are listed on the MPI Sintel benchmark webpage where our method is denoted as *ContFusion* and the baseline is denoted as *COF_2019*. While the error slightly increases from 6.171 to 6.263 (by 1.5%) on the clean pass, it decreases from 8.065 to 7.857 (by 2.6%) for the final pass. Minor improvements on the training data can also be reported for the KITTI benchmark 2012 where the AEE decreases from 10.68 down to 10.47 (by 2%), while for the KITTI 2015 and Middlebury benchmarks there is no significant improvement. Overall, however, the improvements on most benchmarks show that our novel strategy of the simultaneous estimation and fusion of motion candidates is also beneficial in a quantitative sense.

Optimization of More Parameters

So far, for simplicity reasons, we have only optimized the very crucial parameters N_{cand}, α_1 and γ while keeping the $\delta = 1$ for all benchmarks and all other parameters fixed

Table 4.2: Results of Continuous Fusion Flow and its baseline on training data from different benchmarks whereby all parameters have been optimized.

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline	2.73	0.229	6.375	4.084	10.68%	24.25%
$N_{\text{cand}} = 1$	2.73	0.229	6.365	4.082	10.60%	**24.18%**
$N_{\text{cand}} = 2$	2.72	0.229	6.028	4.081	10.55%	24.25%
$N_{\text{cand}} = 3$	2.72	0.227	**5.743**	3.972	10.53%	24.39%
$N_{\text{cand}} = 4$	2.72	0.227	5.757	3.960	10.47%	24.41%
$N_{\text{cand}} = 5$	2.71	0.229	5.859	3.910	**10.45%**	24.46%
$N_{\text{cand}} = 6$	2.71	0.226	5.849	**3.905**	10.46%	24.37%
$N_{\text{cand}} = 7$	**2.70**	**0.225**	5.831	3.975	10.47%	24.49%

for all sequences (see Appendix A.6). The other parameters λ_{cand} (overall weight of the candidate models) and λ_{cpl} (overall weight of the coupling term), however, are not negligible. To better see the full potential of our method, we conduct a fifth experiment that also involves these parameters in the parameter optimization. Tab. 4.2 contains the achieved results when optimizing more parameters. We can see that in this case, we can also achieve consistent improvements for the Middlebury benchmark and further minor improvements for the other benchmarks (except for KITTI 2015). In particular for large sets of candidate models ($N_{\text{cand}} \geq 5$), we avoid the otherwise significant decreases in performance.

4.9.5 Limitations

As we have seen before, our method is not able to capture large displacements in the presence of illumination changes, since the illumination-invariant gradient constancy assumption (GCA) alone is not resilient enough to reliably estimate large displacements. The behavior at occlusions is another limitation of our method. This can be seen both visually at the large displacement sequences (in Figs. 4.7 and 4.8) and quantitatively at the unmatched EPE in the public results of the MPI Sintel benchmark (that increases compared to the baseline). Additionally to regions with mismatched objects, occluded regions produce potentially high data costs. Since our confidence function heavily relies on data costs, accurate smooth flows are replaced by less smooth candidate flows that lead to a smaller local data energy while, however, being potentially meaningless.

4.10 Summary

In this chapter, we pushed the limits of variational approaches that are minimized using a standard coarse-to-fine warping scheme a little bit further w.r.t. relative large displacements. We have shown that many large displacement cases from the literature can be estimated without the need for descriptor matches. The weaknesses of existing variational methods in these cases are not due to weak data representations on coarse resolutions but due to a weight balancing of the data term and the smoothness term that is inappropriate for the estimation of relative large displacements.

Using multiple instances of the baseline model and appropriate choices of weighted coupling terms, we show that we can estimate different scales of motions in a regularized way within a single variational model that simultaneously estimates and fuses candidate flows with different smoothness weights. The findings were confirmed by the evaluation which showed a good performance for relative large displacements and a quantitative improvement over its baseline method on different benchmarks. Moreover, we demonstrated that concept of data constraint normalization is particularly helpful in re-balancing the data term and the smoothness term when estimating relative large displacements.

Limitations include the behavior at occluded regions, where advanced occlusion handling would be necessary, and the handling of severe illumination changes, where the BCA is not applicable at all and the GCA alone cannot help to estimate relative large displacements correctly. The latter case will later be addressed in Chapter 6.

Chapter

5 Optical Flow and Illumination Compensation

Another important data challenge is given by changes in the illumination between the frames of an image sequence, particularly in outdoor scenarios such as driver assistance systems or video surveillance tasks. Along with this, the robustness of optical flow methods under uncontrolled illumination is a major target in recent research. In order to support this research, real-world benchmarks such as the different editions of the KITTI Vision Benchmark Suite [52, 92] have been designed that ideally reflect such scenarios. The sequences depict automotive scenarios in an urban environment where typical illumination changes such as camera re-adjustments or shadows and highlights as instances of physical illumination effects appear.

5.1 Illumination Invariance

A very intuitive way to handle illumination changes is to consider features of an image that are *invariant* under the assumed type of illumination changes. This holds for any type of computer vision problem that considers different depictions of the same scene. It is hence not surprising that a lot of research has been done w.r.t. invariant image features. We have already seen some examples of such features, among which the gradient can be considered as *the* starting point in the research on illumination-invariant features due to its simple computation and its invariance under global additive changes. In the context of motion estimation, the gradient comes into play in several variants: On the one hand, it has been used directly in terms of the gradient constancy assumption (GCA) within the data term of variational optical flow models [26, 29, 165, 25, 160]; similar constancy assumptions have also been constructed from higher order derivatives [75]. On the other hand, it is the basis for a lot of more advanced features like SIFT [80] or HOG [36] which are used either for feature matching [25, 27, 70] or as a constancy assumption of a variational model [79, 109]. Among additional invariances w.r.t. other aspects such as geometry or scale, such advanced features often gain even higher degrees of

illumination invariance by further computations like e.g. normalization steps performed on whole patches of gradients.

A more direct implementation of higher degrees of illumination invariances is given by specifically tailored features that e.g. come from photometric invariants of color images [144, 94, 164] and mutual information [66], such as the normalized cross correlation (NCC) [155]. Moreover, there are small patch-based descriptors based on the comparison of neighboring pixel values, such as the census transform [124, 106, 23] and the (complete) rank transform [162, 40] which are invariant even under any type of monotonic illumination changes. Surveys that compare some of these methods can be found e.g. in [125, 145]. A similar goal is achieved by methods that discard illumination-relevant information in a preprocessing step. This includes the concept of structure-texture decomposition [150] or the concept of derivative-type filters [132].

Discarding Information. Building on illumination invariance always means to discard potentially valuable information at the same time. If illumination changes are only moderate or not even present, such an omission of brightness or contrast information may significantly deteriorate the results. A good example is given by the previously discussed *Continuous Fusion Flow* when comparing the gradient constancy assumption (GCA) with the brightness constancy assumption (BCA): the result of the invariant GCA did by far contain less relative large displacements than the results for the BCA or the combination of both, and it was considerably noisier at the same time (see Chapter 4, Fig. 4.10). Moreover, most invariants are meaningless at homogeneous regions, since they rely on illumination-invariant parts of the information about the local contrast, which, however, is missing there. Summarizing: Since discarding potentially valuable information destroys the ability of the data term to steer the estimation of the optical flow at many locations – which can have negative effects on the overall result –, it would be desirable to keep this information and exploit it in the estimation process.

5.2 Estimating Illumination Changes

Another possibility is not to *discard* such information at all but to estimate it, i.e. to treat it as a further unknown in the problem. This comes down to an explicit estimation of illumination changes. By an online compensation of one of the images of an image sequence for the illumination changes w.r.t. the other image, the BCA can be made valid. This is possible in a *joint* estimation of the illumination changes together with the optical flow, i.e. together with the latter we estimate the so-called *brightness transfer function* (BTF) between both frames – which is known as relative intensity transfer function in [38]. It describes a mapping of intensities between corresponding pixels of both frames and, hence, allows to compensate a frame for the apparent illumination changes and thus to eliminate these changes without omitting the remaining illumination information.

5.2.1 Spatial Properties

Fortunately, illumination changes do not need to be considered pixelwise, since they typically affect entire image regions. Otherwise, the problem would be unsolvable, since any unconstrained variation of the brightness at a location in the image sequence could solely be expressed by an illumination change and the estimation of an optical flow would neither be possible nor meaningful. Instead, we can rely on regions of similar illumination which comes down to using neighborhood information in the estimation. No matter whether we consider shadows or global illumination changes, these conditions all apply to regions of smaller or bigger sizes.

Types and Magnitudes. When we discuss the locality of illumination changes, we should also regard the different aspects of illumination changes: they can be of a different *type* and they can have a varying *magnitude*. Hence, some questions arise: Do we consider a global type of changes with a constant magnitude? Do we consider a global type with a varying magnitude? Or do we consider varying types of illumination changes with varying magnitudes?

5.2.2 Parametrizations

All of these aspects come into play at different stages of the joint estimation. Typically, the type of illumination changes appears in terms of a corresponding parametrization while the magnitude of the illumination changes is given by the coefficients w.r.t. this chosen parametrization. This also hints that the type of illumination changes is usually determined offline by choosing a suitable parametrization while the magnitude of the illumination changes is estimated online by estimating the corresponding coefficients which we call *illumination coefficients*.

Parametrizations can be obtained in different ways, either by explicit modeling e.g. via additive [35] or affine [53] illumination changes or by directly learning them from training data.

5.2.3 Coefficients

The estimation of the optical flow itself is already highly ill-posed whereby the aperture problem plays an important role [16]. The usage of regularization in variational optical flow estimation alleviates the problem a lot. When additionally illumination changes are supposed to be estimated in terms of coefficients for a given parametrization, additional regularization is required in order to avoid an arbitrary description of the change of brightness at a pixel by a change of the illumination. Only a well-balanced regularization scheme for both the optical flow field and the fields of illumination coefficients can lead

to a meaningful separation of brightness changes at the pixels into motion-induced brightness changes and illumination-induced brightness changes.

5.3 Related Work

Since in this context different research fields are important, namely optical flow methods that jointly also estimate illumination changes and parametrization learning in the context of BTFs, we will subdivide the related work w.r.t. the different research fields.

5.3.1 Optical Flow Approaches

The idea to jointly estimate optical flow and illumination changes has already driven some methods in the literature. This includes both approaches that estimate a single *global* BTF [38] as well as methods that estimate coefficients for a given parametrization in order to determine *local* illumination changes. The variety of such explicitly modeled illumination changes on the one hand comprises simple additive [35, 95] and affine [53, 98, 76, 60] terms and on the other hand reaches up to complex brightness models derived from physics [65]. A combination of local and global ideas has also been proposed in [60]: First, a local affine model is used to estimate the correspondences in a PatchMatch-like approach [10] and, second, a single global BTF is estimated from these correspondences.

In order to distinguish real illumination changes and motion-induced brightness variations, it is not only necessary to have an appropriate regularization scheme that prevents arbitrary solutions but also to have an appropriate model for the illumination changes. So far, however, such models were either designed ad-hoc [35, 53] or are based on a certain physical process [62]. An investigation to determine the most suitable model for a given set of data, however, is missing. Moreover, there are no efforts to analyze the design of the regularizer of the coefficient fields which is responsible for a good separation of brightness effects due to motion and due to illumination changes. Finally, existing variational methods with parametrized illumination models rely on simple concepts for data and smoothness terms [53, 38]. Hence, it is of considerable interest to see how a more sophisticated joint method performs on challenging benchmark data that contain significant illumination changes. In this thesis, we will develop such a method with an advanced model that allows for flexible parametrizations of the illumination.

5.3.2 Basis Learning

Also in the context of estimating brightness transfer functions and finding suitable representations a lot of work has been done in the literature. Considering the estimation

of BTFs or camera response functions, there are some methods that address these issues, mainly in the context of HDR imaging. These include the computation of the BTF using histogram specification as proposed by Grossberg and Nayar [56] as well as the estimation of the camera response function as proposed by Debevec and Malik [37] and Grossberg and Nayar [57]. The latter work uses learned basis functions for this task. In the context of representing appearance changes, there are works that make use of basis functions for illumination changes. Hager and Belhumeur [62] used them for template tracking whereas Black *et al.* considered this kind of representation in their work on iconic changes [20]. Another work comes close in spirit: the approach of Tieu and Miller [135] estimates a 3-D basis via PCA to represent *color eigenflows* that represent color changes and map RGB color vectors from one image to the other. Finally, basis functions are also used in optical flow methods both in spatial and in temporal direction for the modeling of the flow itself or the trajectories of points. An early representative is given by the work of Nir *et al.* [100] who over-parametrized the optical flow. Recently, such basis functions are also used in the works of Garg *et al.* who proposed temporal tracking of non-rigid objects with subspace constraints [51] and of Ricco and Tomasi who proposed to learn trajectories with global occlusion reasoning [114]. Hence, apart from developing a more sophisticated model that can make use of flexible parametrizations of the illumination changes it would also be desirable to learn such parametrizations in terms of basis functions from training data.

5.4 Contributions

In this chapter, we tackle the problem of estimating the motion jointly with the illumination changes. In this context, our contributions are fourfold: (i) We present a novel variational approach for the joint estimation of optical flow and illumination changes that can handle variable parametrizations of illumination changes. (ii) We demonstrate how a suitable parametrization can be learned by a principal component analysis (PCA) of the brightness changes in training data. Such a learned parametrization represents the BTF in terms of a basis. In this learning step, we do, moreover, not only estimate one global BTF per image pair but several local ones by clustering different regions of similar illumination changes. (iii) We compare such learned parametrizations with explicitly modeled ones. (iv) We compare different regularization schemes in order to find an optimal combination of regularizers for the optical flow and the illumination coefficients. The contents of this chapter have been published at a conference [41]. In contrast to the PhD thesis of Demetz [39], who is another co-author of our paper and also discusses contents of this paper, we will rather focus on the modeling part than on the learning part.

5.5 Parametrization of Illumination Changes

In order to have a flexible and at the same time tractable way to estimate the illumination changes in terms of coefficients, we take inspiration from the work of Grossberg and Nayar [57] who proposed the use of *parametrized* brightness transfer functions (BTFs) in the context of photometric calibration for HDR imaging. We use such a function, that maps intensities from the first frame to corresponding intensities in the second frame, to account for illumination changes between both frames. Given a set of N_{cIll} basis functions $\phi_j : \mathbb{R} \to \mathbb{R}$ and an intensity I, the corresponding parametrized BTF reads

$$\Phi(\mathbf{b}, I) \quad = \quad \bar{\phi}(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi_j(I)\,, \tag{5.1}$$

where $\bar{\phi}(I) : \mathbb{R} \to \mathbb{R}$ is the so-called mean brightness transfer function and $\mathbf{b} = (b_1, \ldots, b_{N_{\mathrm{cIll}}})^\top$ are linear weights that state the influence of each basis vector. This parametrization is very flexible, since the basis vectors can implement any type of illumination change.

5.6 Variational Model

Let us now define the variational model that is able to estimate the optical flow $\mathbf{w}$ and a coefficient field $\mathbf{b} : \mathbb{N} \times \Omega \to \mathbb{R}^{N_{\mathrm{cIll}} \cdot N_{\mathrm{c}}}$ (where N_{c} denotes the number of image channels) that describes the magnitude of the illumination changes according to a given parametrization. Following the basic approach of Cornelius and Kanade [35], the joint computation of the optical flow and the illumination changes is conducted by minimizing the following energy functional:

$$E(\mathbf{w}, \mathbf{b}) \quad = \quad \int_\Omega \underbrace{D(\mathbf{w}, \mathbf{b})}_{\text{Data Term}} + \underbrace{\alpha \cdot S_{\mathrm{flow}}(\mathbf{w})}_{\text{Flow Regularizer}} + \underbrace{\alpha_{\mathrm{ill}} \cdot S_{\mathrm{ill}}(\mathbf{b})}_{\text{Coefficient Regularizer}} \quad d\tilde{\mathbf{x}}\,, \tag{5.2}$$

where α and α_{ill} are smoothness weights. The variational model comprises three terms: the data term D that is responsible to establish a connection between two consecutive frames geometrically via the optical flow and photometrically via the the parametrized illumination changes (which are encoded in terms of coefficients), the smoothness term S_{flow} which prevents arbitrary fluctuations in the flow, and a regularizer S_{ill} for the illumination coefficients that prevents arbitrary fluctuations within the coefficient field. In the following, let us discuss these terms in detail.

5.6.1 Data Term

Existing data terms for the estimation of the optical flow typically model variations in the brightness between consecutive images as purely induced by motion. In contrast,

our data term allows to explain such variations additionally in terms of illumination changes. Basically, we make use of the data concepts of Bruhn and Weickert [29] (see Chapter 2, Sect. 2.7) and enrich the BCA and the GCA with the illumination coefficients, such that the general data term reads

$$D(\mathbf{w},\mathbf{b}) \quad = \quad D_{\mathrm{BCA}}(\mathbf{w},\mathbf{b}) + \gamma D_{\mathrm{GCA}}(\mathbf{w},\mathbf{b})\,, \tag{5.3}$$

where γ is a positive weight that balances both assumptions. Let us now discuss the two terms in detail.

Compensated Brightness Constancy Assumption

In general, there are two possibilities where to apply the BTF within the data constraint: at the first frame or at the second frame. Since the second frame is evaluated at a displaced position, we decided to apply the BTF to the first frame which reads

$$D_{\mathrm{BCA}}(\mathbf{w},\mathbf{b}) \quad = \quad \Psi_D\left(\sum_{c=1}^{N_c}\left(I^c(\mathbf{x}+\mathbf{w}) - \Phi(\mathbf{b}^c(\mathbf{x}), I^c(\mathbf{x}))\right)^2\right). \tag{5.4}$$

This has some important advantages at the minimization stage: (i) It avoids products of the unknowns in the linearization, (ii) The compensation can be done independently which avoids the question about the detailed procedure when compensating: Is it preferable to first warp the frame and then compensate for illumination or to change the order of compensations?

In order to obtain a consistent notation w.r.t. colors for both the images and the illumination coefficients, we denote by $\mathbf{b}^c(\mathbf{x})$ the coefficient field for the color channel c. Please also note that the brightness changes are modeled to be spatially variant, i.e. with non-constant coefficients $\mathbf{b}$. Hence, we allow different brightness transfer functions Φ at each position.

Temporal Aspects. In the context of camera response functions, the brightness transfer function maps irradiances to intensities. In this context, intensities and irradiances are data at the same time step. In our case, however, we map intensities between frames from different time steps with a temporal distance of Δt. Although we usually consider successive frames to have a fixed non-zero temporal distance (defined as $\Delta t = 1$), the given image frames are treated as slices of a continuous image volume where motions are defined for any temporal distances, including particularly the case $\Delta t \to 0$. In this case, successive image frames converge to be equal, such that $\Phi(\mathbf{b}, I)$ should converge against the identity function. The parametrization in terms of basis functions, however, allows for basis functions where this convergence cannot be guaranteed a priori (in case that $\bar{\phi}(I) \neq I$ which does not have a coefficient) or that do not generalize for other

temporal distances. In general, we thus have to assume the parametrization of the BTF to be time-variant, expressed by adding an index Δt, i.e. we have $\Phi_{\Delta t}$.

This is a necessary remark to understand possible implications in the continuous case. Since our usual case is $\Delta t = 1$, it is, however, in practice sufficient to consider $\Phi := \Phi_{\Delta t=1}$ and to have in mind that $\Phi_{\Delta t=0}(\mathbf{b}, I) = I$ is assumed to be the identity. Any parametrizations at other time steps (i.e. for $\Delta t \notin \{0, 1\}$) are not known and not needed. Please note that in the later learning stage we implicitly assume that the temporal distance $\Delta t = 1$ holds for all used image sequences, i.e. we learn the basis functions particularly for illumination changes between frames with a temporal distance of $\Delta t = 1$.

Compensated Gradient Constancy Assumption

Similar to the BCA, we can also adapt the gradient constancy assumption (GCA) which then reads

$$D_{\mathrm{GCA}}(\mathbf{w}, \mathbf{b}) \quad = \quad \Psi_D \left(\sum_{c=1}^{N_c} ||\nabla I^c(\mathbf{x} + \mathbf{w}) - \nabla \Phi(\mathbf{b}^c(\mathbf{x}), I^c(\mathbf{x}))||^2 \right), \tag{5.5}$$

with ∇ being the spatial gradient operator. This transfers the intended capability to handle the parametrized type of illumination changes also to the gradient constancy.

It may seem surprising to combine both the explicit estimation of illumination changes and a constancy assumption that is based on illumination-invariance. However, its invariance under additive illumination changes together with the robust penalizer can steer the estimation at those locations where an adaptation of the illumination coefficients to the illumination changes is difficult. An example where this is the case are the early stages of the estimation where all unknowns are far away from having converged to their final values. Please keep also in mind that the GCA provides *two* additional constraints at each pixel in the minimization. In a setting where otherwise solely the smoothness terms would resolve the hopeless underdetermination of the equation system (with multiple unknowns at each pixel), additional constraints may stabilize the computations.

Similar to the baseline, the same sub-quadratic Charbonnier penalizer Ψ_D is used for both data constraints.

5.6.2 Regularization Terms

While even a conventional equation system with the two unknowns u and v is highly underdetermined at most locations, since the data constraints locally often fail to provide enough information, an increase of unknowns due to the illumination coefficients worsens the situation. Hence, we need to employ spatial regularization for both the unknowns of the flow and the illumination coefficients. Furthermore, the distribution

of the observed variations in brightness cannot be determined by the data term alone. Besides an appropriate parametrization, that is able to adequately describe the observed illumination changes, there is also the aspect of how to model both regularization terms, which is important to resolve this ambiguity. In the following, we will discuss the design of both smoothness terms.

Flow Regularization

Consistently to the last chapters, we will employ the anisotropic first-order complementary regularizer (see Chapter 2, Sect. 2.8.3) which is given by

$$S_{\text{flow}}(\mathbf{w}) \quad = \quad \sum_{i=1}^{2} \Psi_{Si}\left(\sum_{j=1}^{2}\left(\mathbf{r}_i^{\top}\nabla w_j\right)^2\right), \tag{5.6}$$

where $\Psi_{S1}(s^2) = \epsilon_{S1}^2 \log(1 + s^2/\epsilon_{S1}^2)$ is the Perona-Malik penalizer, $\Psi_{S2}(s^2) = 2\epsilon_{S2}^2\sqrt{1 + s^2/\epsilon_{S2}^2}$ is the Charbonnier penalizer and the direction vectors $\mathbf{r}_1$ and $\mathbf{r}_2 = \mathbf{r}_1^{\perp}$ are defined as in Chapter 2, Sect. 2.8.3, for scenarios with dominant fronto-parallel motion. In case of a dominant non-fronto-parallel motion we will resort to the isotropic second-order regularizer from Chapter 2, Sect. 2.9 which reads

$$S_{\text{flow-AFF}}(\mathbf{w}) \quad = \quad \Psi_S\left(\sum_{j=1}^{2} ||\mathscr{H} w_j||_F^2\right), \tag{5.7}$$

where $\Psi_S(s^2) = 2\epsilon_S^2\sqrt{1 + s^2/\epsilon_S^2}$ is the Charbonnier penalizer.

Coefficient Regularization

As we have seen at hand of the different motion directions (fronto-parallel or non-fronto-parallel), for the flow field different orders of regularization may be appropriate. For the illumination changes, the situation is a little simpler. It typically makes sense to assume that neighboring pixels undergo piecewise similar illumination changes, i.e. the illumination coefficients are piecewise constant. This, for instance, holds for shadows as well as for an adapting camera whose aperture is narrowed or widened when more light is incoming. If there are discontinuities in the coefficient field, it is natural to also assume that they are a subset of the edges in the input images (e.g. shadows) [97]. Consequently, we transfer the successful concept of anisotropic flow regularization to the illumination case and employ the anisotropic first-order complementary regularizer of Zimmer *et al.* [165] on the illumination coefficients:

$$S_{\text{ill}}(\mathbf{b}) \quad = \quad \sum_{i=1}^{2} \Psi_{\text{illum}}^{i}\left(\sum_{c=1}^{N_{\text{c}}}\sum_{j=1}^{N_{\text{cIll}}} \xi_j\left(\mathbf{r}_i^{\top}\nabla b_j^c\right)^2\right). \tag{5.8}$$

In this term, we use the same direction vectors $\mathbf{r}_1$ and $\mathbf{r}_2 = \mathbf{r}_1^\perp$ as in Chapter 2, Sect. 2.8.3. Please note that the associated regularization tensor needs to be computed from the original, i.e. *photometrically uncompensated*, first frame $I(\mathbf{x})$, since illumination edges that are important for the anisotropic regularization might disappear in the compensated version. Moreover, we employ a joint penalization strategy for all coefficient fields whereby a single penalizer per direction is applied, since we assume that any spatial change in the BTF not only leads to a discontinuity in a particular coefficient field but in all of them. Within this joint penalization, we weight the derivative expressions for the different coefficients with weights ξ_j which reflect the different ranges of magnitudes that the coefficient fields may have. The retrieval of these weights is discussed in Sect. 5.7.3, since they are a by-product of the learning process of the basis functions. Similar to the flow regularization case, a good anisotropic behavior is achieved when applying the edge-enhancing Perona-Malik regularizer across edges (in $\mathbf{r}_1$-direction) and the edge-preserving Charbonnier regularizer along them (in $\mathbf{r}_2$-direction) [148].

5.7 Basis Learning for Brightness Transfer Functions

In the previous sections, we have presented the variational model that has to be minimized in order to obtain the optical flow and the illumination changes. Technically spoken, we discussed, how to obtain the illumination coefficients $\mathbf{b}$, i.e. the *magnitude* of the illumination changes. Let us now discuss how to obtain the *type* of illumination changes in terms of the mean BTF $\bar{\phi}$ and the basis functions ϕ_j as well as the associated weights ξ_j that help respecting the orders of magnitude of the illumination coefficients in the corresponding regularizer. In contrast to the magnitude of illumination changes that may vary between different image sequences of a scene, the potential types of illumination changes can be regarded as consistent within a given setting, since they depend on factors that are known a-priori (adaptation behavior of the camera, indoor/outdoor-scenes, lighting conditions etc.). Hence, it makes sense to learn them offline from training data of a given setting. To this end, we take inspiration from the work of Grossberg and Nayar [57], since their *Empirical Model of Response (EMoR)* parametrizes camera response functions of imaging systems in terms of basis functions, and employ a similar learning strategy. However, instead of irradiances, which are an important quantity in imaging systems, our model operates on intensities.

As already mentioned in Sect. 5.5, we relate input intensities and output intensities via

$$\Phi(\mathbf{b}, I) = \bar{\phi}(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi_j(I)\,, \tag{5.9}$$

which allows to represent many kinds of illumination models using appropriate basis functions. This particularly includes polynomial and exponential illumination models such as the affine model of Negahdaripour *et al.* [98] via

$$\bar{\phi}(I) = 0, \quad \phi_1(I) = 1, \quad \phi_2(I) = I,$$

as well as the purely additive models in Cornelius and Kanade [35] and Mukawa [95] via

$$\bar{\phi}(I) = I, \quad \phi_1(I) = 1,$$

and the simple standard case of modeling no illumination changes via

$$\bar{\phi}(I) = I\,,$$

which gives an exemplary overview on how to integrate ad-hoc parametrizations into our framework.

Training Data for the Learning Process

Among all recent benchmarks, the KITTI 2012 Vision Benchmark Suite [52] is a good choice to illustrate our learning strategy, since it provides a rare combination of real-world data and (sparse) ground truth displacements. Due to the large set of image sequences, we are able to consider many real correspondences between input and output intensities in order to analyze true brightness transfer functions with many pixels involved. For simplicity reasons, we will demonstrate the concepts at hand of the scalar intensities of KITTI's grey value images. For color images, the whole learning procedure is either applied to each color channel separately or the BTFs of each channel are combined to learn a joint basis.

5.7.1 General Learning Strategy

We can subdivide our strategy into two phases: an initialization phase to get a first estimate of the basis functions and an iteration phase to get basis functions that are learned from local brightness transfer functions. In the initialization phase, we start by estimating one BTF per image pair via histogram specification [55]. On this set of BTFs, the so-called *observations*, we perform a principal component analysis (PCA) in order to obtain a small set of basis functions that is appropriate to describe the observations.

Iterative Localization of the BTFs. After this initialization phase, we have a first initial set of basis functions based on *global* BTFs. By using them we can proceed with

Algorithm 5.1 Pseudocode for basis learning.

```
initialization:
    setOfBTFs ← {};
    for all image pairs do
        h1, h2 ← compute histograms;
        BTF ← histogram specification on h1, h2;                    //global BTF
        add BTF to setOfBTFs;
    end for
    φ̄, φ, ξ ← perform PCA on setOfBTFs

iteration:
    setOfBTFs ← {};
    for all image pairs do
        b ← computeCoefficients(w^gt, φ̄, φ, ξ);
        segments ← KMeans(b);
        for all segments do
            h1, h2 ← compute histograms on current segment;
            BTF ← histogram specification on h1, h2;                //local BTF
            if not isDegenerated(currentSegment) then
                add BTF to setOfBTFs;
            end if
        end for
    end for
    φ̄, φ, ξ ← perform PCA on setOfBTFs
    goto iteration;
```

the iteration phase, where we can use more *local* BTFs that describe local phenomena (like e.g. drop shadows). To this end, we compute the illumination coefficients using the given ground truth flow and the current version of the basis functions and segment them in the coefficient space using K-Means clustering. Then we estimate a *local* BTF for each segment (again using histogram specification) and apply a PCA on the new set of BTFs over all segments in all image pairs in order to get an improved set of basis functions. These basis functions can then be used for further iterations. An overview in terms of a pseudocode is also given in Alg. 5.1.

This strategy is similar to what Tieu and Miller proposed [135]. They consider shifts in the RGB color space between two images which in total provides the so-called *color flow*. Applying the PCA on the color flows of multiple images then provides a basis which the authors refer to as *color eigenflow*.

5.7.2 Estimating Brightness Transfer Functions

Given a set of m segments from the set of image pairs (which in the *initialization* phase corresponds to one segment per image pair), we now need to estimate m different BTFs $g : \mathbb{R} \to \mathbb{R}$ (one for each segment). In this context, we follow Grossberg and Nayar [56] who estimated *global* BTFs whereby we apply their strategy to local segments instead of global images. To this end, we construct two histograms h_1 and h_2 for each segment, where h_1 accumulates the intensities in the first frame while h_2 accumulates the corresponding intensities in the second frame. We have to ensure to only consider real correspondences between intensities which means that there must be a ground truth flow vector that does not target to a location outside the image domain. Finally, the BTF is given as the result of a histogram specification that transforms the source histogram h_1 to the target histogram h_2.

Restriction to Meaningful Segments. However, we must take care that the segments contain meaningful brightness transfer functions. We assume segments that are too small or that are fully saturated to be harmful in this context. To this end, we filter out any segment that is widely dominated by a single intensity, i.e. in which 80% of the pixel share the same intensity, or that shows a too sparse sampling of the dynamic range, i.e. in which more than one third of all possible intensities are not present.

Representation of the BTF. Please note that we obtain a discrete function as the result of the histogram specification which is given by a vector $\mathbf{g} \in \mathbb{R}^{256} : i \mapsto g_i$ which is not represented as a parametrization in terms of basis functions and the corresponding coefficients. Such basis functions are obtained by the following PCA out of all the BTFs of all remaining segments.

5.7.3 Learning Basis Functions

Based on the results of the last step we obtain $m \leq K \cdot p$ observation vectors $\mathbf{g}_i$ from up to K segments in each of the p training image pairs. From these observation vectors $\mathbf{g}_i$, we want to obtain a common set of basis transfer functions that are able to describe the main aspects behind these observations. To this end, we apply a principal component analysis (PCA) whereby we start by concatenating all observations $\mathbf{g}_i$ $(i = 1, \ldots, m)$ into a so-called *observation matrix*

$$\mathbf{G} \quad := \quad (\mathbf{g}_1 | \ldots | \mathbf{g}_m) \in \mathbb{R}^{256 \times m} . \tag{5.10}$$

From this matrix $\mathbf{G}$ we compute the row-wise mean $\bar{\mathbf{g}}$ that describes the sample mean overall observations. It is then used to derive the covariance matrix $\mathbf{C}$ as

$$\mathbf{C} \quad = \quad \mathbf{U}^\top \mathbf{\Sigma} \mathbf{U} \; = \; \frac{1}{m-1} \sum_{i=1}^{m} (\mathbf{g}_i - \bar{\mathbf{g}})(\mathbf{g}_i - \bar{\mathbf{g}})^\top . \tag{5.11}$$

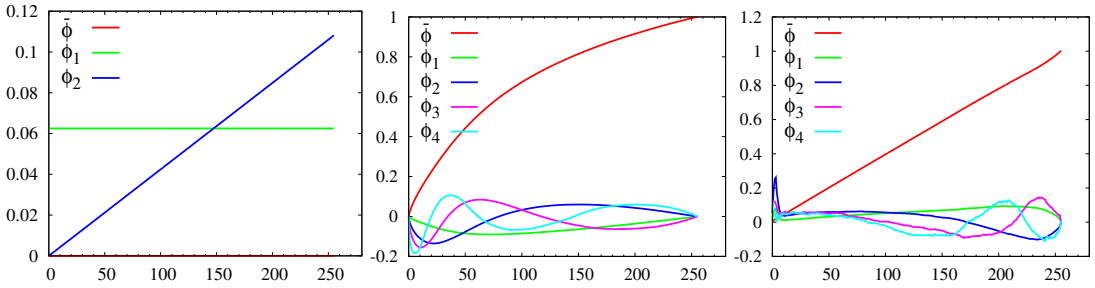


Figure 5.1: Comparison of different basis functions. **From left to right:** (a) Normalized affine basis. (b) EMoR functions [57]. (c) Our basis functions learned from KITTI ground truth data.

Applying a Principal Component Analysis. The eigenvectors of the covariance matrix as the principal components, which are the columns of $\mathbf{U}$, directly serve as the sought basis functions ϕ_j $(j = 1, \ldots, n)$. On top of this, the row-wise mean $\bar{\mathbf{g}}$ is considered to be the 0-th basis function which is called the mean brightness transfer function $\bar{\phi}$. As a by-product, the diagonal matrix $\mathbf{\Sigma}$ is generated by the eigenvalues of $\mathbf{C}$ which express the variance of the given data w.r.t. the principal components. We can consider them as well-suited estimates for the relative magnitudes of the coefficients which are useful to balance the coefficients within the ansiotropic regularization term from Eq. 6.4. For our balancing scheme, we consider the corresponding weights b_j to be the inverse square root of the eigenvalues.

Shapes of Learned vs. Ad-hoc Bases. In Fig. 5.1 we can see the estimated bases for the KITTI Vision Benchmark Suite (2012) in comparison to both an affine basis and the EMoR basis as given by [57]. While the EMoR basis functions model illumination changes in the lower part of the range of intensities, our learned basis functions rather model these in the upper part. A further, big difference can be found in the mean brightness transfer function. The estimated mean BTF is rather linear in contrast to the one of the EMoR basis, since we derive a mapping between intensities where the expected average is given by the identity function. This is not expected when estimating a camera response function as in [57].

5.7.4 Segmenting Illumination Changes

Our goal is now to find *local* brightness transfer functions within the image pairs from the training data, since we assume that different parts of an image can undergo different lighting conditions. To this end, we need to estimate the local brightness transfer function for each pixel. In this context, an obstacle is that for this BTF (which can be arbitrarily complex) there is only *one local constraint* at each pixel: Given the intensity

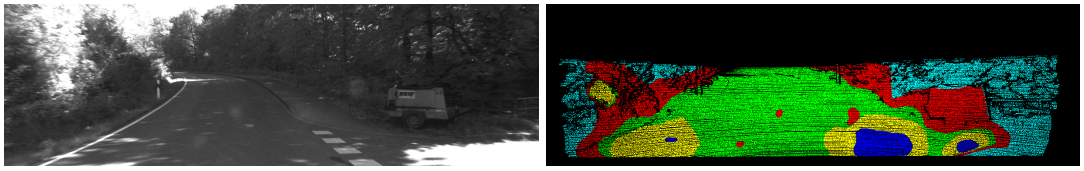


Figure 5.2: Exemplary depiction of the clusters of a coefficient field. **Left:** Frame 1 of the KITTI 2012 training sequence #114. **Right:** Resulting clusters of the K-Means algorithm. While black pixels indicate locations where no ground truth is provided, each color indicates a separate cluster. On the one hand, we can see the three red spots in the depiction that correspond to the inter-reflections at the windshield, while on the other hand, we observe different clusters for the street and the environment which are the result of a stronger brightening effect on the street in contrast to a weaker brightening in the environment.

of the pixel in the first frame as an argument, the unknown BTF must provide the intensity of the corresponding pixel in the second frame.

Estimating Illumination Coefficients. However, this extremely underdetermined problem can be relaxed. Regarding the *types* of illumination changes, the arbitrary complexity of the pointwise BTFs is reduced to those BTFs that are expressible using the basis functions that we have estimated so far. What remains is the determination of the corresponding *magnitudes*. Hence, the problem comes down to an estimation of the coefficient vector $\mathbf{b}$ in each pixel. However, the tools to solve this problem are already given in terms of our model from Sect. 5.6. While keeping the optical flow $\mathbf{w}$ fixed, since we are provided with the ground truth $\mathbf{w}^{\mathrm{gt}}$, the remaining estimation solely concentrates on the coefficients $\mathbf{b}$. In this context, we have to consider that there are some pixels where no ground truth is available (i.e. due to occlusions or due to sparse laser scans) such that we need to disable the data term there. Eventually, we end up in an inpainting scenario using a variational method [153], since at some locations the solution is purely determined by the regularizer of the coefficients. In contrast to conventional inpainting scenarios, the inpainted pixels, where no ground truth is given and thus no correspondence between intensities can be established, are not of interest. The only purpose of this regularization is to enforce a global information flow in order to solve the underdetermination of the otherwise local equation system.

K-Means Clustering. After the coefficients have been determined, they are separated into clusters using K-Means clustering (usually $K = 5$). In this context, we only consider the values within the N_{cIll}-dimensional coefficient space for clustering but intentionally dispense with the spatial coordinates in order to allow spatially unconnected regions to belong to the same segment. We know that all pixels that are part of the same cluster of coefficients do not have substantially different BTFs and thus are assumed to share

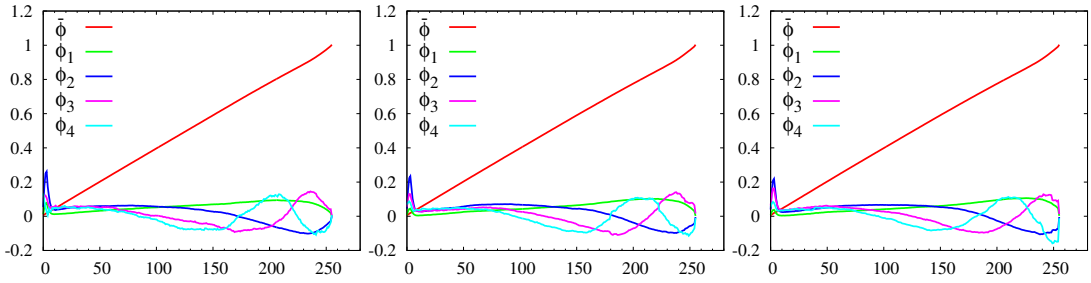


Figure 5.3: Impact of iterating the estimation of the KITTI 2012 basis functions. **From left to right:** (a) Initial basis. (b) After one iteration. (c) After four iterations.

similar lighting conditions. In Fig. 5.2, we find the color-coded result of such a clustering where we clearly see different regions that exhibit different lighting situations.

5.7.5 Iterating the Estimation

After the initial estimates of *global* brightness transfer functions, an iterated estimation of the basis functions using the segmentation step successively localizes the estimations. Hence, local illumination phenomena lead to separate BTFs for the PCA, such that the resulting basis functions can better describe the types of the apparent illumination changes. A clearer separation of these types in terms of more distinct basis functions again allows for a clearer segmentation of the coefficient vectors **b**, since the coefficients can be distributed more clearly among the different basis functions. The impact of iterating the estimation of the basis functions on their shapes can be seen in Fig. 5.3. While the mean BTF remains approximately the identity, the main support of the other basis functions is even further shifted towards the upper end of the dynamic range.

5.8 Aspects of the Minimization

Similar to the methods before, we basically minimize the nonconvex and nonlinear functional using concepts from Chapter 2 which includes the coarse-to-fine warping strategy as described in Sect. 2.6.3 along with the lagged nonlinearity method as described in Sect. 2.3.1. After discretization, the resulting sequences of linear equation systems are solved with a successive overrelaxation scheme (SOR) as hinted in Sect. 2.3. Moreover, we apply constraint normalization as described in Sect. 2.8.1.

5.9 Evaluation

We will split the evaluation of our method for jointly estimating illumination changes and optical flow, which is called *BTFIllum*, into two parts: One part is focused on the

Table 5.1: Comparison of different variants of our method on the full KITTI 2012 training set.

Configuration	Error (BP3)
Baseline (without illumination compensation)	11.17 %
Baseline (only gradient constancy)	10.75 %
Affine basis	11.07 %
EMoR basis	10.64 %
KITTI basis	10.71 %
KITTI basis (iterated)	**10.19 %**
KITTI basis (iterated, only brightness constancy)	10.65 %
KITTI basis (iterated, only gradient constancy)	10.95 %

basis learning stage and directly follows our paper [41]. The second part focuses on the modeling and provides additional experiments. Details on the parameters and their retrieval can be found in Appendix A.7.

5.9.1 Parametrization in terms of a Learned Basis

In the following, we present the results of several experiments that demonstrate the performance of our method which represents illumination changes in terms of learned basis functions. To this end, we evaluate our method on the KITTI 2012 benchmark [52] and compare it to variants with different ad-hoc parametrizations and to methods from the literature. We empirically keep $K = 5$ fixed for the K-Means clustering step and concentrate on a fixed number of $N_{\mathrm{cIll}} = 4$ basis functions, which is a good trade-off between computational effort and complexity. An experiment on the effect of varying the number N_{cIll} of basis functions can be found in the PhD thesis of Demetz [39].

Evaluation of Parametrizations

In our first experiment, we investigate the benefit of jointly estimating illumination changes compared to a purely invariance-driven optical flow computation and compare the performance of different parametrizations of the illumination changes. To this end, we evaluated our method with and without estimating illumination changes. For the former, we used two ad-hoc parametrizations, a learned basis after the initialization phase and our learned basis after four iterations.

In Tab. 5.1, we see the corresponding results. We can observe at hand of our baseline that including a BCA without illumination compensation deteriorates the results compared to a variant where the partially illumination-invariant gradient constancy assumption

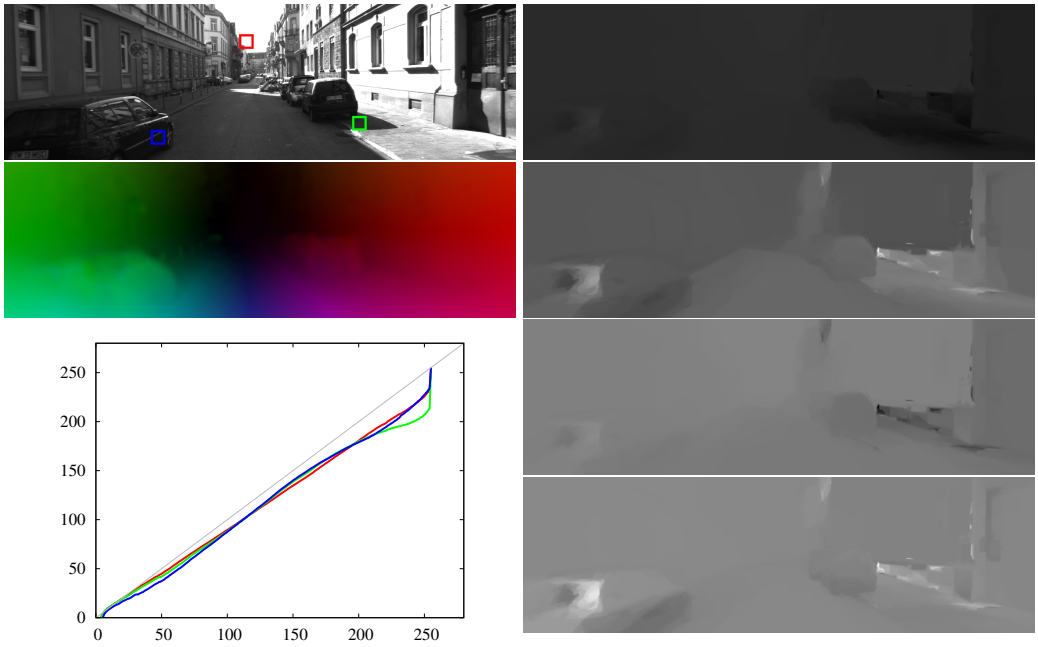


Figure 5.4: Estimated coefficients and BTFs. **Left column, from top to bottom:** Frame 1 of the KITTI 2012 training sequence #15 with three highlighted positions, the estimated optical flow field, a plot of the BTFs where the colors of the BTFs correspond to the colors of the corresponding highlighting boxes. **Right column, from top to bottom:** Estimated coefficient fields b_1 to b_4. The depiction is centered w.r.t. a grey value of 127 which corresponds to a coefficient value of 0. Darker values denote negative coefficients, brighter values represent positive coefficients.

(GCA) is the only data term. Even the usage of an affine model [53] as an example for a rather simple ad-hoc parametrization of the illumination changes does not significantly improve the situation. If we choose the more complex *Empirical Model of Response* [57] as a more advanced example for an ad-hoc parametrization, results begin to improve compared to our baseline method, even if the latter completely relies on the partially illumination-invariant GCA. The same holds for our learned basis after the initialization phase which achieves a comparable result. The best performance, however, is achieved using a basis that is the result of a learning process with several iterations and thus is influenced not only by global but also by local brightness transfer functions at the learning stage. From the last two rows, we furthermore observe that both constancy assumptions - brightness constancy as well as gradient constancy - are necessary to achieve top results. While the BCA does not discard any information and thus is the only assumption that can make use of the estimated illumination changes to their full potential, the GCA improves the initialization at early coarse-to-fine levels where the coefficient fields have not converged, yet.

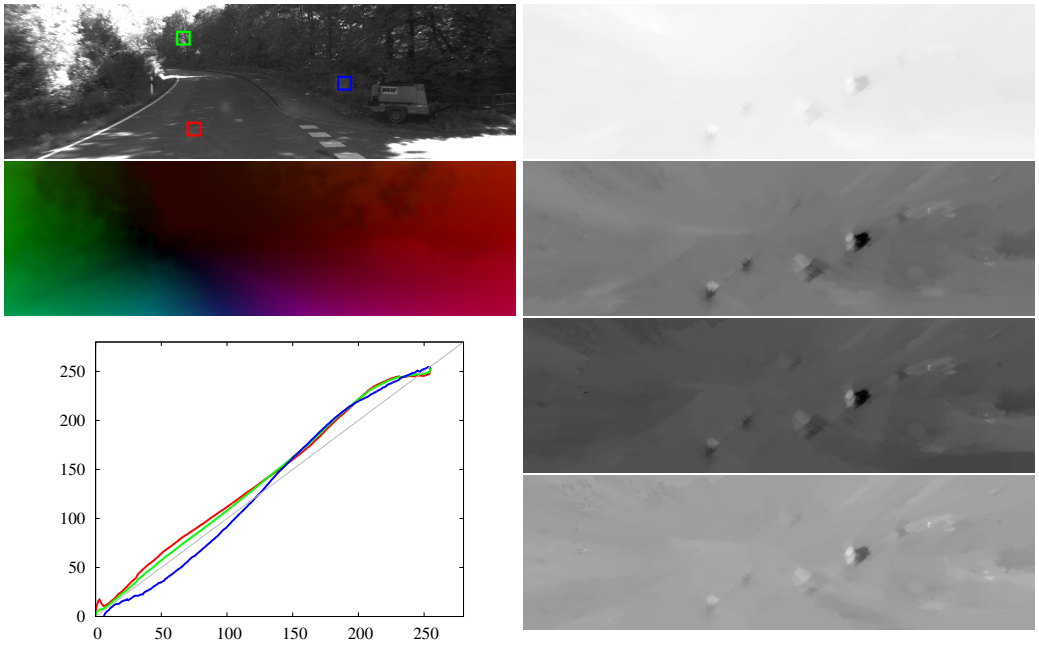


Figure 5.5: Estimated coefficients and BTFs. **Left column, from top to bottom:** Frame 1 of the KITTI 2012 training sequence #114 with three highlighted positions, the estimated optical flow field, a plot of the BTFs where the colors of the BTFs correspond to the colors of the corresponding highlighting boxes. **Right column, from top to bottom:** Estimated coefficient fields b_1 to b_4. The depiction is centered w.r.t. a grey value of 127 which corresponds to a coefficient value of 0. Darker values denote negative coefficients, brighter values represent positive coefficients.

Transfer Functions and Coefficient Fields

In our second experiment, we will have a deeper look at the brightness transfer functions and the corresponding coefficient fields at hand of two example image sequences. To this end, we will show, what effect both rather global and rather local illumination changes have on the BTFs. While the first image sequence, which is depicted in Fig. 5.4, shows rather moderate illumination changes, the second image sequence, depicted in Fig. 5.5 also contains severe illumination changes. In both figures, we display the first frame of each of the sequences along with the flow field and grey scale visualizations of the coefficient fields. In each of these sequences there are locations that are interesting to be analyzed further. We hence highlight locations in the first frame with colored boxes for which we depict the BTFs in a graph below using the respective color. The latter are computed as the linear combinations of the basis functions weighted by the estimated coefficients at these locations.

Table 5.2: Comparison of pure two-frame optical flow methods for the KITTI 2012 evaluation sequences. Superscripts denote the rank of each method in the corresponding column at time of submission (March 7th, 2014). Methods in brackets can not be found anymore in the present state of the ranking or their published results have changed.

Method	Out-Noc		Out-All		Avg-Noc		Avg-All	
DDS-DF	**6.03 %**	1	13.08 %	2	1.6 px	5	4.2 px	3
TGV2ADCSIFT	6.20 %	2	15.15 %	4	1.5 px	2	4.5 px	4
Our method	*6.52 %*	3	***11.03 %***	1	*1.5 px*	2	***2.8 px***	1
Data-Flow	7.11 %	4	14.57 %	3	1.9 px	6	5.5 px	5
(EpicFlow)	7.19 %	5	16.15 %	5	**1.4 px**	1	3.7 px	2
DeepFlow	7.22 %	6	17.79 %	6	1.5 px	2	5.8 px	7
TVL1-HOG	7.91 %	7	18.90 %	10	2.0 px	7	6.1 px	8
MLDP-OF	8.67 %	8	18.78 %	9	2.4 px	9	6.7 px	11
DescFlow	8.76 %	9	19.45 %	11	2.1 px	8	5.7 px	6
CRTflow	9.43 %	10	18.72 %	8	2.7 px	11	6.5 px	9
C++	10.04 %	11	20.26 %	12	2.6 px	10	7.1 px	12
C+NL	10.49 %	12	20.64 %	13	2.8 px	13	7.2 px	13
(IVANN)	10.68 %	13	21.09 %	14	2.7 px	11	7.4 px	14
fSGM	10.74 %	14	22.66 %	15	3.2 px	15	12.2 px	15
TGV2CENSUS	11.03 %	15	18.37 %	7	2.9 px	14	6.6 px	10

The image sequence in Fig. 5.4 contains rather global illumination changes as can be seen at hand of the BTFs. They have a very similar shape which only differs a bit at the upper range of the intensities. Overall, the image sequence becomes darker, which can be seen both from the overall negative BTFs of the highlighted locations as well as from the strongly negative values of the first coefficient field b_1 (which are multiplied with a positive basis function ϕ_1). All plots of the coefficient fields show slight spatial variations which indicate the local adaptations of the illumination estimation at different parts of the scene, including shadows and over- as well as undersaturated regions.

In our second example in Fig. 5.5, we see significantly different BTFs for the highlighted locations of the image sequence. Especially the blue BTF significantly differs from the others, since the corresponding location in the frame contains an inter-reflection in the windshield of the moving car. Such inter-reflections are also clearly visible in the plots of the coefficient fields – sticking out from the rest of these fields.

Table 5.3: Comparison of the bad pixel errors (BP3) for both implementations of our method as well as the corresponding baseline methods. Underlined fonts indicate the best results among a given implementation while bold fonts indicate the overall best result.

Method	Original	New	New (excl. sat.)
Baseline	11.17%	10.19%	–
Baseline (GCA only)	10.75%	10.09%	–
KITTI basis (iterated)	10.19%	9.96%	**9.88%**
KITTI basis (iterated, no GCA)	10.65%	15.04%	10.40%

5.9.2 Comparison to the Literature

In our third experiment, we compare our method to other optical flow approaches from the literature. Tab. 5.2 shows a comparison of the performances of our method and other *pure two-frame* methods *without stereo constraints*. We restrict our comparison, since such constraints are not applicable in other settings where there is dynamic motion that cannot be solely described by camera motion. Moreover, we disregard multi-frame as well as scene flow methods here, since they need additional information compared to our method. At the time of submission, our method achieved state-of-the art performance leading to top results, particularly when considering all pixels (i.e. including also occluded regions). In the latter case, it ranked first, both for the bad pixel measure and for the average endpoint error measure. This clearly demonstrates that the joint estimation of illumination changes and optical flow can outperform other methods that only rely on invariants which discard illumination information.

5.10 Additional Evaluation

As outlined before, the underlying publication [41] of our method was joint work with Demetz (amongst others). Since he focuses on the learning part, further experiments regarding this aspect can be found in his PhD thesis [39]. The current thesis focuses on all other aspects, particularly the modeling part. Hence, we performed additional experiments which include componentwise analyses of the variational model, a novel ad-hoc basis, results on more benchmarks and applications of our approach in more recent methods.

Table 5.4: Comparison of different regularizers for the illumination coefficients.

Regularizer	Error
isotropic	9.91%
anisotropic	**9.88%**

5.10.1 Variational Framework Implementation

Similar to our experiments for our *ALD-Flow* method in Chapter 3, Sect. 3.11, we also build upon a more sophisticated version of our coding framework, which allows to do more advanced experiments on the modeling part. Besides the ability to make use of illumination compensation when using color images, it also offers faster numerical solvers and other minor numerical improvements, such that the results are a bit different. Hence, as a starting point, we will compare the results of our old implementation and our new implementation in our fourth experiment. While improving the implementation, we made the following observation: Over- and undersaturated regions in both images pose a severe problem for the estimation of illumination changes, since there either different intensities are mapped to one intensity or vice versa one intensity is mapped to different intensities. This leads to severe deteriorations in the flow field. Hence, we equipped the data term, that includes both the optical flow and the illumination coefficients, with a spatially variant weight that deactivates it at locations where the grey value of one of the image frames is outside the interval $]0, 255[$. In Tab. 5.3 we see the results of all variants, where *New (excl. sat.)* indicates our new implementation with the mentioned saturation filter. We observe that, even without this filter, the new implementation is superior to the old implementation in all cases except for the case where the GCA is deactivated. When filtering under- and oversaturated regions by deactivating the data term, however, the new variant further improves and is superior in all cases. Nonetheless, we see that including the GCA still remains an important part of our method. Overall, the general tendency that using illumination compensation is beneficial stays the same.

5.10.2 Isotropic vs. Anisotropic Coefficient Regularization

In our fifth experiment, we investigate another important aspect of the model which is the regularizer for the illumination coefficients. It is responsible for disambiguating the otherwise highly underdetermined equation system and for segmenting the coefficients into regions of similar illumination conditions. In order to see if our assumption that discontinuities in the illumination coincide with image edges holds, we compare different regularization strategies for the coefficients. To this end, we juxtapose the results for the complementary regularizer and coefficient-driven isotropic regularization

Table 5.5: Comparison of different schemes for weights ξ_j in the regularizer for the illumination coefficients.

Weights	Error
all equal ($\xi_j = 1$)	9.89%
learned	**9.88%**

in Tab. 5.4. Although there is only a minor difference, anisotropic regularization expectedly is the superior strategy.

5.10.3 Weighted Regularization of the Coefficients

In our sixth experiment, we investigate the importance of the weights ξ_j which balance the regularization of the illumination coefficients b_j^c. From the principal component analysis (PCA) step in the process of basis learning we obtained the eigenvalues which represent the variance of the given data along the basis vectors and serve as the basis to compute the weights ξ_j. Let us now compare our method using these balancing weights to a version where we set $\xi_j = 1$. The results can be found in Tab. 5.5. From these results, we observe that the weighting does not affect the results much. Although the positive effect of using these weights is only of a minor nature, we should keep in mind that the weights come for free as a by-product of the basis learning process.

5.10.4 A Normalized Affine Parametrization

In our seventh experiment, we modify the so far not very convincing affine parametrization for a comparison with our learned basis. While we were in the process of improving our method, we further investigated why the affine parametrization [53] has only a comparably low performance. In this context, it is worth noting that the basis vectors from both the *Empirical Model of Response* [57] and the learned bases are normal vectors except for the mean basis vector. In contrast, the basis vectors from the affine parametrization have norms far bigger than 1. In this case, small spatial variations in the coefficient field **b** lead to potentially big variations in the resulting estimation of illumination changes. Hence, we tested a variant of the affine parametrizations where on the one hand the mean basis vector describes the identity and on the other hand the remaining basis vectors have been normalized, i.e.

$$\bar{\phi}(I) = I, \qquad \phi_1(I) = \frac{I}{n_1}, \qquad \phi_2(I) = \frac{1}{n_2}, \tag{5.12}$$

where n_1 and n_2 are normalization factors such that $||\phi_i(I)||_2 = 1$. In Tab. 5.6, we compare the results of our learned basis with results achieved with the normalized

Table 5.6: Comparison between the learned basis and the normalized affine parametrization.

Parametrization	Error
KITTI basis (iterated)	**9.88%**
Affine basis (norm.)	9.96%

affine parametrization. While the learned basis achieves superior performance, the normalized affine parametrization provides a remarkable trade-off between quality and computational effort, since it only consists of two basis vectors.

5.10.5 Performance on Major Benchmarks

In our eighth experiment, we investigate the performance of our method on all major benchmarks. In contrast to the KITTI 2012 benchmark, all other benchmarks provide color images which requires an appropriate strategy for handling color channels.

Handling Color Channels

When colors come into play, there are a lot of decisions to be made how to handle the different image channels at both the learning stage and the stage of the flow estimation.

Learning Stage. In the learning stage, we can learn the basis functions on grey value versions of the images or on the color images where this can be done either jointly or separately for all channels. If learned separately, there still is the sub-decision to be made whether the clustering of the BTFs shall be conducted jointly or separately for the channels. If both learning and clustering are done separately, this case comes down to treating each of the color channels of the images as grey value images and having a completely independent learning process for each channel.

Estimation Stage. In the stage of optical flow estimation, there again is the option to either use grey value versions of the images or to use the full color spectrum of the original images. When using color images, there is the option to estimate a joint set of illumination coefficients for all image channels or to have separate coefficients for each of the image channels. In the latter case, both the data terms (DT) as well as the smoothness term of the coefficients (ST) offer the options for either a joint robustification over all channels or a separate one.

Results. At hand of the KITTI 2015 benchmark, we determined the results for all possible combinations and state them in Appendix B.

Table 5.7: Results of *BTFIllum* and its baseline on training data of different benchmarks. Here, *basis: gv* denotes a basis that is learned on grey value images, while *basis: jt/jt* denotes a joint basis for all color channels with a joint clustering. In contrast, *basis: sp/jt* denotes a separate basis for each color channel with a joint clustering.

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline (gv.)	2.63	0.214	6.573	4.273	10.19%	23.73%
Baseline (col.)	**2.57**	**0.211**	**6.454**	4.296	–	23.99%
BTFIllum						
1 color channel						
basis: gv	2.69	0.217	6.687	4.335	**9.88%**	**23.55%**
3 color channels *joint coefficients*						
basis: gv	2.61	**0.211**	6.615	**4.110**	–	23.72%
basis: jt/jt	2.61	0.213	6.542	4.438	–	23.87%
basis: sp/jt	2.60	0.212	6.567	4.120	–	23.86%

Major Benchmarks

In order to demonstrate the performance of our method, we do not solely rely on the *overall* best setting for the KITTI 2015 benchmark (see Appendix B), since the characteristics of the Middlebury and the MPI Sintel benchmarks are different from those of the KITTI 2015 benchmark, but we select the most promising options from the results in Appendix B, Tab. 1 and try out different combinations among these options. At the estimation stage, we compare estimations on grey value images with those of color images with joint coefficients. From the learning stage, we obtain bases on grey value images and on color images which comprises both a joint basis for all color channels and a separate basis for each of the channels, using, however, a joint clustering of the coefficients. Hence, we omitted any kind of separate robustification and we omitted to use a separate basis for each of the channels where also the clustering step has been conducted separately for each channel.

In Tab. 5.7 we see the results for these combinations whereby the columns for the KITTI benchmarks contain excerpts from previous experiments. When comparing the baselines with and without color information, we see that color information is helpful in most cases (also for the AEE on the KITTI 2015 benchmark as stated in Appendix B, Tab. 1). The comparison of *BTFIllum* using grey value images with the baseline shows that illumination compensation is beneficial for the KITTI benchmarks

Table 5.8: Impact of the illumination compensation on the variational refinement as proposed by Maurer *et al.* [86] for different benchmarks.

	Sintel (AEE)	KITTI '12 (BP3)	KITTI '15 (BP3)
No Compensation	1.96	9.47%	18.13%
Compensated	**1.94**	**9.29%**	**18.10%**

while deteriorating results on the other benchmarks. When comparing *BTFIllum* using color images and a joint set of coefficients for all image channels with the baseline, there are different observations to be made. For the Middlebury benchmark, which does not contain significant illumination changes, we do not see a significant change in the AEE, while the AAE slightly increases. For the Sintel benchmark, there is a slight decrease in the performance on the subset while results improve for the complete data set. For the KITTI 2015 benchmark, there are also no significant changes in the BP3 error. However, there is some improvement in the corresponding AEE (see Appendix B, Tab. 1). Overall, the joint basis leads to the worst results in most cases, since it covers a basis that is a compromise of representing the different BTFs of the different image channels. Please note that we also tested the use of separate coefficients which, however, demonstrated inferior performance while increasing the workload.

5.10.6 Illumination Compensation for Variational Refinement

In our ninth experiment, we evaluate the influence of the illumination compensation strategy in the context of variational refinement for optical flow. This strategy can not only be used in stand-alone variational methods but also in a variational refinement step of a pipeline approach [111]. Recently, there have been tremendous improvements regarding variational approaches that are used for this step. Amongst other improvements, we propose in [86] to use an illumination-aware data term for refinement using the normalized affine parametrization. Although this method is not in the focus of this thesis, it is interesting to see how the aspect of illumination compensation can help in this context. While the method as a whole has shown superior performance compared to prior work, we will now detail on the effect of the illumination-aware data term. Tab. 5.8 shows the results on three benchmarks with and without illumination compensation. We observe that illumination compensation consistently improves results in all cases whereby the improvement is particularly distinct for the KITTI 2012 benchmark [52]. This is quite remarkable, since illumination changes are already addressed in the matching step of such pipeline approaches by using invariant features. Even on this background, illumination compensation in the refinement step is still useful.

5.11 Summary

In this chapter, we addressed the problem of handling complex illumination changes within a variational optical flow method. To this end, we refrained from solely relying on image features that are invariant under illumination changes, since they discard valuable information and are only invariant under certain types of illumination changes. Instead, we adapted our variational model with a parametrization for illumination changes and a regularization scheme to distinguish illumination-induced from motion-induced brightness changes. This allowed us to estimate the optical flow and the illumination changes jointly.

As a first step, we developed our variational model based on the model of Zimmer *et al.* [164]. Here, we adapted both the brightness constancy assumption and the gradient constancy assumption with a parametrization for illumination changes that consists of basis functions and coefficients. Furthermore, we equipped the model with an anisotropic regularizer for the illumination coefficients to disambiguate the otherwise underdetermined equation system.

In the second step, we described our learning procedure to find a suitable basis for the introduced parametrization. To this end, we extracted brightness transfer functions from image sequences and determined basis functions using a principal component analysis on these BTFs. In order to capture a wider spectrum of such BTFs, we clustered regions of different types of BTFs within the image sequences and iteratively determined basis functions on *local* BTFs.

Our experiments demonstrated the effectiveness of our approach. In any case, where there are substantial illumination changes, we could improve results over the corresponding baseline that only uses invariant image features. Moreover, we demonstrated the benefits of learned bases over ad-hoc bases like an affine parametrization. In the end, our approach does not discard essential information and makes the brightness constancy assumption (BCA), which has many useful properties such as geometric or scale-invariances, also valid in the presence of illumination changes. This will show to be beneficial in the next chapter, where we need the resilience of the BCA to estimate large displacements in the context of illumination changes.

Chapter

6 Large Displacement Optical Flow in the Context of Illumination Changes

In the previous chapters we have seen that large displacements and illumination changes pose severe challenges for variational optical flow methods. We have presented strategies to handle both challenges by keeping a maximal amount of information within the respective variational frameworks. On the one hand, we have a variational framework that jointly estimates and fuses multiple flow candidates. The concept behind that is called de-regularization, i.e. we build on different balances between data term and smoothness term in order to estimate different motion patterns that comprise a different degree of regularity. On the other hand, we have a variational framework that jointly estimates illumination changes along with the optical flow. In contrast, the concept behind this joint estimation is a pronounced regularization of both the optical flow and the illumination changes in order to separate brightness changes into motion-induced changes and real illumination changes.

Combination of Prior Approaches. Due to the importance of the BCA within the de-regularization, our strategy to estimate moderately large displacements is not appropriate in the context of illumination changes (see Chapter 4, Fig. 4.10). Vice versa, our strategy to handle illumination changes did not make use of any concepts to handle large displacements. Hence, in general, it would make sense to combine both strategies in order to be able to handle both challenges at once. However, a straightforward combination of both, i.e. considering multiple instances of our model that jointly estimates optical flow and illumination changes, is not possible, since the requirements w.r.t. regularization are incompatible. A de-regularization strategy would prevent the ability to distinguish motion-induced from illumination-induced brightness changes. In order to illustrate the problem, let us consider a small, clearly isolated region that changes its brightness. This could either be described by a small object that undergoes a large displacement or by a very local illumination change (like a specular reflection

in the extreme case). Without sufficient constraints from the neighborhood the disambiguation of these two descriptions is not possible. A conceivable solution could be to keep a strong regularization of the illumination coefficients while only decreasing the regularization of the flow. However, thereby the problem is that the joint estimation of the illumination changes and the optical flow requires a good balance between three therms: the data term, the regularizer of the flow and the regularizer of the coefficients. A de-regularization applied only to the flow would perturb this balance.

6.1 Contributions

In this chapter, we will, hence, handle the problem of estimating large displacements in the context of illumination changes by decoupling the estimations. We refrain from a joint model and separate the estimation of illumination changes and optical flow into a pipeline of variational methods. In this context, our contributions are threefold in terms of combining ideas from the previous chapters: (i) We prepend a separate illumination compensation step using the approach from Chapter 5 such that the de-regularization strategy can be applied to a modified image pair that does not contain significant illumination changes. (ii) We compute flow candidates using a family of variational methods with varying data terms and varying smoothness weights for the computation of large displacements. (iii) We use a selection strategy similar to the one for *ALD-Flow* in Chapter 3 in order to only integrate helpful candidates into the final estimation.

Regularity Assumption on Illumination Changes. In the whole procedure, we assume that illumination changes are more regular than potential motion patterns, such that we can estimate them from a rather regular flow field that has been computed beforehand using a baseline method (which may make use of illumination-invariant features in the data term). Even if this flow field is invalid at some small regions that may contain relative large displacements, the remaining valid flow vectors are sufficient to allow for an estimation of rather regular illumination changes.

6.1.1 Organization

In order to show the effects of the contributions step-by-step, we will present two variants of such a pipeline: In a first variant, which is a partially decoupled approach, we apply our *ContFusion-Flow* from Chapter 4 (Sect. 4.4) to this modified image pair. That is, we decouple the joint estimation of optical flow and illumination changes into a pipeline of separate steps but keep the *joint* estimation and fusion of multiple flow candidates. In a second variant which is a completely decoupled approach, we will additionally decouple the estimation and fusion of multiple candidates and combine them with a selection strategy similar to the one for *ALD-Flow* in Chapter 3 (Sect. 3.8.2) for the integration into a final flow field.

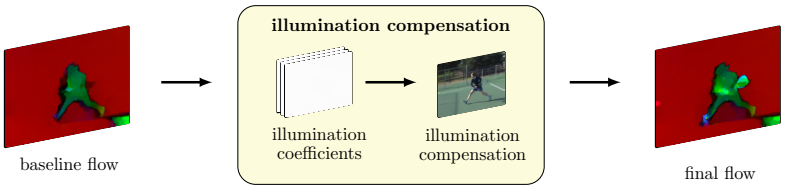


Figure 6.1: Coarse schematic overview of the partially decoupled method at hand of the Tennis sequence [27].

6.2 IC-ContFusion: A Partially Decoupled Method

When we decouple the estimations of both the optical flow and the illumination changes, we need some initial optical flow field as the first step of our pipeline in order to measure the illumination changes between corresponding pixels. In the following, we will describe the three essential steps of our short pipeline: (i) the estimation of the initial flow $\mathbf{w}_{\text{base}}$, (ii) the estimation of the illumination changes based on the pre-computed flow $\mathbf{w}_{\text{base}}$, and (iii) the application of our joint model that estimates and fuses candidate flows as described in Chapter 4 (Sect. 4.4). An overview is given in Fig. 6.1.

6.2.1 Estimation of an Initial Optical Flow

The estimation of the initial flow $\mathbf{w}_{\text{base}}$ is done using our baseline method from Zimmer *et al.* [165, 164] as presented in Chapter 2 (Sect. 2.8).

6.2.2 Separate Estimation of Illumination Changes

The separate estimation of illumination changes can be conducted by a variational approach that is similar to the one in Chapter 5. The main difference is that some optical flow $\mathbf{w}_{\text{base}}$ is now given as input data along with the image sequence. The only unknowns that remain are the illumination coefficients $\mathbf{b} : \mathbb{N} \times \Omega \to \mathbb{R}^{N_{\text{cIll}} \cdot N_{\text{c}}}$, which are associated with a parametrization as in the last chapter, i.e. for each image channel it is given by

$$\Phi(\mathbf{b}, I) \quad = \quad \bar{\phi}(I) + \sum_{j=1}^{N_{\text{cIll}}} b_j \cdot \phi_j(I)\,, \tag{6.1}$$

where $\bar{\phi}$ is the mean transfer function, $\phi_j : \mathbb{R} \to \mathbb{R}$ are the corresponding basis transfer functions, and $\mathbf{b} = (b_1, \ldots, b_{N_{\text{cIll}}})^\top$ is a set of linear weights.

In our decoupled case, they are now estimated as the minimizer of the global energy

$$E_{\text{ill}}(\mathbf{b}) \quad = \quad \int_\Omega D_{\text{ill}}(\mathbf{b}) + \alpha_{\text{ill}} S_{\text{ill}}(\mathbf{b})\, d\tilde{\mathbf{x}}\,, \tag{6.2}$$

which consists of a data term, a smoothness term and a smoothness weight α_{ill}. The data term is given by

$$D_{\text{ill}}(\mathbf{b}) \quad = \quad \Psi_D\left(\sum_{c=1}^{N_{\text{c}}}\left(I^c(\mathbf{x}+\mathbf{w}_{\text{base}}) - \Phi(\mathbf{b}^c(\mathbf{x}), I^c(\mathbf{x}))\right)^2\right) \tag{6.3}$$

where $\mathbf{b}^c$ denotes the part of $\mathbf{b}$ that belongs to image channel c and the penalizer function Ψ_D is the Charbonnier penalizer [33] (as in the baseline method). The smoothness term implements anisotropic regularization and reads

$$S_{\text{ill}}(\mathbf{b}) \quad = \quad \sum_{i=1}^{2}\Psi^i_{\text{illum}}\left(\sum_{c=1}^{N_{\text{c}}}\sum_{j=1}^{N_{\text{cIll}}}\xi_j\left(\mathbf{r}_i^\top\nabla b_j^c\right)^2\right). \tag{6.4}$$

where the direction vectors $\mathbf{r}_1$ and $\mathbf{r}_2 = \mathbf{r}_1^\perp$ are given as in Chapter 2 (Sect. 2.8.3) and ξ_j implement weights to compensate for the potentially different orders of magnitudes that the different coefficients might have. In contrast to Chapter 5, we resort to an affine parametrization, i.e. $\bar{\phi}(I) = 0$, $\phi_1(I) = 1$, and $\phi_2(I) = I$, which is typically a good compromise between quality and complexity. After we have estimated the spatially varying coefficients $\mathbf{b}$, we are finally able to compensate the first frame via $I_{\text{comp}}(\mathbf{x}) \coloneqq \Phi(\mathbf{b}(\mathbf{x}), I(\mathbf{x}))$. This is illustrated in Fig. 6.1.

6.2.3 Final Estimation

Given the modified image sequence with an illumination-compensated first frame, we now apply our *ContFusion-Flow* from Chapter 4 in order to obtain an optical flow estimation that contains large displacements.

6.2.4 Accuracy Issues within the Pipeline

This decoupling strategy 6.2.1 – 6.2.3 has one major drawback: in contrast to a joint estimation, where the estimations of all unknowns mutually benefit from each other, the benefits within a pipeline have a sequential nature. Moreover, each later step relies on the assumption that the results of prior steps are accurate. In the real world, however, this assumption is not fulfilled. This becomes obvious between the first step (optical flow estimation) and the second step (estimation of illumination changes).

Initial Optical Flow Estimation. Occlusions or mismatched objects in the optical flow, which is used as an input to estimate illumination changes, will very likely deteriorate the estimation in the second step, since there are flow vectors that map locations of one object to those of another object. Such incorrect displacements introduce wrong correspondences between grey values, which do not describe correct illumination

changes. It is hence necessary to detect and mask unreliable regions in the flow field before using it for the estimation of illumination changes.

Estimation of Illumination Changes. In contrast to the flow estimation, the computation of the illumination changes does not provide comparable obstacles to the subsequent steps. Given an accurate optical flow with enough reliable matches, which may be indicated by an appropriate mask, we can estimate the illumination changes using the functional in Eq. 6.2 that comprises a data term that is explicit in the unknown illumination coefficients. Hence, the estimation is by far better tractable than the estimation of the optical flow. In case of a set of strictly convex penalizer functions, the separate estimation of illumination changes is even well-posed and thus has a unique solution that continuously depends on the input data. Nonetheless, also for the non-convex Perona-Malik regularizer, which is usually applied in the anisotropic smoothness terms that we use, we usually obtain a meaningful solution. The problem is further relaxed, since we are not actually interested in the coefficients $\mathbf{b}$ themselves but in the resulting brightness transfer function (BTF) $\Phi(\mathbf{b}, I)$ which we want to use in order to compensate the first frame for the estimated illumination changes. So even non-convex functionals are acceptable as long as they lead to useful BTFs. To sum it up: We can expect an accurate estimation of the illumination changes if the given optical flow contains enough reliable matches after having masked out the unreliable ones.

Extending the Pipeline. Hence, we need to assess the quality of the optical flow to mask out unreliable matches while, in contrast, we do not need any further post-processing steps on the estimated illumination coefficients. We thus end up in a four-step pipeline: (i) estimating a basic optical flow, (ii,a) finding and masking unreliable locations in the flow, (ii,b) estimating the illumination changes using the results from the previous steps and (iii) estimating an improved optical flow using an illumination-compensated version of the first frame in the image sequence. We have already seen so far how to deal with Step 1 (see Chapter 2, Sect. 2.8) and Step 3 (see Chapter 4, Sect. 4.4) of this pipeline. For the identification of unreliable locations in the flow, however, let us have a deeper look into the question how to locally measure the quality of the optical flow. Later on, we will also modify the model in Eq. 6.2 to conduct Step 2 properly.

Measuring Flow Quality

In Chapter 3, Sect. 3.8.2, where we proposed the adaptive sparsification strategy, we have already identified regions in the optical flow which need further guidance by feature matches. To this end, we made use of the data energy of a given baseline flow. There, we focused on using the same baseline model twice, first without feature matches and afterwards with an additional similarity term that incorporates the feature matches in the estimation. This was appropriate, since we explicitly wanted to assist the chosen method by providing a set of feature matches tailored to overcome local misestimations

of this method. This set was chosen to be sparse, since the underlying feature matches are not necessarily reliable.

Dense Assessment of the Flow Quality. Now, the situation is different. Instead of *sparsely* complementing the estimation with additional information (i.e. feature matches), we now want to *densely* mark regions in the flow as unreliable and exclude them from further steps. Moreover, we explicitly address illumination changes now, a scenario where the brightness constancy assumption (BCA), which amongst others has been used to compute the data energy so far, is not applicable anymore. This is due to the fact that it is not invariant under any type of illumination changes. We, thus, need a different constancy assumption to assess the flow quality. A possible candidate is the data term of the baseline method. Since, however, this method or at least its data term are assumed to be designed such that it models the occurring brightness changes, we cannot rely on a specific type of data energy. Because of that, we should consider that most data constancy assumptions discard information due to being invariant under certain illumination changes and thus lead to sparse data energies, i.e. they usually are not a *dense* indicator. As we will see in the following, this particularly holds for the gradient constancy assumption (GCA) as a popular example for an invariant constancy assumption. Hence, we compare different types of data energies that arise from different types of data constancy assumptions regarding their usefulness to asses flow quality at different levels of image structuredness. Our goal is to implement a general-purpose strategy that is able to densely indicate unreliable regions based on common properties of energies of common data constancy assumptions.

Comparison of Data Energy Types. Let us consider typical types of data constancy assumptions (brightness-based, gradient-based and patch-based ones) and compare properties of their respective energies for a given baseline flow in order to investigate their usefulness as a quality measure for optical flow. This comparison consists of two parts: (i) In Fig. 6.2, we have a visualization of an image sequence, an exemplary baseline flow that contains mismatched regions and a visualization of such data energies. Here, a feature constancy assumption based on the Geometric Blur feature serves as an example for a patch-based constancy assumption. (ii) Tab. 6.1 roughly summarizes the orders of magnitudes of the corresponding data energies. Having the aperture problem in mind, we remember that the amount of information in image data corresponds to its structuredness. Since different data constancy assumptions rely on different types of information, we juxtapose in Tab. 6.1 the energies of good and poor matches in regions of different structuredness. In this context, a *high* structuredness means that there is a considerable structure at at least one side of the correspondence (origin or target of a match) and a *low* structuredness means that no considerable structure is involved. In general, we hereby assume that a match does not relate different objects with a too similar appearance.

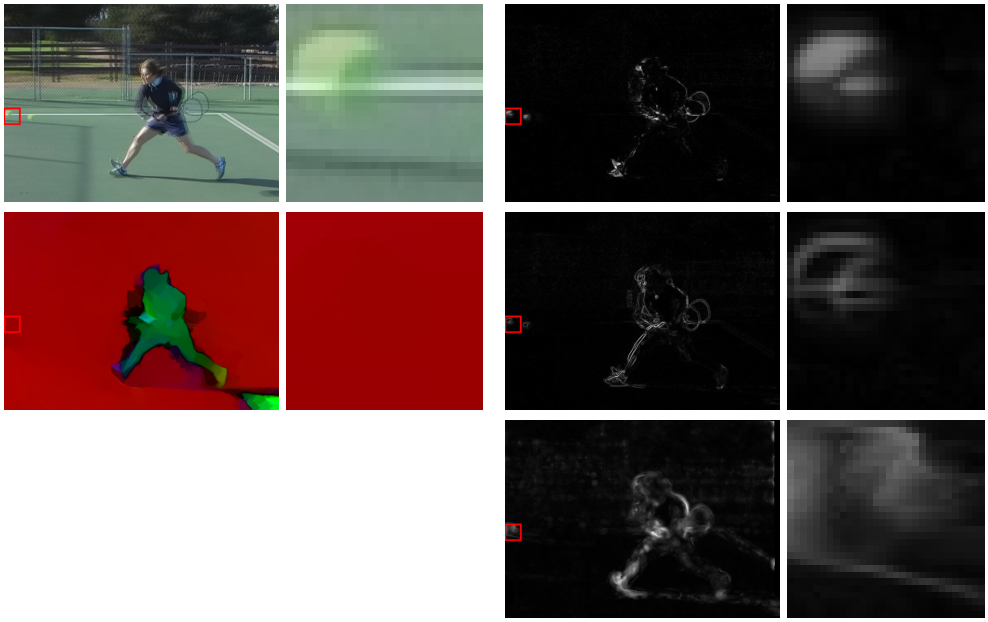


Figure 6.2: Comparison of data term energies (with zoom-ins) for different constancy assumptions evaluated on the same baseline flow for the Tennis sequence. **Left column:** Overlayed input frames, baseline flow field. **Right column:** Data energies evaluated using the BCA, GCA and Geometric Blur constancy. Higher intensity denotes a higher data energy.

Structured vs. Homogeneous Regions. First of all, among all types of energies we can see that poor matches generally lead to high data energies if there is sufficient image structure. However, the first row of Tab. 6.1 (poor + low) shows that in flat regions gradient-based and patch-based data terms may lead to low data energies even if the match is poor. In case of gradient-based assumptions, this is not surprising, since gradients discard essential information and vanish in flat regions. Hence, homogeneous regions of different brightness levels are matched without cost. This can also be seen at hand of the inner part of the tennis ball in the second row of Fig. 6.2 where the homogeneous parts of the tennis ball are matched to the homogeneous tennis court without cost. In case of the patch-based data terms, this problem is related to the use of neighborhood information. High energies may happen if the neighborhood undergoes changes in geometry or scale, or if it covers multiple objects (see Fig. 6.2, last row, where the high energies are only roughly related to the corresponding mismatched object). In contrast, the first column of the table (BCA) shows that the data energy with brightness constancy is the most reliable indicator for the matching quality, since it even makes use of the uniform brightness information that is present in homogeneous regions. This, however, requires the absence of illumination changes, since, otherwise,

Table 6.1: Data energy magnitude for matches of different quality in image regions of different types (without illumination changes).

Match quality	Structuredness	Data Term Energy		
		BCA	Gradient-based	Patch-based
poor	low	high	low	high/low
	high	high	high	high
good	low	low	low	high/low
	high	low	low	high/low

the conventional BCA is not of good use as an indicator for matching quality. Due to our focus on illumination changes, however, we hence rely on the data energy at image edges as a reliable indicator of the flow quality and transfer this indication also to the homogeneous parts of the corresponding objects.

Masking Unreliable Matches

As said before, our approach shall work for any kind of baseline method. In the very general setting, we thus neither know which data term is used nor which particular types of illumination changes are present and to which degree or if there are any at all. That means that we need to prepare for a very general setting.

Extracting Data Energy at Edges. Nevertheless, from Tab. 6.1 we know that for any of the discussed types of data constancy assumptions we can rely on the order of magnitude of the data energy at locations where structure is involved (at the source of the match or at its target). If this data energy shows a high value, we have a mismatched (part of an) object. The structure of an object either origins from its contours or from its texture. It is, however, not sufficient to exclude only incorrect displacements at image edges. Also the homogeneous parts of a mismatched object can deteriorate the estimation of the illumination changes. So the question remains what to do with these parts – where the flow quality cannot necessarily be assessed using the data energy.

Inpainting Data Energy to Flat Regions. At this stage of the pipeline, we start by evaluating the data energy at locations that involve structure (either at the origin of a flow vector in the first frame or at the corresponding end in the second frame) and obtain a sparse map of scalar values where high values indicate bad flow quality. This sparse map is then used as the starting point for a variational inpainting approach. This approach is supposed to inpaint the energy from the edges to the remaining parts of

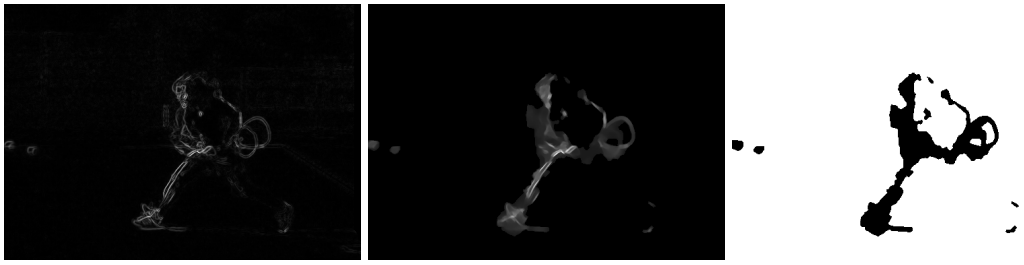


Figure 6.3: Exclusion of poor matches. **From left to right:** (a) Data term energy of the baseline method. (b) Energy after inpainting. (c) Mask χ_{ill} after thresholding.

an object. By thresholding the inpainted result, we obtain a dense binary map that indicates reliable and unreliable regions of the given optical flow.

Required Accuracy of the Quality Measurement. Concerning the accuracy at this stage, our assumption on the regularity of the illumination changes comes into play. Since we do not assume them to be too local but rather regular, we do not necessarily need all flow vectors to estimate the illumination changes within a region and can even afford to locally mask more flow vectors as unreliable than necessary. It is, hence, no problem if the inpainted energy spatially exceeds the concerned object a bit, since there still remain enough unmasked pixels, i.e. reliable matches, to recover all illumination changes. High spatial accuracy is thus not an issue here. Moreover, the accuracy of the values is also not a big issue, since these values are only used for the binary decision whether the respective location is likely to contain a reliable flow vector or not.

Variational Inpainting of the Reliability Mask. In order to obtain a reliable dense quality indication map e, let us start with an evaluation of the baseline data energy in terms of $e_{\text{base}} := E_{\text{Data}}(\mathbf{w}_{\text{base}})$. Additionally, we define a simple edge indicator

$$\chi_{\text{inp}}(\mathbf{x}) \quad = \quad \delta[|\nabla I(\mathbf{x})| > 10 \vee |\nabla I(\mathbf{x}+\mathbf{w}_{\text{base}})| > 10]\,. \tag{6.5}$$

Then, a variational model that inpaints the given energy e_{base} into the denser map e is given by

$$E_{\text{inp}}(e) \quad = \quad \int_{\Omega} \chi_{\text{inp}} D_{\text{sim,e}}(e) + (1-\chi_{\text{inp}}+\epsilon_{\text{inp}})\alpha_{\text{inp}} S_{\text{e}}(e)\; d\tilde{\mathbf{x}}\,, \tag{6.6}$$

where the similarity term

$$D_{\text{sim,e}}(e) \quad = \quad (e-e_{\text{base}})^2 \tag{6.7}$$

enforces the solution to be similar to the data term energy e_{base} of the baseline and the smoothness term

$$S_{\text{e}}(e) \quad = \quad \sum_{i=1}^{2} \Psi_{S_i}\left(|\mathbf{r}_i^{\top}\nabla e|^2\right) \tag{6.8}$$

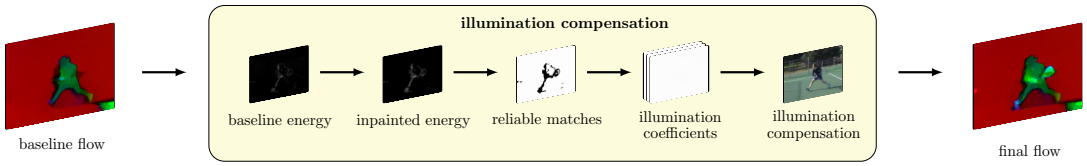


Figure 6.4: Detailed schematic overview of the partially decoupled method at hand of the Tennis sequence [27].

performs first-order anisotropic regularization [164] to align the energy with object boundaries. Thereby, α_{inp} is a smoothness weight, the functions Ψ_{S_i} are given by the Perona-Malik- and the Charbonnier-functions, the direction vectors $\mathbf{r}_i$ are derived from the regularization tensor (see Chapter 2, Sect. 2.8.3) and the small constant $\epsilon_{\text{inp}} = 0.01$ guarantees a minimum amount of regularization. The final mask for excluding unreliable regions is then computed via thresholding the inpainted energy, i.e.

$$\chi_{\text{ill}}(\mathbf{x}) \quad = \quad \delta[e < 0.1 \cdot \max(e)] \,. \tag{6.9}$$

An illustration of this procedure is given in Fig. 6.3.

Adaptive Estimation of Illumination Changes

After discussing the accuracy issues of the decoupled approach and proposing a generalized procedure to mask regions that can deteriorate the estimation of the illumination changes within the scene, we can now adapt the model from Eq. 6.2 to include the reliability mask χ_{ill} which reads

$$E_{\text{ill}}(\mathbf{b}) \quad = \quad \int_{\Omega} \chi_{\text{ill}} D_{\text{ill}}(\mathbf{b}) + \alpha_{\text{ill}} S_{\text{ill}}(\mathbf{b}) \, d\tilde{\mathbf{x}} \,. \tag{6.10}$$

In this model, the data term is deactivated where $\chi_{\text{ill}}(\mathbf{x}) = 0$. The corresponding regions, where thus no correspondences are given, are assumed to undergo illumination changes similar to their environment. Here, the smoothness term fulfills a role beyond disambiguating the otherwise underdetermined equation system: It fills in missing information by propagating it from the neighborhood.

6.2.5 Overview of the Pipeline

Before going on to the evaluation, let us shortly summarize the steps of our partially decoupled method. After having computed an initial baseline flow, we determine the illumination changes within the image pair using this flow. Since outliers in the baseline flow might deteriorate the result, we first determine an inlier mask of the baseline flow by thresholding an inpainted version of the baseline energy. This inlier mask is then used in the estimation of the illumination coefficients. An overview of the pipeline is given in Fig. 6.4.

Figure 6.5: Results for Tennis sequence with artificial illumination changes. **From Left To Right:** (a) Gradient constancy assumption (GCA). (b) Geometric Blur constancy assumption (GBCA). (c) Complete Rank Transform (CRT). (d) BCA and GCA after illumination compensation.

6.2.6 Evaluation

We conduct both qualitative and quantitative experiments to evaluate the performance of our partially decoupled method. They investigate the performance on benchmark data as well as the ability to simultaneously handle illumination changes and large displacements. We use our baseline, the method of Zimmer *et al.* (see Chapter 2, Sect. 2.8), for the initial flow that is used as input for the illumination compensation step. Please note that the parameters δ, γ and α for the data constancy assumptions and the smoothness term appear twice in our pipeline, once for the computation of the initial flow and once for the computation of the final flow (using *ContFusion-Flow*), and they are optimized separately.

Large Displacements

In our first experiment, we investigate the benefits of our method in the context of large displacements. To this end, we use a version of the Tennis sequence [27] which we modified by adding artificial illumination changes. The chosen global additive and multiplicative changes resulted in a significantly darker version of the second image, such that the pure brightness constancy assumption (BCA) is not an appropriate constancy assumption anymore. In Fig. 6.5 we juxtapose the results of our method with

Table 6.2: Comparison of a pure *ContFusion-Flow* (no illumination compensation) and our partially decoupled method on the subset of the Sintel training data (*Sintel (sub.)*).

	ContFusion-Flow	partially decoupled method
Baseline	6.375	-
$N_{\text{cand}} = 1$	6.365	6.121
$N_{\text{cand}} = 2$	6.028	5.998
$N_{\text{cand}} = 3$	**5.808**	5.895
$N_{\text{cand}} = 4$	5.832	5.948

different invariant data constancy assumptions but without illumination compensation and the variant with BCA and GCA data terms and prior illumination compensation. To this end, we used a variational model with the pure GCA as a data term, a model with a constancy assumption based on the Geometric Blur feature [14] (GBCA) and a model that applies the Complete Rank Transform (CRT) [40] in a constancy assumption. All these underlying features have different degrees of illumination invariance, ranging from an invariance under global additive changes (GCA and GBCA) up to a wide-ranging invariance under any monotonic changes (CRT). However, for all of these invariant constancy assumptions we notice many artifacts and for the GCA and the CRT also a limited ability to capture large displacements. For the GBCA, which captures large displacement motions, the flow is not very accurately localized due to the patch-based nature of the constancy assumption. In contrast, our method with illumination compensation is able to capture these motions while being sufficiently local and showing comparably few artifacts, which, however, leave room for further improvements.

Quantitative Evaluation

Now let us have a look, how our modifications change the results on benchmark data. To this end, we compare the pure *ContFusion-Flow* (no illumination compensation) with our partially decoupled method that makes use of illumination compensation at hand of our subset of the MPI Sintel training data [31]. We optimized the parameters for the most promising amounts $N_{\text{cand}} \leq 4$ of candidate models, since more candidates have not improved results further in Chapter 4 (Sect. 4.4). From Tab. 6.2 we can observe that the illumination compensation considerably improves results for the amounts of candidates that only contain a higher degree of smoothness and thus are a little less data dependent ($N_{\text{cand}} < 3$), while results deteriorate a bit when smoothness of the additional candidates is further reduced and the data dependency increases ($N_{\text{cand}} \geq 3$). In this context, we do not achieve a quantitative improvement over *ContFusion-Flow* in terms of the overall best result.

6.2.7 Interim Conclusion

We have seen that our pipeline with a separate estimation of illumination changes and a joint estimation of multiple motion patterns partially improves results, since it is able to estimate large displacements even in the presence of severe illumination changes as well as it is able to improve results quantitatively compared to the *ContFusion-Flow* without illumination compensation for low values of N_{cand}. However, we have also observed that deteriorations come into play when the data dependency is too strong, which can be seen at hand of the slightly worse results quantitative results for $N_{\text{cand}} \geq 3$ as well as at hand of the small artifacts that are visible in the Tennis sequence. In the following, we will hence discuss advanced adaptations to the method in order to prevent such deteriorations.

6.3 ICALD-Flow: A Completely Decoupled Method

One idea to prevent the deteriorations that we have seen so far is to extend our pipeline with a selection strategy that incorporates information about locations where additional guidance is promising and the local structuredness is high enough to allow for meaningful flow vectors in a de-regularized setting (similar to the one in Chapter 3, Sect. 3.8.2). This way, we include less regular candidate motion patterns only at locations where the baseline flow is not appropriate, such that all remaining locations that have been estimated accurately cannot be deteriorated by artifacts from less regularized flows. The embedding of such a strategy is the most straightforward when it is applied to a set of known matches within a sequential pipeline. Since we have already refrained from a completely joint model and initiated a pipeline of separate steps, we may thus also conduct our decoupling strategy to an even higher degree by decoupling the estimation of candidates and their integration into a final flow field, which was done jointly so far by relying on the *ContFusion-Flow* as final step. On the one hand, we can apply the selection step directly after the estimation of the candidates in a straightforward way. On the other hand, the decoupling allows for faster computations, since the workload increases only linearly w.r.t. the value of N_{cand} due, since the number of equations to solve increases, whereas coupling the models not only increases the number of equations but also their sizes (due to a quadratic increase in the size of the motion tensors). Moreover, it allows for some kind of black-boxing where an image sequence and a baseline flow field are given as an input and a set of selected regularized and reliable matches are provided as an output for further integration, whereby we employ a selection strategy similar to the one of *ALD-Flow* (see Chapter 3, Sect. 3.8.2). We call our method *Illumination-Compensated Adaptive Large Displacement Optical Flow (ICALD-Flow)*. It is based on our paper [128].

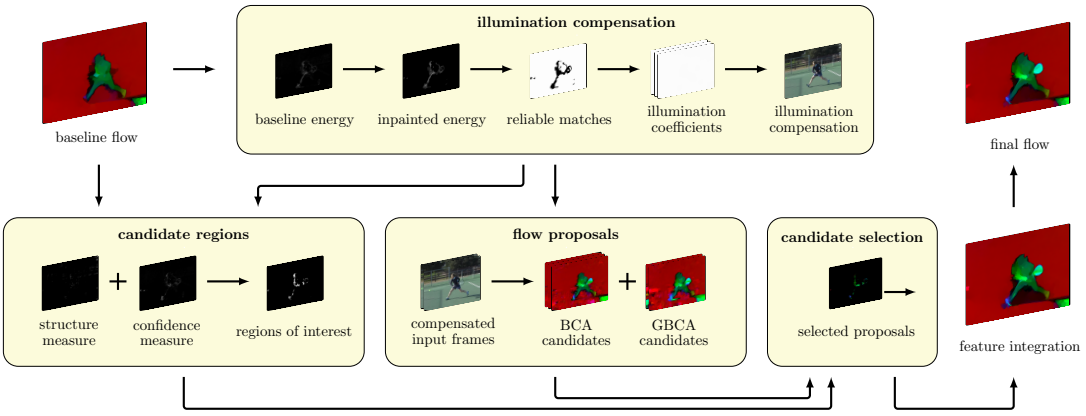


Figure 6.6: Schematic overview of the completely decoupled method (*ICALD*) at hand of the Tennis sequence [27].

6.3.1 Overview

Before detailing on the construction and integration of candidate matches $\mathbf{w}_P$, let us first give an overview of the overall pipeline approach; see Fig. 6.6. The first steps are the same as before: after computing an initial flow $\mathbf{w}_{\text{base}}$ with our baseline, we use this flow to explicitly estimate the illumination changes between both frames. Compensating the first frame for these changes then allows us to rely on illumination-compensated image data in the remaining pipeline. Our modified pipeline now continues with the following steps: (i) Firstly, we identify candidate regions for the integration of flow proposals. (ii) Secondly, we compute different flow proposals from dense variational methods via de-regularization, i.e. by successively reducing the amount of smoothness. (iii) Thirdly, we determine the best candidates from the previously generated flow proposals for each location in the candidate regions.

Usage of Illumination Compensated Image Data. In Step (ii), the compensated image data turns out to be particularly useful, since we can employ the illumination-compensated brightness constancy and the illumination-compensated geometric blur constancy as data terms of dense variational models to calculate the flow proposals. Moreover, it can also be beneficial in the other steps where we need confidence functions that make use of image data.

Confidence Measures. In those steps where we further need to assess the quality of the flow field, we make use of confidence measures: In Step (i) we need a confidence measure to assess the quality of the baseline flow in order to identify candidate regions. In Step (iii) we need a confidence measure to assess the quality of different candidates from the estimated flow proposals relative to each other in order to determine the best candidates.

6.3.2 Confidence Measures for the Optical Flow

In the complete pipeline, there are a lot more stages where we need to assess the quality of flow vectors and each of these steps has different requirements w.r.t. accuracy of the assessment and w.r.t. the ranges of values of the confidence measure in structured and in homogeneous regions (see Fig. 6.2 and Tab. 6.1).

Requirements on Confidence Measures within the Pipeline

The requirements on the spatial as well as the quantitative accuracy of a confidence measure highly depend on its application. While binary decisions between highly distinctive classes require less accuracy in a quantitative sense, a selection among potentially many, less distinctive, classes requires a higher quantitative accuracy. Some stages in our pipeline are binary (when creating exclusion/inclusion maps) while the selection step is n-ary due to selecting from multiple options. The spatial accuracy particularly affects the creation of inclusion/exclusion maps in terms of thresholding confidence values, like e.g. the exclusion of poor matches for estimating the illumination changes or the definition of regions of interest for the integration of candidate flows. If a high spatial locality is necessary in such a map, we need a high spatial accuracy in the confidence measure.

(i) **Excluding Poor Matches for Illumination Estimation.** As in the partially decoupled method, we need to identify poor matches in the baseline flow in order not to deteriorate the estimation of the illumination changes. This is done in the sense of a classification whether a flow vector is good enough to determine illumination changes or not. This classification is binary and it does not require the last bits of accuracy both spatially and quantitatively, since we can even afford to exclude more matches than necessary. A few flow vectors per region with constant illumination conditions are sufficient to determine the respective illumination changes, as long as all harmful matches of mismatched or occluded objects are densely prevented from deteriorating this estimation. Here, accuracy is less important, but the confidence measure must clearly indicate poor matches both in structured and in homogeneous regions to allow for their dense exclusion via thresholding the confidence value.

(ii) **Identification of Regions of Interest.** We need to identify regions of interest where additional guidance is necessary. To this end, we also make a binary decision whether to integrate further flow proposals at some location or not. In contrast to the former stage, however, spatial accuracy is an issue here, since we do not want to integrate matches at locations where the baseline flow is already sufficiently accurate. At such locations, there is no potential for improvement but a potential for deterioration by erroneously considering bad flow proposals.

However, a dense integration of flow proposals is not mandatory as long as each object which has previously been mismatched in the baseline result is covered by enough matches in the final steps. Even if confidence values in homogeneous regions are low for poor matches and, thus, the thresholding step does not mark them as regions of interest, this is typically not a big problem for small objects since their more structured parts are marked as regions of interest which typically is a sufficient covering. Hence, the quantitative accuracy at structured regions is more important than at homogeneous regions.

(iii) **Selection of the Best Candidates.** Among a set of proposals that come from flow fields with varying smoothness, it is our goal to select the most appropriate flow candidate. Here, the decision is not binary anymore, since it becomes a decision among multiple choices. In contrast to the steps before, that are based on thresholding with global thresholds, here we need a local relative ordering of flow proposals according to their quality. In this context, the requirements w.r.t. accuracy are even higher, since the proposals may differ not much w.r.t. their apparent matching quality. Although being substantially different in their displacement due to different degrees of regularization, the aperture problem may let them appear similarly well-suited as motion candidates for a given location.

We have seen that from stage to stage either the locality of the decision increases or the available choices are less distinct. This increases the requirements w.r.t. accuracy on a potential confidence measure that assesses the quality of a flow vector at each stage.

Restrictions on Confidence Measures

At stage (i), where we are about to densely mark unreliable flow regions and can not yet rely on illumination-compensated image data, the inpainting of a (potentially) sparse quality indicator for the baseline flow is the only reliable way for a clear identification of poor matches that works under all circumstances, i.e. with and without illumination changes. Without such an inpainting, the thresholding step could not reliably classify poor matches in homogeneous regions, since the confidence values of poor and good matches are not clearly distinguishable for those types of confidence measures that are invariant under illumination changes (see Fig. 6.2 and Tab. 6.1). For the basic indicator map, the baseline energy is the canonical choice. Compared to the partially decoupled method, we hence do not alter the pipeline at this stage.

Extended Selection at Later Stages. In all later stages of the pipeline, where accuracy is more important and where we can make use of both the original image data as well as the compensated counterpart, there are a lot of choices for the confidence measure that are applicable in the context of illumination changes.

A Selection of Confidence Measures

Given typical constancy assumptions in data terms and the fact that we can access illumination-compensated image data, the following examples are worth investigating as choices for the confidence measure $\rho(\mathbf{w})$:

- First of all, there is the baseline energy (in our case using the BCA and the GCA), which respects the invariances of the underlying baseline model. In this case, evaluating the original data term provides the confidence values. Since it typically consists of a weighted sum of different constancy assumptions, it is beneficial to normalize the energy by these weights γ_i. Hence, we set

$$\rho_{\text{orig}}(\mathbf{w}) \quad = \quad \frac{1}{\sum_i \gamma_i} D(\mathbf{w}) \,. \tag{6.11}$$

- In order to be invariant under additive illumination changes, the evaluation of the gradient constancy assumption (GCA) for the flow proposals is the simplest confidence measure. It is given by

$$\rho_{\text{GCA}}(\mathbf{w}) \quad = \quad \Psi_{\text{D}}\left(\sum_{c=1}^{N_{\text{c}}} |\nabla I^c(\mathbf{x}+\mathbf{w}) - \nabla I^c(\mathbf{x})|^2\right) . \tag{6.12}$$

- An alternative way to using invariant constancy assumptions is to use the brightness constancy assumption (BCA) on the compensated image data, which we call *photometrically-compensated brightness constancy assumption (BCA comp.)*. It reads

$$\rho_{\text{BCA comp.}}(\mathbf{w}) \quad = \quad \Psi_{\text{D}}\left(\sum_{c=1}^{N_{\text{c}}} (I^c(\mathbf{x}+\mathbf{w}) - I^c_{\text{comp}}(\mathbf{x}))^2\right) , \tag{6.13}$$

  and needs illumination-compensated image data to work with illumination changes. While it aims at ignoring the illumination changes, it still keeps the idea of assessing the quality of flow fields by evaluating the matching error. Not only is it illumination-invariant, it is also invariant under changes in geometry and scale due to its locality and allows to detect mismatches in homogeneous regions.

Except for the last choice where using the compensated image data is essential to make the confidence measure illumination-invariant, all other choices can be applied on both the compensated and the uncompensated image data. While the former provides the confidence measure with a higher degree of illumination-invariance, the latter is free from errors that are due to mis-estimated illumination changes. Thus, for each of these measures both variants are worth to be considered. Later on, we will evaluate each of them and investigate their advantages and disadvantages in different contexts. In any case, please note that, by construction, smaller values denote a higher reliability.

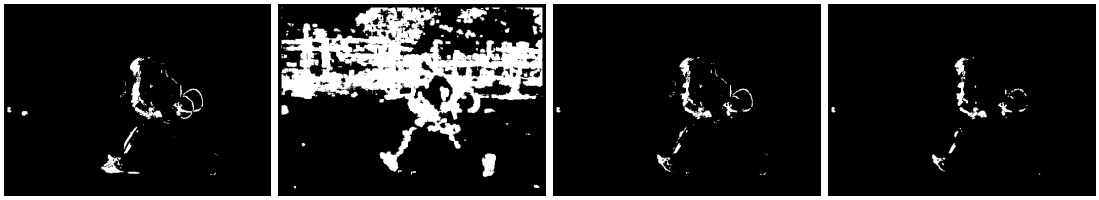


Figure 6.7: Determining the regions of interest. **From left to right:** (a) Confidence mask χ_{reg}. (b) Structuredness mask χ_{struct}. (c) Intermediate mask χ before morphological opening. (d) Final mask χ after opening.

6.3.3 Determining Candidate Regions

Following our method from Chapter 3 (Sect. 3.8), where we selected regions of interest for the integration of feature matches, we will now present a similar scheme for the integration of additional flow candidates. In contrast, however, we allow for different choices of confidence measures as stated above and do not necessarily rely on the original data energy. Nevertheless, for consistency reasons we refer to the corresponding criterion as *energy criterion*, since the confidences are based on data energies. Furthermore, we make also use of a similar local structuredness criterion as in earlier sections, such that the regions of interest are determined by both an energy and a structuredness criterion, i.e. $\chi(\mathbf{x}) = \chi_{\text{reg}}(\mathbf{x}) \cdot \chi_{\text{struct}}(\mathbf{x})$. An overview of this process is given in Fig. 6.7.

Energy Criterion

In order to determine the registration error mask χ_{reg}, we apply a double thresholding strategy on the results of the chosen confidence measure, similarly to Chapter 3 (Sect. 3.8.4). Our goal is to isolate regions where the baseline flow is not appropriate, i.e. where the confidence measure indicates a pronounced registration error. In Chapter 3 (Sect. 3.8.4), we built upon quantile-based thresholds that to some degree gave us control over the relative number of feature matches that are to be integrated. This makes sense, since the feature matches that we used, are known to be unreliable in many cases. Hence, our goal was to keep the number of integrated matches as low as possible by choosing high quantiles.

Thresholds. Now, the context changes and we are about to define regions that are appropriate for the integration of more reliable matches. To this end, we consider thresholds based on multiples of average-values which give more control on the respective properties that are thresholded than on the number of matches that are integrated, i.e. we can define to which degree some energy must exceed the reference value to be regarded as a region of interest. For the energy criterion, we determined the thresholds $\theta_{\text{strict}} = 5 \cdot \bar{\rho}(\mathbf{w}_{\text{base}})$ and $\theta_{\text{soft}} = 1 \cdot \bar{\rho}(\mathbf{w}_{\text{base}})$ to be appropriate, where $\bar{\rho}(\mathbf{w}_{\text{base}})$ denotes the average confidence of the entire flow field. The result is shown in Fig. 6.7 (a).

Structuredness Criterion

In order to compute the structuredness mask, we follow [27] and apply a single thresholding scheme on the smaller eigenvalue λ_2 of the structure tensor [64] which is integrated over a 7×7 region following [27]. Our goal is to find regions that have sufficient structure for the estimation of flow candidates, such that we do not get trapped in the aperture problem when we apply our de-regularization approach. In this context, we chose $\theta_{\text{struct}} = 0.5 \cdot \bar{\lambda}_2$, where $\bar{\lambda}_2$ denotes the average smaller eigenvalue on the entire image domain. The result of this thresholding step is depicted in Fig. 6.7 (b).

Final Mask

As can be seen from Fig. 6.7 (c), the final mask does not contain most of the irrelevant regions that are present in the single masks of the involved criteria. On the one hand, the structuredness mask contains a lot of regions that are structured but already well matched. On the other hand, the energy mask clearly depicts effects that origin from the symmetry of the data term involving source and target frames, as the small objects that are not matched properly appear at different positions in both frames and hence both positions lead to high energies. This can particularly be seen at hand of the tennis racket and the right foot of the player, since both at least partially appear twice in this mask. The combination of both criteria reduces the amount of irrelevant regions a lot. Finally, we eliminate isolated pixels in χ by applying a morphological opening with a squared structuring element of size 3×3 (see Fig. 6.7 (d)).

6.3.4 Generating Flow Proposals

So far, we have computed well-localized regions of interest, which determine where flow proposals are to be integrated in the final estimation. Nevertheless, the computation of these proposals is not restricted to these regions but derived from dense variational methods. Compared to sparse descriptor matching, this allows to incorporate a global communication in terms of regularization into the estimation and thus to compute reliable matches even in homogeneous areas. This in turn improves the robustness of such matches. Indeed, as observed in Chapter 3 and in [44], outliers are the main source of problems when integrating such matches. Moreover, small objects are more likely to prevail in the final estimation, if they are covered by a sufficient amount of matches.

Problems of Coarse-to-fine Schemes. Before we detail on the generation of our flow proposals, let us briefly recapitulate why common coarse-to-fine variational methods have problems with relative large displacements. First of all, small objects smear with their background on that coarse-to-fine level that is appropriate to estimate their displacement. Secondly, large displacements induce large motion discontinuities, *severely* violating the smoothness assumption. The corresponding penalizer functions

are typically not sufficiently robust to handle such jumps – either for convexity reasons or to avoid staircasing. Thus, even if there is enough information remaining to estimate a large displacement, it is typically cheaper to violate the data constancy assumption for a small object than to severely violate the smoothness assumption.

Applying a De-Regularization Strategy. In the following, we address both issues by again relying on a de-regularization strategy, similar to the one in Chapter 4 (Sect. 4.4). In this context, we consider two variational methods to generate the proposals – each equipped with a different constancy assumption.

Large Displacements via De-Regularization

Similar to Chapter 4 (Sect. 4.4), we consider multiple instances of conventional energy functionals consisting of a data term and a smoothness term each. They implement the general concept of de-regularization by applying a separate smoothness weight for each instance. In contrast to Chapter 4 (Sect. 4.4), where we used instances of a baseline model and combined them into a single joint variational model (using an integrated fusion model), we will now consider independent models for the generation of flow proposals. To this end, let us consider a family of energy functionals of the form

$$E_{\text{cand}}(\mathbf{w}_{\text{P}k}) \quad = \quad \int_{\Omega} D_{\text{cand}}(\mathbf{w}_{\text{P}k}) + \alpha_k S_{\text{cand}}(\mathbf{w}_{\text{P}k}) \, d\tilde{\mathbf{x}} \,, \tag{6.14}$$

with successively decreasing smoothness weights α_k with $k = 1, ..., N_{\text{cand}}$. Evidently, decreasing the amount of regularization eases the estimation of large displacements as violating the smoothness term has less impact. Although this strategy deteriorates the average performance, since flow fields typically become very noisy, it significantly helps to improve the performance at locations with large displacements; see Fig. 6.8. Hence, by computing one flow field $\mathbf{w}_{\text{P}k}$ for each of the regularization parameters α_k, we are able to generate a set of flow proposals $\mathbf{w}_{\text{P}1}(\mathbf{x}), ..., \mathbf{w}_{\text{P}N_{\text{cand}}}(\mathbf{x})$ per pixel from which we determine the best candidate $\mathbf{w}_{\text{P}}$ in a final selection step. While we use the same smoothness term S_{cand} as in our baseline method, we consider the following two constancy assumptions for the data term D_{cand} when generating our proposals.

Brightness Constancy Assumption (BCA). By relying on the compensated first frame, we can use pure brightness constancy in the data term:

$$D_{\text{cand, 1}}(\mathbf{w}) \quad = \quad \Psi_{\text{D}}\left(\sum_{c=1}^{N_{\text{c}}} (I^c(\mathbf{x}+\mathbf{w}) - I^c_{\text{comp}}(\mathbf{x}))^2 \right) . \tag{6.15}$$

Apart from being robust against illumination changes, it is also rotation and scale invariant due to its locality. Realizing these properties in a feature descriptor requires much effort. Also another property of the BCA is beneficial. In contrast to gradient-like

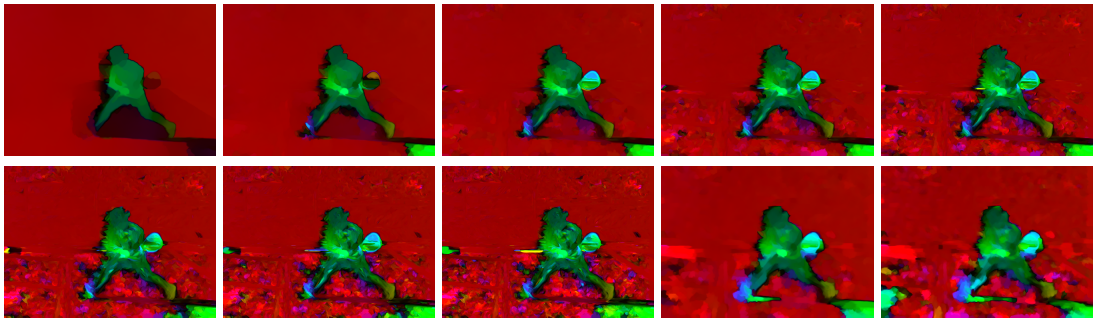


Figure 6.8: Effects of de-regularization: a higher ability to capture large displacements comes along with a higher level of noise. **First row:** BCA data term, candidate sets one to five. **Second row:** BCA data term, candidate sets six to eight, and GBCA data term, candidate sets one and two.

data terms, a potential violation is not only expensive at edges, but also in homogeneous parts of small fast-moving objects. This resilience particularly complements the effect of a decreasing regularization.

Geometric Blur Constancy Assumption (GBCA). At certain locations, however, it may be more appropriate to estimate the motion using feature descriptors – in particular if the local information is not sufficient at the respective coarse-to-fine level. In this context, [120] proposed to expand the local intensity into separate intensity channels, such that each channel is resampled separately and objects at different intensities are not smeared. While this representation is more robust to resampling, it does not add any descriptiveness. In contrast, enhanced descriptiveness can be obtained by regarding neighborhood information such as in the SIFT [80], the HOG [36] and the GB descriptor [14]. Although the latter has a higher descriptiveness compared to HOG as observed in [27], so far only the HOG/SIFT descriptor has been used as data term in variational methods; see [79, 109]. According to [27], the main problem of the GB descriptor is its tendency to produce more false positives in sparse matching. This, however, is not an issue when using it in a constancy assumption of a variational method. Consequently, we propose to use GB descriptors in a feature-based data term:

$$D_{\text{cand, 2}}(\mathbf{w}) \quad = \quad \Psi_{\text{D}}\left(|GB(\mathbf{x}+\mathbf{w}) - GB_{\text{comp}}(\mathbf{x})|^2\right), \tag{6.16}$$

where GB and GB_{comp} denote stacks of image frames that are obtained by applying Geometric Blur feature transforms on the original stack I and on the illumination-compensated variant I_{comp}.

Hence, the GBCA assumes constancy on a feature (i) whose components are resampled separately following the spirit of [120], (ii) which improves descriptiveness over [36], and (iii) whose tendency of false positives as stated in [27] is overcome due to the inherent regularization of the underlying variational model.

Algorithm 6.1 Pseudocode for the fusion of flow candidates at some location $\mathbf{x}$.

initialize state $\{\mathbf{w}_P, \rho^{5\times5}(\mathbf{w}_P)\} \leftarrow \{\mathbf{w}_{base}, \rho^{5\times5}(\mathbf{w}_{base})\}$
for all candidate approaches j **do**
  for all smoothness weights l **do**
    if $\rho^{5\times5}(\mathbf{w}_{Pj,l}) \leq \rho^{5\times5}(\mathbf{w}_P)$ //confidence improves (smaller value!)
      and $\rho^{5\times5}(\mathbf{w}_{base}) > \theta_{base}$ //baseline worse than some lower bound
      and $\lambda_2 > \theta_{struct}$ //enough structuredness
    then
      $\{\mathbf{w}_P, \rho^{5\times5}(\mathbf{w}_P)\} \leftarrow \{\mathbf{w}_{Pj,l}, \rho^{5\times5}(\mathbf{w}_{Pj,l})\}$
    end if
  end for
end for

6.3.5 Candidate Selection

In the previous section, we have generated a set of candidates that can improve the flow estimation. Let us denote them by $\mathbf{w}_{Pi,k}$ where $i \in \{1,2\}$ refers to the model with data term $D_{\text{cand, i}}$ and k relates to the smoothness weight α_k. For each pixel within the candidate regions, it remains now to select the best candidate $\mathbf{w}_P$ out of this set. To this end, we make once again use of the same confidence measure as for the determination of the regions of interest.

Discrete Fusion. During the fusion procedure, we locally keep a current candidate state $\{\mathbf{w}_P, \rho(\mathbf{w}_P)\}$ and successively update it when fusing candidates. It is initialized with the corresponding values for the baseline flow. The updating procedure follows rules that implement the following assumptions: (i) A flow candidate $\mathbf{w}_{Pj,l}$ shall have a better confidence than the currently chosen candidate $\mathbf{w}_P$. In any case, its confidence must improve over the baseline confidence since false positives become more probable. (ii) The baseline confidence must be worse than some lower bound in order to account for noise and to make a significant improvement possible. (iii) A certain level of structuredness (indicated by λ_2) is necessary in order to avoid getting trapped in the aperture problem. An algorithm that describes the process for a location $\mathbf{x}$ is given in Alg. 6.1. Please note that we average the energy within a 5×5 window, similar to [140].

6.3.6 Final Estimation

Finally, the locally best proposal $\mathbf{w}_P$ is integrated into the extended energy functional and a final flow field $\mathbf{w}_{final}$ is estimated. The mask χ guarantees that candidate proposals are only integrated at regions of interest. In order to avoid that single bad proposals deteriorate the result, we re-compute the mask χ by excluding locations where the confidence of the final flow in a local neighborhood of size 5×5 became worse than

the one of the baseline flow or where it exceeds a post-check threshold $\theta_{\text{post-check}} = 4 \cdot \bar{\rho}(\mathbf{w}_{\text{final}})$ based on the average energy of the final flow:

$$\begin{aligned} \chi_{\text{final}}(\mathbf{x}) &= \chi(\mathbf{x}) \\ &\quad \cdot \delta\left[\rho^{5\times5}(\mathbf{w}_{\text{final}}) \leq \rho^{5\times5}(\mathbf{w}_{\text{base}})\right] \\ &\quad \cdot \delta\left[\rho^{5\times5}(\mathbf{w}_{\text{final}}) \leq \theta_{\text{post-check}}\right] . \end{aligned} \tag{6.17}$$

Using χ_{final} we then recompute the final flow field $\mathbf{w}_{\text{final}}$.

6.3.7 Final Variational Model

In this chapter, we will again make use of the method of Zimmer *et al.* [165, 164] as presented in Chapter 2 (Sect. 2.8). Similarly to Chapter 3 (Sect. 3.7), we extend it by a similarity term as proposed by Brox and Malik [27] (see also Chapter 2, Sect. 2.10.1):

$$E(\mathbf{w}) = E_{\text{base}}(\mathbf{w}) + E_{\text{sim}}(\mathbf{w}, \mathbf{w}_{\text{P}}) \tag{6.18}$$

where the additional similarity term E_{sim} is given by

$$E_{\text{sim}}(\mathbf{w}, \mathbf{w}_{\text{P}}) = \beta \int_{\Omega} \chi_{\text{P}}(\mathbf{x}) \Psi_{\text{P}}\left(|\mathbf{w} - \mathbf{w}_{\text{P}}|^2\right) d\tilde{\mathbf{x}} , \tag{6.19}$$

where β is a balancing weight, $\chi_{\text{P}}(\mathbf{x})$ is a binary activation flag and Ψ_{P} is the Charbonnier penalizer [33]. In contrast to Chapter 3 (Sect. 3.7), a local confidence weight is not required and β can be chosen quite large, since, in general, our candidates hardly contain outliers.

6.3.8 Differences to Our Paper

Compared to our paper [128] there are some differences that improve different aspects regarding the consistency and that allow for a deeper analysis of the components: (i) We consider edges from both the reference frame and the successive frame in the energy inpainting step (see Sect. 6.2.4), since edges of mismatched objects appear symmetrically in both frames. (ii) In the estimation of the illumination changes (see Sect. 6.2.2), we now apply complementary instead of homogeneous regularization to be more consistent with our method *BTFIllum* (see Chapter 5, Sect. 5.6.2). (iii) We consider different confidence measures (see Sect. 6.3.2) which allows for a deeper analysis of the components. (iv) Our post-check (see Sect. 6.3.6) now additionally considers the local level of the final data energy in comparison to a threshold that is based on that energy itself (independently from the baseline energy), which improves the behavior at occlusions which also tend to produce high data energies in general.

6.3.9 Aspects of the Minimization

Similar to the methods before, we basically minimize the nonconvex and nonlinear functional using concepts from Chapter 2 which includes the coarse-to-fine warping strategy as described in Sect. 2.6.3 along with the lagged nonlinearity method as described in Sect. 2.3.1. After discretization, the resulting sequence of linear equation systems is solved with a successive overrelaxation scheme (SOR) as hinted in Sect. 2.3 using a multicolor variant [1] that can be parallelized and SIMD vectorized. Moreover, we apply constraint normalization as described in Sect. 2.8.1.

6.3.10 Evaluation

In order to investigate the performance of our novel strategy for generating regularized matches in the presence of illumination changes (*ICALD*), we conducted several experiments, both on common benchmarks as well as on common large displacement sequences. We furthermore tested our method on modified versions of the latter sequences that additionally include illumination changes to demonstrate the robustness of our method against such changes.

Parameters

In order to determine the smoothness weights α_k of the candidate models, we first define basic smoothness weights $\alpha_{\text{cand,BCA}}$ and $\alpha_{\text{cand,GBCA}}$ for both types of candidate models and compute the smoothness weights of candidate number k as $\alpha_{k,\text{(G)BCA}} = \frac{\alpha_{\text{cand,(G)BCA}}}{2.5 \cdot k}$. In each of the following experiments on training data from benchmarks, we optimized the following parameters: the smoothness parameter α and the weight γ of the GCA among the set of the baseline parameters as well as the basic smoothness weights $\alpha_{\text{cand,BCA}}$ and $\alpha_{\text{cand,GBCA}}$ for the generation of the proposals. Details on parameters and their retrieval can be found in Appendix A.8.

Analysis of Components

In order to show the impact of the different components of our method, we start by analyzing them individually in terms of quantitative as well as qualitative experiments.

Confidence Measures. In our first experiment, we evaluate the performance of different confidence measures quantitatively on the subset of the clean pass of the Sintel training data set. In this context, we consider the following measures: the weight-normalized baseline energy (denoted as *norm. orig.*), which covers the BCA and the partially illumination-invariant gradient constancy assumption (GCA), the pure BCA, the pure GCA and the gradient magnitude constancy assumption (GMCA), which in

Table 6.3: Selected results for different confidence measures for *Sintel (sub.)*. Underlined fonts indicate the best results for each value of $N_{\text{cand},BCA}$ while a bold font indicates the overall best result.

	$N_{\text{cand},BCA}$		
Confidence Measure	4	5	6
norm. orig.	5.553	5.541	5.546
norm. orig. (comp.)	5.565	5.521	5.529
GCA	5.552	5.565	5.544
GCA (comp.)	<u>5.493</u>	**<u>5.487</u>**	<u>5.512</u>
GMCA	5.636	5.672	5.659
GMCA (comp.)	5.598	5.621	5.602
BCA	5.812	5.816	5.814
BCA (comp.)	5.812	5.796	5.788

contrast to the GCA is invariant under rotations. We applied these measures both on the original data and on illumination-compensated image data (denoted with *(comp.)*). Please note that the BCA on the original data is not invariant under illumination changes. For the comparison, we consider different sets of candidates from the BCA model and evaluate the respective performances for each choice of confidence measure on this data set. In Tab. 6.3, we display the results of three different numbers of candidates per pixel (covering four to six differently smoothed matches per pixel) of the BCA-based matches for each of the presented confidence measures.

We observe that the results of the invariance-based measures are rather close in general while the versions based on illumination-compensated image data are consistently superior to those on the original image data. Moreover, the BCA shows an inferior performance compared to the invariance-based measures. Hence, we will build on the *GCA (comp.)* as our confidence measure in the following *quantitative* experiments.

In our second experiment, we compare the best invariance-based confidence measure *GCA (comp.)* with the information-preserving *BCA (comp.)* qualitatively at hand of different large displacement sequences. To this end, we demonstrate results using both the visualization of the flow field as well as the motion-compensated second frame, which illustrates registration errors, in Fig. 6.9. For the Tennis sequence, which contains large displacements of very small and structured objects, the results for both confidence measures do not differ substantially. At hand of the Beanbags sequence, however, we observe a weakness of invariance-based confidence measures: they are not very discriminative at large homogeneous areas like the one of the right beanbag. In a good result, we expect to see the following: (i) The motion of the right beanbag must be visualized in yellow and (ii) a beanbag must be visible in the motion-compensated

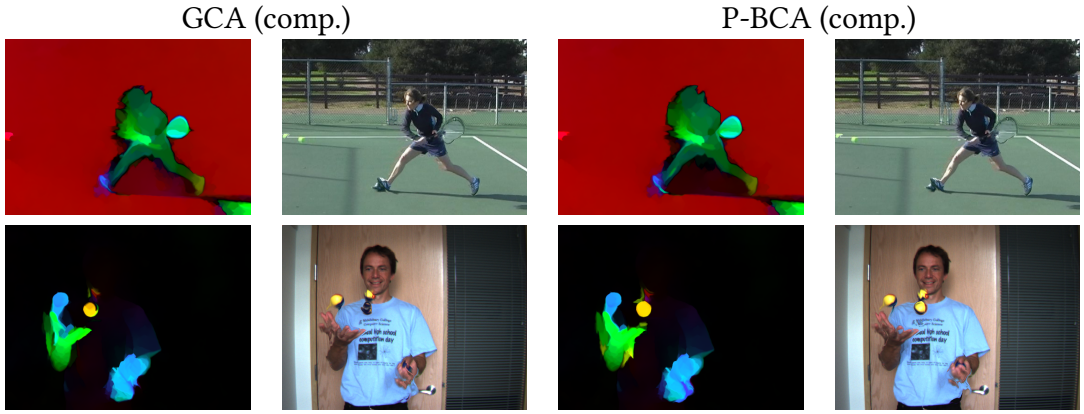


Figure 6.9: Results for two large displacement sequences using different confidence measures. **From left to right:** Flow using the *GCA (comp.)* measure, motion-compensated second frame, flow using the using the *BCA (comp.)* measure, motion-compensated second frame. **From top to bottom:** Tennis sequence (Frame 496) [27], Beanbags sequence [9].

second frame at the corresponding position. Since there is no beanbag at this position using the *GCA (comp.)* measure, we see that this measure is not able to provide enough correct matches for the estimation of the beanbag, since it cannot judge the quality of different matches well at the homogeneous regions of the ball. In contrast, the usage of the *BCA (comp.)* measure clearly guides the estimation into the correct direction, since there is a beanbag at the corresponding position in the motion-compensated image.

Candidate Models and De-Regularization. In our third experiment, we analyze the effect of different numbers of candidates on the overall result. To this end, we consider both types of candidate models (with BCA and with GBCA data terms), where we computed up to nine candidates per pixel using the BCA model and up to five candidates per pixel using the GBCA model. The results of these combinations are given as a matrix in Tab. 6.4, where each row stands for a particular number of candidates for the BCA model and each column stands for a particular amount of candidates for the GBCA model. In this context, we make use of the *GCA (comp.)* measure.

From all these combinations of different candidates, we can see that both sets of proposals have their share on improving the results over the baseline. While it is possible to significantly improve the result with each set independently resulting in improvements of 0.775px (-12.2%) for the GBCA proposals (AEE: 5.600) and of 0.923px (-14.5%) for the BCA proposals (AEE: 5.452), respectively, the combination of both gains another 0.080px (AEE: 5.372). Furthermore, this minimal AEE is embedded into a broad valley of comparably low errors. Even with only three candidates of the BCA proposals per pixel in combination with only two candidates of GBCA proposals, an AEE of 5.406

Table 6.4: Results for de-regularization with BCA, GBCA and the combination on *Sintel (sub.)* (AEE). Underlined fonts indicate the best results using only BCA and only GBCA candidates, respectively, while a bold font indicates the overall best result.

	$N_{\text{cand},GBCA}$					
$N_{\text{cand},BCA}$	0	1	2	3	4	5
0	6.375	<u>5.600</u>	5.711	5.703	5.662	5.712
1	5.913	5.634	5.601	5.687	5.605	5.677
2	5.465	5.471	5.563	5.525	5.553	5.544
3	5.542	5.447	5.406	5.507	5.504	5.503
4	5.493	5.445	5.428	5.512	5.494	5.532
5	5.487	5.458	5.405	5.493	5.499	5.516
6	5.512	5.497	**5.372**	5.439	5.499	5.497
7	5.468	5.484	5.391	5.480	5.478	5.482
8	<u>5.452</u>	5.474	5.378	5.447	5.480	5.466
9	5.461	5.445	5.441	5.469	5.494	5.488

is achieved, which is hardly worse. Also the combinations (BCA: 8/GBCA: 2) with an AEE of 5.378 and (BCA: 7/GBCA: 2) with an AEE of 5.391 yield results similar to the top result. This shows that our method is quite robust in terms of the numbers of candidates that are used as long as each type of proposals is represented.

Since the information-preserving *BCA (comp.)* confidence measure has been superior in the qualitative second experiment, let us investigate its performance using all the combinations of matches as given in this experiment. Indeed, the results (without table) between both confidence measures are much closer than before: The best obtained result for the *BCA (comp.)* measure is an AEE of 5.425 (BCA: 7/GBCA: 3). Even using a comparable setting of six candidates per pixel for the BCA-matches and one candidate for the GBCA-matches leads to an AEE of 5.443 which is hardly worse compared to the best results of both measures.

In our fourth experiment, we investigate the influence of the proposals from each of the candidate models on the final result at hand of the Tennis sequence. Fig. 6.10 shows results obtained using different sets of candidate proposals. On the one hand, one can see that the Geometric Blur constancy assumption (GBCA) is able to capture translational and slight rotational motion, i.e. the tennis ball, the tennis racket and the arm. On the other hand, one can observe that the BCA data term is able to capture the strong rotational motion of the right foot more accurately, which complements the benefits of the GB constancy. Thus, not surprisingly, combining both proposal sets also yields the best results here.

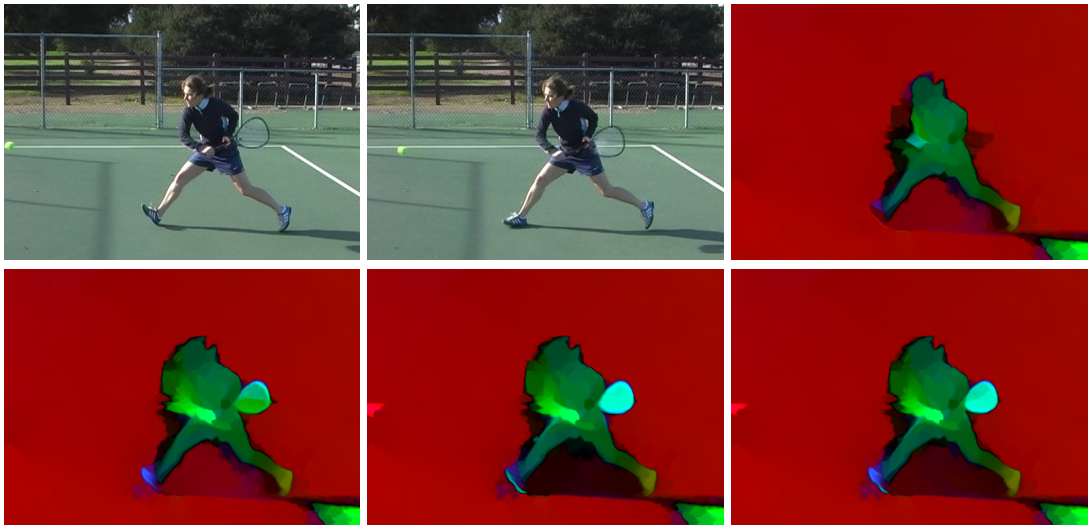


Figure 6.10: Influence of the candidate flows. **First row, from left to right:** First frame, second frame, baseline result. **Second row, from left to right:** Our result using proposals from only the BCA data term, only the GBCA data term, both data terms.

Illumination Compensation. In our fifth experiment, we analyze the importance of the illumination compensation on the overall result. In this context, we compare a variant *without* illumination compensation, a variant where the illumination changes have been computed on *all* vectors (including poor matches) of the baseline flow, and several variants that *excluded poor matches* from the baseline flow before computing the illumination changes.

The corresponding results are listed in Tab. 6.5. On the one hand, one can see that omitting the illumination compensation (AEE: 5.533) deteriorates the accuracy of the results by 0.161px (+3%). This demonstrates that using illumination compensation is indeed useful when selecting and integrating feature matches. On the other hand, it becomes evident that simply using the entire baseline flow for estimating the illumination changes is also not a good idea. In fact, in the this case, the result deteriorates significantly (+422%) with an AEE of 28.019. In general, one can observe that, when estimating the illumination changes, a moderate amount of regularization is beneficial ($\alpha_{\text{ill}} = 4000$). While a too small value for α_{ill} (Tab. 6.5, Rows 1 and 2) interprets all registration errors as local illumination changes, a too large value (Tab. 6.5, Rows 4 and 5) only allows the estimation of global illumination changes. Please note that the semi-local nature of the illumination changes is also reflected in the choice of the inpainting weight when excluding poor matches before the illumination estimation ($\alpha_{\text{inp}} = 3000$). Finally, when replacing the affine parametrization with its normalized

Table 6.5: Impact of the illumination compensation on *Sintel (sub.)*.

	α_{ill}	AEE
Illumination compensation excluding poor matches	400	5.855
	1600	5.536
	4000	**5.372**
	16000	5.486
	40000	5.575
Illumination compensation using all flow vectors	4000	28.019
No illumination compensation	–	5.533

	α_{ill}	AEE
Illumination compensation excluding poor matches using a normalized basis	239	5.570

counterpart – which turned out to be beneficial in a joint model for the estimation of optical flow and illumination changes (see Chapter 5, Sect. 5.10.4) – results slightly deteriorate (AEE: 5.570). Hence, for the separate estimation of illumination changes on a given optical flow field, the original affine parametrization is the better choice.

In our sixth experiment, let us have a look at the Tennis sequence [27] where we added artificial global illumination changes containing both additive as well as multiplicative changes. In Fig. 6.11, we provide the results of our method in two versions: first the results of our complete method and second results obtained when deactivating the illumination compensation. Obviously, the illumination compensation is necessary to obtain meaningful results that contain the large displacements. While the flow vectors of the Tennis ball spread widely, the motions of the right arm and of the tennis racket are not covered correctly. A detailed depiction of the respective candidate flows is given in Fig. 6.12. As one can see, the candidate flow fields using the BCA data term are totally useless while the candidate flows using the GBCA term are still quite meaningful. Nevertheless, the confidence measure using the original, uncompensated first frame (which comes down to an uncompensated BCA confidence measure) is not able to determine the quality of the flow vectors everywhere. Thus, one cannot rely on the selection strategy without illumination compensation. In this context, one might rely on invariance-based confidence measures to circumvent that problem. But as we have seen before, they have problems in homogeneous areas.

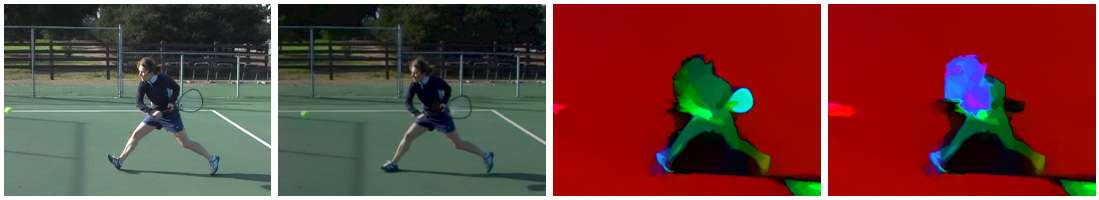


Figure 6.11: Tennis sequence with artificial illumination changes. **From left to right:** First frame, second frame, our method, our method w/o illumination compensation.

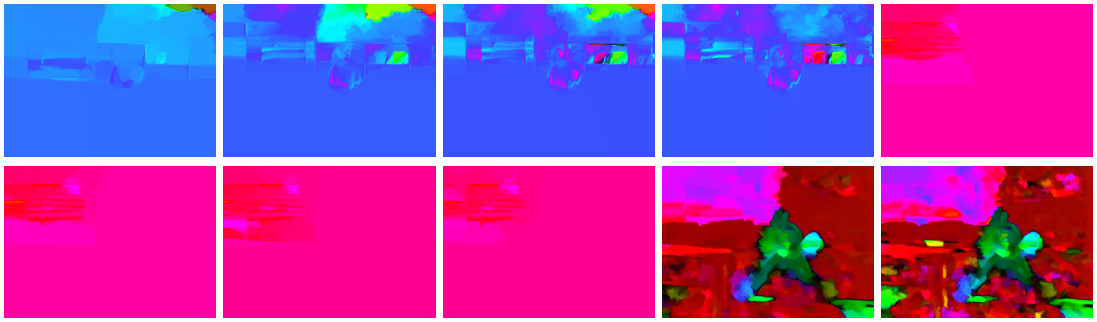


Figure 6.12: Flow candidates for the Tennis sequence with artificial illumination changes when illumination compensation is deactivated. **First row:** BCA data term, candidate sets one to five. **Second row:** BCA data term, candidate sets six to eight, and GBCA data term, candidate sets one and two.

Candidate Regions. In our seventh experiment, we investigate the importance of selecting candidate regions on the final result. To this end, we evaluate a variant of our method, where our selection strategy determines a flow proposal for each pixel of the image instead of restricting the selection only to the candidate regions. In this case, we achieve an overall AEE of 5.983 which deteriorates results by 0.611 (+11.4%). This demonstrates that our strategy to determine candidate regions is beneficial when integrating flow proposals.

Post-Check. In our eighth experiment, we evaluate the impact of the post-check on the final flow field, i.e. the check that removes candidates that had a negative impact on the result. By deactivating it, an AEE of 5.586 is achieved. Hence, it improves results by 0.214 (-3.8%) which demonstrates that it has a considerable influence.

More Large Displacement Sequences

In our ninth experiment, we investigate the ability of our pipeline approach to handle large displacements on different image sequences from the important literature on large

Figure 6.13: Results for large displacement sequences. **From Top to Bottom:** Tennis (Frame 496) [27], Baseball [160], Beanbags [9]. **From Left to Right:** First frame, second frame, baseline, our method.

displacement optical flow [27, 160]. The results are depicted in Fig. 6.13 where they are compared to the results of the baseline method. As one can clearly see when comparing the corresponding results, our strategy allows us to reliably capture all the apparent large displacements at the limbs, the racket, the tennis ball (in the Tennis sequence), the bat (in the Baseball sequence) and the beanbags (in the Beanbags sequence). This ability is also reflected in terms of a decrease of the average photometrically-compensated registration errors for these sequences compared to the respective baseline flow fields (see Tab. 6.6).

In Fig. 6.14, we depict the results of further image sequences that contain large displacements together with the results of the baseline method. The advances are also clearly visible for these sequences: the motion of the right foot (in the Football sequence), the motion of the left arm (in the Tennis sequence, Frame 502) and the motions of the right arm and the tennis ball (in the Tennis sequence, Frame 577) are covered correctly. This demonstrates that our method is quite consistent in estimating large displacements.

Moreover, we evaluate the ability to handle large displacements in the context of illumination changes. Hence, we change brightness and contrast settings for the same sequences as shown in Fig. 6.13 and depict the adapted second frames as well as the corresponding results in Fig. 6.15. As one can see, these results are only slightly worse than without illumination changes, but the large motion is still recovered correctly.

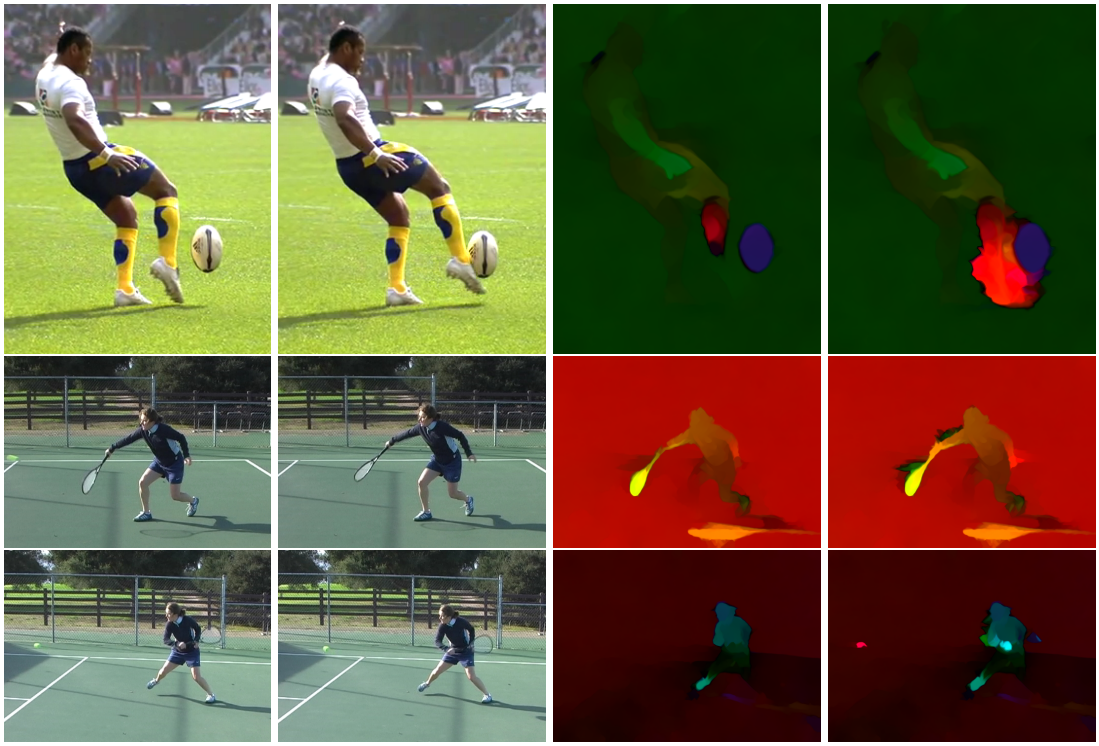


Figure 6.14: Results for further large displacement sequences. **From Top to Bottom:** Football [160], Tennis (Frame 502) [27], Tennis (Frame 577) [27]. **From Left to Right:** First frame, second frame, baseline, our method.

Table 6.6: Registration error for large displacement sequences.

	Tennis	Baseball	Beanbags
Baseline	0.031	0.013	0.050
Our method	**0.009**	**0.007**	**0.019**

MPI Sintel Training Data

In our tenth experiment, we evaluate the performance of our strategy on the clean data set of the MPI Sintel training benchmark data, using both the subset of 69 sequences and the complete data set, by comparing our novel method with the baseline. In this context, we also include a modified baseline without the illumination-invariant GCA constancy assumption in this comparison in order to demonstrate the importance of handling illumination changes of these data in general.

The outcome in Tab. 6.7 shows the superiority of our method compared to its baseline (-15.7% for the subset, -11.4% for the complete data set). Moreover, the results for a

Figure 6.15: Results for large displacement sequences with illumination changes. **From Top to Bottom:** Tennis [27], Baseball [160], Beanbags [9]. **From Left to Right:** first frame, second frame with illumination changes, our method.

Table 6.7: Overall results on MPI Sintel training data (AEE). This comprises the results for the subset that we have chosen for parameter optimization and the complete data set.

Data set	Baseline		Our method
	(only BCA)	(BCA + GCA)	(full)
Subset (clean)	7.091	6.375	**5.372**
All (clean)	4.896	4.084	**3.617**

baseline with pure brightness constancy are even clearly inferior to the ones of the full baseline (+11.2% / +19.9%). This shows that illumination changes are quite present in the data and require proper handling.

Figure 6.16: Exemplary results from the MPI Sintel final evaluation data [31]. **From top to bottom:** First frame, baseline, our result, ground truth. Please note that these results are obtained from the public webpage and thus use the corresponding color code that is different from our usual color code.

MPI Sintel Evaluation Data

In our eleventh experiment, we compare our approach to similar methods from the literature. To this end, we submitted our results to the MPI Sintel benchmark. Fig. 6.16 shows exemplary flow fields of that data set for our method and its baseline. Apparently, the results of our method are less noisy without losing details.

As one can see from Tab. 6.9, on the clean data set, our method shows a performance which is comparable to similar methods from the literature, which however comprise external matches. On the final data set (see Tab. 6.8) it clearly outperforms comparable approaches such as WLIF, MDP-Flow or LDOF. In this context, it is important to recall that we purely consider matches from dense variational methods in contrast to using sparse descriptor matches. This demonstrates that flow proposals from dense variational methods can be a serious alternative to sparse descriptor matches and matches from other large displacement methods like [34]. Moreover, our method shows results that are consistently superior to our *ContFusion-Flow* from Chapter 4 (Sect. 4). To the best of our knowledge, these are the leading variational methods that do not make use of any external, non-variational methods for the generation of candidate matches.

Table 6.8: Ranking on the MPI Sintel final evaluation data.

Method	Feature Matches	AEE
Deep+R [44]	sparse, regularized	6.769
Deep Flow [154]	sparse	7.212
Our Method	*dense, regularized*	*7.640*
ContFusion (Chapter 4)	none	7.857
WLIF [140]	dense, regularized	8.049
Baseline	none	8.065
MDP-Flow2 [160]	sparse	8.445
LDOF [27]	sparse	9.116

Table 6.9: Ranking on the MPI Sintel clean evaluation data.

Method	Feature Matches	AEE
Deep+R [44]	sparse, regularized	5.041
Deep Flow [154]	sparse	5.377
WLIF [140]	dense, regularized	5.734
MDP-Flow2 [160]	sparse	5.837
Our Method	*dense, regularized*	*5.851*
Baseline	none	6.171
ContFusion (Chapter 4)	none	6.263
LDOF [27]	sparse	7.563

6.3.11 Major Benchmarks

In our final experiments, we evaluate our method on all major benchmarks. Similar to previous chapters, we use first-order regularization for the Middlebury and MPI Sintel benchmarks and second-order regularization for the KITTI benchmarks. For the latter cases, we also have the choice of using first- or second-order regularization for the candidate models. First-order regularization typically leads to sufficiently accurate results in structured areas where there is an actual correspondence within the image plane (i.e. the displacement does not target outside the image) even for divergent motions as present in the KITTI benchmarks. Since our candidates origin in such areas, i.e. they represent actual correspondences between pixels in both frames, and furthermore guide the estimation only via a *soft* constraint, it is reasonable to consider first-order regularization in the candidate models. Nonetheless, second-order regularization is

Table 6.10: Results of *ICALD-Flow* and its baseline on training data from different benchmarks. The additional *(socr)* indicates second-order regularization of the candidate models.

	Middlebury		Sintel (sub.)	Sintel	KITTI '12	KITTI '15
	(AAE)	(AEE)	(AEE)	(AEE)	(BP3)	(BP3)
Baseline	2.73	0.229	6.375	4.084	10.68%	24.25%
ContFusion	2.72	0.231	5.808	3.974	10.47%	24.18%
ICALD-Flow	**2.65**	**0.219**	**5.372**	**3.617**	**10.38%**	24.00%
ICALD-Flow (socr)	–	–	–	–	10.67%	**23.69%**

more consistent with the baseline regularization strategy in these cases. In Tab. 6.10, we provide the results for all benchmarks, whereby the results of a second-order regularization of the candidates for the KITTI benchmarks can be found in the last row.

Comparison to the Baseline. When comparing *ICALD-Flow* to the baseline method, we achieve a consistent improvement. For the Middlebury benchmark, the AAE improves by 0.08 (2.9%) while the AEE improves by 0.010 (4.4%). For the MPI Sintel benchmark, the errors decrease by 1.003 (15.7%) on the subset as well as by 0.467 (11.4%) on the complete data set. For the KITTI benchmarks, the first-order regularized candidates drop the error on 2012's edition by 0.3 (2.8%) and on 2015's edition by 0.25 (1%). Regarding the second-order regularized candidates – denoted as *ICALD-Flow (socr)* –, the results for *ICALD* and the baseline are similar on KITTI 2012 with a drop by 0.01 (0.09%) while on KITTI 2015 we have a drop of 0.56 (2.3%).

Comparison to ContFusion-Flow. When comparing *ICALD-Flow* to *ContFusion-Flow* (see Chapter 4), we also observe an improvement in almost all cases. For the Middlebury benchmark, the AAE improves by 0.07 (2.6%) while the AEE improves by 0.012 (5.2%). For the MPI Sintel benchmark, the errors decrease by 0.357 (9%) on the subset as well as by 0.436 (7.5%) on the complete data set. For the KITTI benchmarks, the first-order regularized candidates drop the errors on 2012's edition by 0.09 (0.9%) and on 2015's edition by 0.18 (0.7%). Regarding the second-order regularized candidates (socr), there are ambivalent observations: While the results deteriorate on KITTI 2012 by 0.2 (1.9%), the results on KITTI 2015 show an improvement of 0.49 (2%). Hence, we see that adapting the order of regularization may improve results in some cases but leaving a first-order regularization on the candidates provides a lot of helpful matches.

Please note, that for all benchmarks we consistently used six candidates per pixel from the BCA-matches and two candidates from the GBCA-matches and did not optimize these numbers separately per benchmark (in contrast to the results of *ContFusion-Flow*). Furthermore, we even kept the initial smoothness weight $\alpha_{\text{cand,BCA}}$ for the BCA-matches

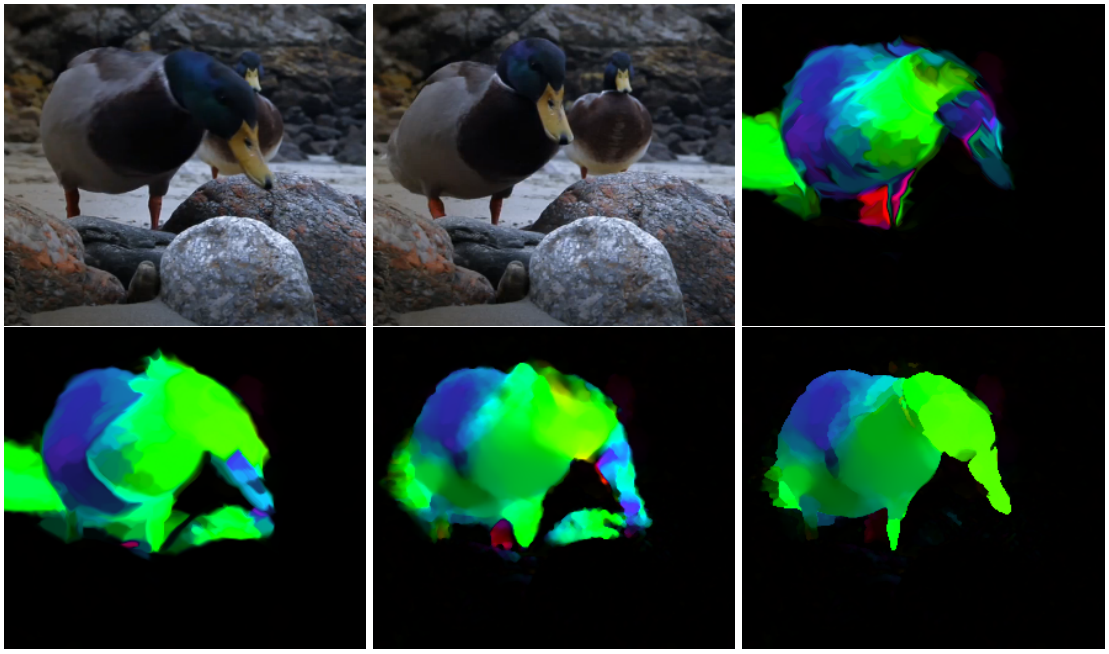


Figure 6.17: Limitations of our method at hand of the Bird sequence [160]. **From left to right, top to bottom:** First frame, second frame, baseline, our method, MDP-Flow without occlusion handling, MDP-Flow with occlusion handling [160].

and the weight β for the similarity term fixed for all settings, the latter one at a very large value (see A.8.2). Even the optimized initial smoothness weights $\alpha_{\text{cand,GBCA}}$ lie in the same order of magnitude for all benchmarks and could likely be fixed as well. This shows that our regularized matches consistently improve results on all benchmarks without adapting the numbers of candidates, changing the order of their regularization or optimizing overly many parameters.

Limitations

Finally, we show an example that illustrates the limitations of our method. To this end, we consider the Bird sequence [160] (see Fig. 6.17) which is particularly challenging, since it contains large and complex motion of the bird's head. While the motion of the body is recovered correctly, the motion of the nib is not recovered well using our approach. In contrast, the MDP-Flow [160] method which is based on feature matching is able to handle this motion better. This, however, is only the case when its particular capabilities to handle occlusions are enabled (Fig. 6.17, bottom right). Otherwise, the results (Fig. 6.17, bottom center) are similar to our method (Fig. 6.17, bottom left).

6.4 Summary

In this chapter, we developed a pipeline of variational methods that is able to handle large displacements using regularized matches even in the context of illumination changes. To this end, we combined the best concepts from the previous chapters which comprise the match generation procedure using variational methods (see Chapter 4), the estimation of illumination changes and their compensation to make this estimation possible in the context of illumination changes (see Chapter 5) and a pipelined approach using an adaptive sparsification strategy to determine promising locations for their integration (see Chapter 3). Simply combining only the first two of these concepts, i.e. directly using the joint model for match generation and fusion from Chapter 4 on illumination compensated image data, did not scale well with increasing numbers of candidates, both in terms of workload and in terms of results.

In contrast, embedding all ideas into a sequential pipeline of variational approaches led to a superior performance. To this end, we first computed an initial flow using the baseline method, which serves as the basis to estimate the illumination changes within the sequence. As an intermediate step, we computed a coarse mask that identifies unreliable regions of that initial flow in order to not deteriorate the estimation of the illumination changes. After having estimated these changes, we compensated the first frame of the image sequence for them. Based on the modified image sequence, we then computed several candidate flow fields using a de-regularization strategy to capture different scales of displacements. In this context, it allowed us to include variational models with selected data terms that are specifically tailored for the generation of matches which can capture particular motion patterns like large translations or fine-grained rotations. Adaptive sparsification and selection strategies extracted a field of small regions with candidate flows that guided the final estimation to achieve accurate results with moderately large displacements.

Our method showed convincing results both in quantitative experiments on benchmarks as well as in qualitative evaluations on large displacement sequences, with and without illumination changes. An in-depth analysis of the involved components demonstrated their individual effectiveness on the overall result.

Chapter

7 Tensors for Point Constraints

This chapter is dedicated to an extended tensor notation for linear and linearized data constraints and other point constraints. It comprises the motion tensors as proposed by [28, 47, 30] and tensors for all presented pointwise constraints in this thesis. We will show how powerful this tensor notation is, since it allows for a widely generalized variational framework that can find minimizers for different models without much implementation effort. In order to extend such a framework to handle another linear(ized) point constraint, it is sufficient to make the corresponding tensor known to the framework. In particular, it is even sufficient to only implement the corresponding constraint vector(s), since the tensor can derived from them automatically.

To make this possible, we provide an overview of the so far presented data constraints and of some other terms from recent literature with their corresponding tensor notation.

7.1 Structure of Linear(ized) Data Terms

We can basically formulate any linear oder linearized data term that consists of one or more data constraints as

$$D(\mathbf{w}) \quad = \quad \sum_i (\mathbf{w}^\top \mathbf{p}_i)^2 \,, \tag{7.1}$$

where $\mathbf{w}$ denotes the unknown flow and $\mathbf{p}_i$ are the generating constraint vectors. Thereby, often a (subquadratic) penalizer function Ψ is applied in order to make the constraint robust against outliers, such that the final constraint is given by

$$D(\mathbf{w}) \quad = \quad \Psi\left(\sum_i (\mathbf{w}^\top \mathbf{p}_i)^2\right) . \tag{7.2}$$

From Chapter 2 (Sect. 2.5.4) we know that such a data term can be rewritten in terms of motion tensor J via

$$\begin{aligned} D(\mathbf{w}) &= \Psi\left(\sum_i (\mathbf{w}^\top \mathbf{p}_i)^2\right) \\ &= \Psi\left(\sum_i (\mathbf{w}^\top \mathbf{p}_i)(\mathbf{w}^\top \mathbf{p}_i)^\top\right) \\ &= \Psi\left(\sum_i \mathbf{w}^\top \underbrace{\mathbf{p}_i \mathbf{p}_i^\top}_{=:J_i} \mathbf{w}\right) \\ &= \Psi\left(\sum_i \mathbf{w}^\top J_i \mathbf{w}\right) \\ &= \Psi\left(\mathbf{w}^\top \left(\sum_i J_i\right) \mathbf{w}\right) \\ &= \Psi\left(\mathbf{w}^\top J \mathbf{w}\right) . \end{aligned} \tag{7.3}$$

In the following, we will derive a broad variety of tensors J from different linear(ized) data terms and other point constraints that can be formulated in terms of such tensors. In this context, we consider a constraint to be (spatially) pointwise, if it does not include any spatial neighborhood information. This guarantees that the constraints keep their pointwise properties after discretization.

We will consider all tensors both in a non-incremental version, which can be used to express linear data terms, and in an incremental version, which can be used in contexts where linearizations are postponed to the numerics (coarse-to-fine warping schemes).

7.2 Organization

First of all, in Sect. 7.3 we will start by recapitulating the motion tensors [30] that describe the linearized versions of the BCA and the GCA, since they are used in our baseline method, the approach of Zimmer *et al.* [165, 164]. These tensors furthermore provide the foundations for the extended motion tensors that in the context of a joint estimation of optical flow and illumination changes (see Chapter 5) define the constraints on both the unknown flow and the unknown illumination coefficients. In Sect. 7.4, we move on to tensors for similarity constraints between a prior and the unknown functions. These tensors are helpful to integrate pre-computed feature matches or candidate flows into the estimation of the flow (see Chapter 3 and Chapter 6). In this context, we do not only derive a tensor that seeks a 1:1 correspondence between the prior and the

unknown functions but also two variants that only aim at steering the unknowns to have a direction similar to the one of the prior. In Sect. 7.5, we derive tensors that are related to similarity tensors but implement a coupling between unknowns instead of a similarity between a prior and an unknown function. Such tensors easily allow for a fusion of candidates within a variational model for the joint estimation and fusion of candidate flows (see Chapter 4). Moreover, they allow for a trajectorial regularization as proposed in the work of Volz *et al.* [148]. In this context, a first-order coupling tensor aims at a 1:1 correspondence between two unknowns by penalizing a first-order derivative in trajectorial direction while a second-order coupling tensor couples three unknowns in order to implement a second-order derivative in trajectorial direction. In the context of trajectorial regularization, we also present tensors for two variants of a directional constraint that restrict the coupling to aiming at a similar direction of the flows along a trajectory as proposed in our paper [88].

7.3 Motion Tensors

Let us start by briefly recapitulating the motion tensors for our baseline method from Chapter 2 (Sect. 2.8.4). These comprise formulations for the linearized versions of the brightness constancy assumption (BCA) and of the gradient constancy assumption (GCA). Both tensors have been proposed in [30].

7.3.1 BCA Motion Tensor

The linearized version of the BCA as already used by [68] reads

$$D_{\mathrm{BCA}}(\mathbf{w}) \quad = \quad \left(I_x u + I_y v + I_t\right)^2 , \tag{7.4}$$

such that the corresponding constraint vector reads $\mathbf{p}_{\mathrm{BCA}} \coloneqq (I_x, I_y, I_t)^\top$. Via the relation $J \coloneqq \mathbf{p}\,\mathbf{p}^\top$ the motion tensor for the BCA is given by

$$J_{\mathrm{BCA}} \quad \coloneqq \quad \mathbf{p}_{\mathrm{BCA}} \mathbf{p}_{\mathrm{BCA}}^\top \quad = \quad \begin{pmatrix} I_x I_x & I_x I_y & I_x I_t \\ I_y I_x & I_y I_y & I_y I_t \\ I_t I_x & I_t I_y & I_t I_t \end{pmatrix} , \tag{7.5}$$

as we have already seen in Chapter 2 (Sect. 2.5.4).

Incremental Formulation

Within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$, the tensor has the same structure. We only need to define $I_x = I(\mathbf{x} + \mathbf{w}^k)_x$, $I_y = I(\mathbf{x} + \mathbf{w}^k)_y$ and $I_t = I(\mathbf{x} + \mathbf{w}^k) - I(\mathbf{x})$.

7.3.2 GCA Motion Tensor

Similarly to the BCA case, we start by stating the linearized version of the GCA as proposed by [26] which reads

$$D_{\mathrm{GCA}}(\mathbf{w}) = \left(I_{xx}u + I_{xy}v + I_{xt}\right)^2 + \left(I_{yx}u + I_{yy}v + I_{yt}\right)^2 . \tag{7.6}$$

Here, we have two constraint vectors, one for the constraint on I_x given by $\mathbf{p}_{\mathrm{GCA},x} = (I_{xx}, I_{xy}, I_{xt})^\top$ and one for the constraint on I_y given by $\mathbf{p}_{\mathrm{GCA},y} = (I_{yx}, I_{yy}, I_{yt})^\top$. Hence, the motion tensor is given as the sum of the motion tensors for the two constraints which reads

$$\begin{aligned} J_{\mathrm{GCA}} &= J_{\mathrm{GCA},x} + J_{\mathrm{GCA},y} \\ &= \mathbf{p}_{\mathrm{GCA},x}\,\mathbf{p}_{\mathrm{GCA},x}^\top + \mathbf{p}_{\mathrm{GCA},y}\,\mathbf{p}_{\mathrm{GCA},y}^\top \\ &= \begin{pmatrix} I_{xx}I_{xx} & I_{xx}I_{xy} & I_{xx}I_{xt} \\ I_{xy}I_{xx} & I_{xy}I_{xy} & I_{xy}I_{xt} \\ I_{xt}I_{xx} & I_{xt}I_{xy} & I_{xt}I_{xt} \end{pmatrix} + \begin{pmatrix} I_{yx}I_{yx} & I_{yx}I_{yy} & I_{yx}I_{yt} \\ I_{yy}I_{yx} & I_{yy}I_{yy} & I_{yy}I_{yt} \\ I_{yt}I_{yx} & I_{yt}I_{yy} & I_{yt}I_{yt} \end{pmatrix} \\ &= \begin{pmatrix} I_{xx}I_{xx} + I_{yx}I_{yx} & I_{xx}I_{xy} + I_{yx}I_{yy} & I_{xx}I_{xt} + I_{yx}I_{yt} \\ I_{xy}I_{xx} + I_{yy}I_{yx} & I_{xy}I_{xy} + I_{yy}I_{yy} & I_{xy}I_{xt} + I_{yy}I_{yt} \\ I_{xt}I_{xx} + I_{yt}I_{yx} & I_{xt}I_{xy} + I_{yt}I_{yy} & I_{xt}I_{xt} + I_{yt}I_{yt} \end{pmatrix}, \end{aligned} \tag{7.7}$$

as we have already seen in Chapter 2 (Sect. 2.6.2).

Incremental Formulation

For the GCA and the BCA, the relations between the incremental version and the non-incremental version of the motion tensor are similar. Here, we need to define $I_x = I(\mathbf{x} + \mathbf{w}^k)_x$ and $I_y = I(\mathbf{x} + \mathbf{w}^k)_y$ as before. The temporal derivatives are given by $I_{xt} = I_{tx}$ and $I_{yt} = I_{ty}$ with $I_t = I(\mathbf{x} + \mathbf{w}^k) - I(\mathbf{x})$.

7.3.3 Tensors with Illumination Compensation

When deriving the extended motion tensors for both data constraints that involve components for illumination compensation, we will for the sake of simplicity resort to the case of grey value images (i.e. $N_c = 1$) and hence define $\mathbf{b}(\mathbf{x}) := \mathbf{b}^0(\mathbf{x})$. The motion tensors for the color case can then be derived analogously.

7.3.4 BCA Motion Tensor with Illumination Compensation

The linearized version of the BCA with illumination compensation is given by

$$D_{\mathrm{BCA}}(\mathbf{w}, \mathbf{b}) = \left(I + I_x u + I_y v + I_t - \bar{\phi}(I) - \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi_j(I)\right)^2 . \tag{7.8}$$

Given a unified solution vector $\mathbf{w}_{\text{uni}} = (u, v, b_1, \dots, b_{N_{\text{cIll}}}, 1)^\top$ that includes all sought functions and allows us to represent this term as $D_{\text{BCA}}(\mathbf{w}, \mathbf{b}) = \left(\mathbf{p}_{\text{BCA,ill}}^\top \mathbf{w}_{\text{uni}}\right)^2$, the corresponding constraint vector is given by

$$\mathbf{p}_{\text{BCA,ill}} = \left(I_x, I_y, -\phi_1, \dots, -\phi_{N_{\text{cIll}}}, I_{\tilde{t}}\right)^\top, \tag{7.9}$$

with $I_{\tilde{t}} = I_t + I - \bar{\phi}(I)$. Hence, the corresponding motion tensor reads

$$\begin{aligned} J_{\text{BCA,ill}} &= \mathbf{p}_{\text{BCA,ill}} \mathbf{p}_{\text{BCA,ill}}^\top \\ &= \begin{pmatrix} I_x^2 & I_x I_y & -I_x\phi_1 & \dots & -I_x\phi_{N_{\text{cIll}}} & I_x I_{\tilde{t}} \\ I_y I_x & I_y^2 & -I_y\phi_1 & \dots & -I_y\phi_{N_{\text{cIll}}} & I_y I_{\tilde{t}} \\ -\phi_1 I_x & -\phi_1 I_y & \phi_1^2 & \dots & \phi_1\phi_{N_{\text{cIll}}} & -\phi_1 I_{\tilde{t}} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -\phi_{N_{\text{cIll}}} I_x & -\phi_{N_{\text{cIll}}} I_y & \phi_{N_{\text{cIll}}}\phi_1 & \dots & \phi_{N_{\text{cIll}}}^2 & -\phi_{N_{\text{cIll}}} I_{\tilde{t}} \\ I_{\tilde{t}} I_x & I_{\tilde{t}} I_y & -I_{\tilde{t}}\phi_1 & \dots & -I_{\tilde{t}}\phi_{N_{\text{cIll}}} & I_{\tilde{t}}^2 \end{pmatrix} \end{aligned} \tag{7.10}$$

Incremental Formulation

For the incremental version, we consider all sought functions in an incremental version, i.e. $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$ and $\mathbf{b}^{k+1} = \mathbf{b}^k + \mathbf{db}^k$, or $\mathbf{w}_{\text{uni}}^{k+1} = \mathbf{w}_{\text{uni}}^k + \mathbf{dw}_{\text{uni}}^k$, respectively. Given a warped image $I_2 = I(\mathbf{x} + \mathbf{w}^k)$ and a non-warped image $I_1 = I(\mathbf{x})$, the corresponding constraint reads

$$D_{\text{BCA}}(\mathbf{dw}^k, \mathbf{db}^k) = \left(I_2 + I_{2,x} du^k + I_{2,y} dv^k - \bar{\phi}(I_1) - \sum_{j=1}^{N_{\text{cIll}}} (b_j^k + db_j^k) \cdot \phi_j(I_1)\right)^2, \tag{7.11}$$

such that the corresponding constraint vector on a unified incremental solution vector $\mathbf{dw}_{\text{uni}} = (du, dv, db_1, \dots, db_{N_{\text{cIll}}}, 1)^\top$ is given by

$$\mathbf{p}_{\text{BCA,ill}} = \left(I_{2,x}, I_{2,y}, -\phi_1, \dots, -\phi_{N_{\text{cIll}}}, I_{\tilde{t}}\right)^\top, \tag{7.12}$$

with $I_{\tilde{t}} = I_2 - \bar{\phi}(I_1) - \sum_{j=1}^{N_{\text{cIll}}} b_j \cdot \phi_j(I_1)$. The incremental version of the motion tensor hence reads

$$\begin{aligned} J_{\text{BCA,ill}} &= \mathbf{p}_{\text{BCA,ill}} \mathbf{p}_{\text{BCA,ill}}^\top \\ &= \begin{pmatrix} I_{2,x}^2 & I_{2,x} I_{2,y} & -I_{2,x}\phi_1 & \dots & -I_{2,x}\phi_{N_{\text{cIll}}} & I_{2,x} I_{\tilde{t}} \\ I_{2,y} I_{2,x} & I_{2,y}^2 & -I_{2,y}\phi_1 & \dots & -I_{2,y}\phi_{N_{\text{cIll}}} & I_{2,y} I_{\tilde{t}} \\ -\phi_1 I_{2,x} & -\phi_1 I_{2,y} & \phi_1^2 & \dots & \phi_1\phi_{N_{\text{cIll}}} & -\phi_1 I_{\tilde{t}} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -\phi_{N_{\text{cIll}}} I_{2,x} & -\phi_{N_{\text{cIll}}} I_{2,y} & \phi_{N_{\text{cIll}}}\phi_1 & \dots & \phi_{N_{\text{cIll}}}^2 & -\phi_{N_{\text{cIll}}} I_{\tilde{t}} \\ I_{\tilde{t}} I_{2,x} & I_{\tilde{t}} I_{2,y} & -I_{\tilde{t}}\phi_1 & \dots & -I_{\tilde{t}}\phi_{N_{\text{cIll}}} & I_{\tilde{t}}^2 \end{pmatrix}. \end{aligned} \tag{7.13}$$

7.3.5 GCA Motion Tensor with Illumination Compensation

The linearized version of the GCA with illumination compensation is given by

$$
\begin{aligned}
D_{\mathrm{GCA}}(\mathbf{w},\mathbf{b}) &= \left|\nabla I + \nabla I_x u + \nabla I_y v + \nabla I_t - \nabla\Phi(\mathbf{b},I)\right|^2 \\
&= \left|\begin{pmatrix} I_x + I_{xx}u + I_{xy}v + I_{xt} - \big(\Phi(\mathbf{b},I)\big)_x \\ I_y + I_{yx}u + I_{yy}v + I_{yt} - \big(\Phi(\mathbf{b},I)\big)_y \end{pmatrix}\right|^2 \\
&= \left|\begin{pmatrix} I_x + I_{xx}u + I_{xy}v + I_{xt} - \left(\bar{\phi}(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi_j(I)\right)_x \\ I_y + I_{yx}u + I_{yy}v + I_{yt} - \left(\bar{\phi}(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi_j(I)\right)_y \end{pmatrix}\right|^2 \\
&= \left|\begin{pmatrix} I_x + I_{xx}u + I_{xy}v + I_{xt} - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,x} \cdot \phi_j(I) \ \ldots \\ \qquad \ldots - \left(\bar{\phi}'(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi'_j(I)\right) \cdot I_x \\ I_y + I_{yx}u + I_{yy}v + I_{yt} - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,y} \cdot \phi_j(I) \ \ldots \\ \qquad \ldots - \left(\bar{\phi}'(I) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j \cdot \phi'_j(I)\right) \cdot I_y \end{pmatrix}\right|^2 .
\end{aligned}
\tag{7.14}
$$

Due to the apparent gradient $\nabla\mathbf{b}$ which is present in terms of the expressions $b_{j,x}$ and $b_{j,y}$, the data term comprises constraints on the neighborhood of $\mathbf{b}$. This raises problems w.r.t. the motion tensor notation both in the continuous as well as in the discrete domain. In the continuous case, we know from the Euler-Lagrange equations that $\mathbf{b}$ and $\nabla\mathbf{b}$ are treated separately whereby gradient-expressions are not a direct part of the solution vector $\mathbf{w}$ but serve as a regularizer on the solution. In the discrete case, such constraints are not pointwise anymore due to the finite difference discretizations of the derivatives while the solution vector only contains the unknowns of the current point. Without all unknowns being part of the solution vector $\mathbf{w}$, we cannot directly formulate a motion tensor notation for this constraint and must resort to the incremental formulation.

Incremental Formulation

In the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$ and $\mathbf{b}^{k+1} = \mathbf{b}^k + \mathbf{db}^k$, we can get rid of the gradients of the sought functions by considering *different* time steps for the flow $\mathbf{w}$ and the illumination coefficients $\mathbf{b}$. That means that we consider the flow $\mathbf{w}$ at the current time step $k+1$ (via $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$) while we consider the illumination coefficients at the old time step k (using only $\mathbf{b}^k$ instead of $\mathbf{b}^{k+1} = \mathbf{b}^k + \mathbf{db}^k$). This way,

the incremental version of the constraint reads

$$
\begin{aligned}
D_{\mathrm{GCA}}(\mathbf{dw}^k, \mathbf{db}^k) &= \left| \nabla I_2 + \nabla I_{2,x} du^k + \nabla I_{2,y} dv^k - \nabla \Phi(\mathbf{b}^k, I_1) \right|^2 \qquad (7.15)\\
&= \left| \begin{pmatrix} I_{2,x} + I_{2,xx} du^k + I_{2,xy} dv^k - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,x}^k \cdot \phi_j(I_1) \ \dots \\ \dots - \left(\bar{\phi}'(I_1) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j^k \cdot \phi_j'(I_1) \right) \cdot I_{1,x} \\ I_{2,y} + I_{2,yx} du^k + I_{2,yy} dv^k - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,y}^k \cdot \phi_j(I_1) \ \dots \\ \dots - \left(\bar{\phi}'(I_1) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j^k \cdot \phi_j'(I_1) \right) \cdot I_{1,y} \end{pmatrix} \right|^2 .
\end{aligned}
$$

Within this formulation, all the parts that affect the illumination coefficients are absorbed by those weights of the constraint vectors $\mathbf{p}_{\mathrm{GCA,ill,x}}$ and $\mathbf{p}_{\mathrm{GCA,ill,y}}$ that correspond to the temporal (and thus constant) part of the solution vector $\mathbf{dw}_{\mathrm{uni}}$. Due to the application of the chain rule, these weights become even more complicated. We hence define the abbreviations

$$
I_{\tilde{t},x} = I_{2,x} - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,x}^k \cdot \phi_j(I_1) - \left(\bar{\phi}'(I_1) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j^k \cdot \phi_j'(I_1) \right) \cdot I_{1,x} \qquad (7.16)
$$

and

$$
I_{\tilde{t},y} = I_{2,y} - \sum_{j=1}^{N_{\mathrm{cIll}}} b_{j,y}^k \cdot \phi_j(I_1) - \left(\bar{\phi}'(I_1) + \sum_{j=1}^{N_{\mathrm{cIll}}} b_j^k \cdot \phi_j'(I_1) \right) \cdot I_{1,y} , \qquad (7.17)
$$

such that the constraint vectors are given by

$$
\mathbf{p}_{\mathrm{GCA,ill,x}} = \left(I_{2,xx}, I_{2,xy}, 0, \dots, 0, I_{\tilde{t},x} \right)^\top , \qquad (7.18)
$$

$$
\mathbf{p}_{\mathrm{GCA,ill,y}} = \left(I_{2,yx}, I_{2,yy}, 0, \dots, 0, I_{\tilde{t},y} \right)^\top , \qquad (7.19)
$$

where the zero-weights at the positions of the basis functions indicate our choice to not consider the illumination coefficients at the latest time step.

The final motion tensor for the incremental GCA constraint with illumination compensation hence reads

$$
\begin{aligned}
J_{\mathrm{GCA,ill}} &= \mathbf{p}_{\mathrm{GCA,ill,x}} \mathbf{p}_{\mathrm{GCA,ill,x}}^\top + \mathbf{p}_{\mathrm{GCA,ill,y}} \mathbf{p}_{\mathrm{GCA,ill,y}}^\top \qquad (7.20)\\
&= \begin{pmatrix} I_{2,xx}^2 & I_{2,xx} I_{2,xy} & 0 & \dots & 0 & I_{2,xx} I_{\tilde{t},x} \\ I_{2,xy} I_{2,xx} & I_{2,xy}^2 & 0 & \dots & 0 & I_{2,xy} I_{\tilde{t},x} \\ 0 & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 \\ I_{\tilde{t},x} I_{2,xx} & I_{\tilde{t},x} I_{2,xy} & 0 & \dots & 0 & I_{\tilde{t},x}^2 \end{pmatrix}
\end{aligned}
$$

$$
\begin{aligned}
&+ \begin{pmatrix}
I_{2,yx}^2 & I_{2,yx}I_{2,yy} & 0 & \dots & 0 & I_{2,yx}I_{\tilde{t},y} \\
I_{2,yy}I_{2,yx} & I_{2,yy}^2 & 0 & \dots & 0 & I_{2,yy}I_{\tilde{t},y} \\
0 & 0 & 0 & \dots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \dots & 0 & 0 \\
I_{\tilde{t},y}I_{2,yx} & I_{\tilde{t},y}I_{2,yy} & 0 & \dots & 0 & I_{\tilde{t},y}^2
\end{pmatrix} \\
&= \begin{pmatrix}
\sum\limits_{l\in\{x,y\}} I_{2,lx}^2 & \sum\limits_{l\in\{x,y\}} I_{2,lx}I_{2,ly} & 0 & \dots & 0 & \sum\limits_{l\in\{x,y\}} I_{2,lx}I_{\tilde{t},l} \\
\sum\limits_{l\in\{x,y\}} I_{2,ly}I_{2,lx} & \sum\limits_{l\in\{x,y\}} I_{2,ly}^2 & 0 & \dots & 0 & \sum\limits_{l\in\{x,y\}} I_{2,ly}I_{\tilde{t},l} \\
0 & 0 & 0 & \dots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \dots & 0 & 0 \\
\sum\limits_{l\in\{x,y\}} I_{\tilde{t},l}I_{2,lx} & \sum\limits_{l\in\{x,y\}} I_{\tilde{t},l}I_{2,ly} & 0 & \dots & 0 & \sum\limits_{l\in\{x,y\}} I_{\tilde{t},l}^2
\end{pmatrix}.
\end{aligned}
$$

7.3.6 Further Motion Tensors

The work of Papenberg *et al.* [103] provides an overview of further data constancy assumptions that are similar in spirit to the ones that we have seen so far. All of them can similarly be formulated in terms of motion tensors both with and without the incremental formulation. Please note that such tensors can encode arbitrary numbers of constraints per pixel: While the BCA provides one constraint per pixel and the GCA provides two constraints per pixel – one on the x-derivative and one on the y-derivative of the image –, higher order constancy assumptions (like the Hessian constancy assumption from [103]) can even provide more constraints whereby the corresponding individual tensors are summed up to form the final motion tensor.

7.4 Similarity Tensors

Another instance of linear tensors are similarity tensors. They allow for a direct integration of candidate solutions or for an information flow between different solution candidates. They can implement a variety of similarity constraints such as similarities to feature matches (see Chapter 3 and Chapter 6), similarities between auxiliary solutions in alternating optimizations [126] or between jointly estimated solution candidates (see Chapter 4), or (higher-order) similarities between solutions at subsequent times t [148]. While the former can be considered as special data terms with candidate solutions as given data, the latter two are considered as coupling terms that couple different unknowns and will be presented later.

7.4.1 Candidate Similarity Tensor

A similarity tensor defines a constraint that couples the final solution $\mathbf{w} = (u, v, 1)^\top$ to an already given candidate solution $\mathbf{w}_\mathrm{P} = (u_\mathrm{P}, v_\mathrm{P}, 1)^\top$. It is derived from the similarity assumption [27] which reads

$$D_\mathrm{sim}(\mathbf{w}) \quad = \quad |\mathbf{w} - \mathbf{w}_\mathrm{P}|^2 \;=\; (u - u_\mathrm{P})^2 + (v - v_\mathrm{P})^2 \,. \tag{7.21}$$

Since this term involves two quadratic expressions, we need two constraint vectors that constrain the flow vector $\mathbf{w} = (u, v, 1)^\top$. The constraint can be rewritten in terms of such vectors as

$$\begin{aligned} D_\mathrm{sim}(\mathbf{w}) &= (u - u_\mathrm{P})^2 + (v - v_\mathrm{P})^2 \\ &= \left(\mathbf{p}_\mathrm{sim,u}^\top \mathbf{w}\right)^2 + \left(\mathbf{p}_\mathrm{sim,v}^\top \mathbf{w}\right)^2 , \end{aligned} \tag{7.22}$$

where the constraint vectors are given by

$$\mathbf{p}_\mathrm{sim,u} \;=\; (1, 0, -u_\mathrm{P})^\top , \tag{7.23}$$

$$\mathbf{p}_\mathrm{sim,v} \;=\; (0, 1, -v_\mathrm{P})^\top . \tag{7.24}$$

Using these constraint vectors, the similarity tensor J_sim is then given by

$$\begin{aligned} J_\mathrm{sim} &= J_\mathrm{sim,u} + J_\mathrm{sim,v} \\ &= \mathbf{p}_\mathrm{sim,u}\mathbf{p}_\mathrm{sim,u}^\top + \mathbf{p}_\mathrm{sim,v}\mathbf{p}_\mathrm{sim,v}^\top \\ &= \begin{pmatrix} 1 & 0 & -u_\mathrm{P} \\ 0 & 0 & 0 \\ -u_\mathrm{P} & 0 & u_\mathrm{P}^2 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & -v_\mathrm{P} \\ 0 & -v_\mathrm{P} & v_\mathrm{P}^2 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & -u_\mathrm{P} \\ 0 & 1 & -v_\mathrm{P} \\ -u_\mathrm{P} & -v_\mathrm{P} & u_\mathrm{P}^2 + v_\mathrm{P}^2 \end{pmatrix} . \end{aligned} \tag{7.25}$$

Incremental Formulation

Within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$, the corresponding assumption reads

$$D_\mathrm{sim}(\mathbf{dw}^k) \quad = \quad (u^k + du^k - u_\mathrm{P})^2 + (v^k + dv^k - v_\mathrm{P})^2 \,, \tag{7.26}$$

and thus the constraint vectors are given by $\mathbf{p}_{\text{sim,u}} = (1, 0, u^k - u_\text{P})^\top$ and $\mathbf{p}_{\text{sim,v}} = (0, 1, v^k - v_\text{P})^\top$. The incremental similarity tensor hence reads

$$\begin{aligned} J^k_{\text{sim}} &= J^k_{\text{sim},u} + J^k_{\text{sim},v} \\ &= \mathbf{p}^k_{\text{sim,u}} {\mathbf{p}^k_{\text{sim,u}}}^\top + \mathbf{p}^k_{\text{sim,v}} {\mathbf{p}^k_{\text{sim,v}}}^\top \\ &= \begin{pmatrix} 1 & 0 & u^k - u_\text{P} \\ 0 & 0 & 0 \\ u^k - u_\text{P} & 0 & (u^k - u_\text{P})^2 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & v^k - v_\text{P} \\ 0 & v^k - v_\text{P} & (v^k - v_\text{P})^2 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & u^k - u_\text{P} \\ 0 & 1 & v^k - v_\text{P} \\ u^k - u_\text{P} & v^k - v_\text{P} & (u^k - u_\text{P})^2 + (v^k - v_\text{P})^2 \end{pmatrix}. \end{aligned} \tag{7.27}$$

7.4.2 Directional Similarity Tensor

In this thesis, seeking a 1:1 correspondence between a prior $\mathbf{w}_\text{P}$ and the final flow $\mathbf{w}$ was the goal in Chapter 3 and Chapter 6. In some cases which are not present in this thesis, however, it may be helpful to only enforce a correspondence w.r.t. direction but not w.r.t. velocity. Inspired by our paper [88] that proposes a directional coupling term, we will, hence, also propose two variants of a directional similarity term and derive the corresponding tensor notations.

In general, a directional similarity can be achieved by minimizing a term that involves the scalar product of a normalized prior vector and a normalized version of the estimated $\mathbf{w}$. This comes down to minimizing a term that contains the cosine of the angle between both vectors.

Orientation-Variant Directional Similarity. A first variant of such a directional similarity term that respects the orientations of the prior $\mathbf{w}_\text{P}$ and $\mathbf{w}$ directly involves the cosine $\cos(\angle_{\mathbf{w}_\text{P},\mathbf{w}})$. Using the auxiliary vector $\mathbf{s}_1 = \frac{\mathbf{w}_\text{P}}{|\mathbf{w}_\text{P}|}$ as the normalized prior, it is given by

$$D_{\text{simDir1}}(\mathbf{w}) = \left(1 - \cos(\angle_{\mathbf{w}_\text{P},\mathbf{w}})\right)^2 = \left(1 - \mathbf{s}_1^\top \frac{\mathbf{w}}{|\mathbf{w}|}\right)^2, \tag{7.28}$$

which respects the orientations of all involved vectors and evaluates to a range between 0 (if both $\mathbf{w}_\text{P}$ and $\mathbf{w}$ have the same direction) and 4 (if both vectors have opposite directions).

Orientation-Invariant Directional Similarity. For the case that the orientation does not matter, we can deduce a second variant of a directional similarity term. It is possible to resort to the sine of the corresponding angle and penalize $\left(\sin(\angle_{\mathbf{w}_\text{P},\mathbf{w}})\right)^2$ instead, which evaluates to 0 if both vectors are parallel and to 1 if they are orthogonal.

However, since we eventually want to obtain a tensor notation for such a constraint, we need a formulation in terms of a scalar product of the flow $\mathbf{w}$. In contrast to the cosine-function, the sine-function does not have a direct formulation in terms of a scalar product of the normalized vectors that span the angle. We hence seek for an equivalent constraint that involves a cosine-expression that can be expressed in terms of a scalar product of $\mathbf{w}$.

By considering the relation $\sin(\alpha) = \cos(90° - \alpha) = -\cos(90° + \alpha)$, we see that $(\sin(\alpha))^2 = (\cos(\alpha \pm 90°))^2$ holds. It is hence possible to replace the sine-expression involving $\mathbf{w}_\mathrm{P}$ and $\mathbf{w}$ by a cosine-expression of an auxiliary vector $\mathbf{s}_2 = \frac{\mathbf{w}_\mathrm{P}}{|\mathbf{w}_\mathrm{P}|}^\perp$ (that is orthogonal to $\mathbf{w}_\mathrm{P}$) and $\mathbf{w}$. Hence, the orientation-invariant constraint reads

$$\begin{aligned} D_{\mathrm{simDir2}}(\mathbf{w}) &= \left(\sin(\angle_{\mathbf{w}_\mathrm{P},\mathbf{w}})\right)^2 \\ &= \left(\cos(\angle_{\mathbf{w}_\mathrm{P},\mathbf{w}} \pm 90°)\right)^2 \\ &= \left(\cos(\angle_{\mathbf{s}_2,\mathbf{w}})\right)^2 \\ &= \left(\mathbf{s}_2^\top \frac{\mathbf{w}}{|\mathbf{w}|}\right)^2, \end{aligned} \tag{7.29}$$

which evaluates to a range between 0 (if $\mathbf{w}_\mathrm{P}$ and $\mathbf{w}$ are parallel) and 1 (if both are orthogonal).

Motion Tensor Notation. Since both versions are not linear (due to the vector normalizations), we will directly consider the incremental formulations.

Incremental Formulation

In the incremental formulation, we can resort to the flow at an old time step in the vector normalization which comes down to a lagged nonlinearity strategy for this constraint. Please note that we also introduce a small constant $\epsilon_{\mathrm{vecNorm}}$ to avoid divisions by zero (similar to those in Chapter 2, Sect. 2.8.1). Hence, within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$, the corresponding assumptions read

$$\begin{aligned} D_{\mathrm{simDir1}}(\mathbf{dw}^k) &= \left(1 - \frac{\mathbf{w}_\mathrm{P}^\top}{|\mathbf{w}_\mathrm{P}|} \frac{\mathbf{w}^{k+1}}{|\mathbf{w}^k|}\right)^2 \\ &= \left(1 - \frac{\mathbf{w}_\mathrm{P}^\top}{|\mathbf{w}_\mathrm{P}|} \frac{\mathbf{w}^k + \mathbf{dw}^k}{|\mathbf{w}^k|}\right)^2 \\ &\approx \left(1 - \underbrace{\frac{1}{|\mathbf{w}_\mathrm{P}||\mathbf{w}^k| + \epsilon_{\mathrm{vecNorm}}}}_{=:\theta_{\mathrm{proj1}}} \mathbf{w}_\mathrm{P}^\top (\mathbf{w}^k + \mathbf{dw}^k)\right)^2 \end{aligned} \tag{7.30}$$

$$
\begin{aligned}
&= \left(1 - \theta_{\text{proj1}} \mathbf{w}_{\text{P}}^{\top} \mathbf{w}^k - \theta_{\text{proj1}} \mathbf{w}_{\text{P}}^{\top} \mathbf{dw}^k\right)^2 \\
&= \left(1 - \theta_{\text{proj1}} (u_{\text{P}} \cdot u^k + v_{\text{P}} \cdot v^k) - \theta_{\text{proj1}} (u_{\text{P}} \cdot du^k + v_{\text{P}} \cdot dv^k)\right)^2
\end{aligned}
$$

and

$$
\begin{aligned}
D_{\text{simDir2}}(\mathbf{dw}^k) &= \left(\mathbf{s}^{\top} \frac{\mathbf{w}^{k+1}}{|\mathbf{w}^k|}\right)^2 \\
&= \left(\mathbf{s}^{\top} \frac{\mathbf{w}^k + \mathbf{dw}^k}{|\mathbf{w}^k|}\right)^2 \\
&\approx \left(\underbrace{\frac{1}{|\mathbf{w}^k| + \epsilon_{\text{vecNorm}}}}_{=:\theta_{\text{proj2}}} \mathbf{s}^{\top} (\mathbf{w}^k + \mathbf{dw}^k)\right)^2 \\
&= \left(\theta_{\text{proj2}} \mathbf{s}^{\top} \mathbf{w}^k + \theta_{\text{proj2}} \mathbf{s}^{\top} \mathbf{dw}^k\right)^2 \\
&= \left(\theta_{\text{proj2}} (s_1 \cdot u^k + s_2 \cdot v^k) + \theta_{\text{proj2}} (s_1 \cdot du^k + s_2 \cdot dv^k)\right)^2 ,
\end{aligned}
\tag{7.31}
$$

such that the corresponding constraint vectors are given as

$$
\mathbf{p}_{\text{simDir1}} = (-\theta_{\text{proj1}} u_{\text{P}}, -\theta_{\text{proj1}} v_{\text{P}}, 1 - \theta_{\text{proj1}} (u_{\text{P}} \cdot u^k + v_{\text{P}} \cdot v^k))^{\top} \tag{7.32}
$$

and

$$
\mathbf{p}_{\text{simDir2}} = (\theta_{\text{proj2}} s_1, \theta_{\text{proj2}} s_2, \theta_{\text{proj2}} (s_1 \cdot u^k + s_2 \cdot v^k))^{\top} . \tag{7.33}
$$

The incremental directional candidate similarity tensors hence read

$$
\begin{aligned}
J_{\text{simDir1}}^k &= \mathbf{p}_{\text{simDir1}} \mathbf{p}_{\text{simDir1}}^{\top} \\
&= \begin{pmatrix}
(\theta_{\text{proj1}} u_{\text{P}})^2 & (\theta_{\text{proj1}})^2 u_{\text{P}} \cdot v_{\text{P}} & -\theta_{\text{proj1}} u_{\text{P}} \cdot p_{\text{simDir1},3} \\
(\theta_{\text{proj1}})^2 u_{\text{P}} \cdot v_{\text{P}} & (\theta_{\text{proj1}} v_{\text{P}})^2 & -\theta_{\text{proj1}} v_{\text{P}} \cdot p_{\text{simDir1},3} \\
-\theta_{\text{proj1}} u_{\text{P}} \cdot p_{\text{simDir1},3} & -\theta_{\text{proj1}} v_{\text{P}} \cdot p_{\text{simDir1},3} & (p_{\text{simDir1},3})^2
\end{pmatrix}
\end{aligned}
\tag{7.34}
$$

and

$$
\begin{aligned}
J_{\text{simDir2}}^k &= \mathbf{p}_{\text{simDir2}} \mathbf{p}_{\text{simDir2}}^{\top} \\
&= \begin{pmatrix}
(\theta_{\text{proj2}} s_1)^2 & (\theta_{\text{proj2}})^2 s_1 \cdot s_2 & \theta_{\text{proj2}} s_1 \cdot p_{\text{simDir2},3} \\
(\theta_{\text{proj2}})^2 s_1 \cdot s_2 & (\theta_{\text{proj2}} s_2)^2 & \theta_{\text{proj2}} s_2 \cdot p_{\text{simDir2},3} \\
\theta_{\text{proj2}} s_1 \cdot p_{\text{simDir2},3} & \theta_{\text{proj2}} s_2 \cdot p_{\text{simDir2},3} & (p_{\text{simDir2},3})^2
\end{pmatrix} .
\end{aligned}
\tag{7.35}
$$

7.5 Coupling Tensors

Another concept that can be formulated in terms of point constraints are pointwise coupling terms. They share similarities in spirit to the previously presented similarity assumptions. Now, we also formulate some kind of similarity, but in contrast to the previous similarity assumptions the similarity now is not formulated between an estimated function and a given solution but between two functions that are estimated simultaneously. In our thesis, such a coupling term is used in Chapter 4 to couple competing flow candidates with a final solution in a joint estimation where all use the same data. In a different context, coupling terms have been proposed by Volz *et al.* [148] in order to regularize the estimation in trajectorial direction by coupling solutions for data from subsequent time steps. For the definition of the coupling terms and the derivation of the corresponding coupling tensors, we will now consider flow vectors $\mathbf{w} = (u_1, v_1, u_2, v_2, ..., 1)^\top$ that contain multiple related displacement vectors $(u_i, v_i)^\top$.

7.5.1 First-Order Coupling Tensor

A first-order coupling intends two displacement vectors to be equal. A direct manifestation of it has been introduced in terms of the coupling term in Chapter 4 (Sect. 4.5.2). Moreover, it has found application as a first-order trajectorial regularization term in [148]. W.l.o.g. we assume to have $\mathbf{w} = (u_1, v_1, u_2, v_2, 1)^\top$ such that the corresponding coupling assumption is given by

$$D_{\text{cpl,1st}}(\mathbf{w}) \quad = \quad |\mathbf{w}_2 - \mathbf{w}_1|^2 \quad = \quad (u_2 - u_1)^2 + (v_2 - v_1)^2 \,, \tag{7.36}$$

which can be interpreted as a first-order derivative in direction of the unknowns (e.g. the direction of a trajectory) with the stencil $(-1, 1)$.

Since this term involves two quadratic expressions, we need two constraint vectors that constrain the flow vector $\mathbf{w}$. The constraint can be rewritten in terms of such vectors as

$$D_{\text{cpl,1st}}(\mathbf{w}) \quad = \quad \left(\mathbf{p}_{\text{cpl,1st,u}}^\top \mathbf{w}\right)^2 + \left(\mathbf{p}_{\text{cpl,1st,v}}^\top \mathbf{w}\right)^2 \,, \tag{7.37}$$

where the constraint vectors are given by

$$\mathbf{p}_{\text{cpl,1st,u}} \quad = \quad (-1, 0, 1, 0, 0)^\top \,, \tag{7.38}$$

$$\mathbf{p}_{\text{cpl,1st,v}} \quad = \quad (0, -1, 0, 1, 0)^\top \,. \tag{7.39}$$

Using these constraint vectors, the final first-order coupling tensor is then given by

$$\begin{aligned} J_{\text{cpl,1st}} \quad &= \quad J_{\text{cpl,1st,u}} + J_{\text{cpl,1st,v}} \\ &= \quad \mathbf{p}_{\text{cpl,1st,u}} \mathbf{p}_{\text{cpl,1st,u}}^\top + \mathbf{p}_{\text{cpl,1st,v}} \mathbf{p}_{\text{cpl,1st,v}}^\top \end{aligned}$$

$$
\begin{aligned}
&= \begin{pmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \qquad (7.40)
\end{aligned}
$$

Compared to the corresponding similarity constraint, we see the similar structure in the individual constraints whereby $-u_{\mathrm{P}}$ and $-v_{\mathrm{P}}$ are replaced by -1.

Incremental Formulation

Within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$, the corresponding assumption reads

$$
D_{\mathrm{cpl,1st}}(\mathbf{dw}^k) = (u_2^k + du_2^k - (u_1^k + du_1^k))^2 + (v_2^k + dv_2^k - (v_1^k + dv_1^k))^2 , \quad (7.41)
$$

and thus the constraint vectors are given by $\mathbf{p}_{\mathrm{cpl,1st,u}} = (-1, 0, 1, 0, u_2^k - u_1^k)^\top$ and $\mathbf{p}_{\mathrm{cpl,1st,v}} = (0, -1, 0, 1, v_2^k - v_1^k)^\top$. For the sake of readability, let us define $r_u^k = u_2^k - u_1^k$ and $r_v^k = v_2^k - v_1^k$ as the accumulated remainders of the constraint from the previous scale k. The incremental version of the first-order coupling tensor is then given by

$$
\begin{aligned}
J_{\mathrm{cpl,1st}}^k &= J_{\mathrm{cpl,1st,u}}^k + J_{\mathrm{cpl,1st,v}}^k \\
&= \mathbf{p}_{\mathrm{cpl,1st,u}}^k {\mathbf{p}_{\mathrm{cpl,1st,u}}^k}^\top + \mathbf{p}_{\mathrm{cpl,1st,v}}^k {\mathbf{p}_{\mathrm{cpl,1st,v}}^k}^\top \\
&= \begin{pmatrix} 1 & 0 & -1 & 0 & -r_u^k \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & r_u^k \\ 0 & 0 & 0 & 0 & 0 \\ -r_u^k & 0 & r_u^k & 0 & (r_u^k)^2 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & -r_v^k \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & r_v^k \\ 0 & -r_v^k & 0 & r_v^k & (r_v^k)^2 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & -1 & 0 & -r_u^k \\ 0 & 1 & 0 & -1 & -r_v^k \\ -1 & 0 & 1 & 0 & r_u^k \\ 0 & -1 & 0 & 1 & r_v^k \\ -r_u^k & -r_v^k & r_u^k & r_v^k & (r_u^k)^2 + (r_v^k)^2 \end{pmatrix}. \qquad (7.42)
\end{aligned}
$$

7.5.2 Second-Order Coupling Tensor

A second-order coupling intents two changes between pairs of displacement vectors to be equal, which has been particularly useful in the context of trajectorial regularization [148]. Here, we need at least three displacement vectors $i = 1,...,3$ and w.l.o.g. we assume to have $\mathbf{w} = (u_1, v_1, u_2, v_2, u_3, v_3, 1)^\top$. The corresponding coupling assumption is then given by

$$D_{\text{cpl,2nd}}(\mathbf{w}) = (u_3 - 2u_2 + u_1)^2 + (v_3 - 2v_2 + v_1)^2 , \tag{7.43}$$

which can be interpreted as a second-order derivative in direction of the unknowns (e.g. the direction of a trajectory) with the stencil $(1, -2, 1)$.

Since again this term involves two quadratic expressions, we need two constraint vectors that constrain the flow vector $\mathbf{w}$. The constraint can be rewritten in terms of such vectors as

$$D_{\text{cpl,2nd}}(\mathbf{w}) = \left(\mathbf{p}_{\text{cpl,2nd,u}}^\top \mathbf{w}\right)^2 + \left(\mathbf{p}_{\text{cpl,2nd,v}}^\top \mathbf{w}\right)^2 , \tag{7.44}$$

where the constraint vectors are given by

$$\mathbf{p}_{\text{cpl,2nd,u}} = (1,0,-2,0,1,0,0)^\top , \tag{7.45}$$

$$\mathbf{p}_{\text{cpl,2nd,v}} = (0,1,0,-2,0,1,0)^\top . \tag{7.46}$$

Using these constraint vectors, the final second-order coupling tensor is then given by

$$\begin{aligned}
J_{\text{cpl,2nd}} &= J_{\text{cpl,2nd,u}} + J_{\text{cpl,2nd,v}} \\
&= \mathbf{p}_{\text{cpl,2nd,u}} \mathbf{p}_{\text{cpl,2nd,u}}^\top + \mathbf{p}_{\text{cpl,2nd,v}} \mathbf{p}_{\text{cpl,2nd,v}}^\top \\
&= \begin{pmatrix} 1 & 0 & -2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 4 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 4 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & -2 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & 0 \\ -2 & 0 & 4 & 0 & -2 & 0 & 0 \\ 0 & -2 & 0 & 4 & 0 & -2 & 0 \\ 1 & 0 & -2 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} .
\end{aligned} \tag{7.47}$$

Incremental Formulation

Within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$, the corresponding assumption reads

$$\begin{aligned} D_{\text{cpl,2nd}}(\mathbf{dw}^k) &= (u_3^k + du_3^k - 2(u_2^k + du_2^k) + u_1^k + du_2^k)^2 \\ &\quad + (v_3^k + dv_3^k - 2(v_2^k + dv_2^k) + v_1^k + dv_1^k)^2 \,, \end{aligned} \tag{7.48}$$

and thus the constraint vectors are given by $\mathbf{p}_{\text{cpl,2nd,u}} = (1, 0, -2, 0, 1, 0, u_3^k - 2u_2^k + u_1^k)^\top$ and $\mathbf{p}_{\text{cpl,2nd,v}} = (0, 1, 0, -2, 0, 1, v_3^k - 2v_2^k + v_1^k)^\top$. Again, for the sake of readability, let us define $r_u^k = u_3^k - 2u_2^k + u_1^k$ and $r_v^k = v_3^k - 2v_2^k + v_1^k$ as the accumulated remainders of the constraint from the previous scale k. Using them, the incremental version of the second-order coupling tensor is then given by

$$\begin{aligned} J_{\text{cpl,2nd}}^k &= J_{\text{cpl,2nd,u}}^k + J_{\text{cpl,2nd,v}}^k \\ &= \mathbf{p}_{\text{cpl,2nd,u}}^k \mathbf{p}_{\text{cpl,2nd,u}}^{k\,\top} + \mathbf{p}_{\text{cpl,2nd,v}}^k \mathbf{p}_{\text{cpl,2nd,v}}^{k\,\top} \\ &= \begin{pmatrix} 1 & 0 & -2 & 0 & 1 & 0 & r_u^k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 4 & 0 & -2 & 0 & -2r_u^k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 0 & 1 & 0 & r_u^k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ r_u^k & 0 & -2r_u^k & 0 & r_u^k & 0 & (r_u^k)^2 \end{pmatrix} \\ &\quad + \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & r_v^k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 4 & 0 & -2 & -2r_v^k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 & r_v^k \\ 0 & r_v^k & 0 & -2r_v^k & 0 & r_v^k & (r_v^k)^2 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & -2 & 0 & 1 & 0 & r_u^k \\ 0 & 1 & 0 & -2 & 0 & 1 & r_v^k \\ -2 & 0 & 4 & 0 & -2 & 0 & -2r_u^k \\ 0 & -2 & 0 & 4 & 0 & -2 & -2r_v^k \\ 1 & 0 & -2 & 0 & 1 & 0 & r_u^k \\ 0 & 1 & 0 & -2 & 0 & 1 & r_v^k \\ r_u^k & r_v^k & -2r_u^k & -2r_v^k & r_u^k & r_v^k & (r_u^k)^2 + (r_v^k)^2 \end{pmatrix}. \end{aligned} \tag{7.49}$$

7.6 Directional Regularization Tensors

The previously introduced coupling tensors are particularly helpful in the estimation of large displacements when they directly couple competing candidates with a final solution in a joint estimation, or for a trajectorial regularization if the underlying motion actually undergoes a trajectorially constant or affine motion. In this context, however, having such trajectories is not a realistic assumption in many cases and thus, such terms unnecessarily constrain the velocity of moving objects. A less restrictive soft constraint has been proposed in our work [88] which only enforces a consistent direction along the trajectory. Again, there are two different ways to formulate such a constraint. W.l.o.g. we assume to have $\mathbf{w} = (u_1, v_1, u_2, v_2, 1)^\top$, i.e. we consider trajectories involving the flows $\mathbf{w}_1$ and $\mathbf{w}_2$.

A constraint that directly involves both vectors is given by

$$R_{\text{dirCons1}}(\mathbf{w}) \quad = \quad (1 - \cos(\angle_{\mathbf{w}_1,\mathbf{w}_2}))^2 \quad = \quad \left(1 - \frac{\mathbf{w}_1^\top}{|\mathbf{w}_1|}\frac{\mathbf{w}_2}{|\mathbf{w}_2|}\right)^2, \tag{7.50}$$

which evaluates to a range between 0 (if both $\mathbf{w}_1$ and $\mathbf{w}_2$ have the same direction) and 4 (if both vectors have opposite directions).

Indirect Variant. Similar to the case of Sect. 7.4.2, there is also a variant that involves the sine of the corresponding angle between both parts $\mathbf{w}_1$ and $\mathbf{w}_2$ of the trajectory [88]. Since in this case, however, there is no pre-defined target direction which serves as the basis for an orthogonal vector $\mathbf{s}$, we have to design a meaningful one. Since the focus is on designing a constraint that is based on directions and should not favor one of the two involved directions, $\mathbf{s}$ should be normal and orthogonal to a vector that shows in the average direction of $\mathbf{w}_1$ and $\mathbf{w}_2$. By defining $\mathbf{n}_1 = \frac{\mathbf{w}_1}{|\mathbf{w}_1|}$ and $\mathbf{n}_2 = \frac{\mathbf{w}_2}{|\mathbf{w}_2|}$ as the normalized versions of the involved vectors, we define a prior direction $\mathbf{w}_\text{P} = \frac{\mathbf{n}_1+\mathbf{n}_2}{|\mathbf{n}_1+\mathbf{n}_2|}$ and the auxiliary vector $\mathbf{s}$ is given by $\mathbf{s} = \mathbf{w}_\text{P}^\perp$. Due to the symmetry of the constraint among both flows $\mathbf{w}_1$ and $\mathbf{w}_2$, it actually involves two terms for the regularization of each of the flows. Instead of each covering the full angle $\angle_{\mathbf{w}_1,\mathbf{w}_2}$, each term only covers the angle between the involved flow vector and the prior direction $\mathbf{w}_\text{P}$. Hence, the constraint reads

$$\begin{aligned} R_{\text{dirCons2}}(\mathbf{w}) &= \left(\sin(\angle_{\mathbf{w}_\text{P},\mathbf{w}_1})\right)^2 + \left(\sin(\angle_{\mathbf{w}_\text{P},\mathbf{w}_2})\right)^2 \\ &= \left(\cos(\angle_{\mathbf{w}_\text{P},\mathbf{w}_1} \pm 90°)\right)^2 + \left(\cos(\angle_{\mathbf{w}_\text{P},\mathbf{w}_2} \mp 90°)\right)^2 \\ &= \left(\cos(\angle_{\mathbf{s},\mathbf{w}_1})\right)^2 + \left(\cos(\angle_{\mathbf{s},\mathbf{w}_2})\right)^2 \\ &= \left(\mathbf{s}^\top \frac{\mathbf{w}_1}{|\mathbf{w}_1|}\right)^2 + \left(\mathbf{s}^\top \frac{\mathbf{w}_2}{|\mathbf{w}_2|}\right)^2, \end{aligned} \tag{7.51}$$

which evaluates to a range between 0 (if $\mathbf{w}_1$ and $\mathbf{w}_2$ have equal direction) and 2 (if they have opposite directions).

7.6.1 Relation between Both Variants

By construction of $\mathbf{w}_P$ as the normalized mean vector of $\mathbf{w}_1$ and $\mathbf{w}_2$, the relation $\angle_{\mathbf{w}_P,\mathbf{w}_1} = -\angle_{\mathbf{w}_P,\mathbf{w}_2} = \frac{1}{2}\angle_{\mathbf{w}_1,\mathbf{w}_2}$ holds. We can thus reformulate $R_{\text{dirCons2}}(\mathbf{w})$ as

$$\begin{aligned} R_{\text{dirCons2}}(\mathbf{w}) &= \left(\sin(\angle_{\mathbf{w}_P,\mathbf{w}_1})\right)^2 + \left(\sin(\angle_{\mathbf{w}_P,\mathbf{w}_2})\right)^2 \\ &= \left(\sin(\frac{1}{2}\angle_{\mathbf{w}_1,\mathbf{w}_2})\right)^2 + \left(-\sin(\frac{1}{2}\angle_{\mathbf{w}_1,\mathbf{w}_2})\right)^2 \\ &= 2\left(\sin(\frac{1}{2}\angle_{\mathbf{w}_1,\mathbf{w}_2})\right)^2 . \end{aligned} \tag{7.52}$$

Since this formulation now contains a fraction of the original angle $\angle_{\mathbf{w}_1,\mathbf{w}_2}$ between the flows $\mathbf{w}_1$ and $\mathbf{w}_2$ as in the directional constraint $R_{\text{dirCons1}}(\mathbf{w})$, we can elaborate the relation between both constraints by applying trigonometric addition theorems:

$$\begin{aligned} R_{\text{dirCons2}}(\mathbf{w}) &= 2\left(\sin(\frac{1}{2}\angle_{\mathbf{w}_1,\mathbf{w}_2})\right)^2 \\ &= 2\sqrt{\frac{1-\cos(\angle_{\mathbf{w}_1,\mathbf{w}_2})}{2}}^2 \\ &= 2\cdot\frac{1-\cos(\angle_{\mathbf{w}_1,\mathbf{w}_2})}{2} \\ &= 1-\cos(\angle_{\mathbf{w}_1,\mathbf{w}_2}) \\ &= \sqrt{R_{\text{dirCons1}}(\mathbf{w})}\,. \end{aligned} \tag{7.53}$$

This shows that both variants are equally expressive w.r.t. the orientation between the involved flows. This is not surprising, since the constraint R_{dirCons2} applies the sine-function on half-angles and thus maps the range of angles from the interval $[0, 360°]$ to $[0, 180°]$, such that the sine-function only evaluates to zero if $\angle_{\mathbf{w}_1,\mathbf{w}_2} = 0° = 360°$ i.e. if $\mathbf{w}_1$ and $\mathbf{w}_2$ have the same direction and the same orientation. In contrast to that, the comparable directional candidate similarity constraint D_{simDir2} from Sect. 7.4.2 operates on full angles and is thus orientation-invariant.

7.6.2 Tensors

Since both variants are highly non-linear, we will directly consider the incremental formulations, where we use appropriate time steps for the involved unknowns to make the problem linearly tractable (similar to the case in Sect. 7.4.2). Please note that we also introduce a small constant $\epsilon_{\text{vecNorm}}$ to avoid divisions by zero (similar to those in Chapter 2, Sect. 2.8.1). Hence, within the incremental formulation $\mathbf{w}^{k+1} = \mathbf{w}^k + \mathbf{dw}^k$,

the corresponding assumptions read

$$
\begin{aligned}
R_{\text{dirCons1}}(\mathbf{dw}^k) &= \left(1 - \frac{1}{2}\left(\frac{\mathbf{w}_2^{k\top}}{|\mathbf{w}_2^k|}\frac{\mathbf{w}_1^{k+1}}{|\mathbf{w}_1^k|} + \frac{\mathbf{w}_1^{k\top}}{|\mathbf{w}_1^k|}\frac{\mathbf{w}_2^{k+1}}{|\mathbf{w}_2^k|}\right)\right)^2 \qquad (7.54)\\
&\approx \left(1 - \underbrace{\frac{1}{2(|\mathbf{w}_1^k||\mathbf{w}_2^k| + \epsilon_{\text{vecNorm}})}}_{=:\theta_{\text{proj1}}}\left(\mathbf{w}_2^{k\top}\mathbf{w}_1^{k+1} + \mathbf{w}_1^{k\top}\mathbf{w}_2^{k+1}\right)\right)^2\\
&= \left(1 - \theta_{\text{proj1}}\left(\mathbf{w}_2^{k\top}\mathbf{w}_1^{k+1} + \mathbf{w}_1^{k\top}\mathbf{w}_2^{k+1}\right)\right)^2\\
&= \left(1 - \theta_{\text{proj1}}\left(\mathbf{w}_2^{k\top}\left(\mathbf{w}_1^k + \mathbf{dw}_1^k\right) + \mathbf{w}_1^{k\top}\left(\mathbf{w}_2^k + \mathbf{dw}_2^k\right)\right)\right)^2\\
&= (1 - \theta_{\text{proj1}}(2\cdot(u_2^k\cdot u_1^k + v_2^k\cdot v_1^k)\\
&\qquad + u_2^k\cdot du_1^k + v_2^k\cdot dv_1^k + u_1^k\cdot du_2^k + v_1^k\cdot dv_2^k))^2
\end{aligned}
$$

and

$$
\begin{aligned}
R_{\text{dirCons2}}(\mathbf{dw}^k) &= \left(\mathbf{s}^{k\top}\frac{\mathbf{w}_1^{k+1}}{|\mathbf{w}_1^k|}\right)^2 + \left(\mathbf{s}^{k\top}\frac{\mathbf{w}_2^{k+1}}{|\mathbf{w}_2^k|}\right)^2 \qquad (7.55)\\
&= \left(\underbrace{\frac{1}{|\mathbf{w}_1^k| + \epsilon_{\text{vecNorm}}}}_{=:\theta_{\text{proj2,1}}}\mathbf{s}^{k\top}\mathbf{w}_1^{k+1}\right)^2 + \left(\underbrace{\frac{1}{|\mathbf{w}_2^k| + \epsilon_{\text{vecNorm}}}}_{=:\theta_{\text{proj2,2}}}\mathbf{s}^{k\top}\mathbf{w}_2^{k+1}\right)^2\\
&= \left(\theta_{\text{proj2,1}}\mathbf{s}^{k\top}\left(\mathbf{w}_1^k + \mathbf{dw}_1^k\right)\right)^2 + \left(\theta_{\text{proj2,2}}\mathbf{s}^{k\top}\left(\mathbf{w}_2^k + \mathbf{dw}_2^k\right)\right)^2\\
&= \left(\theta_{\text{proj2,1}}(s_1\cdot u_1^k + s_2\cdot v_1^k) + \theta_{\text{proj2,1}}(s_1\cdot du_1^k + s_2\cdot dv_1^k)\right)^2\\
&\quad + \left(\theta_{\text{proj2,2}}(s_1\cdot u_2^k + s_2\cdot v_2^k) + \theta_{\text{proj2,2}}(s_1\cdot du_2^k + s_2\cdot dv_2^k)\right)^2,
\end{aligned}
$$

such that the corresponding constraint vectors are given as

$$
\mathbf{p}_{\text{dirCons1}} = \left(-\theta_{\text{proj1}}u_2^k, -\theta_{\text{proj1}}v_2^k, -\theta_{\text{proj1}}u_1^k, -\theta_{\text{proj1}}v_1^k, p_{\text{dirCons1,3}}\right)^\top \qquad (7.56)
$$

with

$$
p_{\text{dirCons1,3}} = 1 - 2\theta_{\text{proj1}}\cdot(u_2^k\cdot u_1^k + v_2^k\cdot v_1^k) \qquad (7.57)
$$

and

$$
\mathbf{p}_{\text{dirCons21}} = \left(\theta_{\text{proj2,1}}s_1, \theta_{\text{proj2,1}}s_2, 0, 0, p_{\text{dirCons21,3}}\right)^\top, \qquad (7.58)
$$

$$
\mathbf{p}_{\text{dirCons22}} = \left(0, 0, \theta_{\text{proj2,2}}s_1, \theta_{\text{proj2,2}}s_2, p_{\text{dirCons22,3}}\right)^\top \qquad (7.59)
$$

with

$$p_{\text{dirCons21},3} = \theta_{\text{proj2},1}(s_1 \cdot u_1^k + s_2 \cdot v_1^k)\,, \tag{7.60}$$

$$p_{\text{dirCons22},3} = \theta_{\text{proj2},2}(s_1 \cdot u_2^k + s_2 \cdot v_2^k)\,. \tag{7.61}$$

The incremental directional regularization tensors hence read

$$J_{\text{dirCons1}} = \mathbf{p}_{\text{dirCons1}}\mathbf{p}_{\text{dirCons1}}^\top \tag{7.62}$$

$$= \begin{pmatrix} (\theta_{\text{proj1}}u_2^k)^2 & \theta_{\text{proj1}}^2 u_2^k v_2^k & \theta_{\text{proj1}}^2 u_2^k u_1^k & \theta_{\text{proj1}}^2 u_2^k v_1^k & \xi u_2^k \\ \theta_{\text{proj1}}^2 v_2^k u_2^k & (\theta_{\text{proj1}}v_2^k)^2 & \theta_{\text{proj1}}^2 v_2^k u_1^k & \theta_{\text{proj1}}^2 v_2^k v_1^k & \xi v_2^k \\ \theta_{\text{proj1}}^2 u_1^k u_2^k & \theta_{\text{proj1}}^2 u_1^k v_2^k & (\theta_{\text{proj1}}u_1^k)^2 & \theta_{\text{proj1}}^2 u_1^k v_1^k & \xi u_1^k \\ \theta_{\text{proj1}}^2 v_1^k u_2^k & \theta_{\text{proj1}}^2 v_1^k v_2^k & \theta_{\text{proj1}}^2 v_1^k u_1^k & (\theta_{\text{proj1}}v_1^k)^2 & \xi v_1^k \\ \xi u_2^k & \xi v_2^k & \xi u_1^k & \xi v_1^k & p_{\text{dirCons1},3}^2 \end{pmatrix}$$

with

$$\xi := -\theta_{\text{proj1}} \cdot p_{\text{dirCons1},3} \tag{7.63}$$

and

$$J_{\text{dirCons2}} = \mathbf{p}_{\text{dirCons21}}\mathbf{p}_{\text{dirCons21}}^\top + \mathbf{p}_{\text{dirCons22}}\mathbf{p}_{\text{dirCons22}}^\top \tag{7.64}$$

$$= \begin{pmatrix} (\theta_{\text{proj2},1}s_1)^2 & \theta_{\text{proj2},1}^2 s_1 s_2 & 0 & 0 & \xi_1 s_1 \\ \theta_{\text{proj2},1}^2 s_2 s_1 & (\theta_{\text{proj2},1}s_2)^2 & 0 & 0 & \xi_1 s_2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \xi_1 s_1 & \xi_1 s_2 & 0 & 0 & p_{\text{dirCons21},3}^2 \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (\theta_{\text{proj2},2}s_1)^2 & \theta_{\text{proj2},2}^2 s_1 s_2 & \xi_2 s_1 \\ 0 & 0 & \theta_{\text{proj2},1}^2 s_2 s_1 & (\theta_{\text{proj2},1}s_2)^2 & \xi_2 s_2 \\ 0 & 0 & \xi_2 s_1 & \xi_2 s_2 & p_{\text{dirCons22},3}^2 \end{pmatrix}$$

$$= \begin{pmatrix} (\theta_{\text{proj2},1}s_1)^2 & \theta_{\text{proj2},1}^2 s_1 s_2 & 0 & 0 & \xi_1 s_1 \\ \theta_{\text{proj2},1}^2 s_2 s_1 & (\theta_{\text{proj2},1}s_2)^2 & 0 & 0 & \xi_1 s_2 \\ 0 & 0 & (\theta_{\text{proj2},2}s_1)^2 & \theta_{\text{proj2},2}^2 s_1 s_2 & \xi_2 s_1 \\ 0 & 0 & \theta_{\text{proj2},1}^2 s_2 s_1 & (\theta_{\text{proj2},1}s_2)^2 & \xi_2 s_2 \\ \xi_1 s_1 & \xi_1 s_2 & \xi_2 s_1 & \xi_2 s_2 & r_{\text{dirCons2}} \end{pmatrix}$$

with

$$\xi_1 := \theta_{\text{proj2},1} \cdot p_{\text{dirCons21},3}\,, \tag{7.65}$$

$$\xi_2 := \theta_{\text{proj2},2} \cdot p_{\text{dirCons22},3}\,, \tag{7.66}$$

$$r_{\text{dirCons2}} := p_{\text{dirCons21},3}^2 + p_{\text{dirCons22},3}^2\,. \tag{7.67}$$

7.7 Summary

In this chapter, we derived motion tensors for all data constraints that have been introduced in this thesis. This shows that it is straightforward to implement these constraints within a generalized variational framework. Moreover, we embedded recent concepts from the literature like trajectorial regularization or directional trajectorial regularization into this notational framework. In this context, we also investigated different variants of the directional regularization constraints and provided a detailed background on their derivation and properties. The same holds for the novel directional similarity constraints whose development was inspired by directional regularization constraints, since such similarity and coupling constraints are close in spirit. In both cases, we furthermore elaborated to which degree these variants respect the orientations between the involved vectors.

Summarizing, we have seen that the motion tensor notation [28, 30, 47] can be extended to a larger family of tensor notations that can express a wide family of point constraints.

Chapter

8 Summary & Outlook

8.1 Summary

In this thesis, we focused on improving variational optical flow methods regarding their handling of relative large displacements and illumination changes. We started by devoting separate chapters to each of these data challenges and ended up in a chapter about handling large displacements in the context of illumination changes. Since handling these data challenges required additional or modified data terms, we embedded each of these data terms within a common notational framework based on the motion tensor notation which allows for an easy integration into variational optical flow frameworks.

8.1.1 Large Displacements

We started with a deep analysis what large displacements are and why they are so difficult to be handled. This analysis allowed us to further sub-categorize them into moderately large displacements and arbitrarily large displacements. For each of these categories, we demonstrated how to adapt variational methods to achieve a robust handling that does not suffer from the negative impact of unregularized false matches.

Arbitrarily Large Displacements

In order to estimate arbitrarily large displacements (as a sub-category of relative large displacements), variational methods need guidance by external feature matches. Feature matching does not comprise restrictions due to downsampled image data or regularization steps in the estimation, since it solely focuses on the uniqueness of image features of an object. This allows to estimate an unrestricted displacement size, which, however, comes at the cost of (arbitrarily large) false positive matches due to lacking uniqueness at some locations. Thus, we developed a strategy of determining promising locations for

the integration of feature matches into variational optical flow methods. This strategy does not only respect the structuredness of the image data with the goal to assemble a descriptive and unique feature but also considers deficiencies of the baseline flow with the goal to not deteriorate regions with an already accurate flow. Restricting both the computation and the adaptive integration of such matches to these locations improved the estimation quality while decreasing the workload of the matching step at the same time. This procedure did not only show improvements using conventional features such as Histogram of Oriented Gradients (HOG) or Geometric Blur (GB) which have originally been introduced in contexts other than optical flow but also on the more modern Deep Features that are dedicated to assist the estimation of optical flow and that were published after our paper in [129].

Moderately Large Displacements

We dedicate the (sub-)category of moderately large displacements to those relative large displacements that can actually be handled by variational methods with appropriate modifications. We analyzed that the reason why relative large displacements can not be handled with conventional variational optical flow methods can not only be found in a lack of descriptive data on coarse levels within the coarse-to-fine warping scheme. Additionally, there is a local balancing problem between the data term and the smoothness term at these coarse levels where the small objects are indistinguishable from noise. We resolved this problem using an extended variational model that couples several instances of a baseline variational method, each with a different balance between both terms. This allowed us to estimate multiple flow candidates among all levels of the coarse-to-fine scheme from which the correct displacement is drawn at a level where the data is descriptive enough, i.e. the respective object is unique enough and not mixed up with noise, to reliably conduct such a selection. This procedure allowed to handle many of the large displacement cases from the literature without omitting regularization which is a good way to prevent false matches while at the same time improving results on benchmark data.

8.1.2 Illumination Changes

In the context of illumination changes, we refrained from making the estimation of the optical flow depending on illumination-invariant image features. Invariances always come along with a loss of valuable information. An alternative to the usage of invariances is the estimation of the illumination changes along with the optical flow. This allows to keep the brightness constancy assumption (BCA), which uses the complete spectrum of the available information, valid. To this end, on the one hand, we developed an offline learning strategy to find a suitable parametrization of the brightness transfer functions within the data which can appropriately describe the types of illumination

changes that are present in a data set in terms of basis functions. On the other hand, we extended our variational method such that it can make use of such a parametrization and is able to determine the magnitudes of each type of illumination changes online in terms of illumination coefficients that correspond to the basis functions. This method did not only show improvements on benchmark data, it is also an essential part when estimating relative large displacements in the context of illumination changes in a regularized way.

8.1.3 Large Displacements in the Context of Illumination Changes

It is quite likely that large displacements and illumination changes come together within an image sequence, since both can be consequences of temporal undersampling of the scene. A straightforward, direct combination of the joint estimation of optical flow and illumination changes with a de-regularization strategy, however, is not a good solution in this case, since the de-regularization would break the important balance between the regularizers for the optical flow and the illumination coefficients which is necessary to distinguish motion-induced from illumination-induced brightness variations. In this difficult context, it has proven valuable to combine the main concepts from the previous methods within a sequential pipeline: the estimation of illumination changes (using a pre-computed baseline flow) to account for illumination changes without losing image information, the de-regularization strategy on top of illumination-compensated data to obtain reliable flow candidates for different types of motion patterns and a careful selection strategy to make the integration of these candidates as robust as possible. This strategy has shown consistent improvements on all benchmarks and lead to one of the best variational methods for large displacement optical flow that does not make use of external, non-variational algorithms to obtain flow candidates.

8.2 Future Work

Every step forward in research answers one question but gives birth to several more. We have pushed the limits of variational methods w.r.t. both handling illumination changes and estimating relative large displacements. Recent literature on optical flow has moved away from purely relying on variational methods in order to estimate complex motion patterns – this, however, before having a deep understanding about the potential that such methods have. Nonetheless, variational methods still fulfill an important role as a refinement step in pipeline methods [111] where particularly the estimation of illumination changes is still successfully applied [86, 84, 85, 88]. While, hence, improvements in this aspect could lead to further improvements in state-of-the-art methods, there are also other steps in such pipelines, like e.g. the matching step,

that leave room for improvements. It, hence, also remains an interesting question if flow candidates from variational methods can improve results at this matching step (e.g. in the presence of repetitive patterns where regularization is a crucial concept).

8.2.1 Large Displacements

Relative large displacements remain a tough problem in general. Some ideas for their improved handling apply to methods for handling relative large displacements in general while others affect methods that are particularly dedicated to the handling of one of the sub-categories arbitrarily large displacements or moderately large displacements.

Relative Large Displacements in General

While handling occlusions is always an important topic as recognized by many works in the literature where different more or less complex methods have been proposed, it is particularly important in the context of relative large displacements. Any object whose relative motion exceeds its size, leads to an occlusion-/disocclusion effect that is maximal w.r.t. that object. Since there is no overlap between the old and the new position relative to its background, an area of the size of the fast-moving object is occluded in one frame and disoccluded in the other. Moreover, occlusions lead to high data energies due to missing visual correspondences. Since the level of the data energy is the basis for many methods from this thesis, it could help a lot to know whether a high data energy is the result of a mismatched object or the result of an occlusion.

Arbitrarily Large Displacements

Since our strategy to handle arbitrarily large displacement can be considered as kind of an add-on that matches a sparse set of objects, it could be interesting to not only select locations based on high data energies and do the matching in the image space, but to also do the feature matching on sparser image data that might be derived from or weighted by the data energy, since in such a representation mismatched objects are highly amplified in terms of high energies.

Such an amplification of certain positions in the image data could also be transferred to the coarse-to-fine warping scheme. As soon as some high data energy is detected when going up to fine levels, data could be amplified according to the data energy, downsampled again and the scheme turns to coarser levels again in order to find better estimates at the mismatched parts of the image sequence. This could be considered as some kind of W-cycle (known from numerical multigrid methods [28]) within the coarse-to-fine warping scheme.

Moderately Large Displacements

In the context of a de-regularization scheme, it could also be possible to refrain from the estimation of multiple candidates using a conventional coarse-to-fine scheme but to replace the global smoothness weights by local counterparts $\alpha(\mathbf{x})$ and to also use this kind of a W-cycle as described before. In this context, a detected mismatch could (additionally) lead to an adaptation of the local smoothness weight (potentially including some neighborhood around the mismatched object). This might lead to multiple flow candidates on particular coarse-to-fine levels which need an appropriate fusion scheme at this part of the optimization.

Since relative large displacements introduce a high uncertainty w.r.t. the velocity of objects (e.g. due to missing regularity in general or due to deceleration effects by air resistance as a physical consequence of fast motions), it might be interesting to transfer the estimation of large displacements to the multi-frame domain and measure the consistency of the estimations with directional priors as introduced in [88]. In such a multi-instance model, it could be interesting to compare the directions within potential trajectories for each level of regularization and to additionally consider directional consistency of a flow candidate in the fusion term.

8.2.2 Illumination Changes

Interesting questions arise when estimating illumination changes for color images. On the one hand, this concerns the joint or separate handling of the different channels where we already presented some aspects w.r.t. the learning step, the amount of channels of the estimated illumination coefficients as well as the robustification of the data term and the smoothness term. These aspects require a deeper analysis using more data, maybe considering camera information or the recording environments in general.

Moreover, it is also the representation of color information that raises interesting questions. There are several different color spaces besides the straightforward RGB representation that we have used so far. Alternative color spaces like e.g. the HSV color space carry some types of illumination invariances in their channels whereby the combination of all channels does *not* discard any image information. It is, hence, an interesting question how to learn parametrizations based on such alternative color spaces and which effect the estimation of illumination coefficients for each of the very different color channels has. In many contexts, it might e.g. make sense to only compensate the value channel (and maybe the saturation channel) within the HSV representation. The far-reaching illumination-invariance of the hue channel could help reducing the computational effort within the estimation without losing too many capabilities w.r.t. illumination compensation.

8.2.3 Large Displacements in the Context of Illumination Changes

It is possible to cut-off the pipeline of our last approach after the matching- or the selection-step resulting in a set of variational flow candidates. These could be the basis for a sophisticated combination with matches from external algorithms like Deep Matching [112], Coarse-to-fine PatchMatch [70] or Discrete Flow [93]. Using e.g. our or another sophisticated fusion scheme, it may be possible to select the best matches from these algorithms and combine their strengths such as the potential of external matching steps to estimate arbitrarily large displacements and the inherent sub-pixel accuracy and the robustness at repetitive patterns of variational methods.

Index

Bibliography

[1] L. Adams and J. Ortega. A multi-color SOR method for parallel computation. In *Proc. International Conference on Parallel Processing (ICPP)*, pages 53–46. IEEE Computer Society Press, 1982.

[2] L. Alvarez, J. Esclarín, M. Lefébure, and J. Sánchez. A PDE model for computing the optical flow. In *Proc. XVI Congreso de Ecuaciones Diferenciales y Aplicaciones*, pages 1349–1356, 1999.

[3] L. Alvarez, J. Weickert, and Sánchez. Reliable estimation of dense optical flow fields with large displacements. *International Journal of Computer Vision (IJCV)*, 39(1):41–56, 2000.

[4] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: an empirical evaluation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2294–2301. IEEE Computer Society Press, 2009.

[5] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM Journal on Applied Mathematics*, 60(1):156–182, 1999.

[6] C. Bailer, B. Taetz, and D. Stricker. Flow Fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 4015–4023. IEEE Computer Society Press, 2015.

[7] C. Bailer, K. Varanasi, and D. Stricker. CNN-based patch matching for optical flow with thresholded hinge embedding loss. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3250–3259. IEEE Computer Society Press, 2017.

[8] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. P. Lewis, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1–8. IEEE Computer Society Press, 2007.

[9] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision (IJCV)*, 92(1):1–31, 2011.

[10] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *Proc. European Conference on Computer*

Vision (ECCV), volume 6313 of *Lecture Notes in Computer Science*, pages 29–43. Springer, 2010.

[11] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision (IJCV)*, 12(1):43–77, 1994.

[12] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

[13] A. Behl, O. H. Jafari, S. K. Mustikovela, C. Rother, and A. Geiger. Bounding boxes, segmentations and object coordinates: How important is recognition for 3D scene flow estimation in autonomous driving scenarios? In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 2593–2602. IEEE Computer Society Press, 2017.

[14] A. Berg and J. Malik. Geometric blur for template matching. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 607–614. IEEE Computer Society Press, 2001.

[15] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proc. European Conference on Computer Vision (ECCV)*, volume 588 of *Lecture Notes in Computer Science*, pages 237–252. Springer, 1992.

[16] M. Bertero, T. A. Poggio, and V. Torre. Ill-posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–889, 1988.

[17] J. Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 13(8):775–790, 1991.

[18] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 292–302. IEEE Computer Society Press, 1991.

[19] M. J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise smooth flow fields. *Computer Vision and Image Understanding (CVIU)*, 63(1):75–104, 1996.

[20] M. J. Black, D. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. *Computer Vision and Image Understanding (CVIU)*, 78(1):8–31, 2000.

[21] F. A. Bornemann and P. Deuflhard. The cascadic multigrid method for elliptic problems. *Numerische Mathematik*, 75(2):135–152, 1996.

[22] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23(11):1222–1239, 2001.

[23] J. Braux-Zin, R. Dupont, and A. Bartoli. A general dense image matching framework combining direct and feature-based costs. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 185–192. IEEE Computer Society Press, 2013.

[24] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Applied Mathematics*, 3(3):492–526, 2010.

[25] T. Brox, C. Bregler, and J. Malik. Large displacement optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 41–48. IEEE Computer Society Press, 2009.

[26] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proc. European Conference on Computer Vision (ECCV)*, volume 3024 of *Lecture Notes in Computer Science*, pages 25–36. Springer, 2004.

[27] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(3):500–513, 2011.

[28] A. Bruhn. *Variational Optic Flow Computation: Accurate Modelling and Efficient Numerics*. PhD thesis, Department of Mathematics and Computer Science, Saarland University, Saarbrücken, Germany, July 2006.

[29] A. Bruhn and J. Weickert. Towards ultimate motion estimation: Combining highest accuracy with real-time performance. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 749–755. IEEE Computer Society Press, 2005.

[30] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *International Journal of Computer Vision (IJCV)*, 70(3):257–277, 2006.

[31] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *Proc. European Conference on Computer Vision (ECCV)*, volume 7574 of *Lecture Notes in Computer Science*, pages 611–625. Springer, 2012.

[32] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 8(6):679–698, 1988.

[33] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 168–172. IEEE Computer Society Press, 1994.

[34] Z. Chen, H. Jin, Z. Lin, S. Cohen, and Y. Wu. Large displacement optical flow from nearest neighbor fields. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2443–2450. IEEE Computer Society Press, 2013.

[35] N. Cornelius and T. Kanade. Adapting optical-flow to measure object motion in reflectance and X-ray image sequences. *Computer Graphics*, 18(1):24–25, 1984.

[36] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893. IEEE Computer Society Press, 2005.

[37] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Proc. SIGGRAPH*, Annual Conference Series, pages 369–378. ACM Press, 1997.

[38] D. Dederscheck, T. Müller, and R. Mester. Illumination invariance for driving scene optical flow using comparagram preselection. In *IEEE Intelligent Vehicles Symposium*, pages 742–747. IEEE Computer Society Press, 2012.

[39] O. Demetz. *Feature Invariance versus Change Estimation in Variational Motion Estimation*. PhD thesis, Department of Mathematics and Computer Science, Saarland University, Saarbrücken, Germany, September 2015.

[40] O. Demetz, D. Hafner, and J. Weickert. The complete rank transform: A tool for accurate and morphologically invariant matching of structures. In *Proc. British Machine Vision Conference (BMVC)*, pages 50.1–50.12. BMVA Press, 2013.

[41] O. Demetz, M. Stoll, S. Volz, J. Weickert, and A. Bruhn. Learning brightness transfer functions for the joint recovery of illumination changes and optical flow. In *Proc. European Conference on Computer Vision (ECCV)*, volume 8690 of *Lecture Notes in Computer Science*, pages 455–471. Springer, 2014.

[42] R. Deriche, P. Kornprobst, and G. Aubert. Optical-flow estimation while preserving its discontinuities: a variational approach. In *Proc. Asian Conference on Computer Vision (ACCV)*, volume 1035 of *Lecture Notes in Computer Science*, pages 69–80. Springer, 1995.

[43] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. FlowNet: Learning optical flow with convolutional

networks. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766. IEEE Computer Society Press, 2015.

[44] B. Drayer and T. Brox. Combinatorial regularization of descriptor matching for optical flow estimation. In *Proc. British Machine Vision Conference (BMVC)*, pages 42.1–42.12. BMVA Press, 2015.

[45] L. E. Elsgolc. *Calculus of Variations.* Pergamon, 1962.

[46] G. Farnebäck. Fast and accurate motion estimation using orientation tensors and parametric motion models. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 175–139. IEEE Computer Society Press, 2000.

[47] G. Farnebäck. Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 171–177. IEEE Computer Society Press, 2001.

[48] P. Fischer, A. Dosovitskiy, and T. Brox. Descriptor matching with convolutional neural networks: a comparison to SIFT. Technical Report 1405.5769, arXiv, 2014.

[49] S. Fučik, A. Kratochvil, and J. Nečas. Kačanov-Galerkin method. *Commentationes Mathematicae Universitatis Carolinae*, 14(4):651–659, 1973.

[50] D. Gadot and L. Wolf. PatchBatch: A batch augmented loss for optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4236–4245. IEEE Computer Society Press, 2016.

[51] R. Garg, A. Roussos, and L. Agapito. A variational approach to video registration with subspace constraints. *International Journal of Computer Vision (IJCV)*, 104(3):286–314, 2013.

[52] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE Computer Society Press, 2012.

[53] M. A. Gennert and S. Negahdaripour. Relaxing the brightness constancy assumption in computing optical flow. Technical Report 975, Artificial Intelligence Laboratory, Massachusetts Institiute of Technology, 1987.

[54] P. Golland and A. M. Bruckstein. Motion from color. *Computer Vision and Image Understanding (CVIU)*, 68(3):346–362, 1997.

[55] R. C. Gonzalez and R. E. Woods. *Digital Image Processing (3rd Edition).* Prentice-Hall, Inc., 2006.

[56] M. D. Grossberg and S. K. Nayar. What can be known about the radiometric response from images? In *Proc. European Conference on Computer Vision (ECCV)*, volume 2350 of *Lecture Notes in Computer Science*, pages 189–205. Springer, 2002.

[57] M. D. Grossberg and S. K. Nayar. Modeling the space of camera response functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26(10):1272–1282, 2004.

[58] F. Güney and A. Geiger. Deep discrete flow. In *Proc. Asian Conference on Computer Vision (ACCV)*, volume 10114 of *Lecture Notes in Computer Science*, pages 207–224. Springer, 2016.

[59] P. Gwosdek, S. Grewenig, A. Bruhn, and J. Weickert. Theoretical foundations of gaussian convolution by extended box filtering. In *Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, volume 6667 of *Lecture Notes in Computer Science*, pages 447–458. Springer, 2011.

[60] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Non-rigid dense correspondence with applications for image enhancement. *ACM Transactions on Graphics*, 30(4):70:1–70:9, 2011.

[61] D. Hafner, C. Schroers, and J. Weickert. Introducing maximal anisotropy into second order coupling models. In *Proc. German Conference on Pattern Recognition (GCPR)*, volume 9358 of *Lecture Notes in Computer Science*, pages 79–90. Springer, 2015.

[62] G. D. Hager and P. N. Belhumeur. Real-time tracking of image regions with changes in geometry and illumination. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 403–410. IEEE Computer Society Press, 1996.

[63] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

[64] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Alvey Vision Conference*, pages 23.1–23.6. Alvey Vision Club, 1988.

[65] H. W. Haussecker and D. J. Fleet. Estimating optical flow with physical models of brightness variation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23(6):661–673, 2001.

[66] G. Hermosillo, C. Chefd'Hotel, and O. Faugeras. Variational methods for multimodal image matching. *International Journal of Computer Vision (IJCV)*, 50(3):329–343, 2002.

[67] P. W. Holland and R. E. Welsch. Robust regression using iteratively reweighted least-squares. *Communications in Statistics - Theory and Methods*, 6(9):813–827, 1977.

[68] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[69] Y. Hu, Y. Li, and R. Song. Robust interpolation of correspondences for large displacement optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4791–4799. IEEE Computer Society Press, 2017.

[70] Y. Hu, R. Song, and Y. Li. Efficient coarse-to-fine PatchMatch for large displacement optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5704–5712. IEEE Computer Society Press, 2016.

[71] P. J. Huber. *Robust Statistics*. Wiley, 1981.

[72] T. Hui, X. Tang, and C. C. Loy. LiteFlowNet: A lightweight convolutional neural network for optical flow estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8981–8989. IEEE Computer Society Press, 2018.

[73] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2462–2470. IEEE Computer Society Press, 2017.

[74] S. Ju, M. Black, and A. Jepson. Skin and bones: multi-layer, locally affine, optical flow and regularization with transparency. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 307–314. IEEE Computer Society Press, 1996.

[75] T. H. Kim, H. S. Lee, and K. M. Lee. Optical flow via locally adaptive fusion of complementary data costs. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 3344–3351. IEEE Computer Society Press, 2013.

[76] Y.-H. Kim, A. M. Martínez, and A. C. Kak. Robust motion estimation under varying illumination. *Image and Vision Computing*, 23:365–375, 2005.

[77] S.-H. Lai and B. C. Vemuri. Reliable and efficient computation of optical flow. *International Journal of Computer Vision (IJCV)*, 29(2):87–105, 1998.

[78] V. Lempitsky, S. Roth, and C. Rother. FusionFlow: Discrete-continuous optimization for optical flow estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE Computer Society Press, 2008.

[79] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(5):978–994, 2011.

[80] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[81] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. International Joint Conference on Artificial Intelligence*, volume 2, pages 674–679. Morgan Kaufmann Publishers Inc., 1981.

[82] M. Lysaker, A. Lundervold, and X. C. Tai. 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, 2003.

[83] D. Maurer. *Depth-Driven Variational Methods for Stereo Reconstruction.* Master's thesis, University of Stuttgart, Stuttgart, Germany, 2014.

[84] D. Maurer and A. Bruhn. ProFlow: Learning to predict optical flow. In *Proc. British Machine Vision Conference (BMVC)*, pages 277.1–277.13. BMVA Press, 2018.

[85] D. Maurer, N. Marniok, B. Goldluecke, and A. Bruhn. Structure-from-motion-aware PatchMatch for adaptive optical flow estimation. In *Proc. European Conference on Computer Vision (ECCV)*, volume 11212 of *Lecture Notes in Computer Science*, pages 575–592. Springer, 2018.

[86] D. Maurer, M. Stoll, and A. Bruhn. Order-adaptive and illumination-aware variational optical flow refinement. In *Proc. British Machine Vision Conference (BMVC)*, pages 662.1–662.13. BMVA Press, 2017.

[87] D. Maurer, M. Stoll, and A. Bruhn. Order-adaptive regularisation for variational optical flow: Global, local and in between. In *Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, volume 10302 of *Lecture Notes in Computer Science*, pages 550–562. Springer, 2017.

[88] D. Maurer, M. Stoll, and A. Bruhn. Directional priors for multi-frame optical flow. In *Proc. British Machine Vision Conference (BMVC)*, pages 377.1–377.13. BMVA Press, 2018.

[89] D. Maurer, M. Stoll, S. Volz, P. Gairing, and A. Bruhn. A comparison of isotropic and anisotropic second order regularisers for optical flow. In *Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, volume 10302 of *Lecture Notes in Computer Science*, pages 537–549. Springer, 2017.

[90] S. Meister, J. Hur, and S. Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *Proc. AAAI Conference on Artificial Intelligence*. AAAI Digital Library, 2018.

[91] E. Mémin and P. Pérez. A multigrid approach for hierarchical motion estimation. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 933–938. IEEE Computer Society Press, 1998.

[92] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3061–3070. IEEE Computer Society Press, 2015.

[93] M. Menze, C. Heipke, and A. Geiger. Discrete optimization for optical flow. In *Proc. German Conference on Pattern Recognition (GCPR)*, volume 9358 of *Lecture Notes in Computer Science*, pages 16–28. Springer, 2015.

[94] Y. Mileva, A. Bruhn, and J. Weickert. Illumination-robust variational optical flow with photometric invariants. In *Proc. German Symposium on Pattern Recognition (DAGM)*, volume 4713 of *Lecture Notes in Computer Science*, pages 152–162. Springer, 2007.

[95] N. Mukawa. Estimation of shape, reflection coefficients and illuminant direction from image sequences. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 507–512. IEEE Computer Society Press, 1990.

[96] H.G. Musmann, P. Pirsch, and H.J. Grallet. Advances in picture coding. *Proceedings of the IEEE*, 73(4):523–547, 1985.

[97] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 8(5):565–593, 1986.

[98] S. Negahdaripour and C.-H. Yu. A generalized brightness change model for computing optical flow. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 2–11. IEEE Computer Society Press, 1993.

[99] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.

[100] T. Nir, A. M. Bruckstein, and R. Kimmel. Over-parameterized variational optical flow. *International Journal of Computer Vision (IJCV)*, 76(2):205–216, 2008.

[101] N. Ohta. Optical flow detection by color images. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 801–805. IEEE Computer Society Press, 1989.

[102] Y. Ono, E. Trulls, P. Fua, and K. Yi. LF-Net: Learning local features from images. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, volume 31, pages 6234–6244. Curran Associates, Inc., 2018.

[103] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision (IJCV)*, 67(2):141–158, 2006.

[104] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 12(7):629–639, 1990.

[105] R. Ranftl, K. Bredies, and T. Pock. Non-local total generalized variation for optical flow estimation. In *Proc. European Conference on Computer Vision (ECCV)*, volume 8689 of *Lecture Notes in Computer Science*, pages 439–454. Springer, 2014.

[106] R. Ranftl, S. Gehrig, T. Pock, and H. Bischof. Pushing the limits of stereo using variational stereo estimation. In *IEEE Intelligent Vehicles Symposium*, pages 401–407. IEEE Computer Society Press, 2012.

[107] A. Ranjan and M. J. Black. Optical flow estimation using a spatial pyramid network. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2720–2729. IEEE Computer Society Press, 2017.

[108] H. A. Rashwan, M. A. García, and D. Puig. Variational optical flow estimation based on stick tensor voting. *IEEE Transactions on Image Processing*, 22(7):2589–2599, 2013.

[109] H. A. Rashwan, M. A. Mohamed, M. A. García, B. Mertsching, and D. Puig. Illumination robust optical flow model based on histogram of oriented gradients. In *Proc. German Conference on Pattern Recognition (GCPR)*, volume 8142 of *Lecture Notes in Computer Science*, pages 354–363. Springer, 2013.

[110] Z. Ren, O. Gallo, D. Sun, M.-H. Yang, E. B. Sudderth, and J. Kautz. A fusion approach for multi-frame optical flow estimation. In *Proc. IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2077–2086. IEEE Computer Society Press, 2019.

[111] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. EpicFlow: Edge-preserving interpolation of correspondences for optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1164–1172. IEEE Computer Society Press, 2015.

[112] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. DeepMatching: Hierarchical deformable dense matching. *International Journal of Computer Vision (IJCV)*, 120(3):300–323, 2016.

[113] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3017–3024. IEEE Computer Society Press, 2011.

[114] S. Ricco and C. Tomasi. Dense Lagrangian motion estimation with occlusions. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807. IEEE Computer Society Press, 2012.

[115] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, 47(1):7–42, 2002.

[116] C. Schnörr. Determining optical flow for irregular domains by minimizing quadratic functionals of a certain class. *International Journal of Computer Vision (IJCV)*, 6(1):25–38, 1991.

[117] C. Schnörr. Segmentation of visual motion by minimizing convex non-quadratic functionals. In *Proc. International Conference on Pattern Recognition (ICPR)*, volume 1, pages 661–663. IEEE Computer Society Press, 1994.

[118] R. Schuster, C. Bailer, O. Wasenmüller, and D. Stricker. FlowFields++: Accurate optical flow correspondences meet robust interpolation. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 1463–1467. IEEE Computer Society Press, 2018.

[119] T. Schuster, L. Wolf, and D. Gadot. Optical flow requires multiple strategies (but only one network). In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6921–6930. IEEE Computer Society Press, 2017.

[120] L. Sevilla-Lara, D. Sun, E. G. Learned-Miller, and M.J. Black. Optical flow estimation with channel constancy. In *Proc. European Conference on Computer Vision (ECCV)*, volume 8690 of *Lecture Notes in Computer Science*, pages 423–438. Springer, 2014.

[121] J. Shi and C. Tomasi. Good features to track. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600. IEEE Computer Society Press, 1994.

[122] D. Shulman and J. Hervé. Regularization of discontinuous flow fields. In *Proc. Workshop on Visual Motion*, pages 81–86. IEEE Computer Society Press, 1989.

[123] L. Sigal and M. Black. HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision (IJCV)*, 87(1):4–27, 2010.

[124] F. Stein. Efficient computation of optical flow using the census transform. In *Proc. German Symposium on Pattern Recognition (DAGM)*, volume 3175 of *Lecture Notes in Computer Science*, pages 79–86. Springer, 2004.

[125] F. Steinbrücker, T. Pock, and D. Cremers. Advanced data terms for variational optic flow estimation. In *Proceedings of the Vision, Modeling, and Visualization Workshop (VMV)*, pages 155–164. DNB, 2009.

[126] F. Steinbrücker, T. Pock, and D. Cremers. Large displacement optical flow computation without warping. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1609–1614. IEEE Computer Society Press, 2009.

[127] M. Stoll, D. Maurer, and A. Bruhn. Variational large displacement optical flow without feature matches. In *Proc. International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, volume 10746 of *Lecture Notes in Computer Science*, pages 79–92. Springer, 2018.

[128] M. Stoll, D. Maurer, S. Volz, and A. Bruhn. Illumination-aware large displacement optical flow. In *Proc. International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, volume 10746 of *Lecture Notes in Computer Science*, pages 139–154. Springer, 2018.

[129] M. Stoll, S. Volz, and A. Bruhn. Adaptive integration of feature matches into variational optical flow methods. In *Proc. Asian Conference on Computer Vision (ACCV)*, volume 7727 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2013.

[130] M. Stoll, S. Volz, and A. Bruhn. Joint trilateral filtering for multiframe optical flow. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 3845–3849. IEEE Computer Society Press, 2013.

[131] M. Stoll, S. Volz, D. Maurer, and A. Bruhn. A time-efficient optimisation framework for parameters of optical flow methods. In *Proc. Scandinavian Conference*

on Image Analysis (SCIA), volume 10269 of *Lecture Notes in Computer Science*, pages 41–53. Springer, 2017.

[132] D. Sun, S. Roth, and M. J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision (IJCV)*, 106(2):115–137, 2014.

[133] D. Sun, S. Roth, J. P. Lewis, and M. J. Black. Learning optical flow. In *Proc. European Conference on Computer Vision (ECCV)*, volume 5304 of *Lecture Notes in Computer Science*, pages 83–97. Springer, 2008.

[134] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8934–8943. IEEE Computer Society Press, 2018.

[135] K. Tieu and E. G. Miller. Unsupervised color constancy. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, volume 15, pages 1327–1334. MIT Press, 2002.

[136] M. Tistarelli. Multiple constraints for optical flow. In *Proc. European Conference on Computer Vision (ECCV)*, volume 800 of *Lecture Notes in Computer Science*, pages 61–70. Springer, 1994.

[137] O. Tretiak and L. Pastor. Velocity estimation from image sequences with second order differential operators. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16–19. IEEE Computer Society Press, 1984.

[138] W. Trobin, T. Pock, D. Cremers, and H. Bischof. An unbiased second-order prior for high-accuracy motion estimation. In *Proc. German Symposium on Pattern Recognition (DAGM)*, volume 5096 of *Lecture Notes in Computer Science*, pages 396–405. Springer, 2008.

[139] D. Tschumperle and R. Deriche. Diffusion tensor regularization with constraints preservation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 948–953. IEEE Computer Society Press, 2001.

[140] Z. Tu, R. Poppe, and R. C. Veltkamp. Weighted local intensity fusion method for variational optical flow estimation. *Pattern Recognition*, 50:223–232, 2016.

[141] N. Ufer and B. Ommer. Deep semantic feature matching. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5929–5938. IEEE Computer Society Press, 2017.

[142] S. Uras, F. Girosi, A. Verri, and V. Torre. A computational approach to motion perception. *Biological Cybernetics*, 60(2):79–87, 1988.

[143] L. Valgaerts, A. Bruhn, H. Zimmer, J. Weickert, C. Stoll, and C. Theobalt. Joint estimation of motion, structure and geometry from stereo sequences. In *Proc. European Conference on Computer Vision (ECCV)*, volume 6314 of *Lecture Notes in Computer Science*, pages 568–581. Springer, 2010.

[144] J. van de Weijer and T. Gevers. Robust optical flow from photometric invariants. In *Proc. IEEE International Conference on Image Processing (ICIP)*, pages 1835–1838. IEEE Computer Society Press, 2004.

[145] C. Vogel, S. Roth, and K. Schindler. An evaluation of data costs for optical flow. In *Proc. German Conference on Pattern Recognition (GCPR)*, volume 8142 of *Lecture Notes in Computer Science*, pages 343–353. Springer, 2013.

[146] C. Vogel, K. Schindler, and S. Roth. 3D scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision (IJCV)*, 115(1):1–28, 2015.

[147] O. Vogel, A. Bruhn, J. Weickert, and S. Didas. Direct shape-from-shading with adaptive higher order regularisation. In *Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, volume 4485 of *Lecture Notes in Computer Science*, pages 871–882. Springer, 2007.

[148] S. Volz, A. Bruhn, L. Valgaerts, and H. Zimmer. Modeling temporal coherence for optical flow. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1116–1123. IEEE Computer Society Press, 2011.

[149] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1663–1668. IEEE Computer Society Press, 2009.

[150] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for TV-L^1 optical flow computation. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, volume 5604 of *Lecture Notes in Computer Science*, pages 23–45. Springer, 2008.

[151] J. Weickert, S. Grewenig, C. Schroers, and A. Bruhn. Cyclic schemes for PDE-based image analysis. *International Journal of Computer Vision (IJCV)*, 118(3):275–299, 2016.

[152] J. Weickert and C. Schnörr. A theoretical framework for convex regularizers in PDE-based computation of image motion. *International Journal of Computer Vision (IJCV)*, 45(3):245–264, 2001.

[153] J. Weickert and M. Welk. Tensor field interpolation with PDEs. In *Visualization and Processing of Tensor Fields*, Mathematics and Visualization, pages 315–325. Springer, 2006.

[154] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1385–1392. IEEE Computer Society Press, 2013.

[155] M. Werlberger, T. Pock, and H. Bischof. Motion estimation with non-local total variation regularization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2464–2471. IEEE Computer Society Press, 2010.

[156] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-L^1 optical flow. In *Proc. British Machine Vision Conference (BMVC)*, pages 108.1–108.11. BMVA Press, 2009.

[157] J. Wulff, L. Sevilla-Lara, and M. J. Black. Optical flow in mostly rigid scenes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6911–6920. IEEE Computer Society Press, 2017.

[158] J. Xu, R. Ranftl, and V. Koltun. Accurate optical flow via direct cost volume processing. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5807–6815. IEEE Computer Society Press, 2017.

[159] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1293–1300. IEEE Computer Society Press, 2010.

[160] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34:1744–1757, 2012.

[161] D. M. Young. *Iterative Solution of Large Linear Systems*. Academic Press, 1971.

[162] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *Proc. European Conference on Computer Vision (ECCV)*, volume 800 of *Lecture Notes in Computer Science*, pages 151–158. Springer, 1994.

[163] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L^1 optical flow. In *Proc. German Symposium on Pattern Recognition (DAGM)*, volume 4713 of *Lecture Notes in Computer Science*, pages 214–223. Springer, 2007.

[164] H. Zimmer, A. Bruhn, and J. Weickert. Optic flow in harmony. *International Journal of Computer Vision (IJCV)*, 93(3):368–388, 2011.

[165] H. Zimmer, A. Bruhn, J. Weickert, L. Valgaerts, A. Salgado, B. Rosenhahn, and H.-P. Seidel. Complementary optic flow. In *Proc. International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, volume 5681 of *Lecture Notes in Computer Science*, pages 207–220. Springer, 2009.

[166] S. Zweig and L. Wolf. InterpoNet, a brain inspired neural network for optical flow dense interpolation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6363–6372. IEEE Computer Society Press, 2017.

Appendix

A Evaluation Details

In this appendix, we provide details on the parameters and the runtimes of the methods that are presented in this thesis. We will start by stating the runtimes of each of the methods and afterwards, we provide the crucial parameters for the final settings of each method for each data set it is applied to.

A.1 Runtimes

In order to compare the runtimes of the methods, we all applied them on the Tennis sequence (Frame 496) of size 530×380. The system uses an Intel Core i7-6950X CPU @ 3.00GHz, the application is written in C++ using SIMD vectorization and OpenMP parallelization where possible and it is executed in a virtual machine with a Xubuntu 18.04 OS. For each setting, we conduct a purely sequential run on a single core and a parallelized run using four cores.

#Cores	Version	*ALD-Flow*	*ContFusion-Flow*	*BTFIllum*	*ICALD-Flow*
1	Full Method	00:40	02:31	01:05	03:04
	Baseline	00:09	00:06	00:09	00:06
4	Full Method	00:31	01:05	00:31	01:52
	Baseline	00:06	00:04	00:06	00:04

A.2 Numerical Parameters

For the different methods, we used the following important numerical parameters. They influence the runtime of the methods and have thus been also adapted to the complexity of the respective variational models to achieve reasonable results without the need of

too much runtime. Please note that we use a cascadic initialization [21] of the SOR solver whenever we employ second-order regularization.

Parameter	*ALD-Flow*	*ContFusion-Flow*	*BTFIllum*	*ICALD-Flow*
Coarse-to-fine η	0.95	0.95	0.95	0.95
Lag.-NonLin. Iterations	10	3	5	3
SOR ω	1.95	1.85	1.85	1.85
Solver Iterations	5	10	10	10

Here, η is the rescaling factor for the image data within the coarse-to-fine warping scheme.

A.3 General Model Parameters

A lot of the model parameters are present in all of the presented methods, since they build upon the same baseline method. There are the following weights in the model:

Name	Parameter	Introduction	Remarks
Weight of the BCA	δ	Sect. 2.8.4	$\delta = 1$, if not stated explicitly
Weight of the GCA	γ	Sect. 2.8.4	
Smoothness weight	α	Sect. 2.8.4	
Differentiability constant	ϵ_D	Sect. 2.6.1	
Normalization constant	ϵ_{cNorm}	Sect. 2.8.1	
Differentiability constant	ϵ_{S1}	Sect. 2.8.3	
Differentiability constant	ϵ_{S2}	Sect. 2.8.3	

When using the isotropic second-order regularizer from Sect. 2.9 for the KITTI benchmarks, we set $\epsilon_S := \epsilon_{S1}$.

A.4 Parameter Optimization

There are a lot of different ways to find good parameters for optical flow methods. Besides manual tuning, there are many automatic parameter optimization strategies, some of them are described and implemented in our work [131]. These include the Downhill Simplex method (DS) [99], the Covariance Matrix Adaptation Evolution Strategy (CMAeS) [63] and the Logarithmic Cascadic Sampling method (LC) [39]. In

general, we follow the methods that have been used when optimizing the parameters for the paper versions of our methods. Since these have changed over time, we apply different strategies in different sections. However, when comparing different methods or different variants of a method on a particular data set within a particular experiment, we consistently used the same parameter optimization method. As a general rule of thumb, we used CMAeS if there are more than three parameters to be optimized, since Downhill Simplex (DS) gets trapped in local minima too easily while the computational effort of Logarithmic Cascadic Sampling (LC) would be intractable.

A.5 ALD-Flow

Our method *ALD-Flow* introduces an additional similarity term which is weighted by the parameter β.

A.5.1 Large Displacement Sequences

For the large displacement sequences, we used anisotropic first-order regularization and obtained the following parameters:

Setting	Optimization	γ	α	β	ϵ_D	ϵ_{cNorm}	ϵ_{S1}	ϵ_{S2}
Tennis	manual	1.5	0.01	140	$3 \cdot 10^{-5}$	0.01	0.05	0.05
others	manual	5	0.06	140	$3 \cdot 10^{-5}$	0.01	0.02	0.02

A.5.2 Major Benchmarks

For the major benchmarks, we consider both the results for the standard variant of *ALD-Flow* using HOG- and GB- descriptors as well as the variant using DeepMatches.

HOG- and GB-Features

Using HOG- and GB-Features, we obtained the following parameters:

Setting	Optimization	γ	α	β	ϵ_D	ϵ_{cNorm}	ϵ_{S1}	ϵ_{S2}
Middlebury	manual	15	0.08	28	$3 \cdot 10^{-5}$	0.1	0.02	0.02
Sintel	LC	2.638	0.01	4.028	$3 \cdot 10^{-5}$	0.1	0.02	0.02
KITTI '12	DS	521	0.431	28	$3 \cdot 10^{-5}$	0.1	0.5	–
KITTI '15	DS	628	0.324	28	$3 \cdot 10^{-5}$	0.1	0.5	–

Deep Matches

Using Deep Matches and a common set of thresholds for all benchmarks, we obtained the following parameters:

Setting	Optimization	γ	α	β	ϵ_D	ϵ_{cNorm}	ϵ_{S1}	ϵ_{S2}
Middlebury	CMAeS	1.25	0.0115	1.664	$3 \cdot 10^{-5}$	0.1	0.02	0.02
Sintel	CMAeS	1.10	0.321	2.68	$3 \cdot 10^{-5}$	0.1	0.02	0.02
KITTI '12	CMAeS	3083	7.32	20.08	$3 \cdot 10^{-5}$	0.1	0.5	–
KITTI '15	CMAeS	908	0.337	51.84	$3 \cdot 10^{-5}$	0.1	0.5	–

A.6 ContFusion-Flow

In the experiments, we optimized only the following parameters: the number N_{cand} of candidates, the data weights δ and γ and the smoothness weight α_1. The remaining parameters are kept fixed throughout all experiments. They are given by $\beta_i = \alpha_f = \alpha_1$, $L = 5$, $\lambda_{\text{cand}} = 1000$, $\lambda_{\text{cpl}} = 1$, $\kappa_s = 0.3$, $\kappa_d = 5$, $\epsilon_D = 0.01$, $\epsilon_{\text{cNorm}} = 0.01$.

A.6.1 Large Displacement Sequences

For the large displacement sequences, we used anisotropic first-order regularization and intentionally chose the data term weights in a way such that $\delta + \gamma = 1$ holds in order to have a convex combination of both data constraints. This made it easy to evaluate the influence of each data constraint on the ability to estimate large displacements. We finally obtained the following parameters:

Setting	Optimization	N_{cand}	δ	γ	α_1	ϵ_{S1}	ϵ_{S2}
All	manual	7	0.5	0.5	2	0.02	0.03

A.6.2 Major Benchmarks

For the major benchmarks, we iterated through different choices of the number of candidates N_{cand} and for each choice, we optimized the parameters γ and α_1 providing the following outcome:

Setting	Optimization	N_{cand}	δ	γ	α_1	ϵ_{S1}	ϵ_{S2}
Middlebury	DS	2	1	67.1	166	0.02	0.03
Sintel	DS	3	1	7.72	24	0.02	0.03
KITTI '12	DS	5	1	31.54	79.77	0.5	–
KITTI '15	DS	1	1	112.32	60.26	0.5	–

Optimization of More Parameters

In this setting, we additionally optimized the weight of the candidate estimations λ_{cand} and the weight of the coupling term λ_{cpl}. In this context, we consider the best results from a joint optimization of γ, α_1, λ_{cand} and λ_{cpl}, a separate optimization of λ_{cand} and λ_{cpl} using the previously determined values for γ and α_1 and the results from the previous experiment. Since the optimization of four parameters heavily increases the dimensionality of the optimization, we switched to CMAeS for the parameter optimization when optimizing all parameters jointly. The best parameters are then given by:

Setting	Optimiz.	N_{cand}	δ	γ	α_1	λ_{cand}	λ_{cpl}	ϵ_{S1}	ϵ_{S2}
Middlebury	CMAeS	7	1	7.39	16.71	0.858	0.0255	0.02	0.03
Sintel	DS	3	1	7.72	24	891	0.523	0.02	0.03
KITTI '12	CMAeS	5	1	24.9	65.1	0.294	0.280	0.5	–
KITTI '15	DS	1	1	112.32	60.26	1000	1	0.5	–

A.7 BTFIllum

Our method *BTFIllum* contains an additional parameter α_{ill} that weights the first-order anisotropic smoothness constraint on the illumination coefficients with fixed differentiability constants $\epsilon_{\text{ill},S1} = 0.01$ and $\epsilon_{\text{ill},S2} = 0.01$. Moreover, the other constants $\epsilon_D = 0.01$ and $\epsilon_{\text{cNorm}} = 0.01$ are fixed as well.

A.7.1 Major Benchmarks

For each the major benchmarks, we state the results of the best setting where illumination compensation is *active* (for the Sintel benchmark, we choose the setting for the best result for the complete data set). In these contexts, we obtained the following set of parameters:

Setting	Optimization	γ	α	α_{ill}	ϵ_{S1}	ϵ_{S2}
Middlebury	LC	2.08	4.67	1.38	0.02	0.03
Sintel	LC	4.13	9.53	24.04	0.02	0.03
KITTI '12	LC	6.96	5.72	1.60	0.5	–
KITTI '15	LC	0.162	0.147	0.215	0.5	–

A.8 ICALD-Flow

Our method *ICALD-Flow* contains an additional parameter β for the similarity term (similar to *ALD-Flow*). Similar to *ContFusion-Flow*, it contains parameters $N_{\text{cand},BCA}$ and $N_{\text{cand},GBCA}$ to determine the numbers of candidates for both the BCA candidate model and the GBCA candidate model, respectively. For each of these models, there is also a base smoothness weight α_{cand}. In all settings, the weights $\alpha_{\text{ill}} = 4000$ and $\alpha_{\text{inp}} = 3000$ and the constants $\epsilon_D = 0.01$ and $\epsilon_{\text{cNorm}} = 0.01$ are fixed.

A.8.1 Large Displacement Sequences

For the large displacement sequences, we used anisotropic first-order regularization and obtained the following parameters:

Setting	Optimization	γ	α	β	α_{cand}		N_{cand}		ϵ_{S1}	ϵ_{S2}
					BCA	GBCA	BCA	GBCA		
All	manual	20	40	900	8	8	8	2	0.02	0.03

A.8.2 Major Benchmarks

For the major benchmarks, we obtained the following set of parameters when using a first-order regularizer on the candidates (non-socr settings):

Setting	Optimiz.	γ	α	β	α_{cand}		N_{cand}		ϵ_{S1}	ϵ_{S2}
					BCA	GBCA	BCA	GBCA		
Middlebury	CMAeS	18.28	37.52	900	8	18.86	6	2	0.02	0.03
Sintel	CMAeS	17.22	51.35	900	8	19.75	6	2	0.02	0.03
KITTI '12	CMAeS	42.58	63.72	900	8	15.97	6	2	0.5	–
KITTI '15	CMAeS	180.91	39.44	900	8	13.48	6	2	0.5	–

Appendix

B Using Color Images in the Estimation of Illumination Changes

In this appendix, we provide the results of an exhaustive evaluation on handling color channels within a joint estimation of optical flow and illumination changes as presented in Chapter 5. To this end, we explain the design choices that can be made both in the learning stage (of the basis functions) and in the estimation stage (of the flow and the illumination coefficients) and provide the results for each combination of choices on the KITTI 2015 benchmark [92].

B.1 Handling of Color Channels

When colors come into play, there are a lot of decisions to be made how to handle the different image channels at different stages.

Learning Stage. In the learning stage, we can learn the basis functions on grey value versions of the images or on the color images where this can be done either jointly or separately for all channels. If learned separately, there still is the sub-decision to be made whether the clustering of the BTFs shall be conducted jointly or separately for the channels. The latter case comes down to treating each of the color channels of the images as grey value images and having a completely independent learning process for each channel.

Estimation Stage. In the stage of optical flow estimation, there again is the option to either use grey value versions of the images or to use the full color spectrum of the original images. When using color images, there is the option to estimate a joint set of illumination coefficients for all image channels or to have separate coefficients for each of the image channels. In the latter case, both the data terms (DT) as well

as the smoothness term of the coefficients (ST) offer the options for either a joint robustification (jt) over all channels or a separate one (sp).

B.2 Results on the KITTI 2015 Benchmark

At hand of the KITTI 2015 benchmark [92], which in contrast to the edition of 2012 makes use of color images, we can show the effects of these decisions on the results. Tab. 1 provides an overview of the results obtained using all possible combinations of decisions from both the learning stage as well as the estimation stage. To this end, we compared the baseline results (both for grey value as well as for color images) to results from *BTFIllum* using the normalized affine basis, the basis learned for KITTI 2012 – as a representative of a basis learned from grey value data – and different variants of bases learned from the KITTI 2015 color images. We group our conclusions according to each aspect that is of interest. The overall best results are marked using a bold font while the best results for each basis are underlined. Since this is the basis to derive the right options when estimating results for the other benchmarks that contain color images (Middlebury and MPI Sintel), we do not only state the thresholded bad pixel error values (BP3) but also the more continuous average endpoint errors (AEE).

Table 1: Comparison of different strategies to handle colors for different parametrizations, both at the learning stage as well as on the estimation stage. At the learning stage we can learn joint or separate basis functions for the color channels (Column 3). If learning them separate, we can still employ a joint or a separate K-Means clustering among the channels (Column 4). At the estimation stage, we can use color or grey value images (Column 5). If using color images, we can estimate a joint or a separate set of coefficients for each channel (Column 6). If estimating a separate set, we can decide whether to employ a joint or a separate robustification for the data term (Column 7) and the smoothness term (Column 8), respectively. Please note that there is no meaningful way to use separately learned basis functions for grey value images.

Method	Basis			Estimation				Error	
		Learning		Image	Coefficients	Robustif.			
	Type	Bases	Clust.	Channels	Channels	DT	ST	BP3	AEE
Baseline	–	–	–	3	–	jt	jt	23.99%	9.642
				1	–	–	–	23.73%	10.137
BTFIllum	affine (norm.)	–	–	3	3	jt	jt	24.89%	10.698
				3	3	sp	jt	24.86%	10.652
				3	3	jt	sp	24.84%	10.695
				3	3	sp	sp	24.82%	10.654
				3	1	jt	jt	24.70%	10.583
				1	1	–	–	24.23%	11.100
	KITTI '12	–	–	3	3	jt	jt	23.87%	9.504
				3	3	sp	jt	24.09%	9.836
				3	3	jt	sp	23.85%	9.526
				3	3	sp	sp	24.09%	9.848
				3	1	jt	jt	23.72%	**9.208**
				1	1	–	–	**23.55%**	9.866
	KITTI '15	jt	jt	3	3	jt	jt	24.06%	9.651
				3	3	sp	jt	24.11%	9.626
				3	3	jt	sp	23.94%	9.559
				3	3	sp	sp	24.10%	9.643
				3	1	jt	jt	23.87%	9.442
				1	1	–	–	23.73%	10.216
		sp	jt	3	3	jt	jt	24.03%	9.486
				3	3	sp	jt	24.08%	9.523
				3	3	jt	sp	24.01%	9.450
				3	3	sp	sp	24.07%	9.469
				3	1	jt	jt	23.86%	9.576
		sp	sp	3	3	jt	jt	24.08%	9.576
				3	3	sp	jt	24.14%	10.143
				3	3	jt	sp	24.10%	9.942
				3	3	sp	sp	24.16%	10.272
				3	1	jt	jt	23.91%	9.445

Appendix

C Own Publications

C.1 Core Area

1. D. Maurer, M. Stoll, A. Bruhn
   Directional Priors for Multi-Frame Optical Flow.
   In *Proc. 29th British Machine Vision Conference*
   BMVC 2018, Newcastle upon Tyne, UK, September 2018 – H. P. H. Shum, T. Hospedales, L. Shao (Eds.)
   Pages 377.1–377.13, BMVA Press, 2018.

2. M. Stoll, D. Maurer, A. Bruhn
   Variational Large Displacement Optical Flow without Feature Matches.
   In *Proc. Int. Conf. on Energy Minimization Methods in Computer Vision and Pattern Recognition*
   EMMCVPR 2017, Venice, Italy, November 2017 – M. Pelillo, E. Hancock (Eds.)
   Lecture Notes in Computer Science, Vol. 10746, 79–92, Springer, Berlin, 2017.
   Long oral presentation at the EMMCVPR (top 9 paper).

3. M. Stoll, D. Maurer, S. Volz, A. Bruhn
   Illumination-Aware Large Displacement Optical Flow.
   In *Proc. Int. Conf. on Energy Minimization Methods in Computer Vision and Pattern Recognition*
   EMMCVPR 2017, Venice, Italy, November 2017 – M. Pelillo, E. Hancock (Eds.)
   Lecture Notes in Computer Science, Vol. 10746, 139–154, Springer, Berlin, 2017.

4. D. Maurer, M. Stoll, A. Bruhn
Order-Adaptive and Illumination-Aware Variational Optical Flow Refinement.
In *Proc. 28th British Machine Vision Conference*
BMVC 2017, London, UK, September 2017 – K. Mikolajczyk, G. Brostow, T.-K. Kim, S. Zafeiriou (Eds.)
Pages 662.1–662.13, BMVA Press, 2017.

5. M. Stoll, S. Volz, D. Maurer, A. Bruhn
A Time-Efficient Optimisation Framework for Parameters of Optical Flow Methods.
In *Proc. 20th Scandinavian Conference on Image Analysis*
SCIA 2017, Tromsø, Norway, June 2017 – P. Sharma, F. M. Bianchi (Eds.)
Lecture Notes in Computer Science, Vol. 10269, 41–53, Springer, Berlin, 2017.

6. D. Maurer, M. Stoll, A. Bruhn
Order-Adaptive Regularisation for Variational Optical Flow: Global, Local and in Between.
In *Proc. 6th International Conference on Scale Space and Variational Methods in Computer Vision*
SSVM 2017, Kolding, Denmark, June 2017 – F. Lauze, Y. Dong, A. B. Dahl (Eds.)
Lecture Notes in Computer Science, Vol. 10302, 550–562, Springer, Berlin, 2017.

7. D. Maurer, M. Stoll, S. Volz, P. Gairing, A. Bruhn
A Comparison of Isotropic and Anisotropic Second Order Regularisers for Optical Flow.
In *Proc. 6th International Conference on Scale Space and Variational Methods in Computer Vision*
SSVM 2017, Kolding, Denmark, June 2017 – F. Lauze, Y. Dong, A. B. Dahl (Eds.)
Lecture Notes in Computer Science, Vol. 10302, 537–549, Springer, Berlin, 2017.

8. O. Demetz*, M. Stoll*, S. Volz, J. Weickert, A. Bruhn
Learning Brightness Transfer Functions for the Joint Recovery of Illumination Changes and Optical Flow.
In *Proc. 13th European Conference on Computer Vision*
ECCV 2014, Zurich, Switzerland, September 2014 – D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.)
Lecture Notes in Computer Science, Vol. 8689, 455–471, Springer, Berlin, 2014.
(*) Authors have equally contributed to the publication.

9. M. Stoll, S. Volz, A. Bruhn
Joint Trilateral Filtering for Multiframe Optical Flow.
In *Proc. 20th IEEE International Conference on Image Processing*
ICIP 2013, Melbourne, Australia, September 2013
IEEE Computer Society Press, 3845-3849, 2013.

10. M. Stoll, S. Volz, A. Bruhn
Adaptive Integration of Feature Matches into Variational Optical Flow Methods.
In *Proc. 11th Asian Conference on Computer Vision*
ACCV 2012, Daejeon, Korea, November 2012 – K. M. Lee, J. Rheg, Y. Matshushita, Z. Hu (Eds.)
Lecture Notes in Computer Science, Vol. 7727, 1–14, Springer, Berlin, 2013.
Oral presentation at the ACCV.

C.2 Others

1. K. Kurzhals, N. Rodrigues, M. Koch, M. Stoll, A. Bruhn, A. Bulling, D. Weiskopf
Visual Analytics and Annotation of Pervasive Eye Tracking Video.
In *Proc. ACM Symposium on Eye Tracking Research and Applications.*
ETRA 2020, Stuttgart, Germany, *accepted for publication.*

2. K. Kurzhals, M. Stoll, A. Bruhn, D. Weiskopf
FlowBrush: Optical Flow Art.
In *Proc. Symposium on Computational Aesthetics, Sketch-Based Interfaces and Modeling, and Non-Photorealistic Animation and Rendering*
EXPRESSIVE 2017, Los Angeles, USA, July 2017 – H. Winnemoeller, L. Bartram (Eds.)
ACM Digital Library, 2017.
Awarded an EXPRESSIVE 2017 Best Paper Award.

3. M. Stoll, R. Krüger, T. Ertl, A. Bruhn
Racecar Tracking and its Visualization Using Sparse Data.
In *Proc. IEEE VIS Workshop on Sports Data Visualization*
VIS 2013, Atlanta, USA, October 2013 – R. Basole, E. Clarkson, A. Cox, C. Healey, J. Stasko, C. Stolper (Eds.)
IEEE Computer Society Press, 2013.

4. S. Metzger, M. Stoll, K. Hose, R. Schenkel
LUKe and MIKE: Learning from User Knowledge and Managing Interactive Knowledge Extraction.
In *Proc. 21st ACM International Conference on Information and Knowledge Management*
CIKM 2012, Maui, USA, Oct./Nov. 2012 - X.-W. Chen, G. Lebanon, H. Wang, M. J. Zaki (Eds.)
Demo Paper, ACM Press, 2671-2673, 2012.