

Automated Analysis of Time-resolved X-ray Data using Optical Flow Methods

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften /
Doktors der Naturwissenschaften

der Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

vorgelegte

Dissertation

von

Alexey Ershov

aus Jakutsk

Tag der mündlichen Prüfung: 13 November 2015

Erster Gutachter: Prof. Dr. Rüdiger Dillmann

Zweiter Gutachter: Prof. Dr. Tilo Baumbach

Abstract

X-ray imaging is a genuine tool to reveal internal structures of opaque objects. This is possible due to the penetration properties of its probe - X-ray radiation. Modern synchrotron facilities, equipped with high-resolution detector systems, provide X-ray radiation of unique quality and allow to investigate a broad range of *dynamical* processes, both in materials and biological specimens. To perform *automated* and *quantitative* analysis of *time-resolved* X-ray data, a method capable to retrieve dynamical information is required. In this work we develop a general-purpose framework for X-ray data analysis based on *optical flow*.

Optical flow methods traditionally belong to the field of Computer Vision. Finding correspondences between time-lapse images is a key problem in a variety of applications such as robot vision, tracking systems and video analysis. In the scope of this work we adapt *variational* optical flow methods - a specific class of approaches used to determine the optical flow - to the task of X-ray data analysis.

The quality of time-resolved X-ray data is diverse, ranging from high-resolution datasets to low-contrast, noisy images with artifacts. We provide a detailed classification of X-ray data. This taxonomy serves as a reference point for the development of image preprocessing, motion estimation and data analysis techniques. Image preprocessing is employed to enhance the original (raw) X-ray data in order to improve the accuracy of optical flow estimation for the case of challenging data.

To develop an accurate and *robust* motion estimation model, we perform a systematic evaluation of state-of-the-art optical flow techniques and make quantitative performance analysis of their components.

On the top of the optical flow estimation we provide an extensive *data analysis toolkit* including automated tracking, flow analysis, motion-based segmentation, image registration and detection of temporal changes. All the devised techniques can be applied in 4D (3D + time) to enable analysis of tomographic data. The implementation of the developed techniques incorporates advanced numerical schemes and computations on GPU. Thereby, the processing of a vast amount of X-ray data is feasible.

Finally, we present the application of the optical flow methods to a number of scientific problems from various research fields. These examples include flow analysis and particle segmentation in semi-solid alloys, analysis of morphogenesis in living frog embryos, coalescence events estimation and stability studies during the foaming process, and tracking of morphological dynamics in living insects.

Zusammenfassung

Röntgen-Bildgebung ist ein wertvolles Werkzeug, um interne Strukturen von undurchsichtigen Objekten zu untersuchen. Dies wird durch die Penetrationseigenschaften der Röntgenstrahlung ermöglicht. Moderne Synchrotroneinrichtungen, die mit hochauflösenden Detektorsystemen ausgestattet sind, bieten eine besondere Qualität der Röntgenstrahlung und erlauben damit, ein breites Spektrum von dynamischen Prozessen sowohl in Materialien als auch in biologischen Proben zu untersuchen. Um eine automatische und quantitative Analyse von zeitaufgelösten Röntgendaten durchzuführen, benötigt man eine Methode, mit der sich dynamische Information aus solchen Daten gewinnen lassen. In dieser Arbeit wird ein universelles Netzwerk für die Analyse von Röntgendaten entwickelt, das auf dem optischen Fluss basiert.

Methoden des optischen Flusses werden traditionell dem Gebiet Maschinelles Sehen (engl. Computer Vision) zugerechnet. Die Suche nach Zusammenhängen zwischen zeitaufgelösten Bildern ist das Kernproblem in verschiedenen Anwendungsgebieten wie etwa Robot Vision, Objektverfolgung und Videoanalyse. Die variationalen Methoden gehören zu einer spezifischen Klasse von Ansätzen, um den optischen Fluss zu bestimmen. Im Rahmen dieser Arbeit passen wir die variationalen Methoden des optischen Flusses an, um die Analyse von Röntgendaten zu erleichtern.

Die Qualität von zeitaufgelösten Röntgendaten unterscheidet sich sehr stark von hochqualitativen und hochauflösenden Datensätzen bis hin zu kontrastarmen verrauschten Bildern mit Artefakten. Wir stellen eine detaillierte Klassifizierung von Röntgendaten bereit. Diese Taxonomie dient als Bezugspunkt für die Entwicklung von Bildvorverarbeitungs-, Bewegungsabschätzungs- und Datenanalysetechniken. Die Bildvorverarbeitung

wird zur Verbesserung von ursprünglichen (rohen) Röntgendifferenzdaten eingesetzt, um die Qualität der Bestimmung des optischen Flusses im Falle von anspruchsvollen Daten zu erhöhen.

Um ein akkurate und robuste Modell der Bewegungsabschätzung zu entwickeln, führen wir eine systematische Auswertung der modernsten Techniken des optischen Flusses und eine quantitative Leistungsanalyse ihrer Komponenten durch.

Basierend auf den Methoden des optischen Flusses entwickeln wir ein umfangreiches Werkzeugset zur Datenanalyse, das automatische Objektverfolgung, Flussanalyse, bewegungsbasierte Segmentierung, Bildregistrierung sowie die Detektion von zeitlichen Veränderungen enthält. Alle in diesem Zusammenhang entwickelten Methoden können für die Analyse von zeitaufgelösten tomographischen Daten (4D) verwendet werden. Die Implementierung dieser Techniken beinhaltet fortgeschrittene numerische Schemata und Berechnungen auf Grafikprozessoren (GPUs), wodurch die Verarbeitung von sehr großen Röntgendifferenzdatensätzen ermöglicht wird.

Schließlich zeigen wir die Anwendung der Methoden des optischen Flusses anhand mehrerer wissenschaftlicher Fragestellungen aus verschiedenen Forschungsgebieten. Beispiele sind die Flussanalyse und die Segmentierung von Partikeln in halbfesten Legierungen, die Analyse der Gestaltbildung in lebenden Froschembryonen, die Beurteilung von Koaleszenz und Stabilität während der Schaumaufbereitung sowie die Verfolgung dynamischer Prozesse in lebenden Insekten.

Acknowledgements

I would like to thank Prof. Tilo Baumbach for giving me an opportunity to work with a team of dedicated scientists who use X-rays to reveal the hidden structure of things, and for his constant support.

I want to express gratitude to Prof. Rüdiger Dillmann for supervising this work in my beloved field of Computer Science.

The interaction with Anton Myagotin and Artem Starostin was a pivotal point for me to start my work in the fields of X-ray Imaging and Image Analysis. Anton was the first one who successfully used optical flow methods for the analysis of time-lapse X-ray images of metal foams.

Also I want to thank Prof. Joachim Weickert and Prof. Andrés Bruhn for introducing me to elegant and very useful variational optical flow methods.

This work would not be possible without insightful and fruitful collaboration with many scientists from various research fields. Among them are Anton Myagotin, Lukas Helfen, Alexander Rack, Simon Zabler, Ralf Hofmann, Jubin Kashef, Julian Moosmann, Venera Weinhardt, Thomas van de Kamp, Tomy dos Santos Rolo, Tomáš Farago, Dmitry Karpov, Martin Köhl and Tilo Baumbach. It was a pleasure to work with them. I have learned so much along the way.

Many thanks goes to Martin Köhl for proofreading and useful feedback. I incorporated almost all his suggestions.

Igor Karlinskiy made a great contribution by helping me to implement optical flow methods on GPUs and visualization of 3D vector fields. He also assisted to create good-looking diagrams and analysis plots.

I appreciate many pleasing conversations, disputes and our fun teamwork with Dmitry Karpov.

And last but not least, I want to thank my family for their patience and infinite support.

Contents

1	Introduction	11
1.1	Motivation	11
1.2	Outline	12
1.3	X-ray Imaging	13
1.3.1	Introduction to X-ray Imaging	13
1.3.2	Experimental Methods	17
1.4	Variational Optical Flow	20
1.4.1	Related Work	20
1.4.2	General Model	22
1.5	Aims of the Work	24
2	Optical Flow Methods	27
2.1	Modeling	27
2.1.1	Quality Measures	27
2.1.2	Data Constancy Assumptions	29
2.1.3	Robust Data Assumptions	33
2.1.4	Smoothness Assumptions	35
2.1.5	Measures of Confidence	38
2.1.6	Generalization to 3D Optical Flow	42
2.2	Numerical Solution	42
2.2.1	Minimization Procedure	42
2.2.2	Discretization	43
2.2.3	Iterative Methods	45
2.2.4	Large Displacements	46
2.2.5	Multi-level Computation	49
2.2.6	Intermediate Flow Filtering	49
2.3	Summary	50
3	Data Analysis Framework	51
3.1	Data Preprocessing	51
3.1.1	Noise Filtering	52
3.1.2	Brightness Correction	57
3.1.3	Contrast Adjustment	60
3.2	Analysis Based on Optical Flow	62
3.2.1	Motion Analysis	62
3.2.2	Object Tracking	64
3.2.3	Image Registration and Alignment	67

4 Computational Framework	73
4.1 Software Framework	73
4.1.1 Modules	73
4.1.2 Components	76
4.2 High Performance Computing	79
4.2.1 Numerical Schemes	79
4.2.2 GPU Computing	80
4.3 Visualization	85
5 Evaluation on Synthetic Data	87
5.1 Data Taxonomy	87
5.1.1 Image Quality	88
5.1.2 Data Features	89
5.1.3 Image Artifacts	92
5.1.4 Motion Model	93
5.1.5 Results Characterization	95
5.2 Experimental Data	97
5.2.1 Synthetic Datasets	97
5.2.2 Modeling Properties of X-ray Data	101
5.3 Optimization of Parameters	108
5.3.1 Optimization of Individual Parameter	108
5.3.2 Confidence-based Optimization	112
5.3.3 Baseline Method	112
5.4 Quantitative Evaluation	112
5.4.1 Experiments: Noise	113
5.4.2 Experiments: Preprocessing	118
5.4.3 Experiments: Artifacts	123
5.4.4 Experiments: Confidence Measures	131
5.5 Models Comparison	134
5.6 Evaluation Summary	137
6 Applications in X-ray Imaging	141
6.1 Motion Analysis	141
6.1.1 Particles Fluid Dynamics in Semi-Solid Metals	141
6.1.2 Analysis of Morphogenesis in Frog Embryos	150
6.2 Temporal Changes Detection	160
6.2.1 Stability Study During Foaming Process	160
6.3 Tracking	171
6.3.1 Kinematics Analysis of Joint Parts in Living Insect	171
7 Summary and Outlook	179
7.1 Conclusions	179
7.2 Further work	181
A List of Publications	183

Chapter 1

Introduction

In this chapter we give an introduction to our work, discuss the possibilities of time-resolved X-ray imaging methods and arising problems for data analysis. The imaging principles and application fields are shortly described for radiography and tomography experiments. Further we give an introduction to optical flow methods and a general mathematical model behind their principle. We proceed with an overview of related work on the optical flow methods. In the end of the chapter we state the aims of this work.

1.1 Motivation

The discovery of physical phenomena such as X-rays, ultrasound, radioactivity, magnetic resonance, and the development of imaging instruments based on their principles, has revolutionized the fields of medical imaging and materials research. Among other techniques, X-ray imaging is recognized as a genuine tool to image the internal structure of opaque objects, which is possible due to the penetration properties of X-ray radiation. X-rays usually do not require special sample treatment and allow to use dedicated sample environments for *in-situ* experiments. New methods emerged with the development of synchrotron radiation X-ray sources. Modern synchrotron facilities, equipped with high-resolution detector systems provide intense radiation and high spatial resolution. This allows to investigate a broad spectrum of dynamical processes in materials and biological samples.

Conducting an *in-situ* experiment, triggering the process of interest, and then collecting time-lapsed data, enables to visualize and study dynamical phenomena. After the X-ray experiment is performed, all underlying events and internal changes are captured in a sequence of radiographic images or tomographic volumes. This makes time-resolved X-ray imaging a powerful tool that offers the exciting possibility to observe real-time dynamics and reveal internal structural information.

A vast variety of X-ray imaging methods, experiment types and sample forms give rise to a numerous challenges in the subsequent data analysis. Application examples of time-resolved X-ray imaging include evolution of morphological structures, monitoring of fabrication processes, investigation of materials under mechanical stress or heat treatment, velocimetry of liquid flows and diffusion processes, examination of living organisms and many other studies from various research fields .

The common point of interest of these problems is the dynamics of a process. From

the data processing perspective we can identify several topics of interest for time-resolved data analysis. First, it is *motion analysis*, which includes the analysis of various qualitative and quantitative characteristics of the moving scene or objects, such as motion type, existence of sources and sinks, velocity or acceleration profiles. Another task is to perform the detection of objects by the analysis of their motion, the so-called *motion-based segmentation*. The next typical task is *object tracking*, which is a procedure for estimation of the trajectory of an object as it changes its position over time. To perform such a broad spectrum of data analysis tasks, a method capable of retrieving temporal changes and capturing dynamical information is required. In this work we present a general-purpose framework for time-resolved data analysis based on *optical flow*.

Optical flow methods traditionally belong to the field of Computer Vision. A problem of finding correspondences between time-lapse images is useful for a variety of applications such as robot vision, tracking systems, video analysis and stereo reconstruction. The application of optical flow methods become increasingly important in other fields, such as medical imaging, materials research and life sciences. In this work we adapt optical flow methods to the task of time-resolved X-ray data analysis and show that these methods are well suited for a broad range of scientific problems. We restrict ourselves to a specific class of approaches that can be used to determine the optical flow, the so-called *variational methods*. These methods are recognized among the currently most accurate, flexible and robust techniques for the optical flow computation [BSL⁺11, SRB14].

1.2 Outline

This work is organized in the following way.

In the beginning of Chapter 1 we shortly discuss the possibilities of time-resolved X-ray imaging methods and arising data analysis problems. X-ray imaging principles will be described shortly for radiography and tomography experiments. Additionally, we discuss their advantages and shortcoming with respect to time-resolved imaging. Further we give a short introduction to optical flow methods and describe the related work. In the end of the chapter we state the aims of this work.

Chapter 2 is dedicated to an extensive description and evaluation of optical flow methods. Quality and confidence measures for motion estimation are introduced. We present different approaches for the construction of variational models and discuss how these techniques can be applied for particular types of X-ray data. We proceed with an extension of the devised 2D methods into a 3D model, which enables us to perform data analysis of tomographic data. Next, we explain how a variational functional can be solved numerically using iterative methods. Furthermore, we discuss various computational aspects such as the coarse-to-fine strategy.

In Chapter 3 we consider important topics of data preprocessing and motion analysis. Several strategies will be presented to cope with the most important data challenges such as image noise filtering, correction of inhomogeneous brightness and enhancement of features contrast. The influence of the proposed filtering techniques on the performance of optical flow methods results will be discussed in detail. Then, we switch to the analysis of the computed flow field and temporal data. We cover data analysis methods including advanced flow analysis, automated tracking, motion-based segmentation, image registration and detection of temporal changes. All the presented techniques will be tested on

synthetic data in the experimental part of the work (see Chapter 5) and employed in the part devoted to applications (see Chapter 6).

In Chapter 4 we describe our computational framework. First, we present our software modules and describe their components and responsibilities. Then, we discuss an important topic of efficient and high performance computation of the optical flow. A number of optimization strategies will be presented. Next, we provide an efficient implementation of the optical flow algorithms on the GPU computing architecture. The last section of the chapter is dedicated to visualization methods. Here we present our own implementation of vector fields visualization, which allows for interactive inspection of the optical flow results.

In Chapter 5 we perform systematic numerical experiments on synthetic data, designed to test the performance of computational methods. In the beginning of the chapter we provide a taxonomy of X-ray data. We present the quantitative evaluation of the performance, influence of preprocessing, modeling of noise and data outliers in dedicated sections. For all the experiments we outline important observations and draw the respective conclusions, which we summarize at the end of the chapter.

In Chapter 6 we apply our framework to various scientific examples from different research fields. These application examples include flow analysis and particle segmentation in semi-solid alloys, analysis of morphogenesis in living frog embryos, estimation of coalescence events and stability studies during the foaming process in various kinds of foams, and tracking of morphological dynamics in a living insect. For each application we give a short description of a problem and specify our data analysis tasks. We evaluate the dataset according to our data taxonomy. Then we provide a detailed discussion on the choice of the appropriate data preprocessing routines, selection of a suitable optical flow model and the selection of the specific data analysis technique, which will lead to the solution of the declared scientific aim.

In Chapter 7 we summarize the main results of our work and give our conclusions about the developed techniques. We also discuss a number of possible extensions and promising ideas which might further improve the results and extend our capabilities for analysis of challenging X-ray data.

1.3 X-ray Imaging

1.3.1 Introduction to X-ray Imaging

X-rays were discovered by W. C. Röntgen on 8 November 1895 [R96]. In 1901 for this discovery and his investigations he obtained the first Nobel Prize in Physics. X-rays are electromagnetic radiation with a wavelength in the range of 0.01 to 10 nanometers, which corresponds to frequencies in the range 3×10^{16} Hz to 3×10^{19} Hz and energies in the range 100 eV to 100 keV.

Due to their penetrating ability, hard X-rays (with photon energies between 10 - 100 keV) are an invaluable tool to image the internal structure of optically opaque objects. Already shortly after the discovery, X-ray imaging became widely used for medical applications, non-destructive testing and security systems. During propagation through the matter the intensity of X-rays is attenuated along their path of propagation. For 2D radiography imaging, the variations in thickness or properties of materials provide a con-

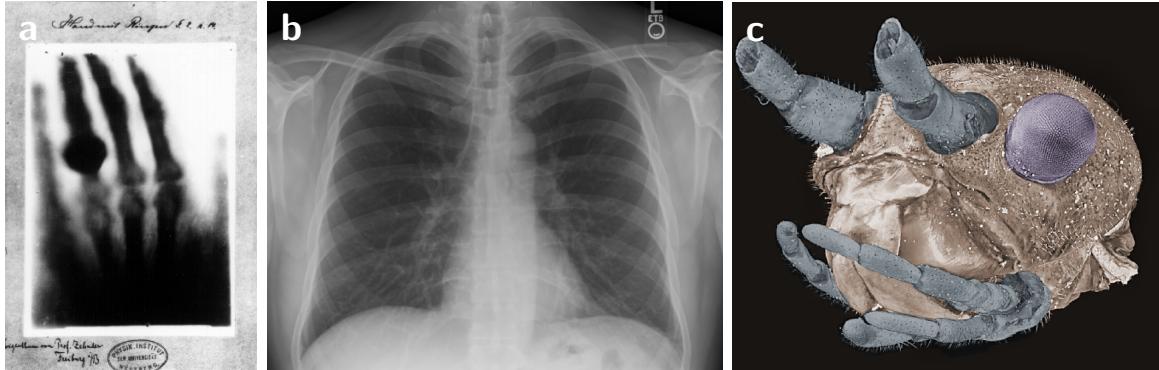


Figure 1.1: Evolution of X-ray imaging techniques. (a) First medical X-ray by Wilhelm Röntgen of his wife’s hand. Image: Wikipedia, public domain. (b) X-ray radiography of a chest. Image: Wikipedia, public domain (c) Volume rendering of the head of the stick insect *Peruphasma schultei* imaged by a high resolution microtomography [vdKEdSR⁺13a].

trast to resolve different structures. However, in this case one obtains only a projected image and loses any depth information due to superimposed structure details.

Computed tomography (CT) overcomes this limitation and provides a three-dimensional information about the internal structure. To achieve this, a series of projections from different angles are acquired. Then, a reconstruction algorithm is used to yield a 3D digital representation of an object. The mathematical foundations for tomographic reconstruction were established by J. Radon in 1917 [Rad86] and the CT technology was developed by G. Hounsfield and A. Cormack in 1960s-1970s. Since then the computed tomography has revolutionized the fields of medical imaging and materials research.

New methods and imaging techniques emerged with the development of *synchrotron radiation* sources. This radiation is generated using charged particle accelerators. Synchrotron radiation is emitted by the charged particles which are accelerated by magnetic fields on a circular trajectory in a storage ring.

Modern 3rd-generation synchrotron radiation sources, equipped with high-resolution detector systems provide small beam size, high radiation intensity and high spatial resolution. With the introduction of various x-ray optics the beam quality can be controlled, providing polychromatic or monochromatic radiation. Synchrotron sources are expensive, not portable and usually are operated by large-scale facilities, which could limit a broad use of this technology. However, the unique quality of its probe make them a powerful tool for a wide range of X-ray experiments and applications.

X-ray Imaging Principles

X-rays interact with matter either by *photoelectric absorption* or *scattering* (Compton and Rayleigh). An absorption effect occurs when the X-ray photon is absorbed in the course of liberating an electron from the inner shells of an atom. This contributes to the radiation dose defined as the amount of absorbed energy per unit of mass. This aspect is crucial for imaging of biological specimens. For a Compton scattering only a part of the X-ray photon energy is used to free an electron from an outer shell and the photon itself changes its direction, i.e. scatters [Dou09].

As a result of these interactions the intensity of the incoming beam is reduced. Various

tissues or materials affect the changes in radiation intensity differently. In a homogeneous object this depends on its thickness l along the direction of propagation and the attenuation coefficient μ of the material. The intensity I of the incident X-ray beam after propagation through the sample of thickness l is related to its initial intensity I_0 by the following expression:

$$I = I_0 e^{-\mu l},$$

which is known as a Beer - Lambert law.

Depending on the main effect of X-ray interaction with matter it is possible to classify imaging techniques into *absorption imaging* and *phase-sensitive imaging*. For the absorption imaging the contrast is produced by the attenuation of x-rays and is described as an amplitude modulation of the wave-field. In the case of phase-sensitive imaging, contrast is governed by the phase modulations due to the differences in spatial distribution of the refractive index. A number of phase sensitive imaging techniques might be used. A simple and popular method exploits the effect of propagation of the X-ray wave-field after its interaction with the sample. Due to interference effects caused by the propagation, the phase modulation can be reconstructed and visualized. For this technique a quantitative retrieval of the phase modulations is possible [CLB⁺99], which allows to extract the real part of the refractive index in the interior of the investigated sample.

Beam Geometry

Depending on the instrumental implementation of a X-ray source a number of different beam geometries are possible:

- *Pencil beam.* The first X-ray source was implemented using a so-called pencil beam. For this geometry the cone of generated X-ray beams is filtered by a pinhole and, thus a point-like X-ray source is produced. A single detector is placed to the opposite side of the source to detect the X-rays.
- *Fan beam.* In this case the beam geometry is represented by a flat fan shape (divergent beam) and a detector system consists of an array of detector elements. Typically, the angle of beam divergence is small, as well as the size of the detection array.
- *Cone beam.* In this geometry the beam is a 3D cone with a wide angle of divergence and a large 2D detector is used. This technique is widely used in a laboratory-based X-ray setups. For this type of geometry the distance between the sample and the source is important, since it influences the object magnification in the detector plane. Additionally, for image reconstruction algorithms the angular distribution of the incoming rays with respect to the rotation axis has to be taken into account [KS01].
- *Parallel beam.* The X-rays are coming parallel to each other and impinge on the sample perpendicular to its rotation axis. The 3D image reconstruction procedure for this type of geometry is the simplest and can be realized in an efficient way [KS01].

In the current work all experimental results presented in the application sections are obtained using the parallel beam geometry. The most important aspect related to data analysis is the quality of 3D reconstructed images. Each beam geometry is associated with its own set of reconstruction artifacts [Buz08]. Therefore, the beam geometry, experimental parameters and the reconstruction algorithm strongly influence the later stages of data processing and analysis. We discuss these aspects in the Section 1.3.2.2 devoted to tomographic reconstruction.

Detector System

After the interaction with an object and propagation to the detector plane, X-rays are registered by a detector system. Two distinct classes of detectors are used in synchrotron radiation imaging [GET01]:

- *Direct photon detectors.* For this type of detector systems X-ray photons are converted directly into an electric charge.
- *Indirect detectors.* For the indirect detection system the incident photons first strike a material layer, called scintillator, which converts X-ray photons into visible-light photons. These photons are then detected by a visible-light detector.

For each detector system a number of technologies are available, such as charge-coupled devices (CCD) and complementary metal-oxide-semiconductor (CMOS) cameras. A substantial advantage of the indirect detector system is that a wide range of commercially available high-performance, visible-light cameras is available. With the recent developments such cameras can achieve a very high spatial resolution (down to $1.0 \mu\text{m}$) and ultra fast data acquisition rates (up to 100.000 frames per second) [RGMS⁺10].

Digital Image

Spatial resolution is the capability of an image to represent object details. It can be characterized as the minimum separation distance of nearby features of the object which can still be recognized on the image as different structures. In general, the spatial resolution of a digital image depends on the resolution of the imaging system described by its point spread function (PSF) and the pixel size of the detector system [Dou09]. The *pixel size* of the digital detector is determined by the sampling frequency used to digitize the image signal and by the physical size and separation of the array elements within a 2D detector. Increasing the pixel size reduces the effective resolution, but can lead to the increased signal sensitivity. The *point spread function* describes the response of the imaging system given a point source or a point object. The width of the PSF function determines the size of the smallest detail that can be resolved. According to the sampling theorem [Nyg28], the pixel size should be less or equal half the width of the point spread function in order to correctly sample an analog signal. For smaller pixel sizes the PSF spreads to several pixel locations and the resolution is reduced.

Another important characteristics of the digital image is the *brightness resolution*. In a digital image differences between structures are represented by variations in brightness values. In X-ray imaging, these variations are determined by or related to the physical properties of a sample, such as sample thickness, material density and chemical composition. The aim of an imaging process is to distinguish between various parts of the

sample, which differ in their structure or material composition. Ideally, this should also be valid if both materials differ from each other only slightly. To achieve this a detector system with a good brightness resolution should be used. This property is also called the *dynamic range* of the detector system. The range of brightness variations which is recorded depends also on the number of bits used for quantization procedure (converting the analog signal to digital). Most modern camera systems allow to record images in 12, 16 or 32-bit value formats. We discuss further aspects related to the brightness resolution in the Section 3.1.3, which is dedicated to preprocessing of low-contrast image data.

1.3.2 Experimental Methods

In this section we briefly describe two imaging techniques, which will be employed in this work to acquire a sequence of time-lapsed X-ray images - radiography and tomography. For both experimental techniques we describe the advantages and shortcomings of its application with respect to its ability to capture spatio-temporal information. Moreover, we introduce typical challenges that arise during data processing and optical flow computation.

1.3.2.1 Radiography

Radiography is a simple, yet powerful technique to visualize internal structures of a sample. It captures differences in attenuation properties or phase variation of X-rays coming through the sample. For this purpose a sample is placed in the beam and properly aligned using manipulation devices (e.g. translation stages), so the optimal position in the field of view is achieved. As soon as the process of interest is triggered, a high speed camera records time-lapse projection images, called "radiograms". Many instrumental modifications for the radiography experiment setup are possible. They include the choice of the X-ray beam properties, positioning devices, source-to-sample and sample-to-detector distances, detector optics and the choice of digital camera or sensor.

We now describe a number of characteristic properties associated with the radiography technique. Among the advantages of this method are:

- High spatio-temporal resolution: Spatial resolution down to the micrometer scale and temporal resolution down to the microsecond range are achievable with modern detector systems [RGMS⁺¹⁰].
- In most cases no complicated alignment of the sample is needed - the sample is simply placed in the field of view of the detector system.
- Since the sample stage does not move during the data acquisition, the usage of complex or bulk sample environments is possible (e.g. furnace, high-pressure chamber, etc).

However, there are also a number of major drawbacks, that limit applicability of the radiography:

- The main drawback of the radiography technique originates from the projection of 3D structures on a single 2D image plane. As a result, any depth information about internal 3D structures is inevitably lost.

- For data processing methods, including correspondence problems (e.g. optical flow, object recognition), the superposition of 3D structures into a 2D plane imposes a critical challenge - the underlying assumptions on data constancy and the affine transformation model are no longer valid. For example, the change in orientation of the object in 3D space causes a redistribution of the projected thickness on the image plane. Thus, no reasonable assumption about the image data can be enforced. In some situations the movement in 3D space does not cause a change in the distribution of grey values in the projected 2D image. One example is an object, which undergoes a translational movement perpendicular to the projected plane and the parallel beam geometry is used.

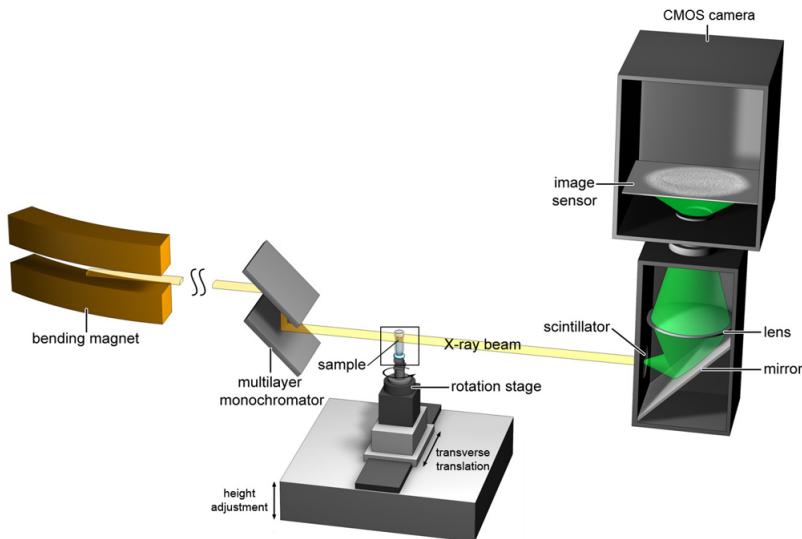


Figure 1.2: Experimental setup for propagation-based phase-contrast X-ray microtomography. A photon beam is generated by the synchrotron. After the beam is shaped and filtered by the monochromator, the X-rays propagate over a large distance to interact with the sample mounted on a rotation stage for tomographic data acquisition. The 2D detector system consists of a scintillator, which converts X-rays into a visible light, a mirror, a lens and a CMOS digital camera. Image: [MEA⁺13]

1.3.2.2 Computed Tomography

The previously described radiographic technique suffers from a major drawback - it only produces a projection of three-dimensional objects on the 2D detector plane. As a result the structures are superimposed with each other, which complicates their identification and analysis. Computed Tomography (CT) is a technique which has been developed to overcome this limitation. It provides a three-dimensional information about the inner structures of an object. This technique consists of two stages. First, a series of radiographic projections from different directions are taken. Then, a dedicated algorithm is used to reconstruct a 3D volumetric image. The task for tomographic reconstruction is to perform the retrieval of the unknown object function from a series of its Radon transformations taken from different angles [KS01]. There are two distinct classes of tomographic reconstructions:

- *Analytical methods* are based on reconstruction steps using the Fourier transformation [KS01]. It is important to note, that such methods yield an exact solution of the inverse problem if a number of requirements are fulfilled. For large tomographic datasets these methods are preferred since they are typically less computationally demanding. Moreover, they are also less sensitive to noise and small data artifacts, thus producing more accurate results. One popular choice for such type of techniques is the *filtered back-projection* (FBP) [KS01].
- *Algebraic methods* represent an object as an array of unknown coefficients and model a ray transmission through the object as a mathematical function of these coefficients. Then a system of algebraic equations is solved to estimate the object. Usually this class of reconstruction methods is used if the requirements for the analytical methods are not satisfied. For instance, if the number of angular projections is insufficient. Or if the beam geometry and tomographic acquisition is hard to be modeled analytically, e.g. for diffraction tomography.

A typical tomography setup is shown on Figure 1.2. Here we list advantages and drawbacks of the computed tomography technique. Among the advantages are:

- Computed tomography provides a 3D structural information about a sample.
- For the computation of optical flow having a 3D representation of a scene and its objects means that there are no occlusions between different structures (as it happens for 2D images). This highly influence the accuracy of motion estimation and leads to better results.

Nevertheless, there are also a number of limitations:

- The need of tomographic acquisition, which involves imaging of an object from multiple directions, can limit temporal resolution. This is especially critical for imaging rapid dynamical processes.
- The rotation of the sample on the tomographic stage could also restrain the use of sample environments (wiring, bulky instrumentation, etc), which cannot be rotated freely around an axis.
- To obtain tomographic volumes of a high quality, a large number of radiographic projections is required. Depending on the sample and properties of the X-ray radiation, it may result in a high dose deposition. This aspect is crucial for imaging living specimens and can limit the applicability of X-ray computed tomography.
- Depending on the number of projections, image quality and specific image artifacts could appear and degrade the resulting 3D data. Compared to such image characteristics, such as noise and low contrast, these artifacts are difficult to correct. The presence of reconstruction artifacts significantly reduces the performance of optical flow methods.

1.4 Variational Optical Flow

Optical flow is one of the major problems in *Computer Vision*. Given a sequence of images the aim is to determine the correspondences between them. A method to find the displacement field which maps all image points from the first frame to their new locations within the second frame is called *optical flow*. Following a formulation from the seminal work of Horn and Schunck [HS81], the motion field is the 2D projection of the actual 3D motion of objects and scene, where the optical flow is the *apparent* motion of the brightness patterns depicted on the image. In our work we extend the definition of optical flow as an actual 3D motion of internal structures when applied to 3D tomographic volumes.

Optical flow is useful for a variety of different applications such as robot vision, driver assistant systems, object tracking and stereo reconstruction, video analysis and processing [Wol96, SL96, KZK03, HTWM04]. In this work we employ optical flow methods to perform data analysis and investigate of a wide range of applications related to the field of X-ray imaging. One example is given on Figure 1.3, where optical flow is used to capture real-time movements of the feeding cockroach.

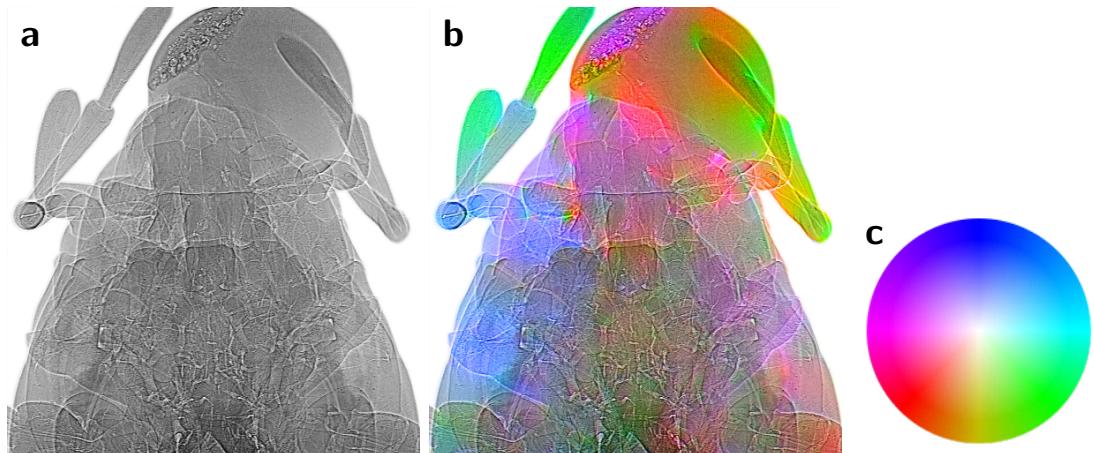


Figure 1.3: Fast radiography of the feeding cockroach *Periplaneta americana*. (a) First frame of the radiographic sequence (the background is removed and contrast is adjusted). (b) Computed flow field, which captures the movements of the insect. (c) Color coding: color represents direction and its brightness represents flow magnitude.

1.4.1 Related Work

The research on optical flow estimation advances is rapidly. The methods become more accurate, robust with respect to challenging data and flexible to model various scene and motion scenarios. In the scope of this work we restrict ourselves to a specific class of approaches used to determine the optical flow, so-called *variational methods*. These methods are recognized as one of the most accurate and best understood techniques for the optical flow computation [BSL⁺11, SRB14].

Recently a number of important contributions to the field of optical flow estimation were made. In the work of Bruhn [Bru06] a systematic taxonomy and quantitative evaluation of variational optical flow methods was presented. A common notation as well as an

extensive list of available models allows to construct an optical flow framework, suitable for a wide range of applications. Development of a novel database for the evaluation of optical flow techniques has led to rapid improvements and understanding of optical flow methods [BSL⁺11]. A vast amount of literature on motion estimation techniques propose sophisticated models and multiple features. This makes it hard to evaluate the performance of the individual components. In Refs. [SRB10, SRB14] the authors provide an overview over current state-of-the-art optical flow methods and reveal the "secrets" behind the most successful models. Recently, some of the authors reported that even the most advanced approaches yield poor results on a challenging real-life data. This highlights the importance of work on robust optical flow methods designed for a particular task or application. A number of application specific optical flow evaluation databases has already emerged [AVBSZ07, GLU12].

Many of the today's state-of-the-art variational optical flow methods still resemble the original formulation by Horn and Schunck [HS81]. This model combines a *data term* that imposes constraints on image features which should be matched and a *smoothness term* which regulates the properties of the flow field. A global energy functional containing both constraints is then minimized to find the solution. In this section we describe important work related to the modeling of data and smoothness assumptions, as well as the optimization procedure. We also give a short outlook of works where optical flow methods has been applied.

There are many approaches and interesting concepts which go beyond the scope of this work. For example, we exclude the discussion of photometric data terms constancy assumption based on color information [MBW07], motion with transparency or layered models [JB93, WA93, JBJ96], incorporation of feature descriptors (i.e. SIFT) to refine motion estimation during coarse-to-fine strategy [XJM10], combination of optical flow estimation and image segmentation [BJ96, ZJK05, XCJ08], or the use of non-convex optimization strategies. We will restrict ourselves only to those approaches which are suitable for the application on the X-ray image data and can be incorporated to the variational framework.

1.4.1.1 Data Assumptions

In the literature, there are two major directions for the formulation of data constraints: design of robust data terms, which improve the performance on problematic image data and incorporation of additional information to model various aspects, such as illumination changes and different motion models. The first approach for the robust data term was proposed in the work of Black and Anandan [BA91, BA96], where the quadratic function of the original Horn and Schunck method was replaced by a non-quadratic variant. The robust modeling of data constraints become an important factor in the construction of accurate optical flow methods [BBPW04, LRR08, XJM10, BSL⁺11, SRB14]. In order to handle varying illumination conditions, higher order constancy assumptions [Sch93, Sch94, BBPW04, PBB⁺06, ZBW11] or special image filtering are proposed [MBW07, WPZ⁺09, ZBW11]. To improve the performance of data constraints for noisy images some of the authors suggested a local-global data term [BWS02, BWS05], which takes into account the information around a local region.

1.4.1.2 Regularization

The regularization of flow fields involves two main aspects: discontinuity preserving motion estimation and incorporation of temporal information. The first approach to use adaptive smoothness was proposed by Nagel et al. [Nag83, NE86]. This idea has been developed in a number of approaches which assume image-driven [Sch93, ASW99] or flow-driven smoothness [SH89, Sch94]. In Ref. [SLB08] the authors combined the advantages of both methods to obtain precise motion discontinuities and avoid oversegmentation problems. In their work the smoothness direction is adjusted to image edges and smoothing amount to the flow gradient. In Ref. [ZBW11] the authors generalized this idea to obtain a *complementary smoothness* term. Such approach regulates the smoothness direction according to the data constraint, instead of image edges.

The idea to assume the smoothness also in temporal direction was first investigated by [MB87] and extended for continuous models in [Nag90]. A discontinuity-preserving spatio-temporal smoothness was proposed in [WS01b] and used for a number of highly accurate optical flow models [BBPW04, ZBW11, VBVZ11].

1.4.1.3 Solution

To find a solution of variational methods in the presence of large displacements it is common to use a coarse-to-fine strategy [BA96, MP98, BPPW04, XJM10]. For the construction of image pyramids and in order to perform image warping a bicubic interpolation is commonly used [SRB10]. In order to represent image gradients a second order derivatives, as well as temporal averaging are among good practices [WPZ⁺09, SRB14, ZBW11]. A number of efficient computation models were proposed, such as multigrid methods [BWF⁺05, BWKS06a]. Furthermore, a popular trend is a fast implementation of variational methods using graphical processing units (GPU) [WPZ⁺09, GZG⁺12, BJKY14].

1.4.1.4 Applications

Optical flow methods are widely used in the native field of Computer Vision, for such applications as video surveillance and analysis [KZK03, HTWM04], video compression [Wol96] and video annotation [SL96]. Recently, the usage of optical flow methods become important for data analysis in the fields of biological imaging and life sciences, as well as in other research fields. Examples include: zebrafish development [AMK13], monitoring of plant growth [BL97], in-vivo blood flow measurements [VLW07], modeling and correction of respiratory motion [MHSK13] and analysis of cardiac images [XPI⁺10].

Various parts of the current work were presented in publications, dedicated to X-ray imaging [BRE⁺09, RGMS⁺10, AER⁺12], applied for *in-situ* studies of advanced materials [ZRGM⁺10, MEH⁺12, ZER⁺13] and for *in-vivo* investigation of various biological questions [BRS⁺08, MEA⁺13, vdKEdSR⁺13b, MEW⁺14, dSREvdKB14].

1.4.2 General Model

A general model for variational methods was proposed by Horn and Schunck in 1981 [HS81]. This technique determines the unknown displacement field as a minimal solution of the *energy functional*. In general, such energy-based formulations are composed of two

parts: a data term which assumes constancy of specific image features, and a smoothness term which regularizes the spatial variation of the flow field.

The classical optical flow approach of Horn and Schunck consists of a *brightness constancy* assumption and a *homogeneous* smoothness term. The first constraint states that after the displacement the brightness value remains constant and the second constraint assumes that the motion field varies smoothly in all directions.

In order to formulate the optical flow model we consider a time-lapse sequence of images $I(x, y, t)$, where (x, y) are the pixel coordinates within a rectangular image domain $\Omega_2 \subset \mathbf{R}^2$ and t denotes time frame. Then, the brightness constancy assumption can be expressed by:

$$I(x + u(x, y), y + v(x, y), t + 1) = I(x, y, t), \quad (1.1)$$

where $(u(x, y), v(x, y))$ is the displacement field from frame t to $t + 1$. To simplify the formulation of model assumptions, we will use the short notation u, v for the displacement field components instead of $u(x, y), v(x, y)$. Assuming that the displacement components are small and changing continuously, we can approximate the non-linear equation (1.1) using a first order Taylor expansion. We obtain the so-called *linearized optical flow constraint*:

$$I_x u + I_y v + I_t = 0. \quad (1.2)$$

The optical flow cannot be uniquely computed at a single point since the displacement is given by two components, while the change in image brightness provides only one constraint. In the literature this is known as the *aperture problem* [BPT88]. There are two different cases of this issue. If only one component of the image gradient can be computed, it is possible to determine the flow component parallel to the direction of image gradient, so-called *normal flow*. In the second case, in homogeneous image regions the brightness gradient vanishes completely, so no appropriate correspondence can be found.

To overcome this problem, the method proposed by Horn and Schunck introduces an additional constraint which assumes the spatial smoothness of the flow field. Such constraint can be expressed by penalizing the spatial flow gradients, given by $\nabla_2 u$ and $\nabla_2 v$. Both, the brightness constraint and the smoothness constraint can be integrated in the energy functional to obtain the classical Horn and Schunck method:

$$E_{\text{HS}}(u, v) = \int_{\Omega_2} (I_x u + I_y v + I_t)^2 + \alpha(|\nabla_2 u|^2 + |\nabla_2 v|^2) dx dy, \quad (1.3)$$

where $|f| = \sqrt{f_x^2 + f_y^2}$ is the magnitude and $\nabla_2 = (\partial_x, \partial_y)$ denotes the spatial gradient. The parameter α is called *regularization* or *smoothness parameter* and controls the amount of required smoothness of the resulting flow field.

The variational optical flow methods based on the Horn and Schunck model can be written in a more general form:

$$E_{\text{general}} = \int_{\Omega_2} (D(I, u, v) + \alpha S(I, u, v)) dx dy,$$

where $D(I, u, v)$ imposes constraints on image features which should be matched and $S(I, u, v)$ regulates the smoothness properties of the flow field. Both parts of this general optical flow framework can be customized and adapted for the specific data properties and

motion models. In Chapter 2 we present different approaches to construct accurate and robust variational optical flow models and discuss how these techniques could be applied for the particular task of X-ray data analysis.

In order to find the solution of the optical flow problem we are interested to have energy functionals which are *convex*. For such functionals a *unique solution* exists and it can be found using *globally convergent* algorithms. In contrast, for non-convex approaches *multiple local minima* can be present and optimization algorithms may suffer from getting trapped in local minima, thus providing inaccurate results. For the Horn and Schunck method linearization of the brightness constraint must be performed in order to obtain a convex functional. Such linearization step does not affect the accuracy of the result as long as only small displacements for the scene are considered. In order to cope with large displacements linearization procedure cannot be applied and the original non-linear variant of the data term must be used. In Section 2.2.4 we will introduce a *coarse-to-fine strategy* based on the theory of warping which allows to solve non-convex energy functionals for our problem.

1.5 Aims of the Work

As it was discussed in the introductory section, time-resolved X-ray imaging methods provide a wide range of possibilities for the research of dynamical processes. It is a challenging task, however, and involves many steps to get from the initial idea of the experiment to the final scientific result. After an experimental setup is prepared and carefully adjusted, the original raw data is recorded by a detector system. The acquired data have to be inspected and evaluated to ensure that the desired image quality is achieved and the events of interest are captured. At this step important adjustments could be introduced to improve the results. The raw experimental data coming from the detector system is usually not suited for the immediate analysis, since images often contain artifacts. Thus, an artifact correction and data enhancement procedure is desirable. The quality of time-resolved X-ray data is immensely diverse. Data related properties, such as image resolution, signal-to-noise ratio, features contrast and underlying motion model, influence the performance of data processing routines. To automatically capture the dynamics of a process an appropriate computation algorithm should be employed. In this work we use *variational optical flow* as the core method to extract temporal information.

The main goal of our work is to cover, investigate and implement all the aspects that are important for automated analysis of time-resolved X-ray data. To achieve this we proceed with the following steps:

- provide a detailed classification of X-ray data, including the description of image quality, motion types and features of interest. This taxonomy should serve as a reference point for the development of image preprocessing, data analysis and motion estimation procedures
- develop image preprocessing methods to enhance the original (raw) X-ray data
- perform systematic evaluation of state-of-the-art optical flow techniques, make quantitative analysis of their components and choose the most robust and accurate approaches

- introduce a flexible framework based on variational optical flow methods to extract dynamical information from time-lapsed datasets
- provide automated confidence measures to ensure the reliability of motion estimation. Such measures should also assist with the automated optimization of model parameters
- develop an extensive data analysis toolkit including automated tracking, flow analysis, motion-based segmentation, image registration and detection of temporal changes
- implement the developed techniques for 4D data (3D + time) to enable analysis of tomographic data
- provide an efficient implementation of optical flow methods using advanced numerical schemes and computations on graphics processing units (GPUs). Thereby, the processing of a vast amount X-ray data - long image sequences and large tomographic volumes - becomes feasible
- apply the devised optical flow and data analysis techniques to investigate a number of scientific problems from various research fields

All the aforementioned topics are essential for the subject of this thesis, therefore we consider them as a complete framework. However, some parts of this work are also useful as isolated topics. As an example - in Section 5.1 we provide an approach for qualitative data taxonomy and in Section 3 we present image processing routines to enhance the original X-ray data. These steps are general for various data analysis problems that are not related to analysis of dynamics (e.g. image segmentation). The variational optical flow methods which we use in this work could be substituted by any other technique solving the same class of problems, i.e. finding correspondences between time-lapse images. Therefore, the information presented in other parts, such as in the chapters dedicated to data analysis or applications are still valuable. The current work is interdisciplinary and closely related to the fields of Computer Science, X-ray imaging, Material and Life Sciences.

Chapter 2

Optical Flow Methods

This chapter is dedicated to a detailed description of optical flow methods. First we introduce quality measures for motion estimation. Using these measures it is possible to assess the accuracy of the result of the optical flow estimation if the ground truth is known. Then, we present different approaches for the construction of variational models and discuss how these techniques could be applied for X-ray data. This part is divided into several sections, which are devoted to (i) the construction of data constancy assumptions, (ii) robust data terms and (iii) smoothness assumptions. Afterwards we introduce of automated confidence measures. Next, we extend the devised 2D optical flow models to 3D, which enables us to perform data analysis of tomographic data. Afterwards, we explain how a variational functional is solved numerically using iterative methods. Furthermore, we discuss various computational aspects such as derivatives discretization and implementation of multi-level computation.

2.1 Modeling

2.1.1 Quality Measures

Before we start with the description of optical flow methods, we first present quality measures which allow for quantitative comparison of different approaches if the ground truth for optical flow is known. We will employ these measures in our extensive performance studies of optical flow methods and data processing techniques in the section 5.4. It is important to note, that the performance of an algorithm cannot be described by a single measure. Instead, to show the performance with respect to different characteristics of an image we employ a set of measures. However, we do not show all the available measures for every evaluation. Instead, we use only those measures which are of interest for the particular evaluation case.

2.1.1.1 Angular Error

In the recent years the most widely used performance measure for optical flow was the *angular error* (AE). It is given by an angle between a computed flow vector $\mathbf{w} = (u, v)$ and a ground truth flow $\mathbf{w}_{GT} = (u_{GT}, v_{GT})$, which can be computed as a dot product of

both vectors, divided by the product of their length:

$$AE = \arccos\left(\frac{u \cdot u_{GT} + v \cdot v_{GT} + 1.0}{\sqrt{u^2 + v^2 + 1.0}\sqrt{u_{GT}^2 + v_{GT}^2 + 1.0}}\right).$$

Note, that angular error contains an additional scaling constant (1.0) to convert from pixels coordinates to the angular space. This measure was introduced in Ref. [FJ90] and was used in one of the first surveys on the performance of optical flow methods by [BFB94]. The main purpose of the AE is to provide a relative performance measure that could be used for all flow scales and is not causing problems for zero motion. However, it is important to note that errors in regions with fast motion are penalized less than errors in slow regions, which is a result of length normalization for this measure. Despite the fact, that the angular error was widely used in the previous work on optical flow methods its properties are not desirable for a robust and unbiased measure.

2.1.1.2 Endpoint Error

Another measure to estimate the performance of flow field computation is the *endpoint error* (EE) [ON94]. This measure captures the absolute difference between two flow vectors $\mathbf{w} = (u, v)$ and $\mathbf{w}_{GT} = (u_{GT}, v_{GT})$:

$$EE = \sqrt{(u_{GT} - u)^2 + (v_{GT} - v)^2}.$$

The endpoint error does not suffer from the drawbacks of angular error measure: it can be applied for the whole range of flow scales and all of them contribute equally. It is also a much more appropriate way to evaluate the differences between vectors if not only the direction, but also the difference in magnitude of the flow is important. Currently this measure is prevalent in the literature [SRB10, BSL⁺11] and will also be used to evaluate the performance of optical flow methods in this work.

2.1.1.3 Error Statistics

In order to evaluate the result of an optical flow computation for a given image sequence one is interested in a quantitative description of the quality of the output. A straightforward statistical measures are the average error and the standard deviation [BFB94]. Despite the fact, that both measures can be used to evaluate the overall performance, they are not optimal for measuring the performance on challenging image data or more complex scenes (e.g. a scene for which the background with little or no motion is prevalent).

Following the ideas presented in Ref. [BSL⁺11] we compute robustness measures [SS02]. For simplicity we also keep their notation for the presented statistical measures and denote by RX the percentage of pixels that have an error measure larger than X . In particular, we compute $R0.5$, $R1.0$, and $R2.0$ for the endpoint error.

2.1.1.4 Statistics Regions

The reliability of optical flow methods differs in various image regions. In particular, it is difficult to compute the flow around motion discontinuities, inside noisy regions, near artifacts or in homogeneous regions. Thus, it is reasonable to distinguish between such regions and evaluate their statistics separately. This can provide insights how different

optical flow models perform for in a particular situation. In the common benchmark for optical flow methods [BSL⁺11] three types of regions were evaluated: all image data (*All*), around motion discontinuities (*Disc*), and in textureless regions (*Untext*). In [BSL⁺11] it is concluded that *Disc* regions contain the most challenging data for the computation of optical flow. In contrast, textureless regions are not hard to recover, as soon as global optimization techniques (employing smoothness constraints) are used. In the Section 5.2.2 to evaluate the performance of different optical flow models on corrupted data we intentionally introduce synthetic artifacts. In this case we are interested to measure error statistics inside these regions and in a local neighborhood around them. We denote such regions with data outliers as *Err* regions.

To compute a mask for the *Disc* regions we take the gradient of the ground truth displacement field, compute its magnitude, apply a threshold for the magnitude, and then dilate the resulting mask with a 9×9 stencil. *Untext* regions are computed in a similar manner, but using a 3×3 stencil size [BSL⁺11].

2.1.2 Data Constancy Assumptions

To solve a correspondence problem one has to assume constancy of certain image features. For the choice of an appropriate constancy assumption all prior knowledge about the acquisition method, imaging conditions and the underlying scene should be taken into account. The selection of suitable data assumptions is essential and should be well adapted for the specific image data. In this section we provide an overview of data constancy assumptions. For each data assumption we discuss the aspects related to its usage on X-ray data. Note, that in the current work we do not aim to provide a complete list of all data constraints available in the literature. We focus only on those approaches, which can be practically used for highly challenging X-ray data. For example, we exclude the discussion of data terms based on color information.

2.1.2.1 Image Smoothing

The computation of optical flow is usually preceded by a *presmoothing* step. The aim of this step is to eliminate image noise. The most common procedure is to convolve the initial image I_0 with a *Gaussian kernel* K_σ with a standard deviation σ . High frequency variations are removed due to *low-pass* filtering effect of Gaussian filter. Furthermore, presmoothed image becomes infinite times differentiable, which is an important property to guarantee *well-posedness* of the inverse problem [HSSW01].

It is important to note that an appropriate choice of the smoothness degree is a crucial task. On the one hand, image presmoothing allows to improve image data and increase signal-to-noise ratio. On the other hand, oversmoothing destroys small image details and edges, which can lead to an inaccurate optical flow estimation. For the description of an optical flow model we always provide the initial smoothness level together with other model parameters.

2.1.2.2 Brightness Constancy Assumption

The simplest assumption is that moving objects within a scene do not change their appearance. In other words, the corresponding pixels have the same brightness values

[LK81, HS81]. This data constraint combines assumptions about the reflectance properties of the scene and illumination changes. For two subsequent image frames on time t and $t + 1$ the brightness constancy assumption can be expressed by:

$$I(x + u, y + v, t + 1) = I(x, y, t). \quad (2.1)$$

Under the assumption that the displacement components are small, this non-linear implicit equation for (u, v) can be linearized by performing a first-order Taylor expansion. For a function $I(x + u, y + v, t + 1)$ around the point $p = (x, y, t)$ the Taylor expansion is given by:

$$I(x + u, y + v, t + 1) \approx I(x, y, t) + I_x u + I_y v + I_t.$$

Using this approximation, the equation (2.1) can be rewritten as:

$$I_x u + I_y v + I_t = 0. \quad (2.2)$$

which is known in the literature as the *linearized optical flow constraint*.

To shorten the formulation of optical flow constraints we follow the idea presented in [BWKS06b] and rewrite the equation (2.2) in the following way:

$$I_x u + I_y v + I_t = \nabla_3^\top I \mathbf{w}$$

where $\mathbf{w} = (u, v, 1)$ and $\nabla_3 = (\partial_x, \partial_y, \partial_t)$ denotes the image gradient.

Since both positive and negative deviations from the data constraint should be taken into account, in the work of [LK81, HS81] the authors incorporate its square to the final version of data term. The brightness constancy assumption in a short notation is given by:

$$D_{grey}(I, \mathbf{w}) = (\nabla_3^\top I \mathbf{w})^2.$$

2.1.2.3 Gradient Constancy Assumption

The brightness constancy assumption can be used as long as illumination conditions and thus the corresponding pixel grey values between successive frames are not changing. If these assumptions are violated, data constraints that are invariant under brightness changes must be imposed. To cope with such situations, spatial brightness derivatives are introduced [UGVT88, Sch93, BBPW04, PBB⁺06], since they remain constant in the presence of *additive* illumination.

Taking spatial derivatives of the image brightness for both spatial components we obtain two constraints given by:

$$I_x(x + u, y + v, t + 1) = I_x(x, y, t),$$

$$I_y(x + u, y + v, t + 1) = I_y(x, y, t).$$

After linearization step is performed, both constraints using a short notation are given by:

$$\nabla_3^\top I_x \mathbf{w} = 0,$$

$$\nabla_3^\top I_y \mathbf{w} = 0.$$

Combining both constraints gives us the required data term using gradient constancy assumption:

$$D_{grad}(I, \mathbf{w}) = (\nabla_3^\top I_x \mathbf{w})^2 + (\nabla_3^\top I_y \mathbf{w})^2.$$

It is important to note - that in contrast to the brightness assumption - the gradient assumption provides two independent constraints instead of one. This gives an additional information and could be used to overcome the aperture problem [HS81]. However, despite the advantages of gradient assumption in the presence of global brightness changes, this constancy assumption is much more sensitive to noise. This property of a data constraint is crucial for noisy data, such as often encountered in time-resolved X-ray imaging. We investigate the influence of noise on the performance of data constancy assumptions in the experimental section 5.4.1.1.

2.1.2.4 High-Order Constancy Assumptions

The use of derivative-based constancy assumptions is not limited to the constancy of the brightness gradients. Higher-order derivatives, such as the Laplacian could also be employed for formulations of constancy assumptions. However, some of them are not suited for all types of motion. This is due to the fact, that some of these assumptions (e.g. brightness gradient) contain directional information, so they are dependent on the spatial orientation of image features. For an extended taxonomy of possible higher-order constancy assumptions the reader is referred to the literature [PBB⁺06]. Here we describe two constraints based on higher-order derivatives which can be potentially useful.

Laplacian Constancy Assumption

As it was mentioned in the previous section, directional information contained in data constraints may introduce some difficulties in the presence of fast rotational motion. To overcome this problem one can restrict to direction-invariant features such as constancy of Laplacian [PBB⁺06], which is given by:

$$\Delta_2 I(x + u, y + v, t + 1) - \Delta_2 I(x, y, t) = 0,$$

where $\Delta_2 I = I_{xx} + I_{yy}$. The linearized data term in a compact form reads:

$$D_{lapl}(I, \mathbf{w}) = (\nabla_3^\top (\Delta_2 I) \mathbf{w})^2.$$

In general, all constraints based on higher-order derivatives are more sensitive to noise and data artifacts than their counterparts based on image intensity. This crucial property must be taken into account for modeling of a suitable energy functional. We compare the performance of the brightness constancy assumption and assumptions based on derivatives in the evaluation section 5.4.1.1.

Gradient Norm Constancy Assumption

Another possibility to deal with fast rotational motion is to use the magnitude of the gradient, instead of spatial gradients themselves ([PBB⁺06]). The obtained data constraint and its corresponding data term are given by:

$$|\nabla I(x + u, y + v, t + 1)| - |\nabla I(x, y, t)| = 0$$

$$D_{grad-norm}(I, \mathbf{w}) = (\nabla_3^\top |\nabla I| \mathbf{w})^2,$$

where $|\mathbf{f}| = \sqrt{f_x^2 + f_y^2}$.

2.1.2.5 Assumptions on Multiple Image Features

To improve the accuracy of optical flow computation it might be a good idea to combine assumptions on various image features. The model based on multiple constancy assumptions provides more flexibility and may lead to a significant improvement of the results. An example for this approach a linear combination of both brightness and gradient constancy assumptions [BBPW04, ZBW11]. Its data term reads:

$$D_{\text{bright+grad}}(I, \mathbf{w}) = \beta_1(\nabla_3^\top I \mathbf{w})^2 + \beta_2((\nabla_3^\top I_x \mathbf{w})^2 + (\nabla_3^\top I_y \mathbf{w})^2),$$

where β_1 and β_2 are parameters to control the contribution of the individual constancy assumption to the overall data term.

Multiple image assumptions might also be useful to incorporate the information obtained by different contrast mechanisms, for example absorption and phase contrast. An alternative approach to include multiple assumptions on image features (and more generally, on different optical flow models) was presented in [LRR08]. In this work a series of candidate flow fields is computed using different algorithms or model parameters. Then a final flow is determined by an additional optimization problem. This idea corresponds to a so-called *fusion* approach.

2.1.2.6 Landmarks-driven approach

It is possible to incorporate user-defined landmarks to an optical flow model. This procedure is especially useful if the manual tracking of some objects or their parts is performed. Thus, it is possible to combine the information specified by an user with an automated flow computation. As it was shown in [Fea03] it is not convenient to embed the information about landmarks into the original energy functional. It is much more straightforward to modify the corresponding Euler-Lagrange equations (see Section 2.2.1) directly. We illustrate this on the example of the Horn and Schunck model. For this purpose we introduce a confidence map $c_{GT}(x, y)$, which equals to 1 or 0, depending on whether for a pixel position (x, y) the ground truth landmark is given.

Embedding the confidence map $c_{GT}(x, y)$ into the Euler-Lagrange equations (2.4) we obtain:

$$(1 - c_{GT})(I_x^2 u + I_x I_y v + I_x I_t - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla u)) + c_{GT}(u - u_{GT}) = 0,$$

$$(1 - c_{GT})(I_y I_x u + I_y^2 v + I_y I_t - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla v)) + c_{GT}(v - v_{GT}) = 0.$$

If the ground truth displacement is not provided we obtain the original version of Euler-Lagrange equations. For the other case the solution is set to the ground truth result $u = u_{GT}, v = v_{GT}$. After the flow field is modified according to information provided by landmarks it is propagated to the neighbouring pixels via the regularization term (see Section 2.1.4). Furthermore, to enhance the influence of the ground truth it makes sense to increase the smoothness constraint around locations where the correct flow is given. This could be regarded as an adaptive smoothness approach and can be implemented in a similar way as presented in Section 2.1.4.5.

2.1.3 Robust Data Assumptions

2.1.3.1 Robust Modeling

To obtain data constraints which take into account both positive and negative deviations from constancy assumptions in the previous sections we used squared data terms. The effect of such a quadratic, energy-like penalty function can be undesired in the presence of image artifacts or other cases, when constancy assumptions cannot be satisfied. As a result, such data outliers provide a large contribution to the overall energy functional, decreasing the quality of optical flow estimation. To overcome this, it is reasonable to penalize such deviations less strictly using a more robust function. There are a number of different penalty functions available in the literature [BA91, BA96, SRB10, BSL⁺11]. The most popular choices are: quadratic function $\Psi(x) = x^2$, the Charbonnier penalty $\Psi(x) = \sqrt{x^2 + \epsilon^2}$ [CBFAB94] and the Lorentzian $\Psi(x) = \log(1 + \frac{x^2}{2\sigma^2})$ [BA96], which is a non-convex sub-quadratic penalty function. The comparison between them is presented in Figure 2.1. An important aspect for the modelling of robust data terms is the convexity of the resulting energy functional. This property implies that a global minimum solution exists and we may use globally convergent algorithms to search for it. For this reason convex penaliser functions are preferred.

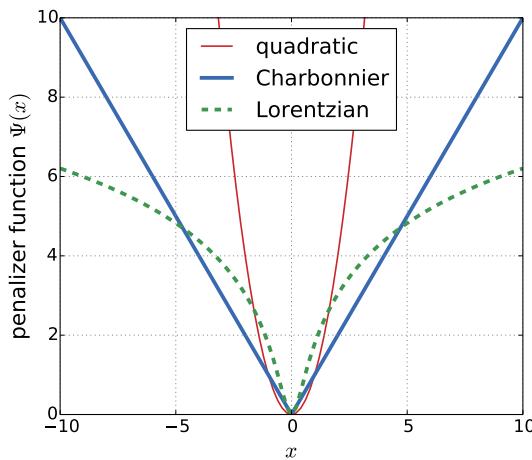


Figure 2.1: Comparison between different penalising functions: quadratic, Charbonnier and Lorentzian.

For our purpose we choose a penaliser function, which is a differentiable variant of L_1 -norm [BBPW04]:

$$\Psi_R(x) = \sqrt{x^2 + \epsilon^2},$$

where ϵ is a small positive constant used to avoid a problem of division by zero after differentiation step.

An example of a robust data term for brightness constancy assumption (see Section 2.1.2.2) is given by:

$$D_R(I, \mathbf{w}) = \Psi_R(\nabla_3^\top I \mathbf{w}).$$

As a result of such robust modelling the quality of optical flow estimation significantly improves [SRB10, BSL⁺11, ZBW11]. We perform quantitative and qualitative evaluation of robust settings in the presence of image artifacts in the experimental sections 5.4.3.3 and 5.4.3.3.

2.1.3.2 Joint and Separate Modeling

In the case when multiple data constraints are used, as we discussed in section 2.1.2.5, an important aspect is how to penalize individual data terms. One approach is to apply a single penalty function on both data constraints. On the example of combined brightness and gradient data constancy assumptions this can be illustrated as:

$$D_{joint}(I, \mathbf{w}) = \Psi_R(\beta_1(\nabla_3^\top I\mathbf{w}) + \beta_2(\nabla_3^\top I_x\mathbf{w} + \nabla_3^\top I_y\mathbf{w})).$$

Evidently, it is not necessary that both constraints should be satisfied at the same time for all data points. As it was previously shown, brightness constraint fails in regions where illumination changes and gradient constraint is sensitive to noise. However, the deviations from both constraints are penalized jointly. Note, that weight parameters β_1, β_2 are coupled as an argument of the penalty function, which slightly changes its mathematical formulation.

In the work of [BW05] the authors introduced a concept of *separate robustification*. Such strategy takes into account all the data constraints independently and allows to use a different penalty function for each counterpart. Using this approach, we rewrite the combined constraint as:

$$D_{sep}(I, \mathbf{w}) = \beta_1 \Psi_{R_1}(\nabla_3^\top I\mathbf{w}) + \beta_2 \Psi_{R_2}(\nabla_3^\top I_x\mathbf{w} + \nabla_3^\top I_y\mathbf{w}).$$

Following the performance analysis from the literature [BW05, ZBW11], if multiple image features are used to impose data constancy assumption we choose a separate robustification strategy. Such setting is more flexible and provides better results in most cases.

2.1.3.3 Combined Local-Global Approach

A useful approach to improve robustness of optical flow methods on noisy data is to consider the information not only from a particular data point, but also in a local neighbourhood. The modified data term would then impose a constancy assumption within a fixed region. Such constraint strongly resembles an original optical flow method by Lucas and Kanade [LK81]. In the work of [BWS02] so-called *combined local-global* (CLG) approach was introduced. The name highlights the fact that such technique combines the local information for the modelling of data term and provides dense results of global variational methods.

It is reasonable to weight contributions within a local region according to a distance from the central location (e.g. using weighted least squares fit). In [BWS02] it is presented how this approach could be incorporated for the construction of robust data terms. For brightness constancy assumption a least square fit is given by convolution with a Gaussian function:

$$D_{CLG}(I, \mathbf{w}) = K_\rho * (\nabla_3^\top I\mathbf{w})^2,$$

with the standard deviation ρ . The standard deviation parameter is also called *integration scale* [BWS02], since it regulates the size of spatial region which is taken into account.

This approach is not frequently used in the recent works since in cases when a high quality, noise-free data is available it could provide more blurry flow results. However, the combined local-global approach can lead to significant improvements for images which are

highly degraded by noise [BWS02, BWS05]. We present numerical experiments on noisy data and discuss the usefulness of the local-global approach in the experimental section 5.4.1.2.

2.1.3.4 Normalization of Data Terms

To provide additional robustness in unreliable regions influenced by noise or artifacts it is possible to further adjust the data term. In the recent work of [ZBW11] the authors used the ideas presented in [SAH91, LV98] to perform *normalization* of data constraints. One can show that image locations with large gradient and smooth regions are weighted differently [ZBW11]. As a result, high-gradient features provide more contribution to overall energy of the data term and thus largely influence the estimation of optical flow. Such behaviour can be undesired since high image gradient may be caused by noise or artifacts. Following [LV98, ZBW11] the original data constraint can be normalized by a factor:

$$n = \frac{1}{|\nabla_2 I(x, y, t)|^2 + \eta},$$

where positive parameter η avoids division by zero and controls the normalisation process depending on the noise scale.

After the normalization procedure the resulting data term is still convex and can be easily incorporated in our variational computation scheme. The normalized brightness constancy constraint is given by:

$$D_{norm}(I, \mathbf{w}) = n(\nabla_3^\top I \mathbf{w})^2.$$

For low contrast and noisy X-ray images the use of normalized data terms can be especially beneficial. We discuss the influence of proposed correction procedure, as well as choice of the appropriate η parameter on various amounts of noise in Section 5.4.1.3.

2.1.3.5 Refinement Model

In the work of [XJM10] authors proposed to explicitly model occlusions (or other data outliers) and reduce the influence of data term in these regions. First, a motion estimation is performed using appropriate model assumptions, then a confidence map is constructed using an occlusion detection. Then, a second run of optical flow algorithm is done using occlusion-aware refinement:

$$E_{refine}(I, \mathbf{w}) = \int_\Omega (c(\mathbf{x}) \cdot D(I, \mathbf{w}) + \alpha S(I, \mathbf{w})) \, d\Omega,$$

where $c(\mathbf{x})$ is a confidence map, which downweights the contribution of data constraint for unreliable image regions, $D(I, \mathbf{w})$ is a data constraint and $S(I, \mathbf{w})$ regulates the smoothness properties of the flow field.

We discuss various approaches for the estimation of confidence maps in Section 2.1.5.

2.1.4 Smoothness Assumptions

As it was mentioned previously, assuming constancy constraints based only on local image data is not sufficient to uniquely find a solution for optical flow. In order to overcome

the aperture problem additional assumptions about the flow field itself are employed. The first idea to impose such assumptions goes back to the seminal work of Horn and Schunck [HS81]. Smoothness constraints allow a *filling-in* [WS01a] of the flow information from the adjacent regions at image locations where no unique correspondence can be determined. Most of the approaches impose smoothness constraint by penalizing the magnitude of the flow gradient. According to the specific properties of the smoothness one may distinguish between *homogeneous*, *isotropic* and *anisotropic* smoothness. For a more detailed description of various models we refer the reader to the work of [WS01a].

2.1.4.1 Homogeneous Smoothness

The original method of Horn and Schunck [HS81] introduces smoothness assumption by penalizing magnitude of the flow gradient:

$$S_{HS}(u, v) = u_x^2 + u_y^2 + v_x^2 + v_y^2 = |\nabla_2 u|^2 + |\nabla_2 v|^2,$$

where $|\mathbf{f}| = \sqrt{f_x^2 + f_y^2}$ is a spatial magnitude and $\nabla_2 = (\partial_x, \partial_y)$ denotes a spatial gradient. This *homogeneous* regularization requires uniformly smooth flow fields and does not adapt to any supplementary information about image data and the flow field. Assuming that depicted objects may move differently from the background or with respect to each other it is evident that such constraint can hardly be fulfilled. Therefore, smoothness assumptions which allow to model discontinuities more thoroughly are required.

2.1.4.2 Image-Driven Smoothness

In order to take into account motion discontinuities one may assume that the flow field coincides with the boundaries of moving object, thus the smoothness degree can be adjusted according to image edges [NE86, ASW99]. To implement this approach one can multiply the original smoothness term by a scalar weighting function $g(x)$ that decreases on image edges [Sch93, ASW99]. One possibility for such function is:

$$g(x) = \frac{1}{2\sqrt{x+\epsilon}},$$

where ϵ is a small positive constant used to avoid a problem of division by zero.

The obtained *image-driven* smoothness term is given by:

$$S_{image}(I, u, v) = g(|\nabla I|^2)(|\nabla u|^2 + |\nabla v|^2),$$

where the magnitude of the spatial gradient $|\nabla I|$ is used to identify image edges [WS01a]. It should be noted, that such smoothness term is isotropic, since it treats all directions in image gradients in the same manner. To enhance the smoothing process along image edges, and suppress across them one can use an anisotropic image-driven approach [NE86].

Despite the fact that image-driven regularization provides sharp edges on the boundaries of moving object, it is prone to errors if the image gradient does not correspond to the actual flow discontinuities. For instance, this occurs in highly textured regions. Image noise is another challenge for this method, which causes the smoothness value to be adjusted to noise variations. In both cases image-driven approaches produce oversegmentation artifacts in the computed flow field.

2.1.4.3 Flow-Driven Smoothness

To preserve motion discontinuities the smoothness can be adapted to the unknown flow field. In this case the smoothness is reduced at flow edges described by the flow gradient [SH89, Sch94]. The regularization based on the *flow-driven* approach is thus given by:

$$S_{flow}(u, v) = \Psi_S(|\nabla u|^2 + |\nabla v|^2),$$

where the smoothness penalty function reads $\Psi_S(x) = \sqrt{x + \epsilon}$ and corresponds to the total variation (TV) regularization [ROF92].

In comparison to image-driven method the flow-driven approach provides less strict estimation of flow edges on the boundaries of the moving objects. However, for high-texture or noisy data the flow-driven approach is more robust and outperforms image-driven flow regularization [BBPW04, PBB⁺06, WPZ⁺09].

2.1.4.4 Spatial-Temporal Smoothness

In case if more than two frames are given for the estimation of optical flow, it is reasonable to additionally assume temporal smoothness of the flow fields [MB87, BA91]. For variational approaches such spatial-temporal smoothness terms were proposed by [Nag90] and [WS01b]. Implementation of the spatial-temporal smoothness term is done by the extension of spatial derivatives to the spatial-temporal dimension:

$$S_{temp}(u, v) = u_x^2 + u_y^2 + u_t^2 + v_x^2 + v_y^2 + v_t^2 = |\nabla_3 u|^2 + |\nabla_3 v|^2,$$

where $\nabla_3 = (\partial_x, \partial_y, \partial_t)$ denotes the spatial-temporal gradient.

The usage of temporal information provides a significant improvement for the estimation of optical flow [BBPW04, ZBW11]. In the case when stationary regions are corrupted by noise the temporal regularization helps to smooth noise contributions and correctly estimate zero flows. This property of spatial-temporal smoothness is especially useful if one is interested in separation between moving objects and a static background, i.e. for motion-based segmentation. For moving objects the applicability of the temporal smoothness depends on the type and magnitude of displacements. In the case of small, linear displacements the extension to the temporal domain will not cause problems since the influence of spatial and temporal smoothness will be approximately equal. However, in the presence of fast motion large discontinuities in a temporal direction may emerge. To appropriately handle them a number of different strategies can be used. One way would be to use the coarse-to-fine computation scheme which allows to solve for large displacements. We describe this technique in details in Section 2.2.4.2. Another option would be to employ discontinuity preserving smoothness terms using robust penalty function presented in section 2.1.4.3. This approach will result into a joint spatial-temporal discontinuity preserving smoothness term:

$$S_{flow-temp}(u, v) = \Psi_S(|\nabla u|^3 + |\nabla v|^3),$$

where $\Psi_S(x) = \sqrt{x + \epsilon}$ and $\nabla_3 = (\partial_x, \partial_y, \partial_t)$.

An important aspect for modeling temporal smoothness is a number of time frames used to impose constraints in temporal direction. In the works of [Bru06], [ZBW11] it was shown that increasing the number of frames significantly improved the results.

However, it was also observed that there is an upper limit after which there is no longer an improvement in the result. The usage of both spatial and temporal information about the flow field allows to decrease the optimal value for smoothness parameter, which result in a sharper flow edges. Additionally, it is reasonable to apply presmoothing step described in section 2.1.2.1 also in temporal direction [Bru06]. The temporal smoothness parameter should be chosen according to constancy of the motion in time and amount of noise. In general, spatial-temporal smoothness provides a robust approach to compute the optical flow, especially for noisy or corrupted data.

2.1.4.5 Adaptive Smoothness

The smoothness term can be further adapted using a number of different strategies. One approach would be to use a weighting function based on segmented masks representing particular data objects. Such objects could be identified during the optical flow computation or can be pre-labeled using an a-priori knowledge. An example of such masks could be a reduction of smoothness weight on the boundary of closely moving objects, which may improve the separation between them. Another example corresponds to the case when the motion in different regions expected to exhibit different properties.

The adaptive variant of the smoothness term is then given by:

$$S_{adapt}(I, u, v) = \alpha_{S_i}(|\nabla u|^2 + |\nabla v|^2),$$

where α_{S_i} represents the smoothness parameter inside or around the segment S_i .

Another strategy is to adapt the smoothness according to the computation procedure. In [ZBW11] authors proposed to adjust the smoothness weight on each level of multiscale computation of optical flow. One may note, that on coarser image levels the data term provides less details resulting from smoothing properties of the downscaling procedure. In this case it makes sense to increase the influence of the smoothness term. This can be done by adjusting the flow regularization according to the warping computation level k : $\alpha(k) = \frac{\alpha}{\eta^k}$, where η is a downscaling factor.

2.1.4.6 Joint Image and Flow Driven Smoothness

As it was discussed in sections 2.1.4.2 and 2.1.4.3 both image- and flow-driven smoothness approaches have their advantages and drawbacks. To improve the performance of optical flow estimation it makes sense to combine the advantages of both methods to obtain precise motion discontinuities and avoid oversegmentation problems in highly textured image regions. The first idea which achieves that was presented in [SLB08]. In this work the smoothness direction is adapted to image edges and smoothing amount to the flow gradient. In [ZBW11] the authors presented a more general approach, which they call *complementary smoothness term*. Such approach regulates the smoothness direction according to the data constraint, which could be different from information given by image gradients. As a result the smoothness is reduced in the direction where data constraint provides useful information and increased in regions where data constraint is vanished.

2.1.5 Measures of Confidence

In Section 2.1.1 we have described measures for quantitative evaluation of optical flow algorithms in the case when the ground truth result is known. In this section we discuss

measures to check the reliability of optical flow based only on the computed flow field and initial image data.

Until the recent works of [SRB10, ZBW11] only little attention was dedicated to the application of confidence measures for optical flow methods. These measures can allow us to determine errors in the computed flow fields caused by artifacts. Moreover, we may use these measures to evaluate the quality of the computed result and automatically optimize the model parameters for the optical flow computation.

2.1.5.1 Gradient-based Measure

The first confidence measure for variational optical flow methods was presented by [BFB94]. It was proposed to relate the quality of the flow estimation with the image gradient, since as it was discussed in section 1.4.2 the correspondence problem can only be solved if there is enough information provided by the local image gradient. The confidence measure based on image data is thus given by:

$$C_{gradient}(\mathbf{x}) = |\nabla I(\mathbf{x})|.$$

Clearly, this measure does not take into account any dynamical information and assumes only local image data. Additionally, this measure is not capable to evaluate global optical flow approaches which use additional smoothness term to overcome the aperture problem. Moreover, the variations in image brightness can be caused by noise or artifacts. These drawbacks make image-based confidence measure not well suited for the evaluation of optical flow estimation.

2.1.5.2 Energy-based Measure

In the work of [BW06, Ers08] the authors presented an intuitive way to evaluate the quality of variational optical flow methods based on energy minimization. Since during minimization procedure such functionals penalize deviations from model assumptions, it is very natural to consider such deviations as a confidence measure.

For the case of method of Horn and Schunck the energy-based confidence measure for a pixel \mathbf{x} reads:

$$C_{energy}(\mathbf{x}) = \frac{1}{(\nabla_3^\top I(\mathbf{x})\mathbf{w})^2 + \alpha(|\nabla_2 u(\mathbf{x})|^2 + |\nabla_2 v(\mathbf{x})|^2)}.$$

At this point one may consider to evaluate energy contributions from both data and smoothness terms jointly or, alternatively, evaluate only the data term which constitutes the optical flow problem, i.e. correspondence between pixels (see Section 2.1.5.3). In the presence of artifacts or other data outliers the model assumptions are violated, which results in a high values of energy in those regions.

As it was discussed in [BW06] energy-based confidence measures have a number of advantages: 1) it is based on the same assumptions as the computational model 2) it is general and could be extended or modified for a particular optical flow model 3) it does not require additional parameters.

2.1.5.3 Data Constancy Error

A similar concept to the energy-based confidence measure is a data constancy error, which evaluates only the data constraint. For the case of brightness constancy assumption, the confidence measure for a pixel \mathbf{x} reads:

$$C_{data}(\mathbf{x}) = (I(\mathbf{x} + \mathbf{w}, t + 1) - I(\mathbf{x}, t)).$$

This measure shares the same set of advantages as the energy-based confidence measure, as it is also directly based on an optical flow model.

2.1.5.4 Motion Uniqueness Criteria

Another approach to evaluate the computed flow field is to examine its spatial distribution. We assume that if the motion is applied (source image is warped towards the target image) all pixels should still be uniformly distributed [BBH03, Ers08, XJM10]. However, in the presence of data outliers or problems with appearing and disappearing information the flow vectors could point to nearly the same position. Such multiple pixels mapping could indicate a location of unreliable motion estimation. This idea is illustrated on Figure 2.2.

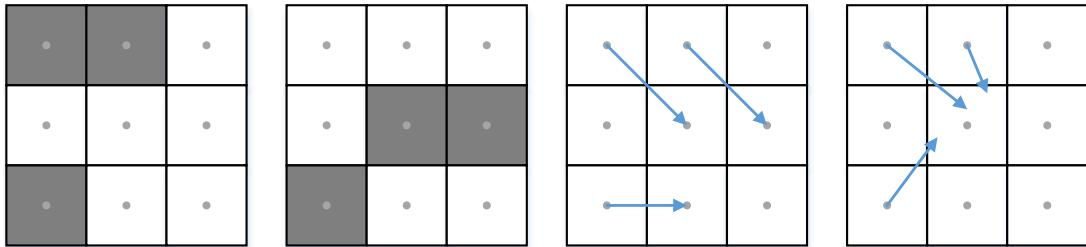


Figure 2.2: Flow uniqueness criteria. **Left:** First image frame. **Middle Left:** Second image frame. **Middle Right:** One-to-one correspondence. Pixels are mapped to uniformly distributed positions. **Right:** Pixels are mapped nearly to the same position, indicating a sink in a vector field.

In order to implement this approach we count a number of arrived pixels $\mathbf{x} + \mathbf{u}(\mathbf{x})$ to the target image position \mathbf{x} . In order to handle real-valued displacements we redistribute contributions from neighboring flow vectors according to their distance to the target pixel location. The confidence measure based on motion uniqueness mapping is then expressed by:

$$C_{unique}(\mathbf{x}) = \frac{1}{|1 - count(\mathbf{x})| + \epsilon},$$

where ϵ is a small positive constant to avoid a division by zero problem. The deviations from the one-to-one mapping will be captured and the confidence degree will be reduced.

2.1.5.5 Forward-Backward Check

A common way to ensure consistency of a motion field solution in the presence of occlusions is to perform a so-called *forward-backward check* (or *cross-checking*) [CCK91, Fua93, BBH03]. In this approach we assume that the flow field computed in forward (from $I(\mathbf{x}, t)$ to $I(\mathbf{x}, t + 1)$) and the backward direction is equal, but opposite:

$$\mathbf{w}^f(\mathbf{x}) = -\mathbf{w}^b(\mathbf{x} + \mathbf{w}^f(\mathbf{x}))$$

where \mathbf{w}^f and \mathbf{w}^b is a flow field computed in a forward and a backward direction respectively.

In the case of occlusions or other image artifacts, forward and backward vectors do not match, which could be exploited to identify problematic regions. For the consistency based confidence measure we estimate the magnitude of the difference vector between forward and backward flow vectors:

$$|\Delta\mathbf{w}(\mathbf{x})| = \sqrt{(\mathbf{w}^f(\mathbf{x}) + \mathbf{w}^b(\mathbf{x} + \mathbf{w}^f(\mathbf{x})))^2}$$

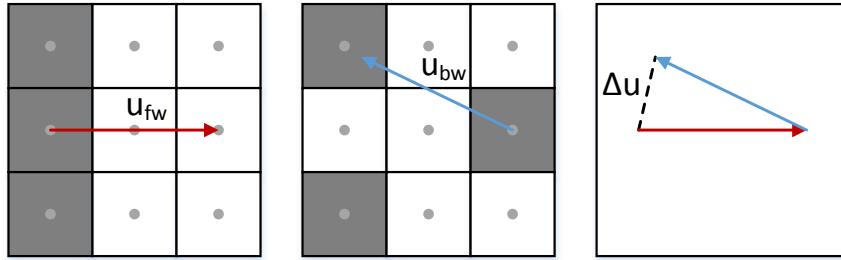


Figure 2.3: Forward-Backward check method. **Left:** First image frame with the forward flow vectors. **Middle:** Backward flow vectors for interchanged images. **Right:** Discrepancy vector between forward and backward displacement vectors.

The idea for forward-backward check approach is illustrated in Figure 2.3.

As a result we consider pixels, where consistency check failed as locations with a low confidence value:

$$C_{consist}(\mathbf{x}) = \frac{1}{|\Delta\mathbf{w}(\mathbf{x})| + \epsilon}.$$

We will employ this measure in the section dedicated to analysis of temporal changes in metal foams of various kind in the application Section 6.2.1.

2.1.5.6 Optimal Prediction Principle

Recently, in [ZBW11] the authors proposed a confidence measure based on the principle of optimal prediction (OPP). It states that the flow field computed with optimal model parameters allows for the best flow prediction for the next image frames. For this, one assumes that the objects are moving with a constant velocity and linear trajectory. This can be implemented by scaling the flow vector by 2. In order to estimate the prediction quality a data term between the first and the third frame is evaluated. The resulting data constancy error (DCE) measure is given by:

$$C_{predict}(\mathbf{x}) = (I(\mathbf{x} + 2\mathbf{w}, t+2) - I(\mathbf{x}, t)).$$

Despite its simplicity, confidence measure based on the optimal prediction principle gives good results [ZBW11]. The only drawback is a restriction imposed by the assumption of temporally constant flow field, which can be frequently violated.

We test the performance of automated confidence measures and give conclusions regarding their use in the experimental section 5.4.4.

2.1.6 Generalization to 3D Optical Flow

In order to apply optical flow methods on tomographic data, the extension from the 2D variational model into a 3D model should be performed. This step is done in a straightforward way by including an additional spatial dimension. The advantages for 3D optical flow are discussed in the section, describing computed tomography experiments (see Section 1.3.2.2).

The 2D variational optical flow model using the flow-driven smoothness approach (Section 2.1.4.3) is given by:

$$E_{2D}(u, v) = \int_{\Omega_2} (I(x + u, y + v, t + 1) - I(x, y, t))^2 + \alpha \Psi(|\nabla_2 u|^2 + |\nabla_2 v|^2) dx dy.$$

For a volume image $I(x, y, z, t)$, where (x, y, z) are the voxel coordinates within an image domain $\Omega_3 \subset \mathbf{R}^3$, the extension to the 3D variational model reads:

$$\begin{aligned} E_{3D}(u, v, w) &= \int_{\Omega_3} (I(x + u, y + v, z + w, t + 1) - I(x, y, z, t))^2 + \\ &+ \alpha \Psi(|\nabla_3 u|^2 + |\nabla_3 v|^2 + |\nabla_3 w|^2) dx dy dz. \end{aligned}$$

It is important to mention that some of the problems of 2D optical flow estimation disappear after the extension into a 3D domain. For example, there are no problems related to the projection geometry, such as occlusions between moving objects or divergent motion due to the perspective change.

As a drawback, solving 3D optical flow problem is a resources demanding task. It requires both large memory consumption and intense computational power. We further discuss performance aspects in Chapter 4.2 we describe a number of efficient computation schemes and present high-performance implementation of 3D optical flow methods on GPUs.

2.2 Numerical Solution

In this section we explain how a variational optical flow model can be solved numerically. For this purpose we introduce a minimization procedure and iterative methods. We also discuss important implementation aspects, such as discretization and a derivatives approximation. In the first part of the section we derive underlying equations assuming small displacements between image frames. Then we present an extension of the energy functional and its minimization procedure which allows to handle large displacements. To shorten the derivation of the corresponding formulas we assume 2D images. The extension to 3D domain can be performed in similar manner.

2.2.1 Minimization Procedure

In the previous chapter we have discussed the construction of a variational energy functional. Next, our aim is to find such a displacement field $u(x, y)$ and $v(x, y)$ which minimizes deviations from the model assumptions.

According to calculus of variations [Els62], the minimum of the convex variational functional of the general form:

$$E(u, v) = \int_{\Omega_2} F(x, y, u, v, u_x, u_y, v_x, v_y) dx dy.$$

can be found at the location where its first-order derivatives vanish. Such coupled differential equations are called *Euler-Lagrange equations* and given by:

$$\begin{aligned} F_u - \frac{\partial}{\partial x} F_{u_x} - \frac{\partial}{\partial y} F_{u_y} &= 0 \\ F_v - \frac{\partial}{\partial x} F_{v_x} - \frac{\partial}{\partial y} F_{v_y} &= 0 \end{aligned} \quad (2.3)$$

with reflecting Neumann boundary conditions $\mathbf{n}^\top \nabla u = 0$ and $\mathbf{n}^\top \nabla v = 0$, where \mathbf{n} is a normal vector.

We show an example of minimization procedure on a flow-driven optical flow model with brightness constancy assumption. First, we calculate partial derivatives with respect to the unknown functions u and v :

$$\begin{aligned} F_u &= 2I_x^2 u + 2I_x I_y v + 2I_x I_t \\ F_v &= 2I_y I_x u + 2I_y^2 v + 2I_y I_t \\ F_{u_x} &= 2\alpha \Psi'(|\nabla u|^2 + |\nabla v|^2) u_x \\ F_{u_y} &= 2\alpha \Psi'(|\nabla u|^2 + |\nabla v|^2) u_y \end{aligned}$$

Embedding these terms into the Euler-Lagrange equations (2.4) we obtain:

$$\begin{aligned} I_x^2 u + I_x I_y v + I_x I_t - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla u) &= 0, \\ I_y I_x u + I_y^2 v + I_y I_t - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla v) &= 0. \end{aligned}$$

Solving this coupled system of equations with respect to the unknown displacement functions u and v gives us the minimum of our energy functional.

2.2.2 Discretization

To solve Euler-Lagrange equations numerically we should perform a discretization procedure. We consider the unknown functions $u(x, y, t)$ and $v(x, y, t)$ on a rectangular image with the grid size of h_x in x -direction and h_y in y -direction. The values u_{ij} and v_{ij} denote the approximation of functions $u(x, y)$ and $v(x, y)$ at a pixel position (i, j) with $i = 0 \dots N - 1, j = 0 \dots M - 1$, where N, M indicate image dimensions.

In order to represent image derivatives we must assume a particular approximation of the differential operator. Second order approximations of the spatial image derivatives using central differences are given by:

$$\begin{aligned} [I_x]_{i,j} &= \frac{I_{i+1,j,t} - I_{i-1,j,t}}{2h_x} \\ [I_y]_{i,j} &= \frac{I_{i,j+1,t} - I_{i,j-1,t}}{2h_y} \end{aligned}$$

To improve the accuracy of derivatives approximation one may also use fourth order approximations:

$$[I_x]_{i,j} = \frac{-I_{i+2,j,t} + 8I_{i+1,j,t} - 8I_{i-1,j,t} + I_{i-2,j,t}}{12h_x}$$

$$[I_y]_{i,j} = \frac{-I_{i,j+2,t} + 8I_{i,j+1,t} - 8I_{i,j-1,t} + I_{i,j-2,t}}{12h_y}$$

For further improvement with respect to noise it is common to use a temporal averaging of spatial derivatives [SRB10, ZBW11]. For the second order central difference such a temporal averaging is given by:

$$[I_x]_{i,j} = \frac{(I_{i+1,j,t+1} - I_{i-1,j,t+1}) + (I_{i+1,j,t} - I_{i-1,j,t})}{4h_x}$$

$$[I_y]_{i,j} = \frac{(I_{i,j+1,t+1} - I_{i,j-1,t+1}) + (I_{i,j+1,t} - I_{i,j-1,t})}{4h_y}$$

To compute temporal derivatives we choose a forward difference:

$$[I_t]_{i,j} = \frac{I_{i,j,t+1} - I_{i,j,t}}{h_t},$$

where time step h_t is chosen to be equal to 1.0.

We proceed with the discretization of others terms such as $\text{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla u)$. Additionally to the second-order derivative approximation, we can also employ the discretization approach based on nested central differences with the halved grid sizes $\frac{1}{2}h_x$ and $\frac{1}{2}h_y$. Using this strategy the term $\text{div}(\Psi'\nabla u)$ can be expressed as:

$$\begin{aligned} \text{div}(\Psi'\nabla u) &= (\Psi'u_x)_x + (\Psi'u_y)_y \\ &= \frac{(\Psi'u_x)_{i+\frac{1}{2},j} - (\Psi'u_x)_{i-\frac{1}{2},j}}{2(\frac{1}{2}h_x)} + \frac{(\Psi'u_y)_{i,j+\frac{1}{2}} - (\Psi'u_y)_{i,j-\frac{1}{2}}}{2(\frac{1}{2}h_y)} \\ &= \frac{\frac{\Psi'_{i+1,j} + \Psi'_{i,j}}{2} \left(\frac{u_{i+1,j} - u_{i,j}}{2(\frac{1}{2}h_x)} \right) - \frac{\Psi'_{i,j} + \Psi'_{i-1,j}}{2} \left(\frac{u_{i,j} - u_{i-1,j}}{2(\frac{1}{2}h_x)} \right)}{2(\frac{1}{2}h_x)} \\ &\quad + \frac{\frac{\Psi'_{i,j+1} + \Psi'_{i,j}}{2} \left(\frac{u_{i,j+1} - u_{i,j}}{2(\frac{1}{2}h_y)} \right) - \frac{\Psi'_{i,j} + \Psi'_{i,j-1}}{2} \left(\frac{u_{i,j} - u_{i,j-1}}{2(\frac{1}{2}h_y)} \right)}{2(\frac{1}{2}h_y)} \\ &= \frac{\frac{\Psi'_{i+1,j} + \Psi'_{i,j}}{2} \left(\frac{u_{i+1,j} - u_{i,j}}{h_x^2} \right) - \frac{\Psi'_{i,j} + \Psi'_{i-1,j}}{2} \left(\frac{u_{i,j} - u_{i-1,j}}{h_x^2} \right)}{2} \\ &\quad + \frac{\frac{\Psi'_{i,j+1} + \Psi'_{i,j}}{2} \left(\frac{u_{i,j+1} - u_{i,j}}{h_y^2} \right) - \frac{\Psi'_{i,j} + \Psi'_{i,j-1}}{2} \left(\frac{u_{i,j} - u_{i,j-1}}{h_y^2} \right)}{2}, \end{aligned}$$

where the nonlinear terms $[\Psi']_{i,j}$ are given by:

$$[\Psi']_{i,j} = \Psi'(|\nabla u|^2 + |\nabla v|^2) = \Psi'([u_x]_{i,j}^2 + [u_y]_{i,j}^2 + [v_x]_{i,j}^2 + [v_y]_{i,j}^2)$$

The discretization of the term $\text{div}(\Psi'\nabla v)$ is done in a similar way.

Finally, after all necessary discretization schemes are introduced we can write down the discrete Euler - Lagrange equations:

$$\begin{aligned} [I_x^2]_{i,j} u_{i,j} + [I_x I_y]_{i,j} v_{i,j} + [I_x I_t]_{i,j} - \alpha \sum_{l \in x,y} \sum_{(\bar{i},\bar{j}) \in N_d(i,j)} \frac{[\Psi']_{\bar{i},\bar{j}} + [\Psi']_{i,j}}{2} \left(\frac{u_{\bar{i},\bar{j}} - u_{i,j}}{h_d^2} \right) &= 0 \\ [I_y I_x]_{i,j} u_{i,j} + [I_y^2]_{i,j} v_{i,j} + [I_y I_t]_{i,j} - \alpha \sum_{l \in x,y} \sum_{(\bar{i},\bar{j}) \in N_d(i,j)} \frac{[\Psi']_{\bar{i},\bar{j}} + [\Psi']_{i,j}}{2} \left(\frac{v_{\bar{i},\bar{j}} - v_{i,j}}{h_d^2} \right) &= 0, \end{aligned}$$

where $N_d(i,j)$ denotes neighbor pixels for the position (i,j) in the direction $d \in (x,y)$.

Now, this coupled system of equations has to be solved with respect to the unknown displacements u_{ij}, v_{ij} .

2.2.3 Iterative Methods

After the discretization of the Euler-Lagrange equations is performed we obtain a linear system of equations, which can be formulated in the following form:

$$A\mathbf{x} = \mathbf{b}$$

To solve it directly, we may invert the system matrix and multiply it by the result vector:

$$\mathbf{x} = A^{-1}\mathbf{b}$$

However, for large systems of equations this procedure is computationally expensive. As an alternative, it is possible to decompose the initial matrix A into two terms:

$$A = A_1 + A_2,$$

and try to approximate the solution iteratively:

$$\begin{aligned} A_1\mathbf{x}^{k+1} &= \mathbf{b} - A_2\mathbf{x}^k \\ \mathbf{x}^{k+1} &= A_1^{-1}(\mathbf{b} - A_2\mathbf{x}^k) \end{aligned}$$

The appropriate choice of matrix A_1^{-1} should be done in such a way, that it is a good approximation to the initial inverse matrix A^{-1} . Additionally, it should be easily computable. This matrix splitting strategy is a core idea for a number of iterative methods such as *Jacobi* and *Gauß-Seidel methods*.

2.2.3.1 Jacobi method

The Jacobi method is an iterative algorithm to solve a system of linear equations [Saa03]. The system of linear equations $A\mathbf{x} = \mathbf{b}$ has an unique solution, if the coefficient matrix A contains no zeros on its main diagonal. On each iteration step, one finds the approximate solution for each diagonal element and use it in the next step. The iterative process is then repeated until it converges.

The Jacobi method can be expressed in a matrix form, using the following decomposition of the initial matrix A :

$$A = D - L - U, \tag{2.4}$$

where D , L and U represent the diagonal, strictly lower triangular and strictly upper triangular matrices of A , respectively. After the substitution of the decomposed matrix A in the equation $A\mathbf{x} = \mathbf{b}$, it can be reformulated in the following way::

$$D\mathbf{x} = (L + U)\mathbf{x} + \mathbf{b}, \quad (2.5)$$

and the corresponding iterative solution can be found via:

$$\mathbf{x}^{k+1} = D^{-1}(L + U)\mathbf{x}^k + D^{-1}\mathbf{b}. \quad (2.6)$$

Using the Jacobi method, the new value of x_i^k on the iteration step $k + 1$ can be estimated in the following way:

$$x_i^{k+1} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^n (a_{ij}x_j^k) \right). \quad (2.7)$$

It is important to mention that the order in which the matrix entries are evaluated is irrelevant since the Jacobi method uses only the matrix values from the old time step. Such property of the method is advantageous for the parallel implementations of the iterative algorithm.

2.2.3.2 The Gauß-Seidel Method

One of the most simple and commonly used iterative solvers for linear system of equations is the Gauß-Seidel method [Saa03]. In this approach the initial matrix A is decomposed into the following parts:

$$A = D - L - U = \underbrace{(D - L)}_{A_1} + \underbrace{(-U)}_{A_2},$$

where D, L, U represent the diagonal, strictly lower triangular, and strictly upper triangular parts of the system matrix A . Using such decomposition we can rewrite the initial equations:

$$(D - L)\mathbf{x} = (\mathbf{b} + U\mathbf{x})$$

The solution using Gauß-Seidel methods is then given by:

$$\mathbf{x} = (D - L)^{-1}(\mathbf{b} + U\mathbf{x}).$$

And the corresponding iterative solution scheme reads:

$$\mathbf{x}^{k+1} = (D - L)^{-1}(\mathbf{b} + U\mathbf{x}^k).$$

2.2.4 Large Displacements

In Section 2.2.2 we assumed that displacements between two consequent images are small, so the linearization of the data constancy assumption can be performed. This procedure is important since it allows us to construct convex energy functionals, which we can minimize using globally convergent algorithms. However, in case of a large displacements the linearization procedure cannot be performed, since no reliable approximation is possible using the Taylor expansion. In order to handle whole range of displacements in this section we introduce an extension of the variational approach and describe the steps related to its minimization.

2.2.4.1 Euler - Lagrange Equations

As a first step, we formulate the energy functional in its original form without linearized data constraints:

$$E(u, v) = \int_{\Omega_2} (I(x + u, y + v, t + 1) - I(x, y, t))^2 + \alpha \Psi(|\nabla_2 u|^2 + |\nabla_2 v|^2) dx dy. \quad (2.8)$$

The corresponding Euler-Lagrange equations, which have to be solved to find the minimum of the energy functional are given by:

$$I_x(x + u, y + v, t + 1)(I(x + u, y + v, t + 1) - I(x, y, t)) - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla u) = 0,$$

$$I_y(x + u, y + v, t + 1)(I(x + u, y + v, t + 1) - I(x, y, t)) - \alpha \operatorname{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2) \cdot \nabla v) = 0$$

Note, that the energy functional (2.8) is not convex, since the data term implicitly depends on the unknown flow components u, v . From the non-convexity property it follows that multiple local minima may exist and numerical algorithms could be trapped in such a local minimum away from the best solution.

In order to find a solution the incorporation of a suitable minimization strategy is required.

2.2.4.2 Coarse-to-fine Strategy

The common approach to deal with large displacements in many optical flow methods is a coarse-to-fine strategy [BA96, MP98, BBPW04]. In the current work we follow this idea and use the incremental hierarchical approach. First, on each computation level we decompose the flow field into an already computed solution from the previous coarser level u, v and a small unknown motion increment for the current level du, dv (see Figure 2.4):

$$\begin{aligned} u^{k+1} &= u^k + du^k, \\ v^{k+1} &= v^k + dv^k. \end{aligned}$$

Putting the incremental definition of the overall flow field we rewrite the image function $I(x + u^{k+1}, y + v^{k+1}, t + 1)$ in the following way:

$$I(x + u^{k+1}, y + v^{k+1}, t + 1) = I(x + u^k + du^k, y + v^k + dv^k, t + 1)$$

Next, we perform a first-order Taylor expansion around the point $I(x + u^k + du^k, y + v^k + dv^k, t + 1)$, which gives us:

$$\begin{aligned} I(x + u^k + du^k, y + v^k + dv^k, t + 1) &= I(x + u^k, y + v^k, t + 1) \\ &+ I_x(x + u^k, y + v^k, t + 1)du^k \\ &+ I_y(x + u^k, y + v^k, t + 1)dv^k \end{aligned}$$

Finally we can write the Euler-Lagrange equations:

$$\begin{aligned} I_x(x + u^k, y + v^k, t + 1)(&I_x(x + u^k, y + v^k, t + 1)du^k \\ &+ I_y(x + u^k, y + v^k, t + 1)dv^k \\ &+ I(x + u^k, y + v^k, t + 1) - I(x, y, t)) \\ &- \alpha \operatorname{div}(\Psi'(|\nabla(u^k + du^k)|^2 + |\nabla(v^k + dv^k)|^2) \cdot \nabla(u^k + du^k)) = 0 \end{aligned}$$

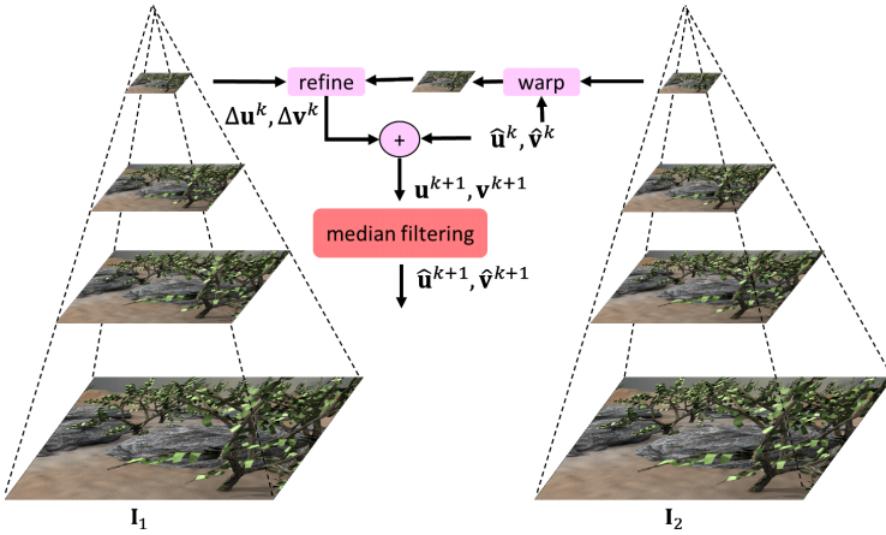


Figure 2.4: A schematic example of a multi-level coarse-to-fine computation strategy. The top level of image pyramids represent the coarsest level and the bottom level of pyramid corresponds to the original image representation. The computation scheme includes image warping 2.2.4.3 and median filtering of intermediate flow results 2.2.6. Image: [SRB14]

$$\begin{aligned}
 I_y(x + u^k, y + v^k, t + 1) &= I_x(x + u^k, y + v^k, t + 1)du^k \\
 &+ I_y(x + u^k, y + v^k, t + 1)dv^k \\
 &+ I(x + u^k, y + v^k, t + 1) - I(x, y, t) \\
 &- \alpha \operatorname{div}(\Psi'(|\nabla(u^k + du^k)|^2 + |\nabla(v^k + dv^k)|^2) \cdot \nabla(v^k + dv^k)) = 0
 \end{aligned}$$

This system of equations has to be solved on each computation level k with respect to the motion increments du^k and dv^k . Since we assume that these increments are small the linearization procedure is valid and can be applied. After the flow increments for the current level are computed we find the overall solution as $u^{k+1} = u^k + du^k, v^{k+1} = v^k + dv^k$. With such a coarse-to-fine strategy we perform the flow computation step-by-step, so instead of a non-convex optimization we solve a series of convex problems and successively refine the result.

In order to implement such a hierarchical approach we have to consider an appropriate implementation to obtain the motion compensated image $I(x + u^k, y + v^k, t + 1)$ at a computation level k and the realization of a multi-level strategy (construction of image pyramids).

2.2.4.3 Motion Compensation

We first discuss the implementation of the motion compensation procedure. The computation of the image function $I(x + u, y + v, t + 1)$ and its derivatives $I_x(x + u, y + v, t + 1), I_y(x + u, y + v, t + 1)$ requires to "apply" the computed flow field on the initial image, so-called *image warping* [MP98]. In this work we follow the approach proposed in [MP98, BBPW04] and perform a *backward registration*. A number of different implementations are possible (see Section 2.2.5). Here we show an example of motion compensation using bilinear interpolation.

We decompose the real-valued computed flow field u and v into two parts: $u = \hat{u} + \epsilon_u$, $v = \hat{v} + \epsilon_v$ where \hat{u} , \hat{v} denote the integer fraction of u , v and ϵ_u , ϵ_v denote the subpixel displacement. Then we compute the motion compensated image $I(x + u, y + v, t + 1)$ by means of bilinear interpolation using:

$$\begin{aligned}[I(x + u, y + v, t + 1)]_{i,j,t+1} = & (1 - \epsilon_u)(1 - \epsilon_v)I_{(i+\hat{u}),(j+\hat{v}),t+1} \\ & + (\epsilon_u)(1 - \epsilon_v)I_{(i+\hat{u})+1,(j+\hat{v}),t+1} \\ & + (1 - \epsilon_u)(\epsilon_v)I_{(i+\hat{u}),(j+\hat{v})+1,t+1} \\ & + (\epsilon_u)(\epsilon_v)I_{(i+\hat{u})+1,(j+\hat{v})+1,t+1}\end{aligned}$$

The motion compensation procedure for the image derivatives $I_x(x + u, y + v, t + 1)$, $I_y(x + u, y + v, t + 1)$ can be performed in the analogous way.

2.2.5 Multi-level Computation

In order to implement a coarse-to-fine strategy we need to specify a transfer function which downscales the original image version to a coarser resolution level and interpolates the obtained results back to the finer computation level. On each computation level k the image size is determined via:

$$N_d^k = N_d^{orig} \eta^k,$$

where η is a warping scale parameter, N_d denotes the image size in the dimension $d \in (x, y)$.

For this purpose a number of different approaches can be used such as nearest neighbour averaging, area-based averaging [BWF⁺05], bilinear or bicubic interpolation. In the current work for the construction of image pyramids we use bicubic interpolation using Coons patches [Coo67]. This method gives good results [SRB14, ZBW11], efficient and easy to implement. For the case of using 3D tomographic data as an input, we employ tricubic interpolation.

2.2.6 Intermediate Flow Filtering

In the work of [SRB10, SRB14] authors reveal a preprocessing step which allows to significantly improve the accuracy and, which is more important, robustness of optical flow estimation [WPZ⁺09]. It is proposed to apply a median filtering to the intermediate flow result prior to the warping step to the next computation level.

The reason for its surprisingly good results is straightforward. The errors in flow estimation which appear on the coarse image scale are propagated via a warping step to the next level. On this level the incorrect flow serves as an initialization for optical flow estimation. This results in an accumulation of errors during incremental computation procedure. The median filtering allows to suppress flow errors already on earlier computation levels. With this approach the current optical flow solution is computed via $\mathbf{w}^{k+1} = M_p(\mathbf{w}^k + \mathbf{dw}^k)$, where M is a median filter and p is its mask size.

In [SRB10, SRB14] it is concluded that optimal size of the median mask is (5×5) , which outperforms both (3×3) and (7×7) settings. In our work we generalize the median filtering approach and use a more adaptive strategy for the flow correction. First, we might observe that on the most coarse computation level, when the image size is the

smallest (around 20-50 pixels) median mask of size (5×5) may oversmooth the flow result, which can lead to inaccurate flow estimation. To avoid that, we apply less strict filtering on coarse levels using a smaller mask of 3×3 pixels.

Additionally, in the presence of high amounts of noise or severe artifacts we use a selective filtering using a larger mask (7×7 and 9×9). We perform such filtering in homogeneous image regions determined using the same median filtering (7×7 or 9×9) on the original image. We present evaluation results in the experimental section 5.4.3.5.

2.3 Summary

In this section we summarize all models and features which are available in our optical flow framework. In Table 2.1 we provide the references to the original papers, where a particular approach was introduced or discussed, as well as give a reference to the corresponding parts of this work.

Table 2.1: A summary of methods implemented within the optical flow framework presented in this work.

	Model	References	Section
Data term	Brightness constancy	[LK81, HS81]	2.1.2.2
	Gradient constancy	[Sch93, BBPW04, PBB ⁺⁰⁶]	2.1.2.3
	Higher-order derivatives	[PBB ⁺⁰⁶]	2.1.2.4
	Multiple image features	[BBPW04, ZBW11]	2.1.2.5
	Combined Local-Global	[BWS02, BWS05]	2.1.3.3
	Robust Modeling	[BA91, BA96, SRB14, BSL ⁺¹¹]	2.1.3.1
	Joint and Separate Modeling	[BW05, ZBW11]	2.1.3.2
	Data Normalization	[LV98, ZBW11]	2.1.3.4
Smoothness	Homogeneous	[HS81]	2.1.4.1
	Image-driven	[Sch93, ASW99, WS01a]	2.1.4.2
	Flow-driven	[Sch94, BBPW04, PBB ⁺⁰⁶]	2.1.4.3
	Spatial-temporal	[Nag90, WS01b]	2.1.4.4
	Adaptive smoothness	[ZBW11]	2.1.4.5
Optimisation	Coarse-to-fine strategy	[BA96, MP98, BBPW04]	2.2.4.2
	Flow median filtering	[WPZ ⁺⁰⁹ , SRB14]	2.2.6
	Multiple results fusion	[LRR08]	2.1.2.5
	User-driven landmarks	[Fea03]	2.1.2.6
	Confidence measures	[Fua93, BBH03, BW06, XJM10]	2.1.5
	Parameters optimization	[ZBW11]	5.3
Compute	2D images	—	2.1.2, 2.1.4
	3D images	—	2.1.6
	Efficient schemes	[BBPW04, Bru06, TBKP12]	4.2.1
	GPU computation	—	4.2.2

Chapter 3

Data Analysis Framework

This chapter is dedicated to analysis of time-resolved X-ray data. It divides into two extensive sections: data enhancement (Section 3.1) and further analysis of the computed optical flow results (Section 3.2).

In Section 3.1 we start the discussion of data pre-processing (or filtering) to correct image defects. This step always precedes any further data treatment and optical flow computation. We consider three topics which are of key importance: image noise filtering, brightness corrections and contrast adjustment.

In Section 3.2 we give a list of data analysis routines which we employ in the application section of the current work. These routines, together with a variety of optical flow methods constitute our framework for time-resolved data analysis. We distinguish five individual *tasks*: analysis of the computed motion field; motion-based segmentation; detection of temporal changes which are not attributed to the motion itself; automated tracking and image registration/alignment.

3.1 Data Preprocessing

In this section we present a first step in our data treatment pipeline, namely data pre-processing. The purpose of this step is to correct various image defects that might be present due to non-optimal experimental conditions, physical properties of the process, imprecise detector system, or artificial errors introduced by an image reconstruction procedure. Such image defects might deteriorate the result of data analysis and thus should be corrected before any further optical flow computation or analysis steps are undertaken.

We assume that data which is provided for the image processing already represents the best possible quality which can be achieved by the data acquisition process. Otherwise, the experimental conditions should be readjusted and optimized. However, this is not always possible and frequently the image acquisition system is employed to its technical limits.

As an example, image noise affects the ability to detect small feature details, especially for the low contrast data - when brightness level differs only slightly for separate image features. To increase signal-to-noise ratio and, thus, to improve data quality, it is necessary to increase the number of detected photons. This can be done by increasing the beam intensity, increasing the exposure time or acquiring and averaging several image frames. All of this, however, will lead to a higher dose deposition, which can affect the viability

of the specimen in the case of biological applications. Another option is to increase the pixel size by combining several neighbor pixels, which will allow to detect more photons, resulting in a higher signal-to-noise ratio. But, in this case, the improvement in signal-to-noise ratio is obtained at the expense of poorer spatial resolution. This illustrates a fundamental trade-off between image quality and dose deposition. Moreover, there is also a trade-off between individual parameters of image quality, i.e. spatial resolution, features contrast and noise. Furthermore, the optimization of experimental setup could be time consuming and, given the limited time for experiments on a synchrotron, dealing with image data as a post-processing step could be more practical. That is why it is reasonable to sacrifice the image quality for the ability to capture physical characteristics of dynamical process, and then utilize data processing as a correction step.

It is important to emphasize, that by the term *data preprocessing* we assume both data restoration (correction) and data enhancement. However, both procedures are not the same and serve different purposes. Unlike data enhancement, which is a subjective routine and aimed on improving visual appearance of certain image details or features, image restoration is based on mathematically justified image correction. There might be cases when the same image processing procedure serves both purposes. For example, correcting image brightness variations could improve visual analysis of image sequences, since image flickering could be distracting and obscure important changes. On the other hand, it could also correct pixel data for the use of certain optical flow models (e.g. grey value constancy assumption).

3.1.1 Noise Filtering

Image noise is the unwanted fluctuations of the pixel brightness in an image. This results in a degradation of image quality and deteriorates data analysis. The noise which is generated within an imaging system is usually a combination of a number of independent noise sources. In general, it is not possible to identify, and thus to filter individual noise contributions separately. To quantify image variations due to the noise, a scene in which the resulting image is expected to have a uniform brightness might be used. Then, a noise content of the image (noise power P_n) is given by the variance, i.e. the square of the standard deviation, of the pixel grey values in local image region. To estimate the noise influence one should compare it with the average amount of intensity in the same region (signal power P_s). The *signal-to-noise* ratio (SNR) is the ratio of the intensity of the signal (pixel brightness) to the noise power $SNR = \frac{P_s}{P_n}$. However, in cases when noise properties cannot be described as random or having *normal* statistical distribution, it is more difficult to estimate the amount of noise.

In the following sections we shortly describe a number of *noise models* which are the most relevant for the processing of X-ray images and optical flow analysis. Then we provide correction techniques (noise filters) which allow us to reduce the amount of noise and to recover useful information. This, in turn, will improve the results of optical flow computation.

3.1.1.1 Noise Models

The common sources of noise in images taken with X-ray imaging systems are photon noise, which originates due to the discrete nature of X-ray radiation (and electromagnetic

radiation in general) and interaction of X-ray with matter, and electronic noise from the digital detector systems. Additionally, the process of digitization also adds noise (quantization noise), but we assume that this type of noise does not contribute much to the overall noise level.

Gaussian noise

A popular noise model which is considered in various fields of Signal Processing and Computer Vision is *Gaussian noise*. It has a probability density function (or normalized histogram), which is given by:

$$G(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

where x is the pixel brightness, μ is the mean brightness value and σ is its standard deviation. This model is a very convenient from the mathematical point of view and simulates many real world random (stochastic) processes. That is why it is widely used in image processing to contaminate images with noise. However, this model does not adequately represent physical properties of image acquisition process in most medical systems, including X-ray imaging, where an image has a spatial and temporal randomness characterized by Poisson statistics.

In the current work we decide to keep this noise model for the sake of having a systematic experimental approach and direct comparability with the literature on optical flow. We start all the experiment on noisy images with this simple model and then proceed with the more suitable ones such as photon and impulse noise models.

Photon noise

Photon noise (also known as quantum noise or shot noise) arises from the statistical properties of electromagnetic radiation - an imaging sensor receives within a time interval of Δt (exposure time) on average N electrons by absorption of photons. The average rate of received photons per unit time λ is given by $\lambda = \frac{N}{\Delta t}$. During each exposure a different number of photons are registered by the detector element. A random process in which an average $\lambda\Delta t$ events are counted is described by a Poisson distribution $P(\lambda\Delta t)$:

$$P(\lambda\Delta t) = \frac{(\lambda\Delta t)^n}{n!} e^{-\lambda\Delta t}, n \geq 0,$$

with the mean and variance: $\mu = \lambda\Delta t$ and $\sigma^2 = \lambda\Delta t$ respectively. An important property of a random process according to Poisson distribution is that it is not independent of a signal and is not additive. Additionally, the signal-to-noise ratio increases as an average brightness level gets larger, which means that the more x-ray photons are detected, the higher the signal-to-noise is and the less noise is contained in the image data. The photon noise is a fundamental and unavoidable source of noise in medical imaging. For an efficient and optimized imaging system it is the dominant source of random fluctuation in image data [Dou09].

An important contribution of our work is that in addition to a classical Gaussian noise model we include an extensive discussion of performance of optical flow methods on a physically justified noise model such as photon noise.

Spike noise

Another type of noise which could be useful to consider for our class of applications is so-called *impulse noise* (also salt-and-pepper or *Spike* noise). This type of noise is usually caused by the errors in data transmission or damaged sensor elements of the digital detector system ("dead" or saturated pixels). The strength of the corruption by impulse noise is usually of much more influence compared to other types of noise, since such type of noise is recorded as an extreme values, so they are equal or close to the maximum and minimum values of dynamic range of an image. This type of noise is usually quantified by the percentage of pixels which are corrupted.

Despite the fact, that sometimes it is possible to correct such corrupted pixels using the dark field correction (which will be described later in Section 3.1.2.2), we choose to include this type of noise in our experiments on noise. We consider this type of noise as very helpful to evaluate the performance of optical flow methods and data correction techniques with respect to data outliers.

3.1.1.2 Noise Filters

In the previous section we introduced several noise models which we will use for our experiments on noisy data (Sections 5.4.1 and 5.4.2.1). Here we present methods to restore image details from the noisy images, with the aim to possibly improve results of optical flow computation. There is a tremendous amount of literature on image denoising methods. For this work we limit our experiments only on those methods which are widely available, easy to implement and are well understood from the mathematical point of view. Here we do not aim to fully cover this topic. For in-depth information on denoising methods the reader is referred to the extensive literature. To enhance image data with respect to noise we consider four filters: Gaussian filter, bilateral filter, median filter, anisotropic diffusion filter.

An important assumption for any denoising method is that the pixel size (resolution) is much smaller than the important details which we aim to recover.

Gaussian Filter

A commonly used approach to filter noise is a *Gaussian smoothing*. It can be implemented as a convolution of the image with the Gaussian function. This filter reduces noise from the image, however, it also blurs image edges and reduces image contrast. It can introduce other artifacts such as merging of nearby structures. Moreover, the data outliers are not completely filtered, but only averaged in a local neighborhood, which means that the corrupted pixel data could be spatially extended. Note, that Gaussian smoothing corresponds to the linear isotropic diffusion process. The implementation of Gaussian low-pass filtering is done via computing a weighted average of pixel values in the neighborhood, with the weights decreasing with the distance from the neighborhood center.

Bilateral Filter

Bilateral filtering removes the noise while preserving image edge information, which is done by a weighted average of intensity values from nearby pixels. An important property of the bilateral filter, is that the weights depend on both geometrical closeness of the pixels and their photometric differences (e.g. grey values) [TM98].

The filtering procedure can be expressed in the following way:

$$I_{bilateral}(\mathbf{x}) = \frac{1}{W_p} \sum_{\mathbf{x}_i \in \Omega} I(\mathbf{x}_i) f_r(|I(\mathbf{x}_i) - I(\mathbf{x})|) g_r(|\mathbf{x}_i - \mathbf{x}|),$$

where the normalization factor is:

$$W_p = \sum_{\mathbf{x}_i \in \Omega} f_r(|I(\mathbf{x}_i) - I(\mathbf{x})|) g_r(|\mathbf{x}_i - \mathbf{x}|),$$

where I is the original image, x is the coordinate of the current pixel being processed; Ω is the spatial window mask centered at location x ; f_r is the range kernel to filter differences in pixel brightness; g_s is the spatial kernel to filter differences in geometrical distances.

Among the advantages of the bilateral filter are simple implementation and non-iterative nature of the algorithm.

Rank Filters

A drawback of smoothing or averaging filters is that they produce undesirable blurring on object boundaries, which are represented by image edges. A possible solution to this problem is to discriminate pixel values by ranking them in a local neighborhood. A common choice is a *median filter* which selects the median value in the ordered list of grey values and uses this value as a new result for the central pixel. Median filter provides excellent performance for certain types of noise, especially impulse noise, which was discussed in the previous section. If a pixel contains an extreme brightness value ("dead" or saturated pixel), it is substituted by an acceptable value from a local neighborhood. The important property of the median filtering is that it is an *edge-preserving filter*. However, for very large neighborhood sizes small image details could be completely eliminated.

To implement a median filtering one may choose between different sizes and shapes of the neighborhood. Several mask patterns are possible: a 4-nearest neighbor cross, a (3×3) square, a (5×5) circular, etc. These different neighborhood patterns are shown in Figure 3.1. Typically, a square mask is easier to implement, however, as the mask size increases the use of a pattern which approximates a circular region is important to produce more isotropic filtering [Rus11]. For implementation of median filtering in the current work we opt to use a circular median filtering mask.

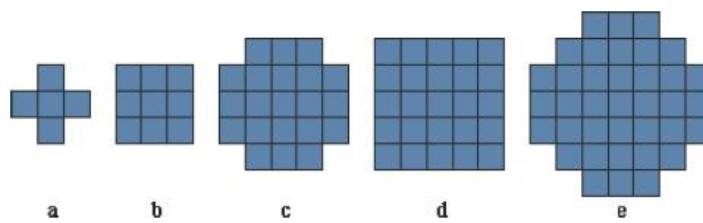


Figure 3.1: Shapes of 2D neighborhood mask for median filtering. (a) 4-nearest neighbor cross. (b) a (3×3) square. (c) (5×5) circular. (d) (5×5) square. (e) (7×7) circular. Image: [Rus11].

Another useful approach is to consider a more sophisticated ranking of neighborhood information. This step is driven by the fact, that median filter removes small details which

are twice smaller than the size of the median mask and also removes sharp corner details. A special case of corner-reserving median filter, is a *hybrid median filter* [NHN87, Rus11]. This filter ranks all the pixels in different sets. In the first group there are 4 nearest neighbours forming a "+" sign and the second group are pixels which are located in diagonal locations with respect to the central pixel (forming a "x" sign). Then, median values from these two sets are compared with the central pixel. The idea is sketched in Figure 3.2.

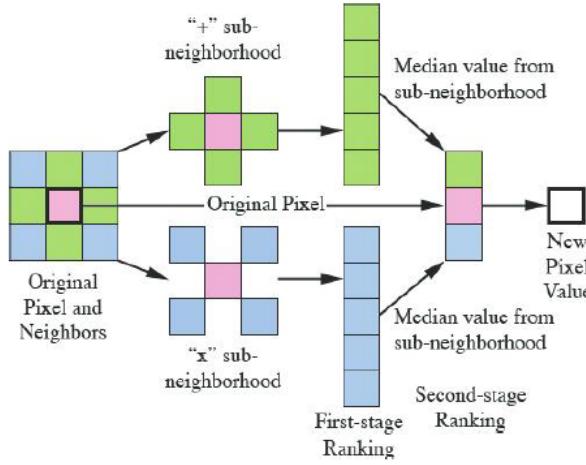


Figure 3.2: A sketch of hybrid median filter, which preserve corners and small coherent structures. Image: [Rus11]

Anisotropic Diffusion

Anisotropic diffusion (also known as Perona - Malik diffusion [PM90]) is a method which aims to reduce image noise without removing image edges representing an important features. Additionally, such types of filters can posses even edge-enhancing properties [Wei98]. The smoothing is done by means of a diffusion process in which the strength of the diffusion is controlled by derivatives of the image brightness values. Anisotropic diffusion is define as:

$$\frac{\partial I}{\partial t} = \operatorname{div}(c(x, y, t) \nabla I),$$

where $I(x, y, t)$ is an image in a domain $\Omega_2 \subset \mathbb{R}^2$, ∇ denotes gradient operator, $\operatorname{div}()$ is a divergence operator, $c(x, y, t)$ function controls the rate of diffusion depending on image data at pixel location (x, y) . In a seminal paper of Perona and Malik the diffusivity $c(x, y, t)$ are:

$$c(|\nabla I|) = e^{-(|\nabla I|/\epsilon)}$$

or

$$c(|\nabla I|) = \frac{1}{1 + (\frac{|\nabla I|}{\epsilon})}$$

where $|\nabla I|$ is a magnitude of the image gradients and ϵ is constant which is choose experimentally or depending on a noise scale.

Anisotropic diffusion is a flexible and highly adjustable filter. With different parameters one may obtain a very distinctive filtering results. However, what might be an advantage to produce better results, also can be a drawback, since it is not easy to optimize

numerous parameters to get the best possible result. Additionally, the implementation is not trivial and the filter is computationally demanding (being an iterative method).

Carefully weighting all pros and cons we use this filter for our quantitative comparison of denosing filters, since anisotropic diffusion is a powerful and flexible method, with many degrees of freedom.

We use an implementation based on the paper of [TD05]. A freely available implementation can be access via a web link:

<http://rsb.info.nih.gov/ij/plugins/anisotropic-diffusion-2d.html>.

3.1.2 Brightness Correction

The most commonly used assumption in image analysis to identify or compare pixel values is that the region representing the same object or feature should have the brightness value. However, in practice for real-world imaging scenarios such strict conditions could not always be fulfilled.

For X-ray imaging the constancy and uniformity of the background brightness distribution can be affected because of various reasons, such as non-uniform beam profile due to spatio-temporal fluctuations of a light source, etc. Usually such effects can be corrected by recording a background image (without an object in the field of view) and then removing non-constant brightness variation (see Section 3.1.2.2). Depending on the image content, this can be implemented as a subtraction or division operation with a background and the image with an object. However, this procedure is not always feasible, especially for *in-situ* or *in-vivo* experiments, when one acquires a continuous sequence of images and brightness variations occur during the recording time. For this case we propose a proper brightness correction procedure. Here we describe the most popular methods to achieve this and later evaluate how different correction procedures influence the accuracy of optical flow estimation (see experimental Section 5.4.2.2).

3.1.2.1 Illumination Models

As we outlined previously, background brightness distribution can be non-constant over time or non-uniform spatially for a sequence of X-ray images. This property of X-ray data is of crucial importance for the choice of appropriate data constancy assumptions for the design of optical flow models (see Section 2.1.2). In the literature on optical flow methods this problem is referred to as "changing illumination conditions".

To correctly model optical flow assumptions it is important to consider realistic illumination scenarios. Two important models in the literature on the optical flow are: *additive* illumination and *multiplicative* illumination [vdWG04, MBW07]. Additionally, one may distinguish between *global* and spatially *local* changes. In section 2.1.2 on the available data terms we already discussed that the gradient constancy assumption is invariant under global additive illumination changes, however it is not suited for the case if there is also a multiplicative part.

It is important to note, that increasing an exposure time while recording an X-ray image corresponds to global multiplicative brightness changes. Thus, to apply optical flow methods, an appropriate choice of the data term which is robust under this type of brightness changes is an crucial aspect. This will be discussed in the experimental section 5.4.3.2.

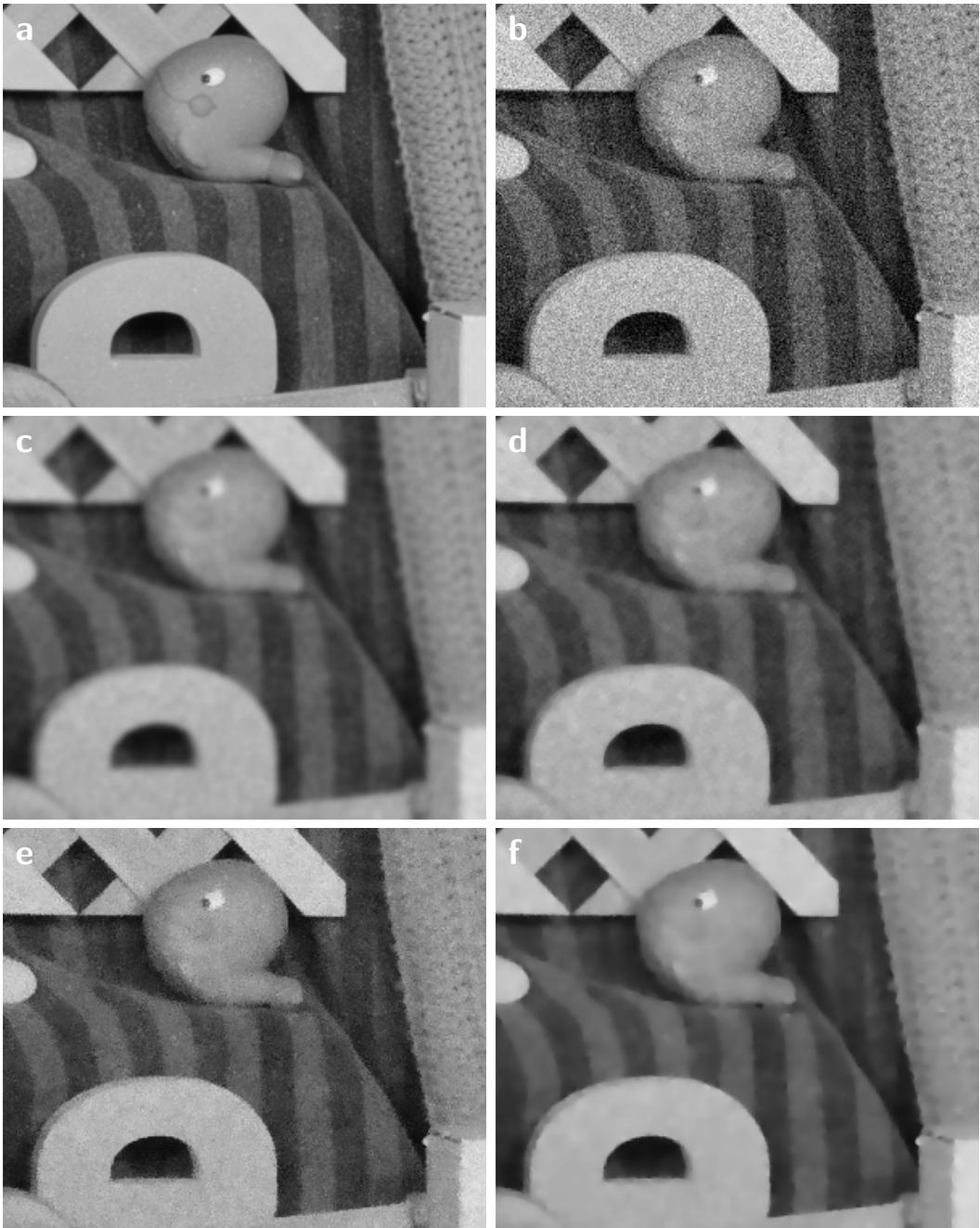


Figure 3.3: Performance of noise filters on *RubberWhale* dataset with Gaussian noise with $n_\sigma=10$. (a) First frame of original image. (b) With Gaussian noise with $n_\sigma=10$ added. (c) Gaussian smoothing filter. (d) Median filter. (e) Bilateral filter. (f) Anisotropic diffusion filter.

Typically, to tackle with varying brightness conditions optical flow methods make use of different photometric invariants derived from color models [MBW07, ZBW11]. Such transformations are invariant under general illumination changes. However, since color information is not available for X-ray images we are enforced to search for another reliable solutions, which we present in the following sections.

3.1.2.2 Flat-field Correction

The simplest approach to correct background non-uniformities is to perform so-called *flat-field correction*. In this case a "flat" image, without an object in the field of view is recorded. Then this image is used to eliminate the brightness pattern from the image with a specimen. In a general form the flat-field correction can be computed as:

$$I_{filt} = \frac{I_{raw} - I_{dark}}{I_{flat} - I_{dark}},$$

where I_{raw} is a recorded raw image, which has to be corrected, I_{flat} is a flat-field image and I_{dark} is a dark-field image. The dark-field image is an image captured with no illumination. In other words, it is an image of the sensor noise. An average of several dark-field frames provides a possibility to correct for the fixed-pattern noise, including dark ("dead") and saturated ("hot") pixels. For the case of X-ray data, since the recorded images represent the absorption of a sample according to the Beer's law (for the case when absorption is dominating the image contrast) we may also perform a logarithm transformation to recover the physical thickness of structures:

$$I_{filt} = \frac{\log(I_{raw} - I_{dark})}{\log(I_{flat} - I_{dark})}$$

However, as we mentioned before, a flat-field correction is not always possible for *in-situ* or *in-vivo* experiments.

3.1.2.3 Fitting Background

One approach to eliminate uneven background from the recorded image, without a flat-field image available is to measure an overall brightness in different parts of the image and then interpolate the result. Then the obtained brightness pattern might be used for image correction. A good choice for the interpolation function is a polynomial interpolation, which gives good results for relatively gradual brightness variations. Another choice is to perform least-squares fitting of sample points distributed over the image domain. The distribution of sample points could be automated (for example using a grid or object detection) or manual.

3.1.2.4 Image Leveling

In the case when the background changes more irregularly, so the fitting of a simple function cannot be performed, a different approach might be useful. For this method one assumes that the image features have smaller size than the scale of brightness variations. The background image is estimated by applying an averaging or ranking filter with a large spatial mask. Then, this background image is subtracted from the original image frame to remove brightness variations.

In the current work we employ two types of filters, namely Gaussian smoothing filter and median filter (See Section 3.1.1.2). Important parameters for such a filter are the shape and the size of a spatial mask. One should note that brightness variations can be one-dimensional (e.g. horizontal stripes patterns), in this case the filter can be implemented as a 1D stencil. Figure 3.4 shows the application of image leveling methods to correct uneven brightness patterns.

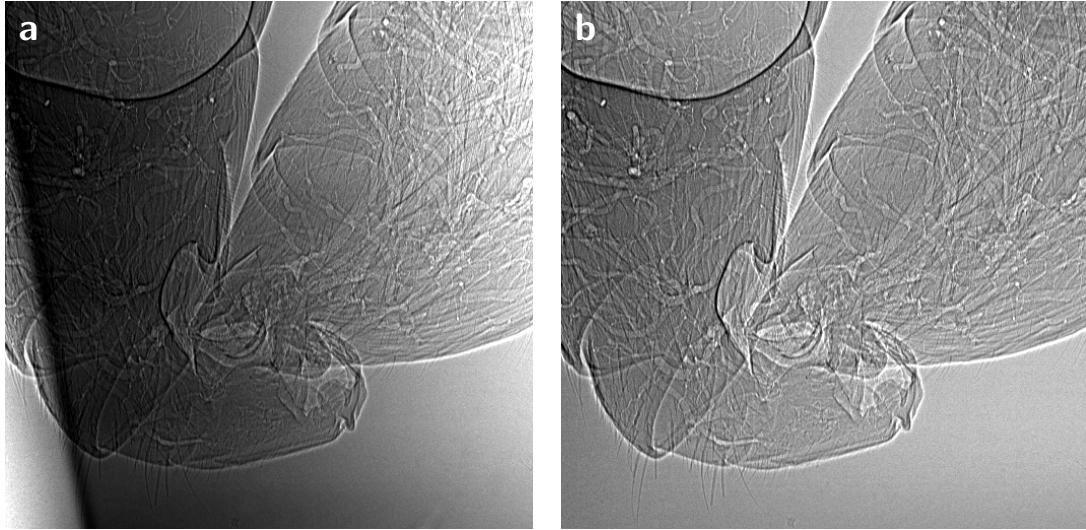


Figure 3.4: Non-uniform background correction using image leveling with the Gaussian smoothing filter with a spatial mask $\sigma = 15$. (a) Original image. (b) Corrected image.

3.1.2.5 Log Transform

Another approach to deal with local brightness variations (including additive and multiplicative intensity changes) is to use a so-called log-derivatives [MBW07]. In this case the original image grey values are transformed into a logarithmic scale, then gradients of these log-derivatives are considered as constancy assumptions for the computation of optical flow. Such transformation is more robust under multiplicative intensity changes.

We evaluate this approach in the experimental Section 5.4.3.2, where we check the performance of different data terms for changing brightness conditions.

3.1.3 Contrast Adjustment

The range of brightness values represented on the image is known as the *dynamic range*. It can be measured as a difference between maximum and minimum values. Ideally, the dynamic range obtained by the interaction of the radiation with the sample - for the case of X-ray imaging - should coincide with the available dynamic range of the detector system. In a case if it is wider than the one of the detector, the image histogram is cutted. The brightness values which fall inside this cutted regions are inevitably lost. Additionally, extreme pixel values which corrupt the range of histogram can be caused by data outliers, such as "dead" or "hot pixels" and can be corrected as a preprocessing step to improve the visibility of image details.

An image property which is related to the dynamic range is a *contrast*. In the case when the dynamic range of an image covers all brightness values produced by the imaging system (including interaction of X-rays with a sample), the image exhibits high contrast. On contrary, when dynamic range is low (only a narrow range of similar grey values present in the image), the image has a low contrast. It is important to note, that the dynamic range and image contrast are related, but not identical concepts. For instance, images with bimodal histogram (peaks on the lower and upper parts of the histogram) exhibit higher contrast than images with smooth, even histogram. For the ideal imaging

session one is interested to both cover the full dynamic range of the image and provide maximum feature contrast.

Since for optical flow methods the pixel grey values are the source of information to solve a correspondence problem, one may try to correct the image contrast not only to enhance the visibility of image details, but also to improve results of optical flow computation. In this section we present a number of techniques to achieve that and in Section 5.4.2.3 we evaluate the influence of these methods on the accuracy of optical flow.

3.1.3.1 Contrast Estimation

In section 3.1.1 we discussed an important characteristic of the image, which is a signal-to-noise-ratio (SNR). Here we emphasize that for the analysis of X-ray images (and medical image in general) a concept which is even more important is a *contrast-to-noise ratio* (CNR):

$$C = \frac{|S_1 - S_2|}{\sigma},$$

where S_1, S_2 are the brightness of structures in the region of interest and σ is the standard deviation of image noise. For medical imaging applications the goal is to distinguish between different structures or between an object and a background. For this case the contrast-to-noise ratio is more descriptive and useful measure, since it relates both image noise and image contrast.

3.1.3.2 Contrast Enhancement

A frequent situation, especially for X-ray imaging with low exposure conditions, is when the images do not have the brightness range that covers the full dynamic range available for a detector system. On Figure 3.5 brightness values on the light and dark ends of the histogram are not used. Expanding the brightness value to cover the full dynamic range may improve the visibility of local image features. This can be implemented with the linear brightness values mapping [GW08]. This operation is often called the *histogram stretching*. The result of this transformation is shown on Figure 3.5b.

In some cases the majority of pixel values occupy a narrow range of a histogram, but a few number of outliers represent extreme values (e.g. hot and dead pixels). In this case the histogram stretching cannot be performed, since the whole histogram range is used. This results in an ineffective use of dynamical range. One way to solve this, is to filter these outliers, e.g. using median filtering. However, this procedure also affects the pixel values in the correct range of brightness range. Alternatively, it is possible to dismiss a certain percentage of pixels on the histogram ends. As a result, these pixels will be saturated with minimum and maximum values, but the rest of histogram range will be used effectively.

Note, that the aforementioned procedure is global and linear. To improve local visibility of image features, one may perform nonlinear, *local contrast enhancement* [PAA⁺87]. This technique perform brightness stretching based on a histogram of a local image region.

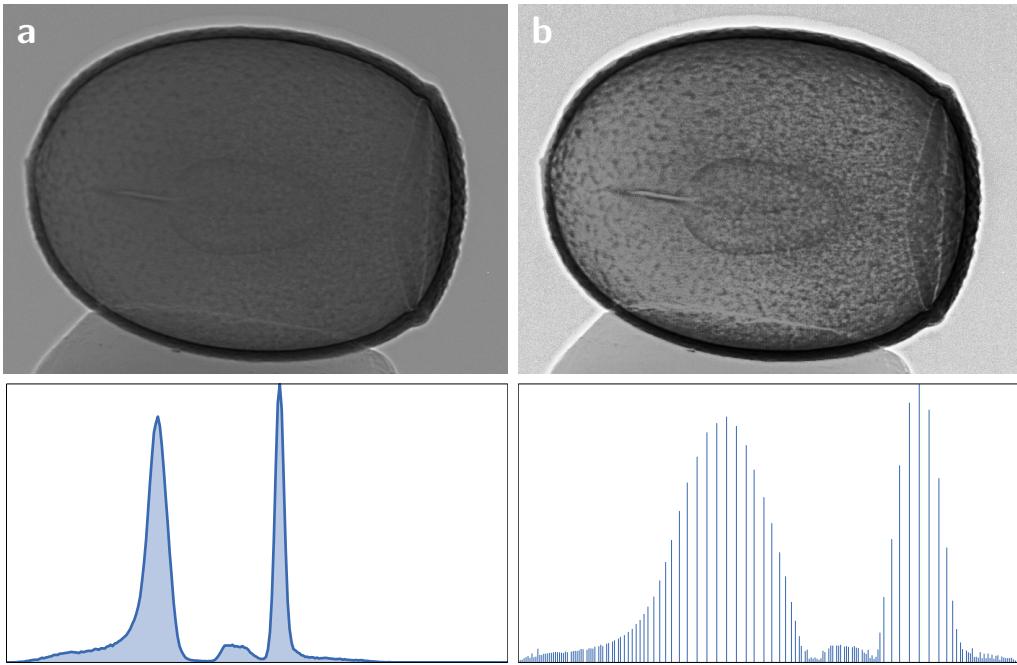


Figure 3.5: Example of a histogram stretching to improve the visibility of local image features. (a) Original radiograph with poor contrast. (b) Radiograph corrected using histogram stretching.

3.2 Analysis Based on Optical Flow

This section is devoted to an overview of data analysis methods, which can be applied after the optical flow results are computed. By further analysis of the obtained motion field one can extract a vast amount of quantitative information about dynamical processes. Although, several techniques can be closely related, in our *data analysis framework* we identify four separate topics:

- Motion / flow analysis
- Motion-based segmentation
- Object tracking
- Image registration and alignment

In the following sections we describe in details each of these analysis methods.

3.2.1 Motion Analysis

In this section we present a common techniques to analyze the motion field after the optical flow is computed. The aim of these methods is to provide quantitative information about dynamical process. All of the presented methods are available in our data analysis framework.

We define a *vector field* as a map $\mathbf{F} : \mathbf{R}^n \mapsto \mathbf{R}^n$ that assigns each coordinate \mathbf{x} a vector $\mathbf{F}(\mathbf{x})$. A vector \mathbf{F} in an n -dimensional Euclidean space can be defined as an ordered list of n real numbers $\mathbf{F} = \{x_1, x_2, \dots, x_n\}$.

3.2.1.1 Magnitude

An important property of a vector is its *magnitude* or length (also referred as velocity). It is commonly defined as a Euclidean norm:

$$|\mathbf{F}| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

Magnitude of the vector field is a useful measure to quantitatively describe physical characteristics of the dynamical process, distinguish between different moving object and analyze the velocity changes over time. To compute the magnitude of the flow field one might be interested to distinguish 4 scales:

- Magnitude of an individual pixel at the location \mathbf{x}
- Average magnitude within a particular object O
- Average magnitude inside a certain region of interest (ROI)
- Average magnitude for the entire image frame $I(\mathbf{x})$

3.2.1.2 Divergence

Divergence is a vector operator that measures as a single scalar value to which extent a vector field behaves as a sink or source at a given point. It can be defined for a continuously differentiable vector field $\mathbf{F} = (F_x, F_y, F_z)$ as follows:

$$\operatorname{div} \mathbf{F} = \nabla \cdot \mathbf{F} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}$$

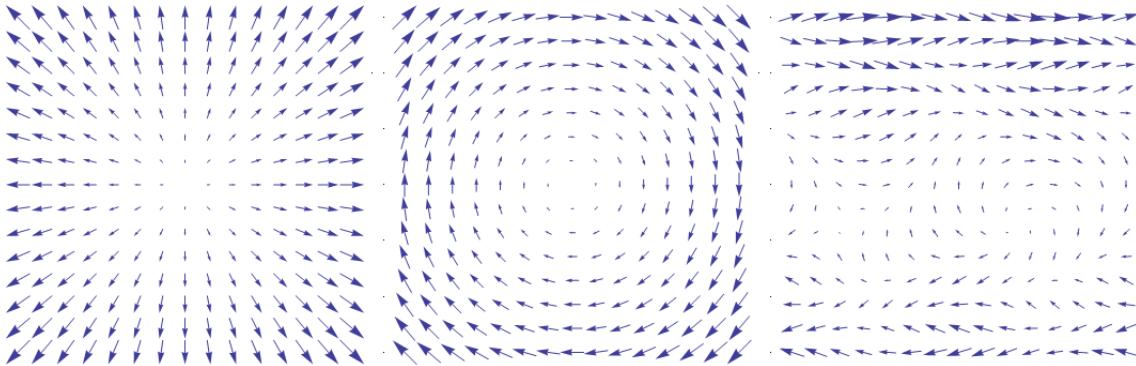


Figure 3.6: Example of different types of vector fields. **Left:** Divergent vector field. **Middle:** Rotational motion field. **Right:** Turbulent flow field with a significant amount of flow discontinuities.

Divergence is a useful measure to analyze changes in velocity magnitude (i.e. acceleration). Additionally, divergence is a good indicator to identify potential errors in the computed optical flow field, since according to the smoothness assumption, the displacement field should vary slowly and violation of this fact can point out to problematic regions.

An example of divergent vector field is given in Figure 3.6.

3.2.1.3 Curl

Curl is a vector operator which describes the rotation of a vector field. At a given point \mathbf{x} of the field $\mathbf{F}(\mathbf{x})$, it is represented as a vector, which characterizes the rotation at that point. The direction of the curl describes the rotations axis and the magnitude of the curl is the magnitude of rotation. For Cartesian coordinates, for a continuously differentiable vector field $\mathbf{F} = (F_x, F_y, F_z)$ the curl is given by:

$$\text{curl } \mathbf{F} = \nabla \times \mathbf{F} = \left(\frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z} \right) \mathbf{i} + \left(\frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x} \right) \mathbf{j} + \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right) \mathbf{k},$$

where \mathbf{i} , \mathbf{j} and \mathbf{k} are the unit vectors for x-,y-, and z-axes. An example of rotational motion and corresponding vector field is given in Figure 3.6.

3.2.1.4 Phase

Another useful measure to analyze the resulting flow field is a magnitude of the flow gradient. This measure is sensitive to any spatial variations of the motion field. For the 2-dimensional vector field $\mathbf{F} = (F_x, F_y)$ the magnitude of the spatial gradient is given by:

$$|\nabla_2 \mathbf{F}|^2 = \left(\frac{\partial F_x}{\partial x} \right)^2 + \left(\frac{\partial F_x}{\partial y} \right)^2 + \left(\frac{\partial F_y}{\partial x} \right)^2 + \left(\frac{\partial F_y}{\partial y} \right)^2,$$

where $|f| = \sqrt{f_x^2 + f_y^2}$ denotes the spatial magnitude and $\nabla_2 = (\partial_x, \partial_y)$ the spatial gradient. For n -dimensional vector field this expression can be generalized in a straightforward way by taking into account additional dimension.

It is important to note, that the given measure is equivalent to the formulation of homogeneous smoothness assumption 2.1.4.1 and can be used to measure discontinuities in the motion field. Figure 3.6 shows an example of a turbulent flow field with a large amount of flow discontinuities.

3.2.2 Object Tracking

Object tracking is an important task in the field of Computer Vision. It is used for a wide range of applications like video surveillance, human-computer interaction, automated video analysis and autonomous vehicle navigation. For our application needs we are interested to track and analyses spatio-temporal evolution of various objects imaged by means of X-ray radiation: migrating cells, moving particles, changing structures, etc.

In the most general way, *tracking* is a procedure of estimating the trajectory of an object within an 2D image plane (or 3D volume for a tomographic data) as it changes its position over time. Tracking involves three key steps: detection of moving objects, tracking of these objects within an image sequence, and analysis of their trajectories to learn about their dynamical behavior.

In order to implement a tracking procedure one should consider a number of important aspects. First, a suitable representation of the object should be defined. The next aspect is to choose an image feature which will be used to recognize the object. And, finally, an appropriate tracking strategy should be chosen.

We describe these aspects step by step in the following sections. Note, that the object tracking is a well established and an extensive topic. Here we cover only those approaches

which will be implemented within our data analysis framework (see Section 3.2), but additional methods could be easily incorporated upon the need. For more details we refer the reader to the literature on tracking methods [YJS06, TP06].

3.2.2.1 Object Representation

Here we describe common approaches to represent an object shape. In general, the choice of object representation determines the tracking algorithm. It is possible to distinguish the following representations:

- *Points.* The simplest representation of an object is using a point. This point could be, for example, a geometrical center of the object (centroid) or some other characteristic feature. Moreover, complex objects can be represented by a set of points. In general, a single point representation is well suited for tracking small objects (e.g. particles, cells).
- *Primitive geometric shapes.* In this case an object is represented by a simple geometrical shape, such as a rectangle or an ellipse. The motion for such shapes is usually assumed to be described by affine transformations. Such geometrical shapes are useful to represent rigid object.
- *Object contours.* Contour representation specify the boundary of an object. This type of presentation is useful for tracking non-rigid objects. The contours are usually described using a special mathematical formulation, known as geodesic active contours or snakes [CKS95]. However, in the scope of our work we do not use this formulation. Instead, to represent a contour we use a multiple points, distributed on the edge of an object. This simplifies and reduces the number of tracking algorithms in our data analysis framework.
- *Articulated shape or skeletal models.* Articulated objects are composed of several parts connected together with a joints mechanism. The relations between different parts are driven by a specified kinematic motion model (e.g. angles relations, stiffness). This representation is useful for tracking human or animal body parts. For our class of applications this model can also be useful, however in the current work we do not implement it.

As a base representation of objects we opt to choose a point representation, which is general and allows to define small objects with a single point, complex objects as a set of points and non-constant objects contours as a set of points distributed along the boundary of an object. Additionally, this allows us to have only one type of tracking algorithm (see Section 3.2.2.3).

3.2.2.2 Features for Tracking

The crucial aspect is a choice of image features or object properties which are used for tracking. The requirements for such features are: they should be unique and provide a good description the object, they should also be robust to noise and other artifacts. The commonly used visual features are:

- *Color.* The primary information about an object is its color or a brightness value. In the visual light imaging scenario the color is mainly determined by the illumination conditions of the scene and surface properties of the object (reflectance, diffusivity). In our case, the color represents the result of interaction of X-rays with a sample (e.g. attenuation of the upcoming radiation).
- *Edges.* Another characteristic property of an object is its boundaries. For this case an edge detection algorithm is used to identify the changes in image intensities, representing an object contour. There are several advantages for this feature. First, the edges are less sensitive to illumination changes. Second, they additionally provide orientational information. There are a number of easy to implement and accurate edge detection techniques available [Can86, BKD01].
- *Optical Flow.* In this work the main feature used for tracking is a displacement vector computed by an optical flow algorithm. This vector is available for every pixel in the image (dense displacement field) and directly shows the new position of the given pixel on the next frame.
- *Texture.* Texture is feature which describes brightness variations on the surface or within an object. These variations might be caused by a pattern in the illumination or properties of the object itself. Usually, to quantify the texture a number of texture descriptors are used. Similar to an edge feature, texture based information is also robust with respect to illumination conditions and noise.

Depending on the application and available data, a combination of aforementioned features can be used. In the scope of this work we use an optical flow as the main source of information to perform tracking. In order to track contours of non-rigid or complex objects we additional may use the object edges to distribute a set of points for further tracking.

3.2.2.3 Object Tracking

After the object representation and features of interest are chosen, the final step for the tracker algorithm is to estimate a trajectory of a moving object by locating its position in every frame of the image sequence.

The tracking procedure starts by detecting or identifying initial position of objects. In our work we mainly use the point representation for small objects and multiple points to represent complex objects. The distribution of points representing an object can be done manually by the user or distributed automatically. In case of automatic selection the points can be distributed randomly, using specified positions on a grid or according to image features discussed in Section 3.2.2.2.

The model which is selected to represent an object constraints the type of motion or deformation it can perform. For a point representation, the object motion is limited to translations. Tracking of an object represented by a single point is trivial, we just follow each point using a computed displacement vector. A non-rigid object represented by multiple points defining its contour can move freely. In this case, by analyzing the new positions of the contour it is possible to learn about object's appearance. For a

complex rigid object represented by multiple points we may introduce a number of motion constraints [YJS06] to reconstruct its affine transformation. These constraints are schematically depicted in Figure 3.7.

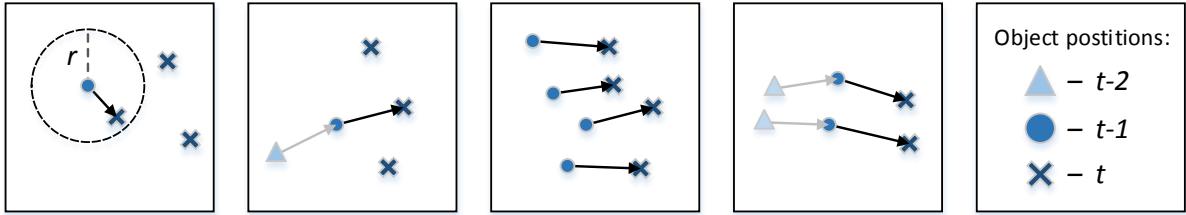


Figure 3.7: Different motion constraints. **Left:** Maximum velocity constrained to the radius r . **Middle Left:** Small velocity change constraint. **Middle Center:** Common motion. **Middle Right:** Rigidity constraints. Triangles denote object position at frame $t-2$, circles denote object position at frame $t-1$, and crosses denote object position at frame t . **Right:** Scheme of pixel codes on different time frames. Image adopted from [YJS06].

- *Maximum velocity* specifies an upper bound of the object velocity which it can undergo. This can be specified by a circular neighborhood around an object.
- *Small velocity change* (temporal motion smoothness) assumes the direction and magnitude of the object's displacement does not change rapidly.
- *Common motion* (spatial motion smoothness) constrains the displacements of nearby points within the object to be similar.
- *Rigidity* assumes that objects are rigid, resulting that the distance between any two points which belong to the object are unchanged after the .

3.2.3 Image Registration and Alignment

Image registration is a process of aligning two or more images of the same scene taken at different times, from different views, or by different imaging techniques. Registration is required in many fields, such as remote sensing, in medicine and in Computer Vision. .

The task for image registration is to find the optimal spatial and intensity transformations so the input images are matched. Image registration can be defined for two-dimensional images as a mapping between a target $I_1(x, y)$ and a reference $I_2(x, y)$ images, where $I(x, y)$ represents image intensity. Then, the mapping between these images can be expressed as following:

$$I_2(x, y) = g(I_1(f(x, y))),$$

where f is a function which maps spatial coordinates (x, y) into new locations (x', y') , such as $(x', y') = f(x, y)$ and g is an intensity transformation.

Depending on the task and application field, it is possible to distinguish four classes of specific registration problems:

- *Multimodal registration (different sensors):* Registration of images of the same scene acquired using different imaging modalities. Example: Combine structural information from CT and functional brain activity from MRI imaging.

- *Template matching*: Match a reference pattern or a model with an image. Example: Registration of an image pattern with the model representation of the scene or atlas.
- *Viewpoint registration (different views)*: Registration of a scene taken from different viewpoints. Example: Depth or shape reconstruction in Computer Vision.
- *Temporal registration*: Registration of images of the same scene taken at different time, under different imaging or physical conditions. Example: Detection and monitoring of changes in a scene, object or specimen.

In our work we mainly focus on the task of temporal registration to expose changes between subsequent images or perform image alignment of object position.

In general, image registration procedure consist of the four stages [ZF03], which are similar to the common steps for object tracking (see Section 3.2.2):

- *Feature detection*. Detection of salient and distinctive features such as objects, characteristic points, edges, contours, lines, corners, etc. Such features can be placed manually or detected automatically.
- *Feature matching*. Establish the correspondence between the features detected in the target and the reference image. To perform a matching procedure a number of spatial constraints and similarity measures are used.
- *Transformation estimation*. A mapping between the target and the reference image is estimated according to established feature correspondences.
- *Image resampling and transformation*. As a final step, the images are transformed towards each other using the estimated mapping function. For the mapping and resampling of intensity values an appropriate interpolation techniques is used.

All the differences within images in terms of pixel positions or intensity values due to the temporal changes or different acquisition conditions are referred as *variations*. It is important to distinguish between distortions and other variations. *Distortions* are variations which are the source of misregistration. Essentially it is the distortions between images which we would like to remove by registration procedure and reveal the changes of interest. These distortions might be caused by the image noise, intensity changes, sample or detector movement or other unwanted effects [ZF03].

Another important aspect, is to distinguish between global/local transformations and global/local variations [Bro92]. A global transformation is described by the same mapping (e.g. single equation) and transforms the whole image. A local transformation maps different pixels according to their spatial localization. The same holds for variations: some variations are due to the global process like camera misalignment or global sample movement, and some variations correspond to local differences. To perform image registration it is crucial to take into account which kind of transformation is applied for which kind of variations. For instance, images may have local variations, but a global transformation can be used to align for a global change to assist in revealing small local changes. We will cover these aspects in a Section 3.2.3.5 where we discuss different registration strategies. In the following section we describe all the relevant steps in image registration.

3.2.3.1 Feature Detection

A crucial step for image registration is the selection of feature descriptors. These features should be distinct, well distributed over the image and robust under different image artifacts (e.g. noise, brightness changes, artifacts). It is possible to categorize most of the registration techniques according to feature detection approach in the following way:

Area-based detection. Area-based methods do not perform a separate feature detection step. Instead, these approaches perform the feature matching procedure directly, according to a specific similarity measure. We discuss these methods in the next section (See Section 3.2.3.2).

Feature-based detection. This approach is based on the detection of salient and distinctive feature to perform image registration. Whole range of characteristic information extracted from the image data can be used for this purpose. Such features might include: specific points, lines, corners, contours, texture, geometric shapes, etc. An important difference from the area-based methods is that this kind of features represent higher level of information and could correspond to a physical model of the scene. The procedure for feature detection is closely related to the similar task for object tracking discussed in Section 3.2.2.2, so the same set of feature detection tools might be used.

3.2.3.2 Feature Matching

After the features are detected, the next step is to perform feature matching. In this section we describe the methods which will be used in the scope of this work.

Area-based matching. A first category of methods are called area-based methods. Such methods are also known as *correlation-like* methods or template matching techniques [ZF03]. As we already mentioned in the previous section 3.2.3.1, these techniques combine the feature detection and matching procedures. The images are registered without characteristic data points, instead the whole image domain is used.

A classical approach for area-based registration is a *cross-correlation*, which is a statistical method to find similarity between images. For a template T and target image I , the two-dimensional cross-correlation function estimates the similarity for each translation (u, v) by:

$$CC(u, v) = \sum_{x,y} T(x, y)I(x - u, y - v).$$

If the best template match with a target image is given at a translation of (i, j) , the cross-correlation function will have its peak at $CC(i, j)$.

In the case, if the brightness of the image and template differs due to the varying imaging conditions, the images can be first normalized. The normalized cross-correlation is then given by:

$$NCC(u, v) = \frac{\sum_{x,y} (T(x, y) - \mu_T)(I(x - u, y - v) - \mu_I)}{\sqrt{\sum_{x,y} (T(x, y) - \mu_T)^2(I(x - u, y - v) - \mu_I)^2}},$$

where μ_T and μ_I are mean values of the template T and the image I respectively.

A similar, but more intuitive measure computes the sum of the squared differences (SSD) between a template and an image:

$$SSD(u, v) = \sum_{x,y} (T(x, y) - I(x - u, y - v))^2.$$

Another variant of area-based techniques is the so-called sequential similarity detection algorithm (SSDA). Which simply estimates an absolute differences between two images being matched. This measure is given by:

$$SSDA(u, v) = \sum_{x,y} |T(x, y) - I(x - u, y - v)|.$$

Area-based techniques are useful for images which are misaligned by rigid transformation, preferably a translation. For more complex geometrical transformation (e.g. fast rotation) the model should be modified, which usually leads to a significant increase in computation load due to the extended search space. Another disadvantage of the area-based methods is a poor performance for noisy data or homogeneous images without prominent details or structures. In these cases no distinct peak position could be estimated. For example, low-contrast data with noisy background or rotated spherical shape of the embryo with a lot of similar cell structures.

Feature-based matching. For this class of techniques we assume that features in the reference and the target images are detected. The next step for feature-based matching is to establish correspondences between them. There are a vast variety of methods to find such correspondences. We do not contemplate to describe these algorithms. For the extensive overview of different matching techniques we refer the reader to surveys on image registration methods [Bro92, ZF03]. One popular choice of feature-based matching technique is based on scale invariant feature transform (SIFT) [Low04]. These methods show good performance for a broad range of transformation models between images. However, their use exceeds the scope of the current work. In this work the feature correspondences for sparse or dense points are provided by the optical flow method and described by the computed displacement vector.

3.2.3.3 Transformation Estimation

After correspondences between features are established the mapping function which overlays the target image with the reference is constructed. The key component of the registration procedure is a type and properties of a spatial transformation which implements such alignment. Since a wide range of transformation types may present within each image, the registration technique is restricted by a particular transformation model or parameters. This transformation should remove only those image variations, which are the result of distortion and should be corrected. Other image difference (changes of interest) should remain unaffected by the registration procedure and will be used for further analysis.

As we already discussed in the Section 3.2.3, transformation can be global and local. The most common types of global transformations are rigid, affine, projective, perspective and polynomial. Within our work the most useful model is rigid transform, which we use

to track objects that preserve their shape. Affine transformation is a more general type of transformation and includes translation, rotation, scaling, similarity transformation, reflection, shear mapping, and compositions of them. Here we enlist three main transformation models:

Global mapping models. The most common global transformation is a rigid transform, which preserves all the geometric properties of the object. According to this transform each point (x_1, y_1) of the first image is mapped to a point (x_2, y_2) of the second image in the following way:

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + s \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix},$$

where t_x, t_y are translation vector components, θ is a rotation angle and s is a scale factor. A rigid transform with a scaling factor could be useful to register datasets acquired with a cone beam geometry (with different object to beam distances).

To estimate a transformation and its parameters global mapping methods based on point matching use a set of point correspondences. In this case, if a sufficient number of control points are provided, it is possible to derive the parameters of the transformation using two general methods:

- *Approximation* procedure estimates the overall transformation mapping in such a way, that matched points are aligned as close as possible. This is usually implemented with statistical optimization methods, e.g. least square approximation. The key assumption in this case, is that the matched points could be inaccurate, so the transformation should satisfy all the matches in an optimal way.
- *Interpolation* procedure finds the transformation mapping so the feature points match precisely and parameters in other locations are estimated using a special relation. This method is useful in the case if fewer, but more accurate matches are provided.

Local mapping models. A significant drawback of global approaches is that these methods do not properly treat the transformations which are local. For this case a number of specific or modified registration methods are used. In this work we do not use this type of mapping models, so we opt to skip the description of such techniques. For the overview of available methods the reader is referred to the works of [Bro92, ZF03].

Elastic registration. A fundamentally different approach for the registration of images with complex or local distortions is not to use any parametric or global mapping functions. The idea of *elastic registration* was introduced in a seminal work of Bajcsy and Kovacic [BK89]. In this method an image is represented as an elastic model (rubber-like) and transformations are described as deformation forces constrained by a stiffness or smoothness. In the current work we use the results of optical flow to compute dense local displacements to perform elastic registration. In our case the data constancy assumption is serving as a metric for feature matching and smoothness constraint is a similar concept as in the case of elastic registration. Note, that the optical flow parameters should be chosen appropriately, depending on the task and amount of local variations to be corrected or revealed.

3.2.3.4 Image Resampling

To apply a mapping transformation, which is computed in the previous steps, an appropriate image resampling and interpolation have to be employed. This transformation can be done in a forward and backward direction. The forward approach is hard to implement and could produce data outliers (hole or overlaps) due to multiple mapping or discretization problems. The backward registration is easier to implement and produce good results.

There is a large number of choices for the interpolation procedure [PKT83, GU98]. For our task we use a backward registration by means of bilinear interpolation described in Section 2.2.4.3. For the volumetric data we use a 3D variant of this transformation - trilinear interpolation.

3.2.3.5 Image Registration Strategies

Here we summarize several strategies of image registration used in our data analysis framework:

- *Dense local registration via optical flow.* Features - dense pixels, matching procedure - dense optical flow, transformation - backward flow field interpolation. Application example: Register two images to obtain the maximum similarity.
- *Dense global registration via optical flow.* Global registration is achieved by using a coarse representation of the flow field or input images, i.e. using an increased smoothness constraints or presmoothing parameter. Features - dense pixels, matching procedure - optical flow, transformation - backward flow field interpolation. Application example: Register two images of the same scene to compensate for global elastic changes to reveal local changes.
- *Feature-based registration via optical flow.* Features - sparse set of landmarks, matching procedure - sparse optical flow correspondences, transformation - global transformation based on approximation or interpolation (see Section 3.2.3.3). Application example: align images based on automatically distributed landmarks.
- *Global image registration with correlation-based technique.* Features - entire image (image-based matching), matching procedure - correlation-based techniques (SSDA is a good choice, see Section 3.2.3.2). Application example: align images to compensate for a global sample movement prior to the optical flow computation, which is sensitive for small changes.

Chapter 4

Computational Framework

In this chapter we describe our computational framework. First, we present the software architecture which is implemented in a modular approach. Then we describe main responsibilities of individual components of each module. The second part is devoted to high performance computing. First, we show a number of efficient numerical techniques to speed up the computation process. Then we present an implementation of optical flow methods on graphical processing units (GPUs), which enables us to process large datasets in a reasonable time. In the last part of the chapter we discuss and show few examples of visualization methods which are implemented or related to our data analysis framework.

4.1 Software Framework

The software for optical flow computation and time-resolved data analysis is implemented using C++ programming language, is modular and follows the object-oriented design paradigm. The requirements for our implementation are high performance, platform independence and strict control for types and memory management. Most of the components are implemented within a single unified framework, if not stated otherwise (e.g. some data analysis or data visualization packages are implemented as external routines). In the following section we describe all the software components in more details.

4.1.1 Modules

Our optical flow computation and data analysis framework is designed as a modular system. Each module is responsible for a specific class of functionality. A central part of the framework is a **CORE** module, which contains classes and methods to perform the optical flow computation. The integral class of this module is a **Runner** class which serves as unifying interface, control and communication routine between the core classes and external modules. With such organisational architecture it is possible to minimize interdependence between different parts and allow to easily extend or modify existing functionality. The overall diagram of our framework is presented in Figure 4.1.

The developed software framework consists of the following modules:

- User interface
- Input / Output Library

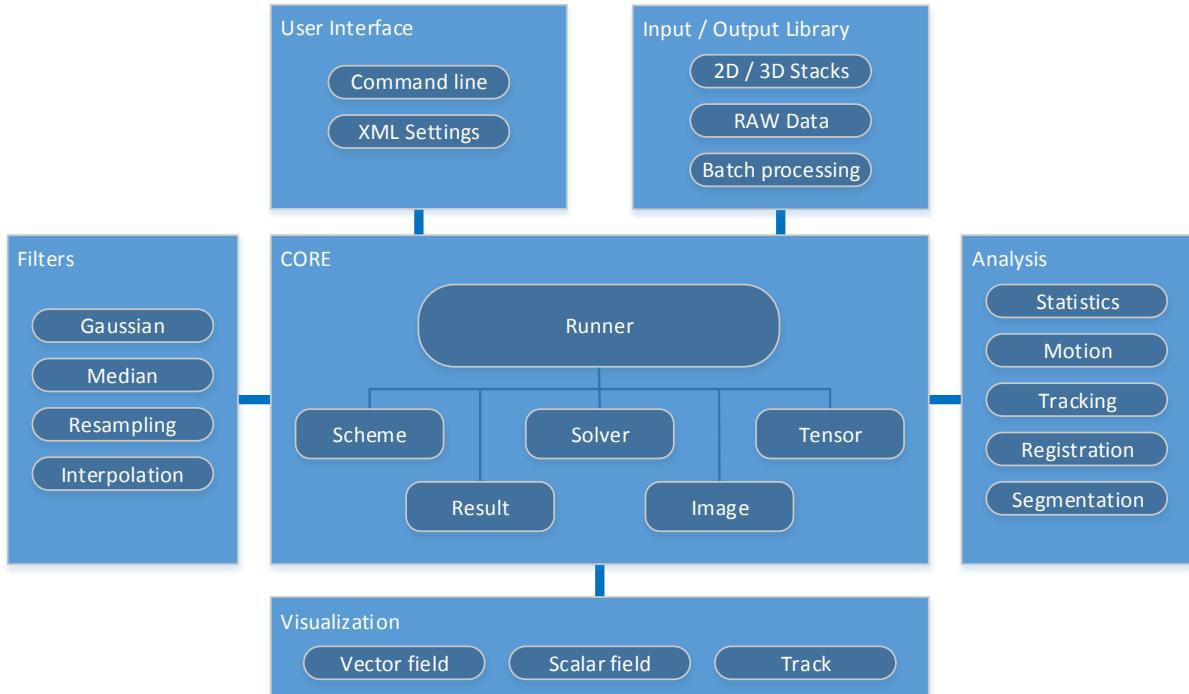


Figure 4.1: Optical flow computation and data analysis framework as a modular system. Each model is responsible for a specific class of functionality. All modules are connected and controlled via the **Runner** class of the **CORE** module.

- Filters
- Core
- Analysis
- Visualization

In the following part we describe the responsibility and essential functionality of each of these modules.

User Interface Module. Provides an universal interface to setup and adjust the overall workflow, select optical flow models, adjust computation parameters, as well as perform further analysis of the optical flow results. Implemented as a command line utility and a settings file in a easy-to-read XML format¹.

Input / Output Library Module. The module provides routines for data input and output. This includes the work with settings files, statistics, image data, resulting vector fields and other related information. The library allows to use the following data inputs:

- 2D / 3D image sequences using common image formats.
- RAW data. Using various data types: 8-bit, 16-bit, 32-bit.

¹<http://www.w3.org/XML/>

- Batch processing. Allows to perform batch processing. Several input modes are possible: masking mode with the specified image indexes; a list of images; a list of image pairs.

Filters Module. The module contains various filter which are used during pre-processing, data analysis or optical flow computation. There list of possible image filters includes:

- Gaussian smoothing. Implements image filtering via convolution with a Gaussian function (See Section 3.1.1.2).
- Median filtering. Implements image filtering using a median filter (See Section 3.1.1.2).
- Image resampling. Using a number of methods: Averaging, Area-based resampling.
- Image interpolation. Available modes: nearest-neighbor, bilinear, trilinear, bicubic interpolations [PKT83].

Core Module. This module is the central part of the framework and contains classes and methods to perform the optical flow computation, as well as to connect and control the classes from external modules.

- Image. Provides functionality to represent an image in various formats and dimensions. The implemented types of images are: `Image2D` `Image3D` and `ImageRGB`.
- Scheme. The main class to impose data constancy assumption via the computation of motion tensor (See Section 2.1.2).
- Tensor. This class stores the entries of a motion tensor for 2D or 3D image data.
- Solver. The main class which performs numerical solution of the optical flow problem via solution of linear equations, which describe the minimum of the energy functional.
- Result. This class stores the results of optical flow computation and contain methods to analyze them. In our framework two versions of the optical flow results are implemented - `Result2D` and `Result3D` - for two-dimensional and three-dimensional cases respectively.

Analysis Module. Contains a set of routines for further analysis based of optical flow results. Some functionality implemented within the same framework and some as external tools.

- Motion. A set of methods for further analysis of displacement fields. See section 3.2.1 for theoretical description and section 4.1.2.4 for the list of available functionality.
- Tracking. A module for points or objects tracking using the computed displacement fields as a feature (See Section 3.2.2.2).
- Registration. A module for motion-based image registration (See Section 3.2.3).

- Segmentation. Module for motion-based segmentation of moving objects. Thresholding methods are implemented as part of our framework, more advanced methods such as region-growing and level sets methods are implemented externally using Fiji / ImageJ software².

4.1.2 Components

4.1.2.1 Scheme Components

Scheme components intend to perform the construction of a data term by computation of the corresponding motion tensor (See Section 2.1.2). A list of available data constancy assumptions implemented as separate `Scheme` classes includes:

- `Scheme`[`DataConstancy`]. Performs a computation of a motion tensor for a given constancy assumption. The complete list of available constancy constraint is presented in Section 2.1.2. The implemented constancy modes are `DataConstancy` = {`Grey`, `Gradient`, `GreyGradient`, `GradientNorm`, `Laplacian`, `LogDerivatives`, `GreyRGB`, `GradientRGB`}.
- `SchemeNormalization`. All the previously presented data constancy assumptions (list `DataConstancy`) can be normalized according to the procedure described in Section 2.1.3.4.
- `SchemeCLG`. All data assumptions `DataConstancy` can be convolved with the Gaussian function to obtain a motion tensor which integrates local information, to obtain the so-called Combined-Local-Global approach (See Section 2.1.3.3).
- `SchemeMultiChannel`. This scheme allows to construct a data term using an arbitrary number of channels as the input (for example as for 3-channel RGB color images). This might be useful to incorporate images obtained using multiple contrast modalities.
- `Scheme3D`. All the presented schemes can be implemented for 2D or 3D images. Moreover, implementation for higher dimensions can be done in a straightforward way.

4.1.2.2 Solver Components

The `Solver` classes perform numerical computation of the optical flow problem solving a system of linear equations. Several algorithms are available in our computational framework (See Section 2.2.3):

- `SolverJacobi`. Performs solution of a system of linear equations using Jacob method (see Section 2.2.3.1). It is not an optimal solver methods in terms of performance and requires large amount of iteration to converge to the optimal solution. However, for the case of parallel computations, where each element of the matrix is evaluated separately, this solver is more suited (see discussion in Section 2.2.3.1).

²<http://imagej.net>

- **SolverGauss.** Solves a system of linear equations using a Gauß-Seidel method presented in Section 2.2.3.2. It speed ups the computation time comparing with the Jacobi methods, since all the entries of the approximation matrix are updated on each iteration.
- **SolverSOR.** Performs the solution using a Gauß-Seidel method with the successive overrelaxation technique, which is presented in the section dedicated to the advanced numerical techniques 4.2.1.1.
- **SolverMultiLevel.** This solver implements a multi-level computation as described in Section 2.2.5 and allows to perform high accuracy optical flow computation of large displacements. On each individual computation level the **SolverJacobi**, the **SolverGauss** or **SolverSOR** numerical solvers can be used.

4.1.2.3 Results Components

The responsibility of this module is to store and analyze the results of optical flow computation. The following measurements are implemented:

- **EndpointError.** This measure is used to compare two vector fields in terms of a distance between two vectors (See Section 2.1.1.2). Presently it the most popular measure in the literature on optical flow to compare the computed results with a ground truth displacement field.
- **AngularError.** Computes the angular differences between two vectors (See Section 2.1.1.1).
- **DataConstancyError.** Computes a confidence measure based on data constancy error (See Section 2.1.5.3).
- **EnergyError.** Computes a confidence measure based on the residual energy of the energy functional (See Section 2.1.5.2).
- **UniquenessError.** Computes a confidence measure based on the motion uniqueness criteria (See Section 2.1.5.4).
- **ConsistencyError.** Computes a confidence measure based on the constancy between forward and backward flow fields (See Section 2.1.5.5).
- **PredictionError.** Computes a confidence measure based on the constancy between forward and backward flow fields (See Section 2.1.5.6).
- **ErrorStatistics.** For each of the error or confidence metrics presented above, a complete set of statistical measure is extracted and can be used. These measures include a minimum (**min**), a maximum (**max**), a mean (**avg**) and standard deviation (**std**). Moreover, for the endpoint error metrics a set of robust statistics measures is also available. They compute the percentage of pixels that have an error measure larger than a certain amount of pixels. In particular, we compute **R0.5**, **R1.0**, and **R2.0** for the endpoint error, which corresponds to half, one and two pixels respectively.

- **ErrorRegions.** For every error or confidence metric it is possible to evaluate their statistics in various image regions to get insights how optical flow methods perform in these regions. We provide the following regions of interest: whole image (**A11**), around motion discontinuities (**Disc**), in textureless regions (**Untext**), around data artifacts (**Err**) or in an arbitrary region (**Roi**). All the regions of interest have to be supplemented to the results analysis routine as binary masks.

4.1.2.4 Motion Analysis Components

These routines allow to perform further analysis of the computed motion field. The following methods are implemented within our analysis framework:

- **Magnitude.** See Section 3.2.1.1.
- **Phase.** See Section 3.2.1.4.
- **Curl.** See Section 3.2.1.3.
- **Divergence.** See Section 3.2.1.2.
- **Components.** The module allows to save individual components of the computed flow field for further analysis.
- **Kinematics.** Allows to extract kinematics components from a motion field. Under assumption of a rigid motion this computational module requires as an input a set of landmarks representing the object of interest, a center of coordinates and an axis of rotation. As an output the component produces a translational (**trans**) and rotational components (**rot**) of the object's motion. An example of such motion analysis is presented in Section 6.3.1.

4.1.2.5 Tracking Components

These components perform object tracking or tracking of individual points using the results of optical flow computation. With the help of these routine it is possible to perform the following tracking operations:

- **Tracks.** The tracking component takes as an input a set of initial points (landmarks) and performs the tracking procedure using the results of optical flow throughout an image sequence, specified by a number of frames. As an output a list of points and their coordinates on each time frame is produced. A possibility to filter the resulting tracks according to confidence measures, e.g. a consistency measure (See Section 2.1.5.5, 4.1.2.3), or according to total distance are also available. An example of this tracking procedure is presented in Section 6.3.1.
- **TracksAmira.** The same as the previous methods, but saves the output in a format compatible with the Amira/Avizo software³.
- **TracksVTK.** The same as the previous methods, but saves the output in a Visualization Toolkit (VTK) format⁴.

³<http://www.fei.com/>

⁴<http://www.vtk.org/>

4.2 High Performance Computing

In this section we discuss such an important aspect as high performance computing. Since optical flow methods are very computationally demanding, an efficient implementation of these techniques should be provided to guarantee that the data processing on a given computational platform is feasible.

First, we discuss several numerical techniques, which allow to speed up the computation time. Then we focus our attention on the implementation of optical flow methods on the GPU architecture.

4.2.1 Numerical Schemes

4.2.1.1 Successive-Overrelaxation

The convergence speed of the Gauß-Seidel method can be substantially improved by performing *Successive Overrelaxation* (SOR) technique [You71, Saa03]. This is done via extrapolation of the Gauß-Seidel result. If we denote by $\bar{\mathbf{x}}$ the result of Gauß-Seidel method for the iteration step $k + 1$, the SOR technique then given by:

$$\mathbf{x}_i^{k+1} = (1 - w)\mathbf{x}_i^k + w\bar{\mathbf{x}}_i^{k+1},$$

where $w \in [0, 2)$ is a *relaxation* parameter. Then, The Gauß-Seidel method can be extended by the overrelaxation technique in the following way:

$$x_i^{k+1} = (1 - w)x_i^k + w \frac{1}{a_{i,i}}(b_i - \sum_{j < i} a_{i,j}x_j^{k+1} - \sum_{j > i} a_{i,j}x_j^k)$$

For the value of relaxation parameter $w = 1$, the SOR method comes down to the original Gauß-Seidel method. The choice of the relaxation parameter has a strong influence on the convergence speed. A parameter w smaller than 1.0 allows to perform underrelaxation which could lead to improved stability of the algorithm. For the values larger than 1.0 the method performs overrelaxation of the result and thus accelerate the convergence. An optimal value of the relaxation parameter should be chosen according to the specific task. For our applications we select values which are close to the maximum value, such as $w \in [1.9..1.95]$. A major drawback of the method is a possible oversmoothning of the resulting displacement fields.

4.2.1.2 Adaptive multi-level strategy

An interesting approach to decrease the computation time is to reduce the input on each computation level. For this purpose the computed flow field is evaluated according to a special criteria. For example one may use the fact that the flow field is varying slowly from one computation level to another. This may highlight that the correct flow has already been estimated. In such regions the computationally expensive energy minimization is substituted by a simple interpolation from the results of previous computation levels. Such approach was presented by the authors in [TBKP12] and allowed to significantly reduce the computation time without serious impact on the accuracy of the overall result. This approach can be especially advantageous for datasets with small or isolated objects and dominance of the static background. In this cases an adaptive computation procedure removes most of the irrelevant computations in the background region.

4.2.2 GPU Computing

This section is dedicated to a GPU-based implementation of the optical flow algorithm. First, we briefly discuss the key features of the GPU architecture [NVia, NVib]. Then, we analyze the algorithm to find possible problems and limitations we may come across during the implementation phase. Then we introduce a GPU-based implementation that uses only one GPU accelerator and is limited to the datasets that can be entirely fitted in the GPU memory. A number of key aspects of the implementation are discussed. Finally, we show how a single GPU version can be extended to process large datasets.

4.2.2.1 GPU Computing Architecture

For the GPU implementation of 3D optical flow methods we choose a NVIDIA CUDA technology⁵. However, all findings presented in the current section are valid or can be implemented by means of other GPU platforms, for example, OpenCL⁶.

Host-device execution model

In general a GPU computing architecture is comprised of two essential parts - a *host* and a *device* [NVia]. Alternative notation is a CPU for the host and a GPU for the device. The host manages the device by scheduling control commands and sending them to the device. The code execution on the host and the device can be done *synchronously* or *asynchronously*. For the case of synchronous execution mode, the host waits until a command is completed on the device and only after the continues to execute its pipeline. If the control commands are scheduled in an asynchronous mode, the host receives the control immediately after sending a command to the device. Such mechanism implements a strategy, for which the host and the device can perform different tasks simultaneously. All the commands sent by the host are queued for sequential execution on the device side. In terminology of CUDA such queues are called *streams*. All the commands are executed in issue-order and there is no possibility to run several commands within the same stream. However, a single device has many processing streams which can be executed in parallel.

Device execution model

The commands to be sent from the host to the device can be memory management commands, such as memory allocation, deallocation or copy, and also control synchronization or profiling commands. Additionally, the host can send a special command to launch a *kernel* on the device. Kernels are the core elements of the GPU computing. A kernel is a user-defined function which can be run on the device. All the kernel's instructions are executed sequentially, while multiple copies of such kernels are run in parallel on the device. The kernel code is written in the CUDA C-based language and then compiled into a binary code for the specific GPU device [NVia].

Multiple kernels are organized in *thread blocks*, which in turn make a grid of thread blocks. Both threads and thread blocks have unique indexes. Thread blocks and a grids can be organized as one-, two- or three-dimensional structures. Only a limited amount

⁵http://www.nvidia.com/object/cuda_home_new.html

⁶<https://www.khronos.org/opencl/>

of thread can be run on a single GPU multi-processor. It is important to note, that the execution order of thread blocks and each thread within a block is not guaranteed.

Memory model

In the GPU computing the memory model is divided into a host memory (system memory) and device memory (GPU memory). Both parts have the direct access only to its own memory resources.

There are several distinct types of GPU memory:

- **Device memory** is a global memory shared between all streaming multi-processors. The access to the device memory has highest access latency, and therefore it is the slowest type of memory. The device memory can be accessed by all threads. Moreover, it can be accessed from the host by using memory management functions. The aim of many optimization strategies is to minimize the amount of memory accesses to the global memory and reduce the data transfer between the host and the device.
- **Texture memory** is optimized for storing and accessing textures. This type of device memory is read-only. Using this memory one can benefit from the built-in hardware functionality (e.g. interpolation).
- **Constant memory** allows to store limited amount of constant data.
- **Shared memory** serves as a low latency storage accessible from all threads within an individual block.
- **Registers** memory belong to the fastest GPU memory type. Each streaming multi-processor has a limited amount of registers which are shared between all threads within a thread block. Registers are used to store local variables of a single thread.

Clearly, each type of GPU memory has its pros and cons. To achieve maximum performance we have to efficiently use all types of GPU memory and adjust the implementation of the GPU-based parallel algorithm accordingly.

4.2.2.2 GPU implementation

To maximize the performance of GPU-based optical flow algorithm, we provide a first GPU implementation, a *full* GPU version. In this version all data is stored in the GPU memory. In this way the data transfer between the host and the device is minimized. Clearly a limitation factor for this kind of implementation is the size of available GPU memory. Now, we estimate the size of input frames which is possible to process using such GPU method. For a 3D image with $width \times height \times depth$ voxels, assuming that each voxel is represented by a single-precision floating-point number, the size of such a 3D image is then equals to $width \times height \times depth \times 4$ bytes. To solve the optical flow problem we need 2 such volumes. Additionally, we need containers for the vector field results, each having 3 flow components u , v and w . Moreover, we need auxiliary containers to store intermediate results of the optical flow algorithm. In total we need to allocate 15

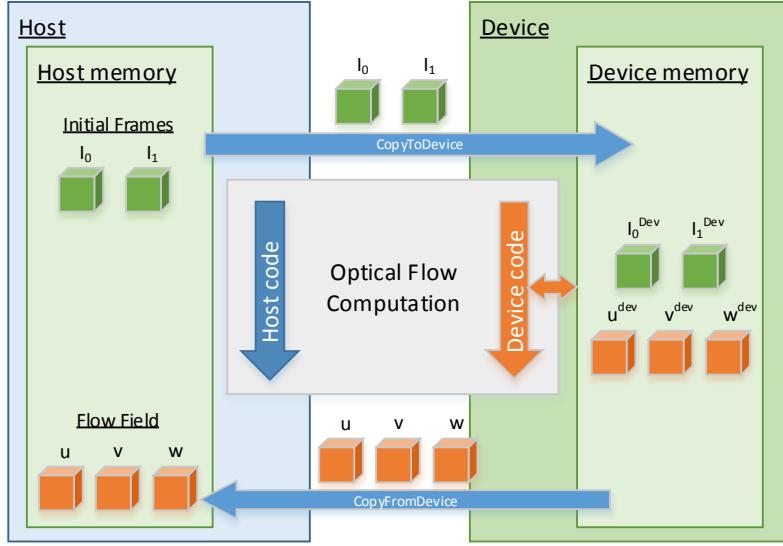


Figure 4.2: Data transfer between the host and the device for the GPU based implementation of the 3D optical flow algorithm. Image: [Kar15].

single-precision floating-point containers with the size $width \times height \times depth \times 4$. Figure 4.2 gives a schematic overview of allocated data and the data transfer procedure.

Then, we can estimate the maximum size of the input frame which is possible to process using full GPU version:

$$15 \times pitch \times height \times depth \times \text{sizeof}(float) \leq N. \quad (4.1)$$

As an example, if we have a GPU with 4GB of memory, we can process input volumes of approximately 273MB, which corresponds to a volume with the sizes about $400 \times 400 \times 400$ voxels. Obviously, such data sizes are not adequate, if we consider the processing of high-resolution datasets. For large datasets it is then necessary to downsample the initial data, which could lead to a significant loss in resolution and accuracy. To overcome this limitation and enable the processing of large datasets in the next section we present an extension of our GPU implementation.

4.2.2.3 Extension for large datasets

In this section we show how the 3D optical flow method can be extended to support large datasets. The main idea of the proposed method is to implement the so-called *data partitioning strategy*. We store the dataset and corresponding vector fields in the host memory and transfer smaller portions to the device for parallel processing on GPU. In this strategy, the host performs managing of the data partitioning, execution on the device and collection of the results after each computation level. To perform data partitioning we make the slicing of the original volumes and extend each data chunk with overlapping regions, solve the data dependency problem. An example of such partitioning (slicing along Z-axis) is presented in Figure 4.3.

Note, that in the previous implementation (*full GPU version*) the whole dataset, the resulting vector fields and all auxiliary containers reside in the GPU memory and there is only one data transfer operation from the host to the GPU. In our *extended* version data

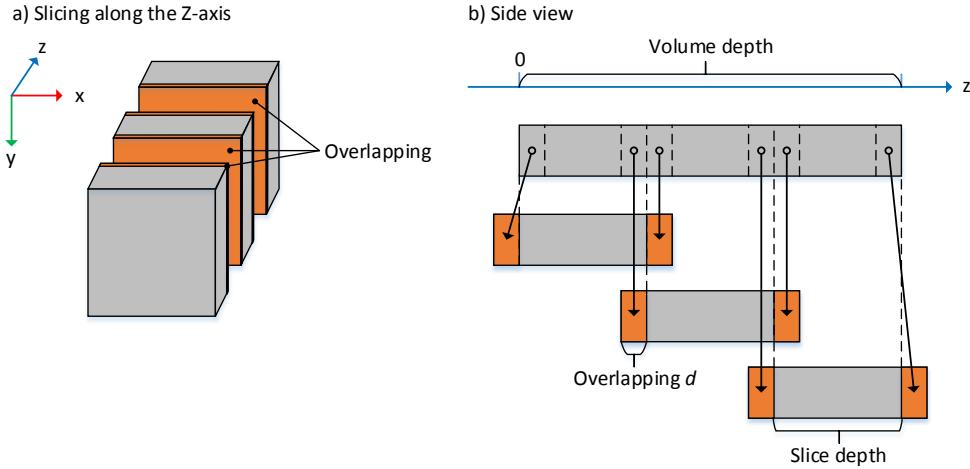


Figure 4.3: Data partitioning strategy. Slicing is done along the Z-axis with the overlapping regions (a). Partitioning of voxels in the full volume with overlapping regions (b). The overlapping regions that are outside the full volume are filled with "mirrored" values. Image: [Kar15].

is transferred between the host and the device before and after processing of each data chunk. Therefore, data transfer can lead to serious performance issues and the bandwidth between the host and the device become a limiting factor. In the next section we provide quantitative performance analysis between CPU and GPU versions, as well as between *full* and *extended* GPU versions.

4.2.2.4 Performance evaluation

To measure the performance of our GPU based 3D optical flow implementation we run the method on datasets with different sizes. Instead of evaluating each individual component of the algorithm, we present the execution time and memory consumption for the whole computation pipeline. Moreover, we verify the correctness of the algorithms compared with the original CPU implementation.

For the evaluation we use the computation hardware with the specifications presented in Table 4.1.

	CPU Test machine	GPU Test machine
CPU	Intel Xeon Processor X5675	Intel Xeon Processor E5-2637 v2
RAM	96 GB	64 GB GK104
GPU	—	(As a single module of GRID K2 Graphics Board)
GPU Memory	—	4 GB
CUDA Cores	—	1536

Table 4.1: Technical specifications of computation machines used for the performance evaluation.

We estimate how the size of the input data influences performance of 3D optical flow method and the memory consumption by the computation routines. For the evaluation we use datasets with the following sizes: 100^3 , 200^3 , 400^3 , 600^3 , 800^3 . As it was highlighted

previously for a GPU device with 4GB of memory the maximum size of the input data is 400^3 . Therefore, datasets with sizes 100^3 , 200^3 and 400^3 can be processed using *full* GPU method; while larger dataset - 600^3 , 800^3 , can be processed only using an *extended* version. Evaluation results for performance measurements and memory consumption are presented in Tables 4.2 and 4.3 respectively.

Size (voxels)	Execution time (hh:mm:ss)			Speedup relative to CPU verion	
	GPU <i>Full</i> version	GPU <i>Extended</i> vesion	CPU Version	GPU <i>Full</i> version	GPU <i>Extended</i> version
100^3	00:00:04	00:00:22	00:00:38	9.5	1.7
200^3	00:00:18	00:01:08	00:06:02	20.1	5.4
400^3	00:02:12	00:06:26	01:13:52	33.5	11.5
600^3	—	00:19:01	02:47:53	—	8.8
800^3	—	00:45:43	07:30:49	—	9.9

Table 4.2: Evaluation of execution time for dataset with different sizes 100^3 , 200^3 , 400^3 , 600^3 , 800^3 .

Size (voxels)	GPU <i>Full</i> version		GPU <i>Extended</i> version	
	Device memory	System memory	Device memory	System memory
100^3	73 MB	19 MB	*	57 MB
200^3	586 MB	153 MB	*	458 MB
400^3	3.64 GB	1.19 GB	*	3.58 GB
600^3	—	—	*	12.07 GB
800^3	—	—	*	28.61 GB

Table 4.3: Memory usage of the implemented framework. Note that the memory allocated on the device is padded to satisfy alignment requirements. (*) For the extended version the usage of the device memory is determined by the current needs for the sliced data and available GPU memory.

From time measurements we see the following results - for small datasets $\leq 400^3$), *full* GPU implementation gives the best performance. Clearly, the *extended* version provides less performance compared to the *full* version due to data transfer operations between the host and the device. However, for larger datasets ($> 400^3$), the *full* version is no longer functional, since the data cannot be fitted into the memory of the GPU, so the data partitioning strategy of the *extended* version has to be used. In this case, the *extended* version significantly outperforms the CPU version of the optical flow method. The maximal performance speedup for the *full* version was 33.5, for the *extended* version – 11.5. Therefore, we conclude that our 3D GPU-based implementation is efficient and provides a significant reduction in the computation time for large 3D datasets.

4.3 Visualization

The section gives a brief summary of visualization methods for time-varying data and results of optical flow computation which are available in our data analysis framework. We implement three main visualization methods:

- Vectors fields using glyphs visualization
- Scalar properties or other metrics using color pseudo-codes.
- Motion trajectories using track visualization

In the following part we describe in more details vector fields visualization using vector glyphs and omit the description of other methods.

Vector Field

A simple yet effective method for visualization of vectors fields is a glyphs method. In this case each vector of the displacement field is depicted as a glyph with a certain shape and color.

For our implementation we use 3D cones. For the coloring of glyphs we use two modes: the first one assigns a color according to the magnitude of a vector and the second mode uses the direction of a vector. Both visualization using different coloring schemes are presented in Figure 4.4.

In order to improve the usability of the 3D visualization tool we provide visualization parameters that can be changed interactively. It is possible to adjust the sample grid G_x , G_y , G_z , the glyphs scaling factor and the magnitude threshold. All different visualization modes are presented in Figure 4.4. Moreover, it is possible to use the so-called *jittered* grid. It allows to avoid possible artifacts associated with the regular grid (artifacts similar to aliasing, when viewed from different angles). This improves the clarity of users perception of different parts of the vector field. Comparison between a regular and a jittered grid is given in Figure 4.4. All the visualization parameters can be adjusted interactively and the changes are immediately rendered.

For the implementation of our visualization tool we use OpenGL⁷ as a 3D graphics library, GLFW⁸ as a library for window and input management, GLM⁹ as a mathematics library for graphics software and AntTweakBar¹⁰ as a light graphical user interface library. All the libraries are cross-platform and free to use.

⁷<https://www.opengl.org/>

⁸<http://www.glfw.org/>

⁹<http://glm.g-truc.net/>

¹⁰<http://anttweakbar.sourceforge.net/>

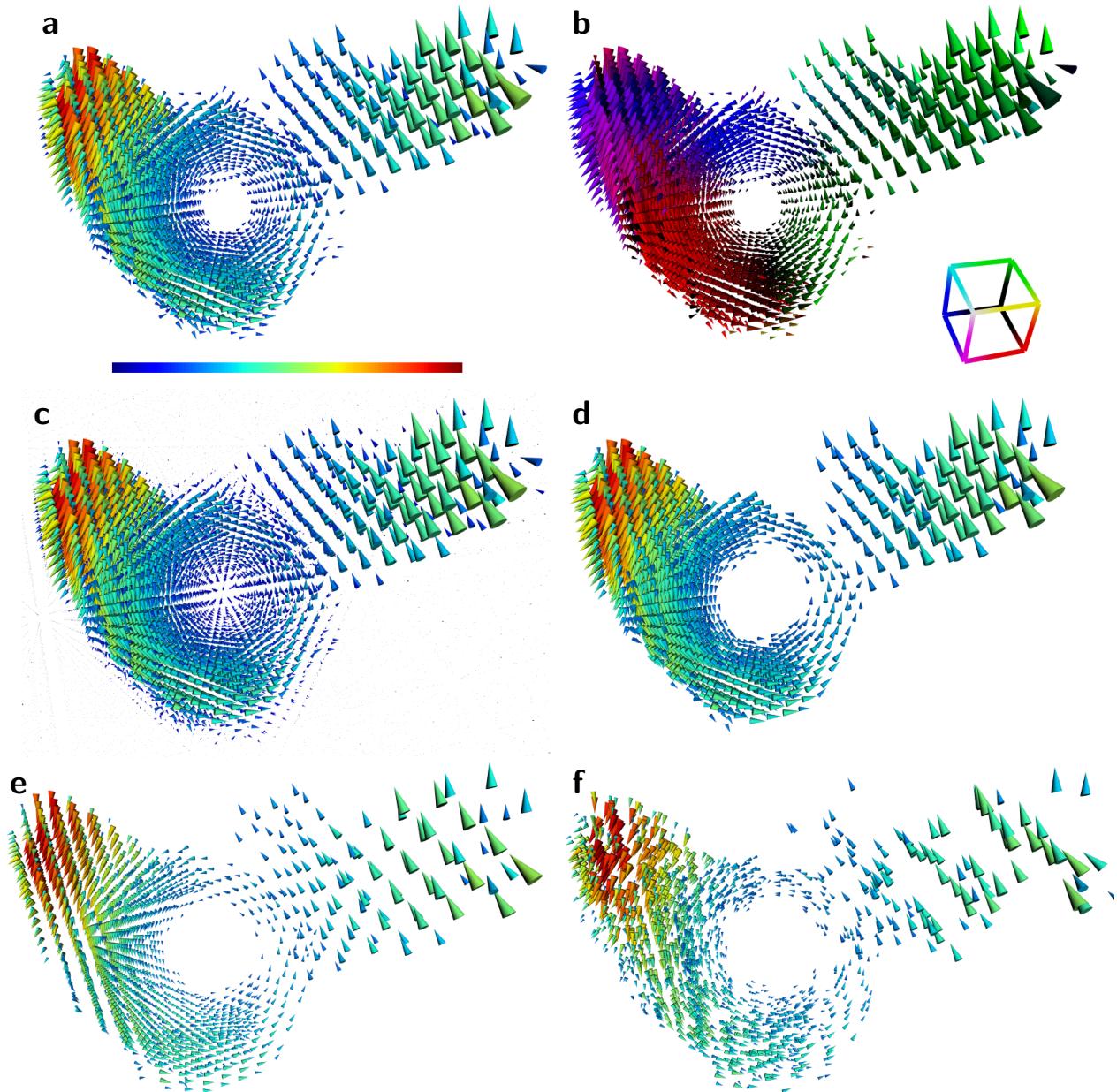


Figure 4.4: Visualization of a 3D vector field. **(a)** Using magnitude of the vectors for coloring. **(b)** Using direction of the vectors for coloring. **(c)** Regular grid, no threshold of magnitude value. **(d)** Regular grid, threshold is applied. **(e)** Regular grid, decreased number of grid points. **(f)** Jittered grid. Image adopted from: [Kar15].

Chapter 5

Evaluation on Synthetic Data

In this chapter we perform systematic experiments on synthetic data, designed to test the performance of optical flow methods. In the beginning of the chapter we provide a taxonomy of X-ray data. Our classification includes evaluation of image quality, description of data features and artifacts, as well as the description of motion models. Then, we present synthetic datasets, which we employ for our experimental evaluation. Further, we discuss an important aspect of parameters optimization and describe the overall quantitative evaluation procedure. We present a discussion on the performance, influence of preprocessing, modelling of noise and data outliers and the use of confidence measures. For all the experiments we outline important observations and draw the conclusions, which we summarize in the end of the chapter.

5.1 Data Taxonomy

Since correspondence problems are highly ill-posed and challenging to solve, before any data processing steps are performed, the input data should be systematically evaluated. Its quality, description of the moving objects or the scene, and the motion model should provided a basis for the choice of appropriate data preprocessing routines and optical flow models. Moreover, we emphasize that the clear definition of the application and description of the requirements on the algorithm should be specified.

One approach to evaluate the data is to perform an explicit quantitative description of the input images. For example, compute a histograms of grey values and velocities, provide the distribution of image derivatives, use noise estimators, etc. But, it is not always possible to provide such an extensive description - noise estimation can be a difficult task, the motion model prior to the optical flow computation is not known, artifacts are difficult to estimate. In our case we aim to provide a systematic, yet simple and consistent data classification model, which is handy to apply for a wide range of datasets and applications.

In the rest of this section we enlist the most important aspects for our data taxonomy classification. The first category is a description of the image quality.

5.1.1 Image Quality

Image Noise

Noise is a fundamental characteristic of image data. The presence of noise affects the image quality and deteriorates data analysis. To describe the image noise we consider two parameters - the noise model and the amount of noise (or noise level).

Noise Model. Is an important characteristic, since it determines the choice of filtering procedure and impose certain requirements on the design of optical flow models. Various noise models that are important for our work are described in Section 3.1.1.1. For our data taxonomy we distinguish the following noise models:

- Gaussian noise. This noise model is an approximation model and is frequently used for synthetic datasets in a large body of work on optical flow methods. However, as we have discussed earlier, this model does not adequately describe a physical process of imaging, especially X-ray imaging.
- Poisson noise. That is the most important source of noise for X-ray images due to the photon statistics and the detection process. Therefore, it is crucial to test the performance of optical flow methods and the processing techniques on the type of noise.
- Spike noise. An important noise model to evaluate the performance of computational methods with respect to data outliers.
- Granular noise. This type of noise is characterized by the varying size and the unknown distribution function. The presence of such noise can be a result of a preprocessing routine, e.g. tomographic reconstruction.

Examples of images contaminated with different noise are shown in Figure 5.1.

Noise Level. This parameter specifies the amount of noise. As it was already mentioned in the previous section, adequate quantitative estimation of a noise level is a challenging task. Even when such procedure is feasible to perform, it is not clear how to employ this quantitative information to guide the design of processing procedures. For this reason, we omit a measurable estimation of a noise level and proceed with a qualitative, empirical estimation approach. For this purpose we distinguish the following noise levels:

- low: no noise or very little noise is present. In this case no special data treatment is required and design choices for the optical flow model are not restricted to methods that are robust with respect to noise.
- average: there is a considerable amount of noise, but the image features are well discernible. A careful consideration of noise is required.
- high: image data is highly deteriorated with noise. An advanced noise filtering and robust optical flow methods are mandatory to obtain reliable optical flow results.

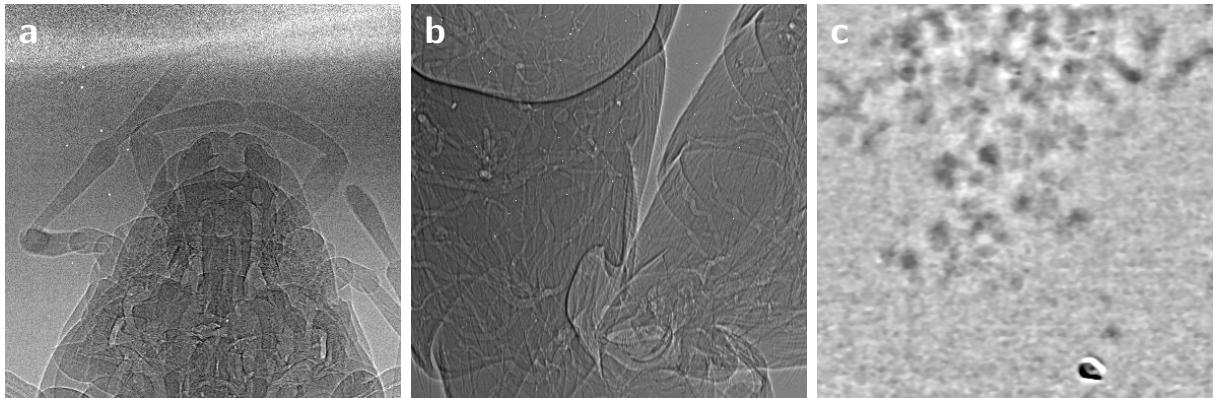


Figure 5.1: Examples of X-ray images containing different types of noise. (a) Radiographic image with a large amount of the Poisson noise, as a result of a short exposure time. (b) X-ray image with a good signal-to-noise ratio, but with a substantial amount of the Spike noise (saturated pixels). (c) Radiographic image with the granular noise resulting from a post-processing step of the raw data.

Image Contrast

We refer to the image contrast, as a general ability to distinguish between different image structures. As we already pointed out in Section 3.1.3.1 the contrast-to-noise ratio (CNR) is a more descriptive and useful measure to evaluate the contrast and its implications on the data analysis, than the signal-to-noise ratio or solely image contrast measures.

In the same manner as for the classification of an image noise, we distinguish three contrast levels:

- high: there is a pronounced difference in contrast of image features. This includes a high amount of textures and regions with large values of image gradients. This type of datasets is preferred for the data analysis and usually leads to an accurate optical flow results.
- average: a contrast is enough to clearly distinguish between features of interest. Moreover, a signal-to-noise ratio for the dataset is acceptable.
- low: different image features can be hardly distinguished. Additionally, a signal-to-noise ratio is poor, which makes it difficult to enhance the contrast (e.g. using histogram stretching) without augmenting the noisy part of a signal. This case is one of the most challenging image analysis scenarios and an advanced data preprocessing procedure is mandatory to retrieve useful information.

Examples of different contrast levels are shown in Figure 5.2.

5.1.2 Data Features

In this section of our data taxonomy we provide a set of parameters to describe a visual appearance of the scene and its objects. Depending on the visual and morphological characteristics of moving objects, different types of optical flow models and their parameters might be optimal. In general, all the information about the scene should be identified

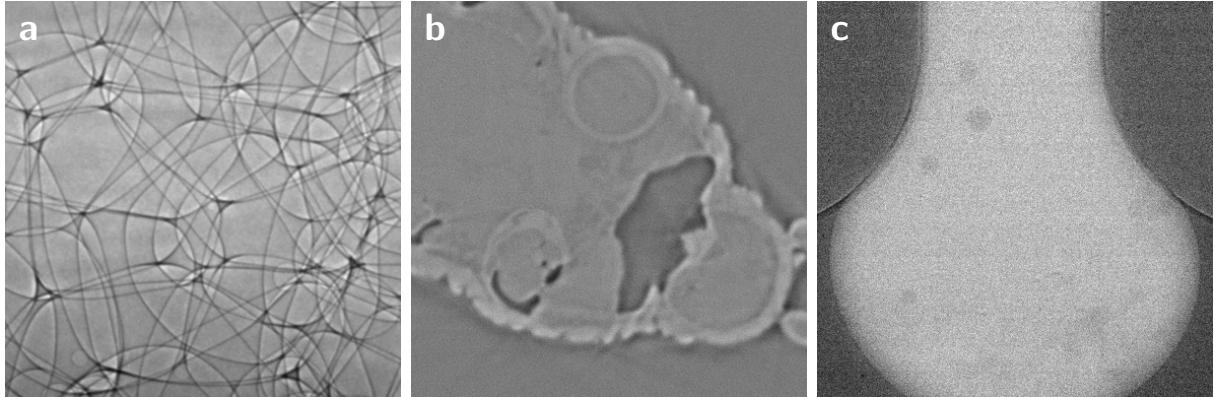


Figure 5.2: Examples of X-ray images with different contrast levels. (a) Radiographic image of aqueous foam with high contrast and good signal-to-noise ratio. (b) Tomographic 2D slice through a *Sitophilus granarius* beetle. The dataset has relatively low contrast between internal structures and poor SNR. (c) Radiographic image of a semi-liquid alloy. Contrast between liquid phase (bright part) and metal particles (dark spots) is very low. Moreover, SNR ratio is very low.

and described in details. First, this imposes constraints on data constancy assumptions. Second, it describes the requirements for the flow field. Finally, it states the task for the optical flow computation - for which parts of the scene, which dynamical information should be computed.

Here we list the main aspects of the description of a scene and its objects, that are used in our data classification:

Object size. One of the basic characteristics of an object is its size. The distribution of object sizes may vary a lot within a given dataset. For our classification we are interested to describe extreme values of object sizes, i.e. small and large objects. Additionally, it is important to identify cases, when both categories of objects are present within the same scene. Thus, we may distinguish the following categories of object sizes:

- **small:** if the objects of interest are small, a special care should be taken when choosing computational methods and their parameters. For example, if we aim to compute the optical flow on small features, the size of a median mask (See Section 2.2.6) should be adjusted accordingly, since it might eliminate the flow around such small features. In a similar way, the large flow regularization parameters may lead to an oversmoothing of the flow field, resulting in a loss of flow details for small objects.
- **average:** by an average size we assume a range of sizes, which do not belong to the extreme values of our categorization approach. In general, such kind of objects do not introduce any restrictions on optical flow models.
- **large:** by a large object we define the object which size exceeds 10-20% of an image size. In general, the computation of optical flow on large objects should not differ from the objects of average sizes. However, there is an exception. If the object is homogeneous (i.e. contains a small amount of details) and the task is to capture

a dense flow field within this object, a large value of the smoothness parameter is required. This can affect the computation in other image regions that contain different data features. That is why the presence of large homogeneous objects should be noted.

- mixed: for this case we consider a mixture of small and large objects. This situation is the most challenging, since it requires the simultaneous optimization of parameters for all objects sizes. It is a common problem in the literature that the small image features get smudged because of a significant contribution from the smoothness term. One way to deal with this problem is to use an adaptive smoothness (See Section 2.1.4.5) in different image regions. Another solution is a fusion of results obtained using different model parameters that are specifically tuned for each object category. In any case, a mixture of different objects sizes introduces a major problem for the optical flow computation.

Object distribution. A spatial relationship between objects, i.e. objects density. To describe the object distribution we use the following cases:

- sparse: in this situation the objects are distant from each other, separated by a homogeneous background. Despite the fact that this situation might seem to be easy to handle by optical flow methods, there are some pitfalls. The problem arises from the homogeneous background which separates the objects. In this region there is no contribution from the data term, i.e. aperture problem exists (See Section 1.4.2), so the smoothness term takes a lead. As a result, on the boundary between an object and a background the oversmoothing problem might occur (which depends on the chosen parameter for the flow regularization α). And if the object of interest also appears to be details-free, even a larger value of the smoothness parameter is required to precisely capture the motion within such object. In this situation severe oversmoothing artifacts may occur. To summarize, a sparse object distribution can be troublesome in conjunction with the low amount of object details (See next characteristic of object features).
- normal: this case describes a normal situation for object distribution. Objects might be close to each other or even share a common boundary, but the length of this interface is much smaller, then the perimeter of the corresponding objects.
- dense: the objects are close to each other and share substantial amount of their boundaries with other objects. Clearly, in this case a special attention should be paid when choosing optical flow parameters that might lead to imprecise localization of object boundaries. Such parameters include, the gradient constancy assumption as a data term, the presmoothin parameter σ , the flow regularization parameter α , the integration scale ρ and the mask size for the flow median filtering.

Object details. This characteristic describes the amount of image features or details for the objects of a scene. We distinguish the following quantities of object details:

- low: small amount or no image details (homogeneous objects). For such objects the optical flow computation may be a challenging task and the inappropriate choice of the regularization parameter may lead to oversmoothness artifacts.

- normal: this case describes an ideal situation. It means that the amount of image gradients is sufficient to provide enough contribution to the data term. So, no aperture problem exists in most image regions.
- high: in general, the more details an object has, the more information is available for the data term. However, there might be some exceptions. For example, in [ZBW11] the authors found out that repeating high-frequency image texture can cause aliasing artifacts during the resampling procedure. As a result, the accuracy of optical flow estimation in these regions can be substantially impaired. Thus, it is important to describe such high-frequency textures and any other periodic structures beforehand.

5.1.3 Image Artifacts

In this section we present a description of various image artifacts, which affect results of data analysis. All artifacts, if existing, should be treated by an appropriate processing step or computed using a dedicated optical flow model. Here we provide a list of possible image artifacts:

- *Brightness variations.* Brightness of X-ray images can be non-uniform spatially and non-constant over time (See Figure 5.3). Since the main component of every optical flow model is a data term, which imposes certain constancy assumptions on image brightness or other image-based features, the presence of brightness variations is crucial for the design of optical flow models.
- *Ring artifacts.* Damaged elements of a detector system cause individual pixels to remain constant during an image acquisition and in particular during the tomographic scanning process. As a result, line structures on sinograms appear. After the reconstruction is performed using the filtered backprojection algorithm these lines form the so-called ring artifacts (See Figure 5.3). Such artifacts can significantly deteriorate the accuracy of data analysis. Thus, such artifacts should be removed prior to motion estimation.
- *Star artifacts.* A non-linear relation between the attenuation coefficient in the material and the intensity values measured by the detector system give rise to the so-called beam-hardening artifacts. These artifacts appear along a geometric path of the respective X-ray beam. After the reconstruction such artefacts appear radially around the boundaries of on object and are called star artifacts (See Figure 5.3). Usually, such artifacts are not so pronounced in the region within an object, however, can lead to the incorrect flow estimation on the boundaries of an object. As a possible remedy one can use a dedicated image reconstruction procedure, which is more robust to the beam-hardening artifacts, i.e. algebraic reconstruction techniques with a spatial image regularisation.
- *Motion blur.* For the conventional reconstruction methods such as the filtered-back projection, it is assumed that the object is not changing during the acquisition process, so one obtains radiographic projections of the same object from multiple angles. But, this assumption is not always fulfilled. This could be a result of sample motion due to poor experimental conditions, e.g. bad fixation of the sample.

Additionally, the motion of a sample could be a part of the experiment, which aims to study dynamical processes. In any case, the inconsistent projection data will result in motion artifacts in the reconstructed volume. The amount of motion blur and the specific steps to take it into account should be described prior to the actual flow computation.

- *Projective geometry.* This kind of artifact is characteristic to the radiographic imaging where 3D structures are projected into a 2D detector plane. This effect is similar to the problem of occlusions in the classical optical flow. Such artifacts are non-existing for tomographic data, in which a 3D information about the scene is available. Projective geometry poses a major problem for optical flow computation, since it deteriorates the data constancy assumption.

For our data taxonomy, if any of the above mentioned artifacts exists, we mark it as a possible data challenge and provide a list of steps to compensate for it.

A list of examples of various image artifacts is shown in Figure 5.3

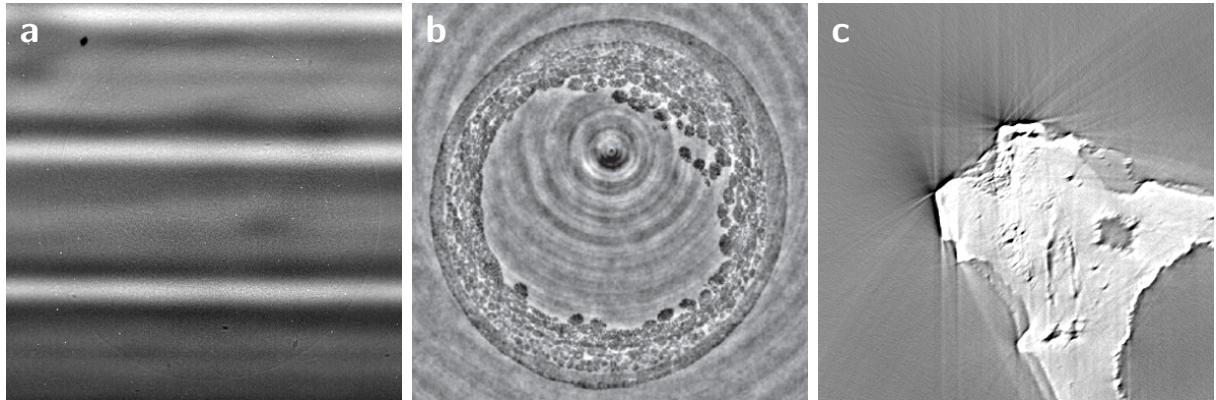


Figure 5.3: Examples of X-ray images with various image artifacts. (a) Radiographic image of living embryo. Non-uniform brightness variations due to monochromator instabilities. (b) Tomographic 2D slice through a *Xenopus laevis* embryo. Ring artifacts are highly pronounced and severely degrade the image data. (c) Tomographic 2D slice through a weevil incorporated in amber. Note star artifacts due to the beam hardening.

5.1.4 Motion Model

In order to compute motion one needs to describe which transformations the objects of a scene undergo. In this section we list data properties, which are related to the motion model. We distinguish the following characteristics:

Motion model. In general, a motion field is a vector-valued function of spatial coordinates. For practical reasons, it is useful to describe a motion field by means of a simplified model with fewer parameters. The most important motion models for our work are:

- translation: the simplest motion model described by a single displacement vector. This transformation preserves the orientation of an object. In most cases this motion model do not pose troubles for the computation of the optical flow.

- rotation: is described by the axis of rotation and the angle of rotation. An important aspect of this transformation is that not all data constancy assumptions are invariant under it. For example, gradient constancy assumption, which contains directional information 2.1.2. Clearly, this aspect should be taken into account for the selection of data term.
- rigid (translation + rotation): this transformation corresponds to the standard motion model of a rigid body.
- non-rigid: this transformation is not represented by a parametric motion model. Instead, the motion assumed to be local with a certain constraint on its smoothness (e.g. elastic transformation) or unconstrained at all.

Motion range. Another important measure to describe the motion model is to specify the range of velocities with which the objects of the scene move. We distinguish the following motion rates:

- small displacement: typically in the range of 1-2 pixels. In this case the linearization of the data term can be performed via the first-order Taylor expansion leading to a convex formulation of the optical flow problem (See Section 1.4.2).
- large displacement: as soon as large values of object displacements are assumed, we cannot employ linear data terms. This is why an appropriate computation strategy have to be used (See Section 2.2.5). Also the optimal size for the coarsest resolution level is an important factor (See Section 5.3.1.3).
- mixed: a scenario when both small and large displacements are present and are of interest for the data analysis. For this case, the influence of the smoothness term should be carefully controlled to preserve small displacements, which tend to be smeared out in the vicinity of a fast motion. Additionally, one may consider the adaptive smoothness approach (See Section 2.1.4.5) or a more sophisticated model to respect small movements.

Motion discontinuities. It is important to describe the amount of motion discontinuities within a scene. This characteristics plays a key role for the choice of the smoothness constraint. Both spatial and temporal discontinuities should be reported. The following scenarios are important for our data taxonomy:

- no discontinuities: for this case we assume a global motion of a scene, described by the same motion model.
- normal amount: this situation corresponds to a normal case of optical flow computation. There are always discontinuities between objects which are moving with different velocities or directions. Moreover, there is always a motion boundary between the moving object and the static background. It is important to note, that the smoothness assumption is always violated to some extent. The only aim of preserving discontinuities via the appropriate choice of a smoothness term is to allow for piecewise-smooth motion fields.

- large amount: the cause of discontinuities is the same. i.e. the differences in motion fields, but the amount of flow gradients is large, which requests a particular attention for the choice of a smoothness approach and the corresponding parameters.
- occlusions: in the literature on optical flow methods occlusions between objects or their parts is a major problem. As a result, in the occluded regions not only the information for the data term is lost, but also strong motion discontinuities occur. For X-ray imaging applications occlusions are not present for tomographic datasets, since a 3D information about objects is available. The case when the occlusion problem displays itself is a radiography method, where the projection of 3D structures into a 2D plane gives similar effects. Another important case is the appearance or disappearance of information. This might be a new object entering the scene or formation of a crack in a material, or the process of cell division. For these cases both the data constancy and the motion consistency assumptions are violated.

5.1.5 Results Characterization

To complete our data taxonomy we introduce another important topic - characterization, or requirements, for the output of the optical flow. At this point we consider the following important aspects:

Accuracy. No real-life optical flow application should be started without defining the requirements for the accuracy of a result. It can be specified as an upper bound for an average or a maximum error using a chosen error metric (e.g. endpoint error). After the flow field is computed, the results have to be evaluated according to the imposed accuracy limits. This can be achieved by the comparison with a manual tracking, or using automated confidence measures (See Section 2.1.5), or by checking the computed flow with the a-priori information (in the case if motion of some objects is known or can be deduced). Verification by visual means may also provide some clues about the accuracy of the computed flow field, but cannot be considered as a reliable way to justify the correctness of the results. Without specification of error limits and subsequent quantitative verification of the result, the computed flow field cannot be conclusive and trustworthy.

Density. Specifies whether a dense (i.e. available at every pixel) flow field is required or a sparse representation, available at specified locations is sufficient for the given task.

- sparse: in some cases the requirement on the density of a flow field can be relaxed, which can give a certain degree of freedom to choose model parameters and strategies for the post-processing. For example, one can perform the averaging of the flow vectors from multiple locations or make interpolation of the result using points of high fidelity. These approaches can be especially beneficial for the case of challenging datasets.
- dense: this is the default scenario for the classical optical flow formulation, where the flow vector is computed for every pixel location.

Motion components. Describes which feature of the motion field we are interested to compute. If only one component of the motion field is of relevance, this might give some flexibility to optimize model parameters.

- magnitude: if one is interested to estimate the amount of motion, the directional information is not very useful. An example of such scenario is the task of computation of a time-to-impact in driver assistance systems.
- direction: if one is aimed to estimate directional distribution of the motion field. For this case, the magnitude of the flow field is not relevant.
- full flow: that is the classical formulation of optical flow, when a flow vector, containing both the direction and the magnitude, is required.

Consistency. This requirement assumes the consistency of the computed flow field in a forward (between the first image and the second) and backward directions. This condition may be important for various tasks such as image registration and video processing. By the consistency we imply that the flow is the same, but opposite in direction.

- forward: the classical formulation of optical flow, which determines the flow field only in a forward direction.
- consistent: the motion field, computed in a forward and backward directions should be consistent.

Motion boundaries. Before starting optical flow computation it is useful to specify how motion boundaries around moving objects should be treated. Again, this property may be different for various applications.

- strict: Accurate recognition of motion discontinuities is required. In this case a spacial attention should be given to the features of a computation model, which may lead to an oversmoothing or a coarsening of the motion boundaries.
- not strict: Overall (average) flow is more important and requirements on the precise boundaries estimation are relaxed.

Computation time. An important requirement for an optical flow method is the time required to perform the computation. This aspect can significantly influence the choice of an optical flow model, its parameters and the overall accuracy. The requirement is governed by a particular application.

- real-time: for some kind of applications such as driver assistant systems the computation time is crucial. That is why the design of optical flow models is restricted to high-performance methods.
- interactive: for some applications the real-time performance is not required, however, some feedback from an user is expected. For instance, if the user wants to inspect the computed flow fields and adjust computation parameters.
- offline: for this case there are no limitation on the processing time, so the computation is completely offline. It means that the time complexity of an algorithm is not taken into account and the accuracy of a result is the top priority.

Data size. Describes the data size of the input and whether it is possible to process using available memory resources and particular implementation. For the optical flow problem at least two image frames and the flow field should be stored in the memory of a computation unit. For more complex methods, such as a multi-level optical flow (See Section 2.2.4), motion increments and resampled input images should be additionally stored on each computation level. For some datasets, especially this is typical for a large tomographic volumes, the input data and all variables required for an algorithm, do not fit into the available memory. Thus, the input should be reduced (i.e. using downsampling or cropping), leading to a possible loss of resolution and accuracy.

5.2 Experimental Data

In order to perform a systematic evaluation of different features of optical flow models, test the results of data preprocessing routines and investigate the influence of image artifacts, we need to have a set of synthetic test data for which a ground truth result is known. To obtain such datasets for X-ray images one may proceed in two directions: generate synthetic images exclusively for the X-ray data (see our data taxonomy Section 5.1) or, alternatively, one may use the existing datasets and model X-ray related image properties, such as image noise, low contrast and artifacts. In this work we proceed with the second approach and use synthetic datasets which are popular in the literature on optical flow methods. Our choice is based on a number of advantages, associated with the second approach. First, these datasets are extensively used and many optical flow methods are already evaluated using them, which makes it easy to compare our results with the previous work. Second, these datasets are publicly available, which can increase the reproducibility of the results.

5.2.1 Synthetic Datasets

Here we describe synthetic datasets, which we choose for our experimental studies. In total we include three datasets: two datasets depicting a real scenery and one synthetic dataset.

First two image sequences were developed and presented in the work of [BSL⁺11]. The authors made an extraordinary effort to obtain a realistic non-rigid scenes for which a highly accurate optical flow result is available. In this technique a scene is built from real-life objects and each element of the scene is moved using a mechanical motion stage. The authors applied a fine spatter fluorescent pattern to surfaces of all scene objects. Then, the camera recorded a pair of high-resolution images under the ambient light and the UV illumination (which revealed fluorescent particles). The ground truth is obtained by tracking small image blocks in the high-resolution UV images to get integer-valued motion correspondences. As a second step the authors refined the ground truth to the sub-pixel precision using the Lukas-Kanade optical flow algorithm [LK81]. As a result, a dense ground truth with a verified precision of about 1/10 pixel in the original size and 1/60 pixel in the downsampled size (used as the final result) is obtained.

Here we describe the selected datasets according to our data taxonomy (See Section 5.1).

RubberWhale dataset. It contains several independently moving objects, various motion types including non-rigid motion and rotation, different amount of textures. Evaluation according to our data taxonomy yields:

- *Image noise*: There is no noise in this dataset, perfect imaging conditions.
- *Contrast level*: The image contrast is high, the separation between the objects is good.
- *Object size*: The scene depicts objects of various sizes, but no small objects are present.
- *Object distribution*: Objects are distributed normally.
- *Object details*: Images contain objects with different amount of textures - some of them are highly textured and some of them are more homogeneous. In general, all objects have normal or high amount of details.
- *Image artifacts*: There are no image artifacts.
- *Motion type*: Several types of motion, including translation, rotation, non-rigid motion are present within this dynamic scene. Note, that non-rigid movements (on the blanket, see Figure 5.4) vary smoothly, so we do not expect it to cause complications for the optical flow estimation.
- *Motion range*: Objects are moving with the mixed range of velocities, including slow and fast motion.
- *Motion discontinuities*: There are occlusions between moving objects, but not very strong ones.

A summary of evaluation of the *RubberWhale* dataset is given in Table 5.2. The corresponding color codes to classify the influence of certain properties of input data according to our data taxonomy are given in Table 5.1.

Table 5.1: Color codes to classify the influence of certain properties of input data according to our data taxonomy.

Color	Description
good	the feature is good for data processing
normal	in general, the feature should not cause problems
challenge	the feature presents a significant challenge

Hydrangea dataset. Contains a large, highly-textured object, which rotates with a fast angular velocity in front of a large background. That background undergoes a translational movement. Evaluation according to our data taxonomy yields:

- *Image noise*: There is no noise in these images, perfect imaging conditions.
- *Contrast level*: The image contrast is high, the separation between the object and the background is good.

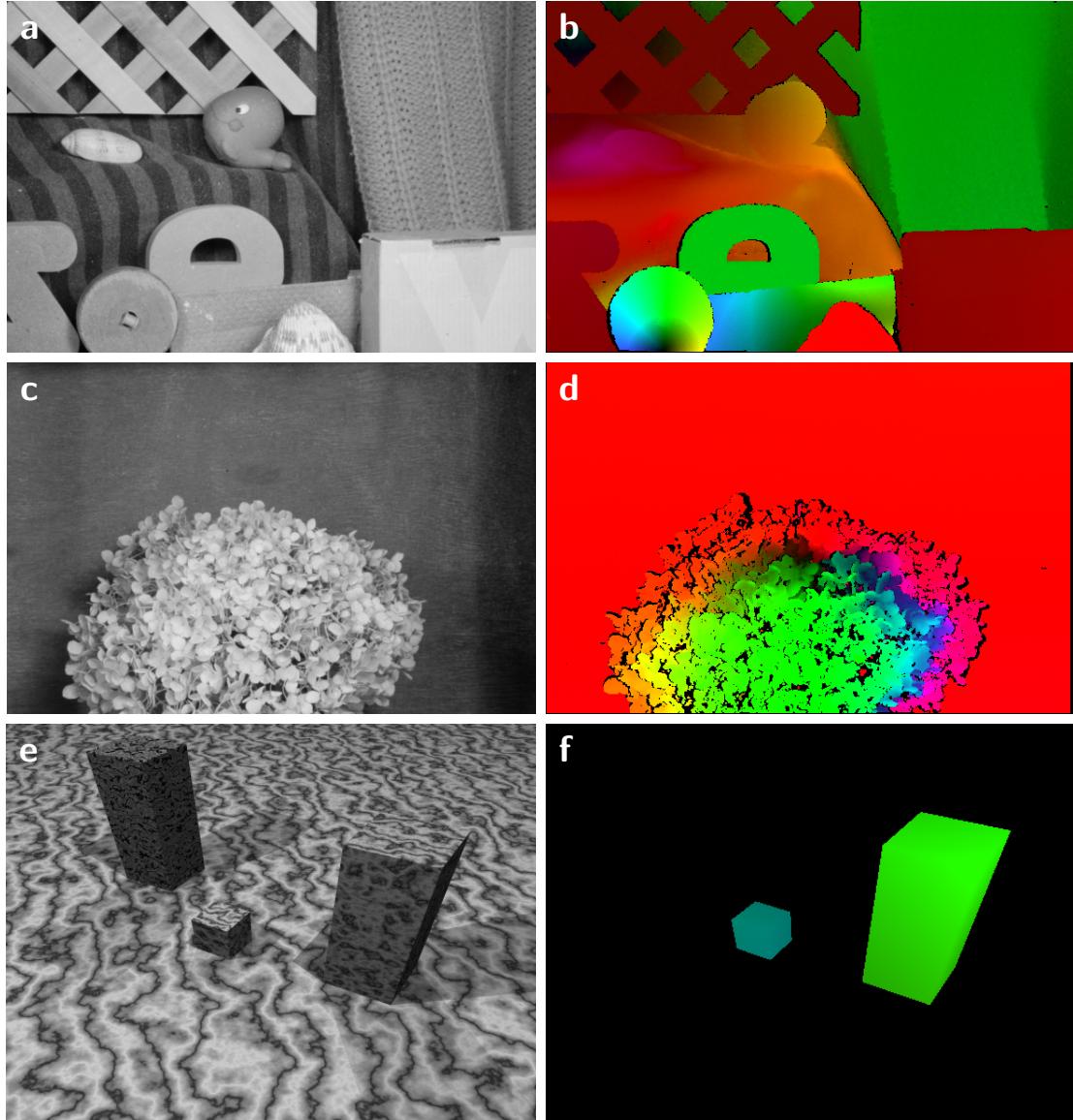


Figure 5.4: Synthetic datasets used in the scope of this work for quantitative evaluation of optical flow methods and preprocessing routines. (a,b) *RubberWhale* dataset. (c,d) *Hydrangea* dataset. (e,f) *New Marble* dataset. Left column: First frame of the corresponding image sequence. Right column: Ground truth motion field using color code.

- *Object size*: The scene depicts one object of a large size and large background.
- *Object distribution*: Only one object is moving in front of the background.
- *Object details*: The object (a plant) contains a lot of details (leaves). Background is highly textured. Individual leaves, however, have not much details.
- *Image artifacts*: There are no image artifacts.
- *Motion type*: Translational motion of the background. Fast rotation of the object.
- *Motion range*: Objects are moving with approximately the same velocity range.

Table 5.2: Evaluation of *RubberWhale* dataset.

Image	Data	Motion
noise: no contrast: high artifacts: no	size: average/large dist: norm detail: normal / high	type: vary, non-rigid range: mixed disc: occlusions

- *Motion discontinuities*: There is a substantial amount of occlusions around and within the rotating object.

A summary of evaluation of the *Hydrangea* dataset is given in Table 5.3.

Table 5.3: Evaluation of *Hydrangea* dataset.

Image	Data	Motion
noise: no contrast: high artifacts: no	size: large dist: norm detail: normal / high	type: rigid, rotation range: small disc: flow discontinuities

New Marble dataset. It is a computer generated synthetic dataset, which contains two marble blocks moving on a marble floor. This sequence is available at http://i21www.ira.uka.de/image_sequences/. Evaluation according to our data taxonomy gives us:

- *Image noise*: There is no noise in these images, the scene is synthetically rendered.
- *Contrast level*: The image contrast is high.
- *Object size*: The scene depicts objects of normal size.
- *Object distribution*: Normal object distribution.
- *Object details*: Objects and background contains a lot of details.
- *Image artifacts*: There are no image artifacts.
- *Motion type*: Translational motion of two blocks, static background.
- *Motion range*: Objects are moving with similar velocities.
- *Motion discontinuities*: There are occlusions around moving objects.

A summary of evaluation of the *Hydrangea* dataset is given in Table 5.3.

All dataset are shown on Figure 5.4. As it is evident from comparing summary tables for different datasets, the *New Marble* image sequence is the easiest sequence to process in terms of image quality, data features and motion model. On the other hand, the *RubberWhale* dataset impose more challenges for the optical flow estimation (different motion types, occlusions), and the *Hydrangea* dataset is the most challenging in terms of accuracy of the results on the boundaries of the moving object or their parts. Indeed, the inspection of the quantitative results, obtained by the state-of-the-art optical flow methods on these datasets, are in a good agreement with the qualitative estimation based on our data taxonomy. This proves usefulness of our simple, yet adequate approach.

Table 5.4: Evaluation of New Marble dataset.

Image	Data	Motion
noise: no	size: average	type: rigid
contrast: high	dist: norm	range: small
artifacts: no	detail: high	disc: occlusions

5.2.2 Modeling Properties of X-ray Data

In the current work to test the performance of data processing routines, choose the appropriate optical flow models and their parameters we use synthetic data for which the ground truth result is known. In this way we have a possibility to perform an extensive quantitative evaluation of the devised techniques and deduce sound conclusions. To achieve that we use synthetic datasets which are popular in the literature on optical flow methods, are publicly available, and extensively studied. As the next step we simulate all X-ray data related image properties, which account for different imaging scenarios. Note, that we do not aim to produce realistically looking synthetic X-ray datasets, however, such work could be useful. In this section we show our approach to model the most important properties of the X-ray data, which are image noise, low-contrast, illumination changes, and image artifacts.

5.2.2.1 Image Noise

To perform experiments under different noise conditions we choose three noise models presented in Section 3.1.1.1. We start our evaluation with a *Gaussian* model, which is a popular choice to introduce noise artifacts. Then we proceed with a more physically justified noise model - *Poisson* noise. It is important to compare how the Gaussian and the Poisson noise models are differ and which implication it makes on the design of the robust optical flow techniques. Then, we use a very specific *Spike* noise model that aims to emulate the presence of data outliers, potentially caused by dead and saturate pixels. Here we describe which parameters and implementations we use to model image noise:

- *Gaussian noise.* For the implementation of the Gaussian noise we used a plugin for an open-source ImageJ/Fiji software [SACF⁺12, SRE12] called RandomJ, which is freely available at the website: <http://www.imagescience.org/meijering/software/randomj/>. We vary standard variation σ to achieve different noise levels. An example of images contaminated with the Gaussian noise is shown in Figure 5.5.
- *Poisson noise.* For the implementation of photon noise according to the Poisson distribution we used the same plugin as for the Gaussian noise. To produce images with different level of Poisson noise we use different values of the mean value of the Poission distribution μ .
- *Spike noise.* For the implementation of the Spike noise we used an open-source ImageJ/Fiji software [SACF⁺12, SRE12]. By default the percentage of corrupted pixels affected by the routine is 5%. To increase the amount of noise we run the method multiple times.

On Figure 5.5 we show an example of *RubberWhale* dataset contaminated with the *Gaussian* noise. Histograms obtained from images with different noise levels reveal dramatic changes in the pixels content caused by noise. To check the performance of optical flow methods on images with varying amount of noise and accuracy of the results depending on the noise level we run a simple experiment on all three datasets. We vary use the Gaussian noise model and vary the standard deviation parameter. For all datasets we show the average endpoint error (AEE) in three regions: for the whole image domain (*All*, on motion discontinuities and within the homogeneous image regions (*Untext*). The evaluation results are shown on Figure 5.6. From these plots and inspection of the quantitative results we can make the following observations and conclusions:

- The *New Marble* dataset is the simplest among our testing dataset and the optical flow computation gives best results. Both errors in *All*, as well as in *Disc* and *Untext* regions are the smallest. In turn, the *RubberWhale* and *Hydrangea* datasets are much more challenging to process. Our data taxonomy allowed to predict such results, since it highlights the characteristic properties of each dataset. Thus we, again, make certain that our approach is practical.
- The performance of optical flow methods on all datasets is quite different, both in terms of accuracy, as well as in influence of noise on the final result. For example, for *RubberWhale* and *New Marble* datasets, the dependence is more or less linear. For *Hydrangea* dataset more interesting behaviour takes place (see next). This confirms the rationale behind our data classification - every dataset should be carefully analysed and treated with the optimum algorithm and parameters.
- As it can be expected, the performance in various image regions is different. The most complicated cases are regions around object and motion discontinuities. However, this also depends on the noise level and original contrast of the input sequence. For example, for *Hydrangea* dataset for noise levels more then $\sigma = 12$ in the region of homogeneous background, high amounts of noise results in a incorrect motion estimation and this becomes the dominant contribution to the overall error.
- There is a substantial statistical variability in the results depending on the concrete noise distribution. Note, that in some cases the results on images with higher amount of noise are more accurate then on images with lower amount of noise. This means that for the evaluation of noise it is not enough to make conclusions from a single noise dataset. Instead, statistical measurements should be performed. As a result, such measurement should include statistically sufficient number of experiments and report the average value and the standard deviation as a result.

Datasets with synthetically added noise will be used for the quantitative evaluation in the following sections :

- In Section 5.4.2.1 to show how noise affects the performance of optical flow and to evaluate different noise reduction approaches presented in Section 3.1.3.
- In Section 5.4.1.1 to make a comparison between simple and higher-order data constancy assumptions and draw the conclusions about which of them are more robust w.r.t. to noisy data.

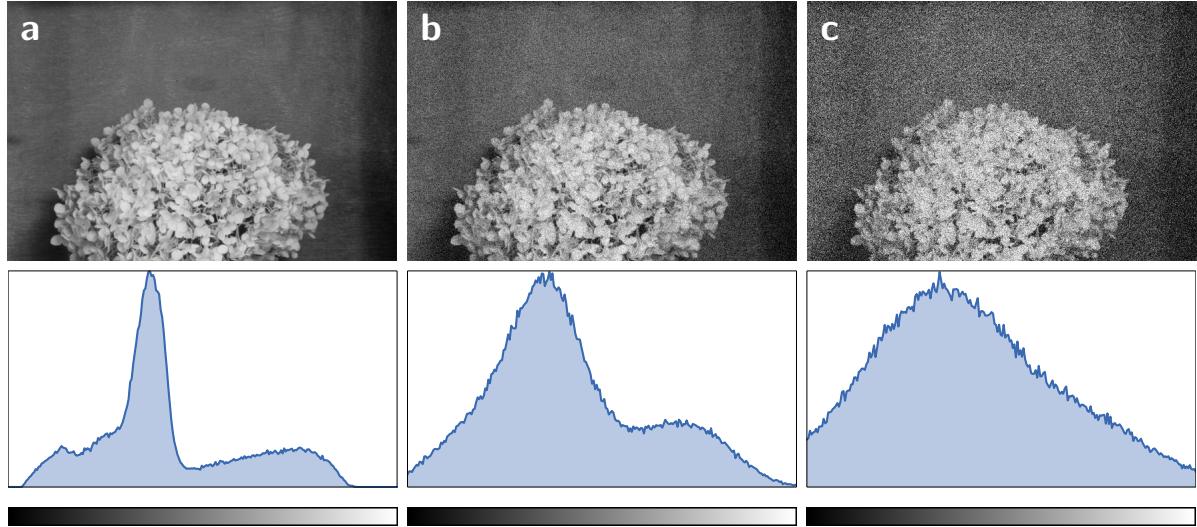


Figure 5.5: Modeling noise for the *Hydrangea* dataset using the *Gaussian* noise model. **(a)** First frame of the original image sequence. **(b)** Image with added Gaussian noise with standard deviation $\sigma = 20$. **(c)** Image with added Gaussian noise with standard deviation $\sigma = 40$. **Bottom Row:** Corresponding histograms. Note a dramatic change in the grey value distribution as a result of noise.

- In Section 5.4.1.2 we evaluate the influence of incorporating the local information into the data term (CLG approach).
- In Section 5.4.1.3 we study the result of data normalisation and its implication on the accuracy of flow computation.

5.2.2.2 Contrast Level

Another important property to evaluate is contrast. Since the X-ray data exhibits a wide range of image contrast levels (depending on imaging parameters), it is crucial to simulate and estimate the influence of image contrast on the performance of optical flow methods.

To implement various contrast levels we take the original image sequence and reduce the contrast via the histogram rebinning while keeping the original values of maximum and minimum of the original histogram. An example of contrast reduction is illustrated on Figure 5.7.

To justify the importance of investigation of contrast changes on the results of motion estimation we perform an experiment in a similar fashion as we did with the previous experiment. We take a fixed optical flow model (*baseline* method) and evaluate its performance on the images with different contrast values, varying it from 100% of contrast to 7.5%. The evaluation plots are shown on Figure 5.6. Analysing them we can make the following conclusions:

- The shrinkage of the dynamic range and a loss of useful information causes a severe degradation of the accuracy of optical flow results. For the substantial decrease in contrast the performance drop is significant and can be compared with the influence of noise. That is why even for low noise image data, an appropriate adjustment of computation parameters is required.

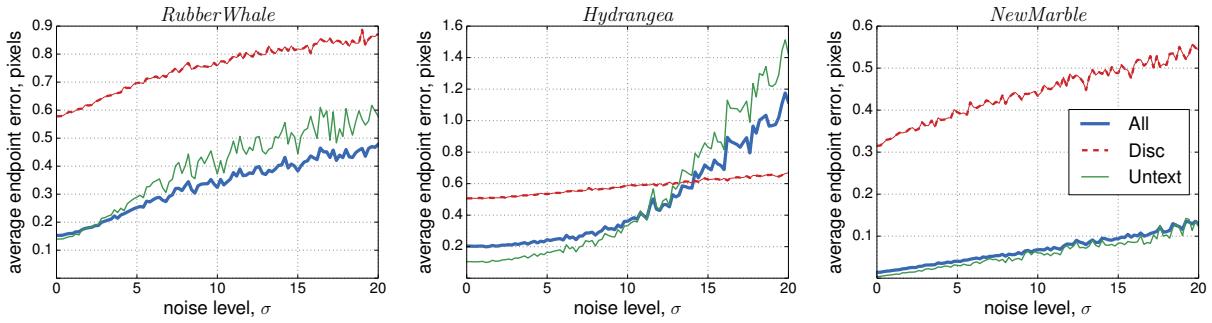


Figure 5.6: Evaluation of optical flow performance on noisy data with *Gaussian* noise. For all datasets an average endpoint error (AEE) is shown in 3 regions: *All* region is depicted as a green plot; *Untext* region as a blue plot and *Disc* region as a red plot. **Left:** *RubberWhale* dataset. **Middle:** *Hydrangea* dataset. **Right:** *New Marble* dataset. Method: *Baseline* algorithm (See Section 5.3.3).

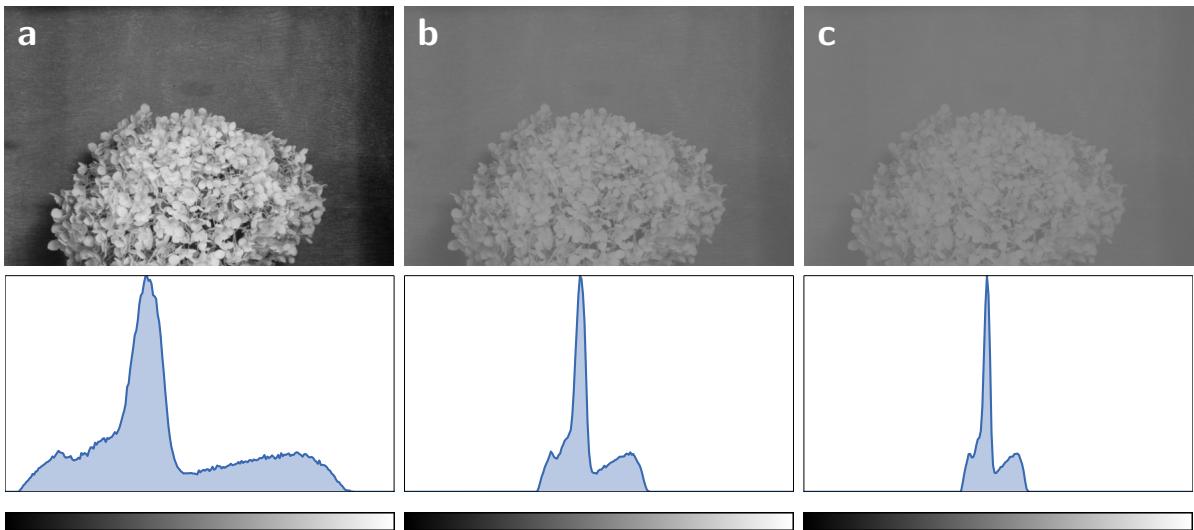


Figure 5.7: Modeling datasets with low feature contrast. **(a)** First frame of the original sequence. Contrast level $C = 100\%$. **(b)** Image with the reduced contrast. Contrast level $C = 25\%$. **(c)** Contrast level $C = 12.5\%$. **Bottom Row:** Corresponding histograms showing a reduction in the dynamic range and, as a result, in the contrast level.

- *Hydrangea* dataset contains a large amount of textural information in all image regions, as a result it is the easiest dataset to process, even in the case of low-contrast data.

We will use the datasets with the low-contrast in the following experiments:

- In Section 5.4.2.3 to show how a decrease in the image contrast affects the performance of optical flow , as well as to evaluate different contrast enhancement approaches presented in Section 3.1.3.
- In Section 5.4.3.1 to check the performance of different data terms, make comparison between them and draw conclusions about the data terms which are more robust with respect to low-contrast data.

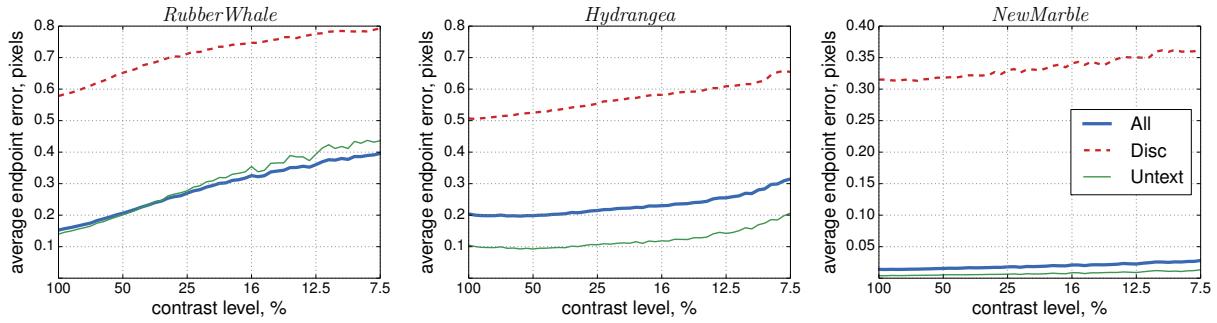


Figure 5.8: Evaluation of optical flow performance on datasets with degraded contrast level. For all datasets an average endpoint error (AEE) is shown in 3 regions: *All* region is depicted as a green plot; *Untext* region as a blue plot and *Disc* region as a red plot. **Left:** *RubberWhale* dataset. **Middle:** *Hydrangea* dataset. **Right:** *New Marble* dataset. Method: *Baseline* algorithm.

- In Section 5.4.3.3 to evaluate the influence of robustification of the data term for low contrast image data.

5.2.2.3 Brightness Changes

In this section we describe different types of illumination changes, which may occur in X-ray data. Since the assumptions about brightness distribution of the scene and objects is the core information for the construction of data terms, any changes in brightness or illumination are important to take into account for the design of optical flow model or data preprocessing.

In this work we consider the following models of illumination changes:

- Global additive illumination. For this case a constant value of A is added to every pixel of the second image of the input sequence.
- Global multiplicative illumination. Every pixel of the second image frame is multiplied by a factor M . It is important to note, that changes in exposure time are corresponding to global multiplicative brightness change, thus is very important to model.
- Smooth spatial variation. We consider two patterns: "*Spot*", resembling changing beam spot and "*Strips*" pattern, simulating horizontal brightness flickering due to instability of a monochromator.

Two spatial patterns intended to simulate the uneven brightness distribution are presented in Figure 5.9.

We use datasets with the simulated brightness changes for the following evaluation experiments:

- In Section 5.4.2.2 to show the performance and usefulness of the brightness corrections techniques.
- In Section 5.4.3.2 to check the performance of different data terms for various brightness changes scenarios.

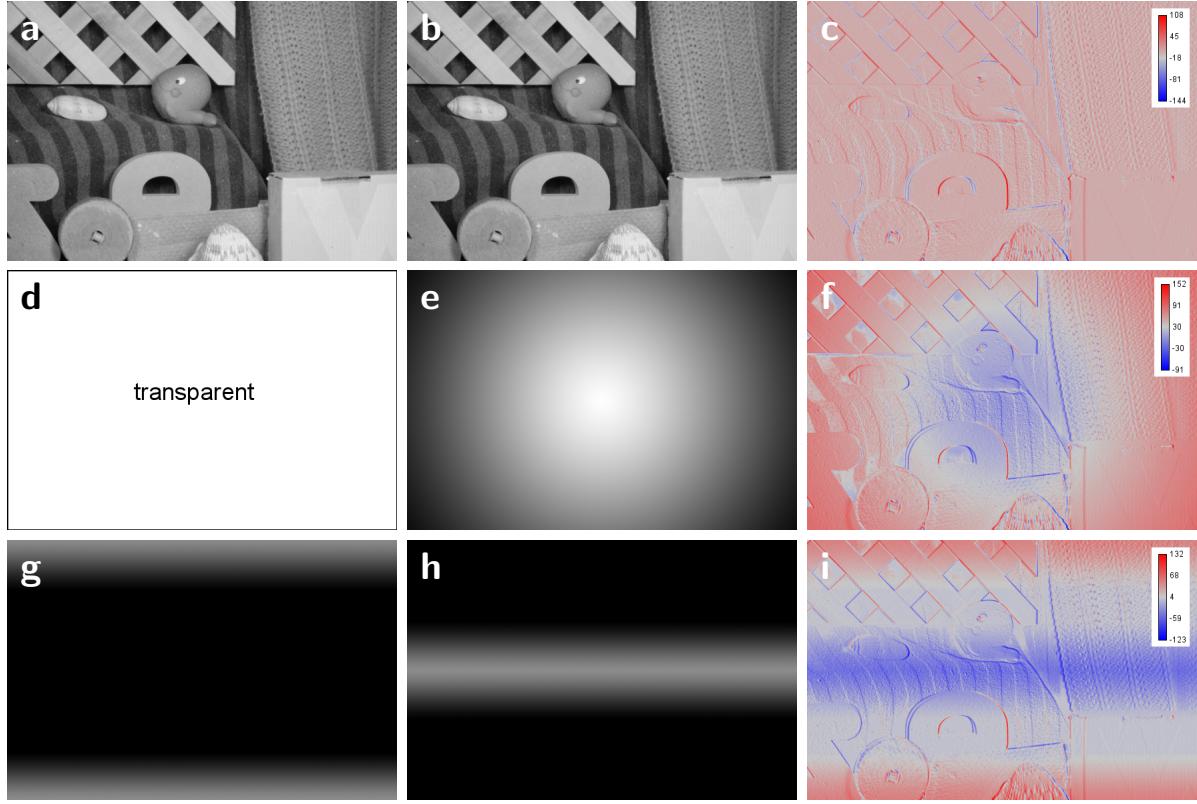


Figure 5.9: Modeling illumination changes. (a) First frame of the original sequence. (b) Second frame. (c) Difference between original images that shows amount of brightness changes in a scene due to the actual movement. (d) The brightness pattern added to the first frame. In this case its is transparent, so there is no modification of pixel values. (e) A "Spot" brightness pattern added to the second frame. (f) Image changes due to the "Spot" brightness pattern. (g) The first part of the "Stripes" brightness pattern, added to the first frame. (h) The second part of the "Stripes" brightness pattern. (i) Changes due to the "Stripes" brightness pattern.

- In Section 5.4.3.3 to evaluate the influence of robustification of the data term for low contrast image data.

5.2.2.4 Image Artifacts

The last step to complete our framework for quantitative evaluation is to provide the simulation of image artifacts. As it was mentioned previously, there are two prevalent artifacts in the X-ray data, which corrupt image data - ring artifacts and star (or streaks) artifacts. The main requirement for the modelling of such artifacts is that their location and the degree to which these artifacts affect the original data must be controlled. In this work we employ a simple model using geometric shapes similar to original effects. We control the placement of such artifacts and gradually increase the thickness of shapes to influence the intensity of the artifacts.

In this work we use the following artifacts structures:

- Line shapes. These artifacts intended to mimic star artifacts (See Section 5.1.3).

- Circular shapes. These artifacts intended to model ring artifacts (See Section 5.1.3).
- Dead pixels, dust and scratches artifacts. These artifacts are modelled using *Spike* noise and were presented in Section 5.2.2.1.

Images with the artificially modelled artifacts intended to simulate image degradation by means of ring or star artifacts are presented in Figure 5.10.

We use datasets with the simulated artifacts to perform quantitative evaluation in the following experiments:

- In Section 5.4.3.4 to check the performance of the robust data term in the presence of image artifacts.
- In Section 5.4.3.5 to evaluate the approach of median filtering and its influence on the accuracy of the optical flow.
- In Section 5.4.3.6 to evaluate the performance of the data refinement procedure.

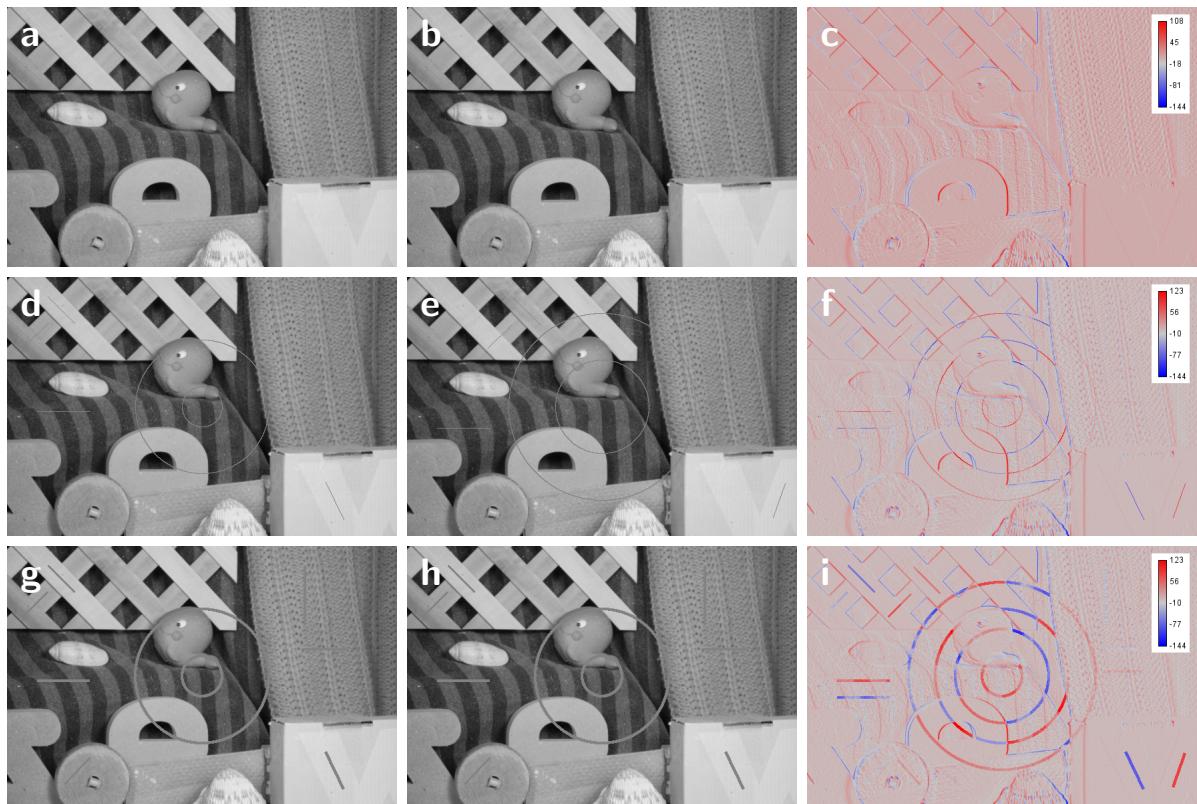


Figure 5.10: Modeling image artifacts. (a) First frame of original sequence. (b) Second frame of original sequence. (c) Difference between original images that shows amount of brightness changes in the original scene due to actual movement. (d) First image frame with artifacts. The thickness of artifacts is $A = 1$ pixel. (e) Second image frame with artifacts $A = 1$. (f) Corresponding image difference which reveals the affected areas. (g) First image frame with artifacts. The thickness of artifacts is $A = 4$ pixels. (h) Second image frame with artifacts $A = 4$. (i) Corresponding image difference.

5.3 Optimization of Parameters

In the section dedicated to data taxonomy we described a variety of X-ray data. In the previous section we showed how the performance of optical flow methods can be governed by the choice of model parameters (on the example of noisy images 5.2.2.1 and low-contrast data 5.2.2.2). Therefore, parameters optimization become a critical and challenging task. For example, from the literature on optical flow methods it is well-known that appropriate choice of the smoothness parameter is important to obtain desirable results. However, there is remarkably little work on methods which allow to select or estimate optimal parameters. Only recently in the work of [ZBW11] the authors presented an approach for automatic smoothness parameters estimation based on the optimal prediction principle (See Section 2.1.5.6). In our work we evaluate more approaches and confidence metrics (See Section 2.1.5). In this section we discuss numerous aspects of parameter optimization.

5.3.1 Optimization of Individual Parameter

First we start with the evaluate and discussion of different model parameters of optical flow computation. The aim is to identify the set of paramters which are general and allow to obtain accurate and robust results. Selecting the best paramters we obtain a basis for our *baseline method* which will be presented later in the chapter (See Section 5.3.3). Moreover, during the evaluation we highlight the importance of such optimization and provide a general guidance to perform it. Here we evaluate the most basis components of optical flow computation, such as derivatives approximation, number of warping levels and warping scale factor.

5.3.1.1 Derivatives Approximation

In Section 2.2.2 a number of alternatives to represent image derivatives were outlined. Here we compare different stencil modes and draw the conclusions about their appropriate usage for different image properties. For that, we consider two important aspects from our Data Taxonomy section 5.1, namely the influence of noise and low contrast (low spatial image gradient). To check the performance on noisy data we restrict our study to *Gaussian noise* model 3.1.1.1, since it introduces sufficient disturbances to the image data. For the evaluation we fix all the model parameters (which were optimized ahead), and vary only the computation schemes for derivatives approximation. The results for noisy data are shown in Table 5.5 and for low-contrast data in Table 5.6.

With respect to noisy data we can make the following observations:

- *Temporal averaging improves the results.* From the evaluation result we conclude that temporal averaging of is useful for both original and noise datasets and provided a robust scheme for derivatives. However, one should be confident that the displacements between corresponding image pairs are small, so such step could be performed. In the presence of large displacements we employ multi-level strategy, discussed in Section 2.2.5. The appropriate choice for the warping scale and number of warping levels we discuss in Section 5.3.1.3 and Section 5.3.1.2 respectively.
- *Increased sensitivity of fourth-order derivatives to noise.* Despite the facts, that fourth-order derivatives approximation may lead to a significant improvements in

accuracy for the original, noise-free images, it does not perform well on noise data. In the presence of noise a second-order approximation performs better for all noise levels. This might be explained by the increased sensitivity of the fourth order-derivative scheme to noise.

Table 5.5: Comparison between spatial derivatives approximation schemes on noisy *RubberWhale* dataset. The following computation schemes are compared: second-order, second-order with temporal averaging, fourth order and fourth-order with temporal averaging. An additive *Gaussian noise* with a zero-mean and standard deviations $\sigma=0, 5, 10, 20$ was introduced.

noise	model	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>
$\sigma = 0$	2nd order	0.172	0.683	0.135
	2nd order + temp	0.161	0.628	0.13
	4th order	0.169	0.673	0.132
	4th order + temp	0.157	0.605	0.127
$\sigma = 5$	2nd order	0.315	0.822	0.289
	2nd order + temp	0.284	0.748	0.264
	4th order	0.343	0.841	0.311
	4th order + temp	0.304	0.751	0.273
$\sigma = 10$	2nd order	0.581	1	0.544
	2nd order + temp	0.506	0.921	0.454
	4th order	0.664	1.037	0.652
	4th order + temp	0.593	0.951	0.564
$\sigma = 20$	2nd order	1.414	1.584	1.508
	2nd order + temp	1.317	1.502	1.416
	4th order	1.494	1.622	1.63
	4th order + temp	1.45	1.576	1.6

We can make the following conclusions about influence of different derivatives approximations on data with low contrast:

- *Usefulness of temporal averaging.* The same as for the case of noisy data, temporal averaging show to be effective to improve the accuracy of the optical flow computation.
- *Improved accuracy using fourth-order derivatives.* The accuracy of optical flow computation is improved using fourth-order derivatives approximation.

The evaluation of derivatives approximation schemes on *Hydrangea* and *New Marble* datasets shows similar outcomes. Summarizing both experimental results we may conclude the following: in the presence of substantial noise we choose second-order derivatives with temporal averaging, if noise level is low or can be significantly reduced using preprocessing step we opt to use fourth-order derivatives with temporal averaging.

5.3.1.2 Warping Levels

In order to deal with large displacements according to our multi-level computation strategy (Section 2.2.5) one has to construct image pyramid. For that purpose a scaling

Table 5.6: Comparison between spatial derivatives approximation schemes on *Rubber-Whale* dataset with low contrast. The following computation schemes are compared: second-order, second-order with temporal averaging, fourth order and fourth-order with temporal averaging. A decrease to a contrast level of $C=100\%$, 50% , 25% , 10% was introduced.

contrast	model	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>
$C = 100\%$	2nd order	0.172	0.683	0.135
	2nd order + temp	0.161	0.628	0.13
	4th order	0.169	0.673	0.132
	4th order + temp	0.157	0.605	0.127
$C = 50\%$	2nd order	0.224	0.681	0.238
	2nd order + temp	0.214	0.667	0.223
	4th order	0.207	0.662	0.213
	4th order + temp	0.193	0.64	0.195
$C = 25\%$	2nd order	0.3	0.729	0.348
	2nd order + temp	0.293	0.721	0.334
	4th order	0.284	0.717	0.325
	4th order + temp	0.275	0.707	0.308
$C = 10\%$	2nd order	0.41	0.797	0.48
	2nd order + temp	0.398	0.791	0.461
	4th order	0.385	0.785	0.451
	4th order + temp	0.372	0.776	0.427

function is used which transfers the original image version to a coarser resolution level and interpolates the obtained results back to the finer computation level. Thus, on each computation level k the image size is determined via:

$$N_d^k = N_d^{orig} \eta^k,$$

where η is a warping scale parameter, N_d denotes the image size in the dimension $d \in (x, y)$. In general, to get the best results and capture whole range of displacements (including the fastest motion) one may downscale the image as much as possible. However, for large images with small or moderate displacement values this strategy could result in a substantial loss of performance in terms of computation time. In fact, the aim is to downscale the original image in such a way, that the large displacements become small-scale on the lowest resolution level, so the linearized data constancy assumption can be employed. As soon as it is achieved, further downscaling provide no additional improvement. We illustrate this situation on the *RubberWhale* dataset in Figure 5.11. Ideally, if the motion model is known or can be deduced from the original image sequence, it is possible to extract the optimal coarsest image size (e.g. number of warping levels) automatically.

Here we propose to use two strategies:

- *Estimation based on maximum velocity.* We compute the coarsest resolution image size via $N^k = N^{orig} * S_{max}$, where $S_{max} = \frac{1}{max(\mathbf{w})}$ is a scale factor and $max(\mathbf{w})$ represents maximum displacement value for the current image pair.

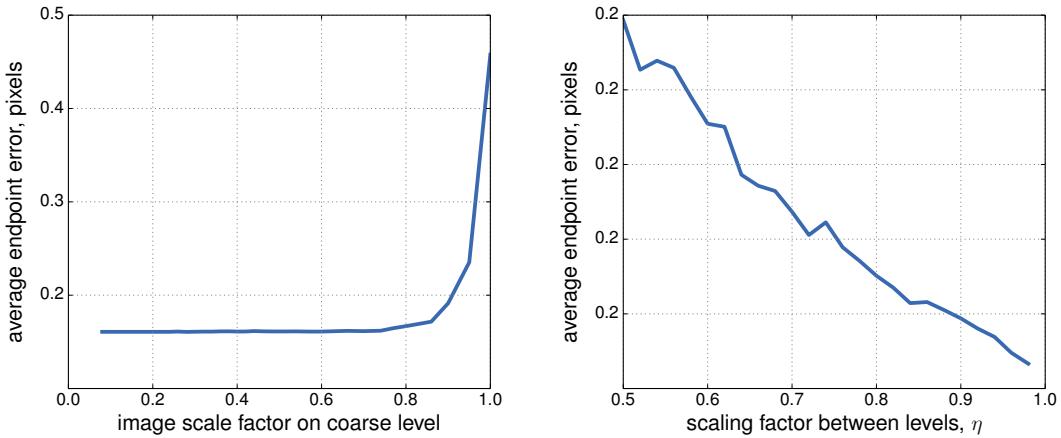


Figure 5.11: Dependence of optical flow accuracy on a number of levels (coarse scale size) and a scaling factor η . **Left:** Improvement of average endpoint error for decreased image size on the coarsest resolution level. All the parameters were fixed as for *baseline* method, warping scale factor $\eta = 0.95$. Note, that there is no further improvement after a certain scale factor is reached. **Right:** Improvement of average endpoint error for increasing scaling factor. All the parameters were fixed as for *baseline* method, a number of required warping levels is chosen according to a minimum possible coarsest image size.

- *Estimation based on mean velocity.* We compute in a similar manner as a previous measure, but using mean velocity. Such measure can be reasonable in case if it is known that objects are moving with similar velocities. To account for velocity variations we include the standard deviation value of the velocities distribution. The final measure reads $N^k = N^{orig} * S_{mean}$, where $S_{mean} = \frac{1}{\text{mean}(\mathbf{w}) + \text{std}(\mathbf{w})}$.

We compare both measures in the Table 5.7. How we compute S_{best} : we increase the amount of warping levels (decreased image size on the coarsest resolution level) as long as there is a substantial improvement (more than 0.001 for AEE measure and 0.1% for RX1.0 measure).

Table 5.7: Comparison of methods to select an appropriate number of warping levels for *RubberWhale*, *Hydrangea* and *New Marble* datasets.

dataset	max	mean	std	S_{best}	S_{max}	S_{mean}
<i>RubberWhale</i>	4.616	1.236	0.505	0.7	0.22	0.57
<i>Hydrangea</i>	11.12	3.48	1.452	0.57	0.09	0.2
<i>New Marble</i>	2.09	1.64	0.166	0.66	0.48	0.55

Conclusions about automated selection of coarsest resolution image size:

- *Maximum-based measure provides good results.* This measure provides a reasonable precision and adapts to the image size and correctly handles largest displacements, but computations on the coarsest levels introduced by the measure can be redundant. This can be seen from the comparison between S_{best} and S_{max} for the *Hydrangea* dataset. By default, if we are interested in computing an accurate optical flow we choose this measure.

- *Mean-value improves the computation time.* This measure improves the computation time by adapting image to the mean velocity magnitudes and consequently considering less image levels. This measure is closer to the S_{best} scale factor, however could provide poor results for large displacements.

5.3.1.3 Warping Scale

In general, the warping scaling factor should be as small as possible to guarantee a smooth transfer between warping levels. However, too small value will result in a very large number of warping levels, hence find compromise. An influence of scaling factors on *RubberWhale* dataset on the overall precision of optical flow is shown in Figure 5.11.

A range of values between 0.9 and 0.95 is a good compromise between accuracy and running time.

5.3.2 Confidence-based Optimization

In the previous two sections we have based our parameters optimization procedure on the use of ground truth data, so we were able to compare the computed results with the real ones. For the real life data (tasks) such information is not available. For this purpose a measure based on the confidence can be used. In Section 2.1.5 a number of possible confidence measures were presented.

Here we demonstrate an example of comparison of metrics to evaluate the correctness of the result using average endpoint error 2.1.1.2 and data constancy error 2.1.5.3. We vary the parameter of the flow regularization α and measure both metrics. The results are presented in Figure 5.15. As it can be observed, even without ground truth result it is possible to judge the reliability of the results and choose a sub-optimal parameter based on confidence-based optimization.

We further evaluate these measures in the experimental section 5.4.4 and draw conclusions about their appropriate usage.

5.3.3 Baseline Method

As it was discussed in the introductory part of the current section, dedicated to parameter optimization, the performance of optical flow largely depends on the set of chosen model parameters. Since advanced optical flow methods tend to incorporate many features into a single model, it is challenging to separate and evaluate the influence of the individual setting. To provide a basis for a systematic comparison of different settings and model parameters here we describe a *baseline* method which we use in the following experiments section. In all parts where we refer to the baseline method we mean this set of settings.

5.4 Quantitative Evaluation

In this section we present a discussion on the performance, influence of preprocessing, modelling of noise and data outliers and the use of confidence measures. For each experimental evaluation we discuss the advantages and shortcomings of the proposed prototypes. All the results are presented in both a quantitative and a qualitative way. For

Table 5.8: A list of all model parameters used for the *baseline* method. If not presented explicitly, all the parameters of the *baseline* method correspond to the ones given in this table.

Model parameter	Parameter name	Symbol	Value / Expression
Constancy assumption	grey value	$D_{grey}(I, \mathbf{w})$	$(\nabla_3^\top I \mathbf{w})^2$
Smoothness assumption	flow-driven	$S_{flow}(u, v)$	$\Psi_S(\nabla u ^2 + \nabla v ^2), \Psi_S(x) = \sqrt{x + \epsilon}, \epsilon = 0.001$
Robust function	Charbonnier penalty	$D_R(I, \mathbf{w})$	$\Psi_R((\nabla_3^\top I \mathbf{w})^2), \Psi_R(x) = \sqrt{x + \epsilon}, \epsilon = 0.001$
Derivatives approximation	second-order and time averaging	I_x, I_y	$\frac{(I_{i+1,j,t+1} - I_{i-1,j,t+1}) + (I_{i+1,j,t} - I_{i-1,j,t})}{4h_x}$
Data normalization	–	$D_{norm}(I, \mathbf{w})$	$\frac{(\nabla_3^\top I \mathbf{w})^2}{ \nabla_2 I(x, y, t) ^2 + \eta}$, disabled
Combined local-global	–	$D_{CLG}(I, \mathbf{w})$	$K_p * (\nabla_3^\top I \mathbf{w})^2$, disabled
Median filtering	–	$M_p(\mathbf{dw}^k)$	$p \in (3, 5)$, disabled
Flow regularization	smoothness parameter	α	3.5
Adaptive regularization	–	$\alpha(k)$	α , disabled
Presmoothing parameter	standard deviation	σ	0.35
Multi-level scheme	warping levels	k	15
Warping accuracy	warping scale	η	0.9

all the experiments we outline important observations and draw the conclusions, which we summarize in the last section of the chapter (See Section 5.6). In the summary table columns *all*, *disc* and *untext* show the *best possible* result for each region of statistics over the whole range of evaluated parameters.

5.4.1 Experiments: Noise

5.4.1.1 Experiment: Data terms for noisy data

In this experiment we compare different data terms with respect to their performance on noisy data. To model noise we choose three models - Gaussian, Poisson and Spike noise (see Section 3.1.1.1), which aim to represent different noise scenarios. While *Gaussian noise* is commonly used in the literature on optical flow to test the robustness of methods in the presence of noise, it is still a simplified noise model. *Poisson noise* is a physically based noise model and correctly describes X-ray data. Finally, *Spike noise* represents a special type of noise with a severe level of image degradation and aims to model data artifacts. Details on the implementation of noise models are given in Section 5.2.2.1. We evaluate the performance of data constancy assumptions on noisy data and present our results in Table 5.9. For the evaluation we use the *baseline* optical flow method with the optimized smoothness parameter α and presmoothing parameter σ .

From the results of quantitative evaluation we can get the following conclusions:

- *Gradient constancy assumption with the robust setting provides good results for both Gaussian and Poisson noise models.* It is reasonable to assume that a derivatives-based constancy assumption will be more sensitive to noise, than a simple grey value

Table 5.9: Performance analysis of data terms on noisy data on the *RubberWhale* dataset. *Gaussian noise* with zero mean and standard deviation values of $\sigma_n = 0, 5, 10, 20$ was added. *Poisson noise*: Mean value of the Poisson random distribution P is 0, 50, 200, 400. *Spike Noise*: we randomly replace a certain percentage of pixels with black and the same amount of pixels with white grey value. For the experiment an amount of $S = 0\%$ (no noise), 2.5%, 5.0%, 7.5% replaced pixels was used. Method: *baseline*, optimized for $\sigma = [0.15..2.5]$, $\alpha = [1.0..40]$.

assumption. In the work of [Bru06] it was shown that, indeed, the gradient based constancy assumption gives worse performance in comparison with the grey value assumption. However, in the evaluation perform in [Bru06] the robust data term was not used. From our performance analysis it is evident, that the gradient constancy assumption in a robust setting provides better results even for large amounts of noise in all image regions.

- *Higher order derivatives are sensitive to noise for the Gaussian, Poisson and Spike noise models.* Both the Laplacian and Gradient norm constancy assumptions are very sensitive to noise and provide poor results. Thus, these data terms are not useful for X-ray data, which usually contains high amounts of noise. However, one should note, that for noise-free images these data terms outperform the grey value constancy assumption.
- *Grey value constancy assumption is more robust for the Spike noise model.* For the Spike noise model the situation is different from the Gaussian noise model. In this case, grey value constancy assumption is more robust to corrupted pixel data. One may notice, that in some cases high order constancy assumptions perform better in *Disc* and *Untext* image regions, however, this situation corresponds to the case when the largest values for smoothing parameters α and σ are chosen. In this case the image was oversmoothed, which is seen from the values of *AEE* measure in all image regions.
- *Grey value constancy assumption is more robust for the Spike noise model.* For this noise model the situation is different as for the Gaussian noise model. In this case, the grey value constancy assumption is more robust with respect to corrupted pixel data. One may notice, that in some cases high order constancy assumptions perform better in *Disc* and *Untext* image regions, however, this situation corresponds to the case when the largest values for smoothing parameters α and σ are chosen. In this case the image was oversmoothed, which is seen from the values of *AEE* measure in all image regions.
- *Grey value constancy assumption is more robust for the Poisson noise model in the regions of image discontinues.* In general, the Poisson noise model is more challenging than the Gaussian noise model. This is due to the fact that the effect of Gaussian noise can be effectively removed with an appropriate image smoothing. Additionally, the evaluation of results shows that the grey value constancy assumption provides better performance in the regions of image discontinues.

5.4.1.2 Experiment: Combined local-global approach

In this experiment we evaluate the performance of the Combined Local-Global (CLG) approach on noisy data. In the literature on optical flow methods such approaches are not commonly used, especially for synthetic datasets. However, as it was shown in the works of [BWS02, BWS05], such approach can be beneficial in the presence of large amount of noise. We extend the evaluation presented in [Bru06] and perform quantitative evaluation for both the grey value and the gradient value constancy assumptions on *Gaussian* and *Spike* noise models. The evaluation results on the *RubberWhale* dataset are presented

in Table 5.10. The evaluation results are similar for the *Hydrangea* and the *New Marble* datasets and excluded from presentation.

Table 5.10: Performance of the Combined Local-Global (CLG) approach on the *Gaussian* and *Spike* noise models on *RubberWhale* dataset. *Gaussian Noise* with zero mean and standard deviation values of $\sigma_n = 5, 10, 20$ was added. For *Spike Noise* an amount S of 2.5%, 5.0%, 7.5% replaced pixels was used. Method: *baseline*, optimized ρ in the range [0.5..2.0], optimized σ in the range [0.35..1.5], optimized α in the range [3.0..40].

model	σ_n	Gaussian noise			Spike noise			
		error			S	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>		<i>all</i>	<i>disc</i>	<i>untext</i>
grey		0.26	0.722	0.261		0.437	0.838	0.576
grey+CLG	5	0.222	0.706	0.219	2.5%	0.424	0.838	0.576
grad		0.233	0.689	0.242		0.543	0.92	0.614
grad+CLG		0.218	0.689	0.235		0.516	0.916	0.588
grey		0.346	0.774	0.359		0.611	0.902	0.683
grey+CLG	10	0.299	0.755	0.3	5.0%	0.609	0.909	0.685
grad		0.331	0.767	0.346		0.67	0.992	0.732
grad+CLG		0.308	0.769	0.328		0.657	0.981	0.744
grey		0.474	0.866	0.567		0.761	0.978	0.733
grey+CLG	20	0.434	0.862	0.51	7.5%	0.711	0.985	0.709
grad		0.462	0.866	0.545		0.761	1.031	0.71
grad+CLG		0.443	0.865	0.516		0.739	1.007	0.714

Performance evaluation of Combined Local-Global (CLG) approach leads to the following conclusion:

- *Integration of local information is useful to improve results on noisy data.* For both constancy assumptions and both noise models used for the evaluation, the Combined Local-Global (CLG) approach gives significantly better results. The performance is better even near the boundaries between moving objects (*disc* region). It can be expected that in these regions a local integration may spoil the result, but it was not the case. Additionally, the CLG model shows better performance on the *Spike* noise model. As a result of this evaluation, we make a definite conclusion, that the CLG model is useful for noisy X-ray images.

5.4.1.3 Experiment: Normalization of data term

In this section we investigate the influence of normalization of data terms. Such approach is relatively novel, but is promising to correct the information in data term, which can be corrupted by noise or image artifacts. For the evaluation we restrict ourselves to a grey value constancy assumption, however, normalization of other data terms can be performed in a similar manner. Furthermore, we show the performance of data normalization only on *Gaussian noise* model. The evaluation results on *RubberWhale* dataset are presented in Table 5.11.

Results of performance evaluation of data term normalization approach are:

- *Data normalization improves results on noisy data.* As it was reasoned in Section 2.1.3.4, image locations with high gradient provide more contribution to the overall

Table 5.11: Performance analysis of normalization of data term on noisy data for the *RubberWhale* dataset. *Gaussian Noise* with zero mean and standard deviation values of $\sigma_n = 0$ (no noise), 5, 10, 20 was added. For the comparison we use a grey value constancy assumption model *without* data normalization and *without* the CLG approach; a model *with* the CLG; model *with* normalization of the data term; model *with* normalization and the CLG approach. Method: *baseline*, model = {grey, grey + CLG, grey + norm, grey + norm + CLG}, optimized ρ in the range [0.5..2.0], optimized η in the range [0.1..5.0], optimized σ in the range [0.35..1.5], optimized α in the range [0.5..40].

noise	model	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>
$\sigma_n = 0$	grey	0.159	0.622	0.13
	grey+CLG	0.142	0.602	0.11
	grey+norm	0.14	0.582	0.111
	grey+norm+CLG	0.143	0.615	0.113
$\sigma_n = 5$	grey	0.26	0.722	0.261
	grey+CLG	0.222	0.706	0.219
	grey+norm	0.226	0.706	0.225
	grey+norm+CLG	0.218	0.711	0.215
$\sigma_n = 10$	grey	0.346	0.774	0.359
	grey+CLG	0.299	0.755	0.3
	grey+norm	0.303	0.764	0.3
	grey+norm+CLG	0.29	0.772	0.29
$\sigma_n = 20$	grey	0.474	0.866	0.567
	grey+CLG	0.434	0.862	0.51
	grey+norm	0.457	0.854	0.523
	grey+norm+CLG	0.448	0.859	0.513

energy of the data term and thus largely influence the estimation of optical flow. Since such image gradient may be caused by data outliers such as noise or image artifacts, the influence of high image gradient can be intentionally suppressed. From the quantitative evaluation one can see, that the normalization step is indeed helps to improve the results of optical flow.

- *Data normalization approach outperforms the Combined Local-Global approach on flow boundaries.* It should be noted, that despite the fact that the CLG model gives better overall performance and improves the result in untextured image regions, the approach based on data normalization gives more accurate results on flow discontinuities. This could be explained by the smoothing properties of local integration in the motion tensor used in the CLG approach.
- *Data normalization can be further combined with the CLG approach to improve results.* An important advantage of the data normalization approach, is that it can be further combined with a CLG approach to improve the result of optical flow computation. For all noise levels a combination of both approaches lead to a significant improvement of the result in all image regions.

5.4.2 Experiments: Preprocessing

5.4.2.1 Experiment: Noise reduction

In this section we evaluate the performance of noise reduction techniques and their influence on the accuracy of optical flow methods. An important question to answer in this section is - does an advanced preprocessing procedure improves the results of optical flow? The second important aspect is to reveal the most suitable filtering procedure for the noise scenarios which are used in our evaluation.

We do not attempt to evaluate a large set of possible filtering procedure existing in the literature. Instead, we restrict ourselves to a number of popular filters, which are easy to implement. We evaluate four denoising filters, presented in Section 3.1.1.2:

- Gaussian smoothing
- Median filter
- Bilateral filter
- Anisotropic diffusion

For each individual filter we optimized its parameters and show only the best achievable result in the final evaluation table. Quantitative comparison of denoising filters and influence on the accuracy of flow computation is presented in Table 5.12 and Table 5.13 and gives us the following results:

- *Noise filtering of initial data can be helpful.* As it can be observed from the resulting tables, for every noise model and all corresponding noise levels, some kind of preprocessing filter always gave better results.
- *For each noise model there is an optimal filtering procedure.* Not surprisingly, a Gaussian smoothing performed significantly better then any other methods on the dataset with *Gaussian* noise. The same situation is true for the median filtering, when it is applied on datasets with *Spike* noise. However, both filtering methods failed to provide good results for datasets, for which they are not well suited.
- *Gaussian filtering is a simple and effective filtering procedure.* Despite its simplicity, Gaussian smoothing still provides relatively good results. For *Gaussian* noise model no filter could outperform it, even if all the parameters are carefully optimized.
- *Gaussian filtering is useful to remove Poisson noise.* Gaussian filtering shows better performance to remove the *Poisson* noise then the other approaches.
- *Usefulness of Anisotropic diffusion filtering for complex noise scenarios.* It is interesting to note, that anisotropic filter which performed relatively poor on both *Gaussian* and *Spike* noise models, provides the best results in the case when both models are combined (see Table 5.13). This could be explained by the fact that anisotropic filtering combines smoothing properties of the Gaussian smoothing and ability to filter data outliers as of the median filtering.

- The appropriate choice of a filtering procedure and its parameters is crucial, however, a challenging task. From the extensive evaluation on multiple datasets and noise models, we conclude that preprocessing of initial (raw) data is useful and allows to achieve more accurate results. However, one should carefully investigate a proper choice of appropriate filtering procedure based on extensive evaluation of the given data.

Table 5.12: Performance analysis of noise reduction techniques on *Gaussian*, *Poisson* and *Spike* noise models. *Gaussian* noise with zero mean and standard deviation values of $\sigma_n = 5, 10, 20, 40$ was added. *Poisson* noise: Mean value of Poisson random distribution $P = 50, 200, 400, 800$. For *Spike* noise an amount $S = 2.5\%, 5.0\%, 7.5\%, 10\%$ replaced pixels was used. Method: *baseline*, filter = {no filter, Gaussian, Median, Bilateral, Anisotropic}, $\sigma = 0.35$, optimized α in the range [0.5..40].

filter	σ_n	Gaussian noise			Poisson noise			Spike noise				
					P				S			
		<i>all</i>	<i>disc</i>	<i>untext</i>		<i>all</i>	<i>disc</i>	<i>untext</i>		<i>all</i>	<i>disc</i>	<i>untext</i>
No filter		0.256	0.719	0.255		0.323	0.761	0.361		0.427	0.835	0.598
Gaussian		0.256	0.719	0.248		0.319	0.761	0.348		0.408	0.83	0.554
Median	5	0.279	0.753	0.263	50	0.339	0.791	0.367	2.5%	0.206	0.7	0.169
Bilateral		0.26	0.73	0.25		0.321	0.762	0.351		0.453	0.849	0.631
Anisotropic		0.388	0.819	0.396		0.393	0.832	0.431		0.377	0.819	0.358
No filter		0.349	0.773	0.392		0.396	0.794	0.501		0.519	0.897	0.671
Gaussian		0.342	0.773	0.346		0.392	0.794	0.478		0.516	0.895	0.65
Median	10	0.359	0.79	0.378	150	0.412	0.816	0.5	5.0%	0.23	0.736	0.19
Bilateral		0.346	0.772	0.354		0.396	0.794	0.466		0.542	0.908	0.692
Anisotropic		0.407	0.817	0.43		0.45	0.834	0.538		0.463	0.867	0.603
No filter		0.489	0.865	0.599		0.468	0.841	0.576		0.666	0.977	0.73
Gaussian		0.471	0.857	0.563		0.448	0.835	0.553		0.666	0.976	0.715
Median	20	0.495	0.867	0.573	400	0.474	0.848	0.591	7.5%	0.263	0.769	0.214
Bilateral		0.483	0.863	0.579		0.45	0.835	0.552		0.692	0.988	0.746
Anisotropic		0.504	0.874	0.583		0.486	0.849	0.599		0.534	0.91	0.65
No filter		0.723	0.97	0.746		0.563	0.9	0.659		0.836	1.096	0.812
Gaussian		0.625	0.937	0.648		0.537	0.889	0.63		0.834	1.093	0.771
Median	40	0.638	0.929	0.644	800	0.554	0.904	0.622	10%	0.375	0.804	0.247
Bilateral		0.639	0.931	0.697		0.542	0.891	0.63		0.85	1.109	0.754
Anisotropic		0.631	0.928	0.647		0.556	0.882	0.659		0.604	0.955	0.73

5.4.2.2 Experiment: Non-uniform brightness correction

As the next step, we test the possibility to correct non-uniform brightness, which is a crucial image degradation process affecting computation of optical flow. For this purpose we compare performance of the grey value data term without preprocessing and the same data term, but with prior non-uniform brightness correction. For the evaluation we use the following brightness changes scenarios, described in Section 5.2.2.3:

- Additive brightness change: for this case a constant value of 30 is added to every pixel of the second image of an input sequence.

Table 5.13: Performance analysis of noise reduction techniques for complex noise scenarios (combination of the *Gaussian* and *Spike* noise models). For *Gaussian Noise* with zero mean and standard deviation values of $\sigma_n = 5, 10, 20, 40$ was added. For *Spike Noise* an amount $S = 2.5\%$ replaced pixels was used. Method: *baseline*, filter = {no filter, Gaussian, Median, Bilateral, Anisotropic}, $\sigma = 0.35$, optimized α in the range [0.5..40].

	noise	filter	error		
			<i>all</i>	<i>disc</i>	<i>untext</i>
$(\sigma=5) + S_1$	No filter	0.468	0.84	0.618	
	Gaussian	0.459	0.839	0.579	
	Median	0.299	0.774	0.274	
	Bilateral	0.475	0.845	0.623	
	Anisotropic	0.412	0.815	0.435	
$(\sigma=10) + S_1$	No filter	0.508	0.866	0.619	
	Gaussian	0.493	0.863	0.579	
	Median	0.384	0.801	0.43	
	Bilateral	0.51	0.868	0.62	
	Anisotropic	0.445	0.837	0.525	
$(\sigma=20) + S_1$	No filter	0.602	0.935	0.697	
	Gaussian	0.579	0.928	0.665	
	Median	0.544	0.902	0.63	
	Bilateral	0.6	0.934	0.694	
	Anisotropic	0.51	0.885	0.6	
$(\sigma=40) + S_1$	No filter	0.799	1.007	0.775	
	Gaussian	0.7	0.965	0.664	
	Median	0.687	0.953	0.656	
	Bilateral	0.741	0.972	0.733	
	Anisotropic	0.651	0.935	0.647	

- Multiplicative brightness change: every pixel of the second image frame is multiplied by a factor of 1.5.
- Pattern "*Spot*": this pattern simulates a spatial variation of beam intensity and is implemented as an addition of the initial image with a spatially varying brightness pattern (See Figure 5.9).
- Pattern "*Stripes*": this pattern aims to model brightness variation due to spatial fluctuations of a light source and changing beam intensity. Different strip patterns are added to the first and the second frame of the input sequence (See Figure 5.9).

For the preprocessing procedure we use the image leveling approach using the Gaussian filter (see Section 3.1.2.4) with an optimized size of the spatial mask.

Comparison between a model with and without preprocessing of initial image sequence is shown in Table 5.14.

From the evaluation of quantitative results we can make the following conclusion:

- *Brightness correction procedure significantly improves results of optical flow in the presence of brightness variations.* It is obvious, that already for the case of additive brightness change, the constancy assumption of image brightness is violated,

Table 5.14: Performance of non-uniform brightness correction on the *RubberWhale* dataset. Four brightness patterns are used for the evaluation: additive, multiplicative, pattern "*Spot*" and pattern "*Stripes*". Method: *baseline*, model = {no filter, Gaussian leveling}, optimized σ in the range [0.15..1.5], optimized α in the range [0.15..40].

brightness	model	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>
additive	grey	1.179	1.234	1.137
	filter + grey	0.128	0.581	0.089
multiplicative	grey	2.125	2.003	2.901
	filter + grey	0.162	0.613	0.103
<i>"Spot"</i>	grey	1.839	2.246	1.699
	filter + grey	0.128	0.581	0.088
<i>"Stripes"</i>	grey	1.062	1.318	0.816
	filter + grey	0.132	0.584	0.093

which results in a very poor performance of this approach. Addition of a simple preprocessing step allows to correct corrupted images and significantly improves the results.

- *Multiplicative brightness change is the most complicated brightness change scenario.* From the analysis of results we see that the multiplicative brightness change is the most challenging brightness variation scenario. It affects the results of optical flow computation more severely. This is seen from the results on both original and preprocessed images (see Table 5.14). The reason why "*Spot*" and "*Stripes*" patterns were easier to correct is because these patterns introduce a smooth, gradual spatial variation, which can be effectively corrected.

Quantitative evaluation of non-uniform brightness correction on the *Hydrangea* and the *New Marble* image sequences gives similar results. We continue the experiments on datasets with changing brightness in Section 5.4.3.2, where we examine data terms suitable for such imaging situations. Additionally, we include results achieved in this section to answer the question - which optical flow model is more useful to deal with changing brightness - using a simple data term (brightness constancy assumption) and a preprocessing step or using more advanced data terms specifically designed for the case of varying brightness conditions.

5.4.2.3 Experiment: Contrast enhancement

In the following section we examine the possibility to correct image contrast in order to enhance the visibility of image details and also to improve the accuracy of optical flow computation. To achieve that we use two contrast correction techniques, presented in Section 3.1.3:

- Histogram stretching
- Local contrast enhancement (CLAHE)

Table 5.15: Performance of local contrast enhancement methods on low-contrast data. Contrast level of $C = 100\%, 50\%, 25\%, 12.5\%$ is used to simulate low-contrast data. For every contrast level two datasets are checked: with no noise and with $\sigma=5$ Gaussian noise added to low-contrast images. Method: *baseline*, optimized σ in the range [0.15..2.0], optimized α in the range [0.1..40].

contrast	filter	no noise			with noise		
		error			error		
		<i>all</i>	<i>disc</i>	<i>untext</i>	<i>all</i>	<i>disc</i>	<i>untext</i>
100%	no filter	0.167	0.622	0.144	0.426	0.826	0.512
	stretch	0.168	0.625	0.146	0.414	0.832	0.523
	CLAHE	0.173	0.614	0.153	0.421	0.818	0.496
50%	no filter	0.185	0.64	0.153	0.528	0.878	0.647
	stretch	0.186	0.645	0.156	0.515	0.878	0.656
	CLAHE	0.188	0.636	0.17	0.507	0.867	0.593
25%	no filter	0.2	0.654	0.162	0.622	0.931	0.747
	stretch	0.201	0.657	0.164	0.577	0.917	0.653
	CLAHE	0.218	0.663	0.205	0.546	0.897	0.603
12.5%	no filter	0.211	0.664	0.176	0.673	0.962	0.722
	stretch	0.211	0.669	0.177	0.654	0.951	0.72
	CLAHE	0.235	0.677	0.228	0.613	0.926	0.708

To check the performance of optical flow on low-contrast data and data with low contrast-to-noise ratio (CNR) we use two datasets: low-contrast image data without noise and low-contrast images with added Gaussian noise with standard deviation of $\sigma = 5$. The quantitative comparison of contrast enhancement techniques is shown in Table 5.15. From the evaluation we can provided the following conclusions:

- *Contrast enhancement is not useful for noise-free images.* Evaluation results show that there is no additional gain in performance when the contrast enhancement is used on a low-contrast, noise-free images.
- *Contrast enhancement could be helpful for low-contrast images with noise.* Even when a small amount of noise is added to low-contrast images, a contrast-to-noise ratio (CNR) (see Section 3.1.3.1) decreases dramatically. This results in a severe degradation of image data. As the evaluation results show, contrast enhancement can potentially be beneficial on the expense of amplifying noise. However, with an appropriate use of robust optical flow model and denoising methods (Gaussian smoothing for the case of the *baseline* method used in this evaluation test), this could lead to the improved accuracy of optical flow computation.
- *Performance of contrast enhancement depends on the statistics of noise distribution.* It is important to note, that performance of local contrast enhancement in the presence of noise can be unstable, depending on statistical properties of noise distribution. This can limit application of contrast filtering techniques.
- *The contrast-to-noise ratio (CNR) is a crucial quality measure for analysis of X-ray data.* For challenging X-ray data, especially for low-exposure or poor photon

statistics conditions, a contrast-to-noise ratio is the most important measure to assess the image quality. This measure relates both image noise and image contrast.

5.4.3 Experiments: Artifacts

5.4.3.1 Experiment: Data terms for low-contrast data

In this section we check the performance of data terms on low-contrast data. One important aspect of this experiment is to evaluate higher-order data terms, since they might be more robust under degrading image contrast and, as a result, the reduced values of image gradients. We model low-contrast data according to the procedure described in Section 5.2.2.2. Numerical results are shown in Table 5.16.

Table 5.16: Performance evaluation of different data terms for low-contrast data. Contrast level of $C = 100\%, 50\%, 25\%, 12.5\%$ is used to simulate low-contrast data. Method: *baseline*, data term = {grey, gradient, Laplacian, gradient norm}, $\sigma = 0.15$, optimized α in the range [0.1..10].

contrast	data term	error		
		<i>all</i>	<i>disc</i>	<i>untext</i>
100%	grey	0.155	0.61	0.128
	grad	0.11	0.524	0.075
	lapl	0.137	0.561	0.083
	norm	0.135	0.6	0.095
50%	grey	0.165	0.612	0.147
	grad	0.121	0.535	0.094
	lapl	0.145	0.569	0.109
	norm	0.143	0.603	0.115
25%	grey	0.184	0.629	0.158
	grad	0.128	0.549	0.112
	lapl	0.161	0.589	0.132
	norm	0.147	0.611	0.134
12.5%	grey	0.219	0.654	0.18
	grad	0.154	0.573	0.144
	lapl	0.191	0.608	0.155
	norm	0.174	0.625	0.163

From the performance evaluation of different data terms we have the following results:

- *Higher-order data terms provide better results for low-contrast images.* As it could be expected, constancy assumptions based on higher-order derivatives provide better results in all image regions, since these data terms are more sensitive to vanishing image gradients.
- *Gradient value constancy assumption provides the best results for low-contrast data.* Data term based on the gradient constancy assumption outperforms all other data terms in all image regions. Thus, we conclude that it is more suited for low-contrast data.

- Lower values of the smoothness parameter are required for low-contrast data. From the optimization procedure for the current evaluation we observe, that the lower the image contrast is, the lower value of smoothness parameter should be chosen. This could be in a contradiction with optimization strategy used to process noisy data, where higher smoothness values allow to achieve better results. This fact should be carefully addressed and taken into account in the case of processing low-contrast or low-features image data with noise.

5.4.3.2 Experiment: Data terms for non-uniform brightness

In the experimental section 5.4.2.2 we evaluated how the preprocessing of initial data could help to overcome problems with non-uniform brightness. In this section we examine data terms and check their performance under different illumination conditions.

For the evaluation we use the same brightness changes scenarios, as were presented in Section 5.2.2.3 and employed in the experimental section 5.4.2.2. Results of quantitative analysis of different data terms on the *RubberWhale* dataset are shown in Table 5.17.

As it was already outlined in the experimental section 5.4.2.2, the most challenging illumination conditions occur for multiplicative brightness changes. Note, that the log-based data term outperformed other data terms for this case. To check the performance of gradient and log-based data terms we evaluate the performance of both approaches for different values of multiplicative factors. Results are shown on Figure 5.12.

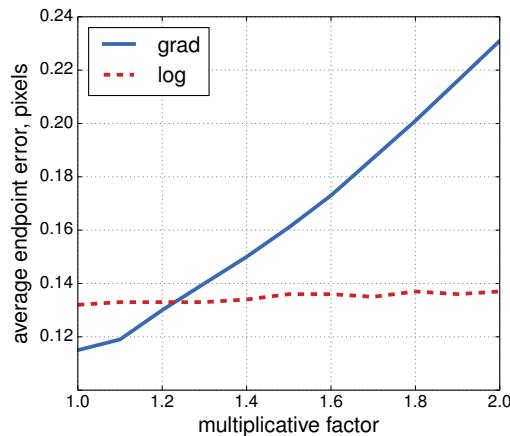


Figure 5.12: Comparison of gradient constancy assumption and log derivatives data terms for different values of multiplicative factor. Log-based data terms are robust under multiplicative brightness changes for all scaling factors.

From the evaluation of data terms and the experiment on the influence of multiplicative factor we can make the following conclusions:

- Gradient constancy assumption is the best data constraint for additive illumination changes. This performance of gradient-based constancy assumption is not surprising, since derivatives are invariant w.r.t. global additive grey value changes. However, such situation is not realistic for X-ray imaging scenarios. For this reason, a performance of data terms on additive brightness change has little value for practical use on X-ray data.

- *Log-derivatives based data term is robust with respect to multiplicative brightness changes.* Transformation of image data into logarithmic scale helps to overcome problems with multiplicative brightness changes. Log-derivatives based data term shows significantly better results in comparison with other data terms.
- *Log-derivatives based data term outperforms gradient data term only for large values of multiplicative factor.* The performance comparison of gradient-based and log-derivatives based constancy assumptions on various multiplicative factors (see Figure 5.12) reveals, that the log-based constancy assumption starts to outperform the gradient constancy assumption for multiplicative factors greater than 1.2. It is important to note, that the brightness changes equivalent to multiplication by a factor greater than 1.5 are unlikely during image acquisition for a stable and well-adjusted imaging setup. However, if it is the case, then the log-based image derivatives data term is a suitable constancy assumption.
- *Data preprocessing of initial image data significantly improves robustness against artifacts related to changing brightness conditions.* In the previous Section 5.4.2.2 we evaluated and discussed the usefulness of a non-uniform brightness correction. In this section we compare the preprocessing approach with a model, which employs advanced data terms designed to provide robustness against such kind of conditions. From the quantitative results presented in Table 5.17 one can observe, that preprocessing approach provides a good general performance. It gives similar results with the gradient constancy assumption on the multiplicative change dataset and outperforms all data terms for patterns "Spot" and "Strips". We want to emphasize that such kind of brightness change conditions are more physically relevant and more frequently occurring for the real life X-ray imaging experiments, then a global additive or strong multiplicative brightness changes.

5.4.3.3 Experiment: Robust data term for low-contrast data

In this experiment we check the influence of robustification of data terms for low-contrast data. Since low-contrast data is characterized by small values of image derivatives, this might affect the computation of a motion tensor. For such situation the quadratic penalization of a data constancy assumption might be beneficial, since it provides more sensitivity to deviations from constancy assumptions. In this experiment we test this hypothesis. For the evaluation we use both the grey value and the gradient value constancy assumptions. The quantitative comparison between the original (quadratic) and the robust penalizer functions of the data term on low-contrast data is shown in Table 5.18.

Results of quantitative comparison of both data terms are:

- *For low-contrast image data the robust data term performs worse than the original (quadratic) data term.* From the performance analysis of robust data terms (both the grey value and the gradient value constancy assumptions), one may observe that the robust setting is not useful for all contrast levels. Starting from a contrast level of $C = 25\%$ the grey value constancy assumption provides worse results and starting from a level 12.5% gradient constancy assumption in a robust setting no longer gives better results.

Table 5.17: Performance of non-uniform brightness correction on *RubberWhale* dataset. Four brightness patterns are used for the evaluation: additive (+30), multiplicative ($\times 1.5$), pattern "Spot", pattern "Stripes". Method: *baseline*, data term = {grey, grey + Gaussian leveling, gradient, log-based derivatives}, optimized σ in the range [0.15..1.5], optimized α in the range [0.005..150].

	brightness	data term	error		
			<i>all</i>	<i>disc</i>	<i>untext</i>
additive	grey	1.143	1.17	1.056	
	filter + grey	0.128	0.581	0.089	
	grad	0.117	0.543	0.084	
	log	0.172	0.643	0.152	
multiplicative	grey	1.212	1.204	1.12	
	filter + grey	0.162	0.613	0.103	
	grad	0.161	0.685	0.09	
	log	0.136	0.554	0.105	
"Spot"	grey	1.238	1.219	1.113	
	filter + grey	0.128	0.581	0.088	
	grad	0.129	0.562	0.099	
	log	0.221	0.723	0.224	
"Stripes"	grey	0.979	1.144	0.807	
	filter + grey	0.132	0.584	0.093	
	grad	0.153	0.594	0.118	
	log	0.242	0.748	0.197	

- *Gradient constancy assumption outperforms the grey value constancy assumption in a robust setting for low-contrast data.* As it was already discussed in the previous experiment, the gradient constancy assumption is more suited for low-contrast data, since it is more sensitive to vanishing image gradients (as a result of contrast degradation). The same holds for a model using the gradient based data term in a robust setting. However, when the contrast drops down to 12.5% both data terms in a robust setting provide worse result in comparison to a simple data term. Therefore, we conclude that for low-contrast image data the original (quadratic) data term should be employed or, alternatively, one may try to enhance the contrast using techniques presented in Section 3.1.3.2 and examined in the experimental Section 5.4.2.3.

5.4.3.4 Experiment: Robust data term for images with artifacts

In this section we test the performance of robust data terms on image data with artifacts. In the literature, the use of robust data term is well-studied for the original data containing natural artifacts, such as occlusions [BA96, SRB10, BSL⁺11, ZBW11]. An important addition of our study is that we introduce another structures, which aim to simulate artifacts frequently occurring for the real X-ray data. We control the placement of such artifacts and regulate their effect by varying their thickness. Moreover, we check the performance of optical flow computation in the regions around the artifact structures (denoted as *outliers*).

Results of quantitative evaluation of the robust data term on different amounts of

Table 5.18: Comparison of original (quadratic) and robust data term for low-contrast data. Contrast level of $C = 100\%, 50\%, 25\%, 12.5\%$ is used to simulate low-contrast data. Method: *baseline*, model = {original, robust}, $\sigma = 0.15$, optimized α in the range [0.1..10].

			contrast	data	model	error		
						<i>all</i>	<i>disc</i>	<i>untext</i>
100%	grey	original	0.169	0.628	0.131			
		robust	0.155	0.61	0.128			
	grad	original	0.139	0.565	0.089			
		robust	0.11	0.524	0.075			
50%	grey	original	0.169	0.629	0.128			
		robust	0.165	0.612	0.147			
	grad	original	0.141	0.567	0.091			
		robust	0.121	0.535	0.094			
25%	grey	original	0.169	0.63	0.125			
		robust	0.184	0.629	0.158			
	grad	original	0.14	0.57	0.096			
		robust	0.128	0.549	0.112			
12.5%	grey	original	0.171	0.634	0.138			
		robust	0.219	0.654	0.18			
	grad	original	0.153	0.585	0.131			
		robust	0.154	0.573	0.144			

artifacts on the example of a grey value constancy assumption are shown in Table 5.19.

From the performance evaluation we conclude that:

- *Robust data term significantly improves optical flow results in the presence of image artifacts.* As it was expected from the result of previous work, robust data terms significantly improve the accuracy of optical flow. The performance improvement is especially remarkable in the *outliers* regions. With the increased amount of artifacts the importance of robust data term becomes even more prominent.

5.4.3.5 Experiment: Flow median filtering

We proceed with the experiment to test the performance of a median filtering of the intermediate flow result (see Section 2.2.6). In the work of [SRB10, SRB14] the authors already shown that the intermediate flow filtering allows to significantly improve the accuracy of optical flow. For this reason we skip such evaluation and proceed with a more challenging datasets. We assess the median filtering procedure in the presence of more severe artifacts. For this purpose we use the same approach to distribute data artifacts as in the experimental section 5.4.3.4. For each artifacts level we evaluate the performance of different mask sizes of the median filter. Moreover, we present evaluation results of the median filtering procedure on two image sequences - *RubberWhale* and *Hydrangea* datasets, since these datasets represent scenes with different sizes of moving objects. For the evaluation we use a *baseline* model (see Section 5.3.3), which includes the usage of a robust data term.

Qualitative results using color coding are shown in Figure 5.13. One can see the improved robustness of optical flow computation, especially in the regions around image

Table 5.19: Comparison of original (quadratic) and robust data term using grey value constancy assumption on images with data artifacts. Artifact structures with thickness of $A = 0$ (no artifacts), 1, 2, 3 pixels are added to original images to simulate data outliers. Method: *baseline*, data term = {original, robust}, optimized σ in the range [0.15..2.0], optimized α in the range [1.0..40].

artifacts	model	error			
		<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>
A_0	original	0.166	0.632	0.127	0.156
	robust	0.157	0.617	0.129	0.152
A_1	original	0.294	0.803	0.302	1.04
	robust	0.173	0.645	0.157	0.217
A_2	original	0.495	0.924	0.586	2.527
	robust	0.221	0.708	0.235	0.3
A_3	original	0.694	1.142	0.797	3.915
	robust	0.263	0.734	0.29	0.328

artifacts (regions *outliers* in experimental Table 5.20)

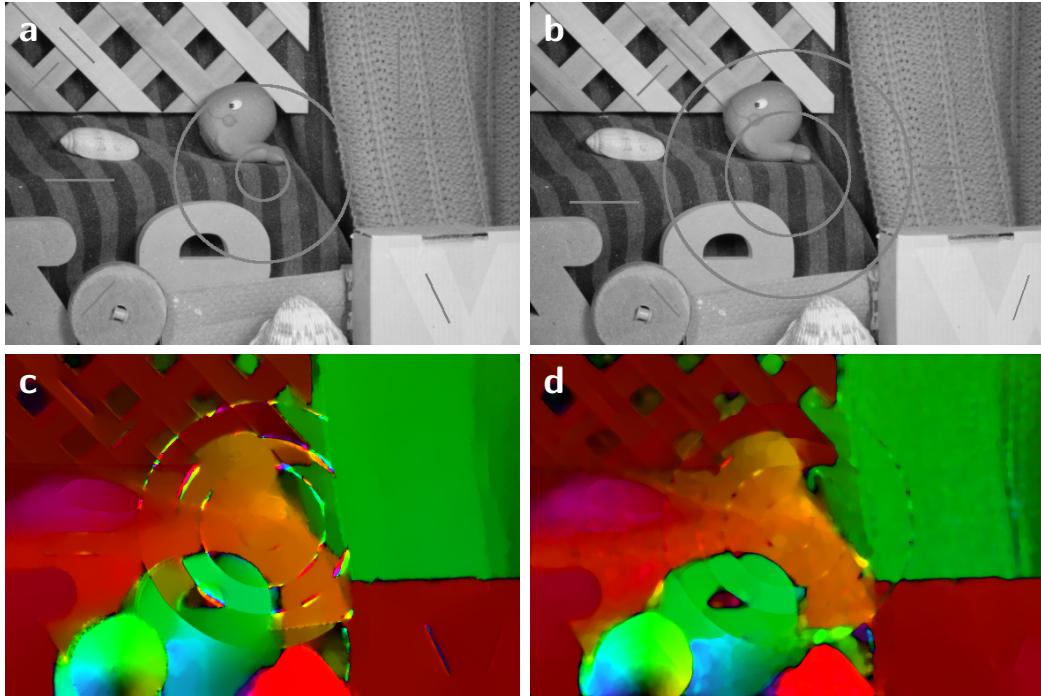


Figure 5.13: Performance evaluation of flow median filtering on *RubberWhale* dataset with image artifacts of thickness A_3 . (a) First frame of degraded image sequence. (b) Second frame of the sequence. (c) *Baseline* model *without* median filtering. (d) Optical flow model *with* median filtering with a spatial mask size $p = 9$. Flow errors due to image artifacts are significantly reduced.

Quantitative results are presented in Table 5.20. Evaluating the obtained results we can make the following conclusions:

Table 5.20: Performance evaluation of the flow intermediate filtering on images with data outliers. For the evaluation two image sequences are used: the *RubberWhale* and the *Hydrangea* dataset, to check the performance of a median filtering for different sizes of moving objects. Artifact structures with thickness of $A = 0$ (no artifacts), 1, 2, 3 pixels are added to the original images to simulate data outliers. Method: *baseline*, size of a median filter mask $p = \{nofilter, 3, 5, 7, 9\}$, optimized σ in the range $[0.15..2.0]$, optimized α in the range $[1.0..40]$.

artifacts	filter	RubberWhale dataset				Hydrangea dataset			
		error				error			
		<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>	<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>
A_0	-	0.157	0.617	0.129	0.152	0.203	0.518	0.102	0.251
	$p = 3$	0.152	0.594	0.126	0.148	0.2	0.51	0.101	0.247
	$p = 5$	0.147	0.576	0.124	0.144	0.198	0.504	0.1	0.243
	$p = 7$	0.144	0.56	0.122	0.141	0.197	0.505	0.1	0.242
	$p = 9$	0.142	0.556	0.121	0.14	0.198	0.511	0.1	0.243
A_1	-	0.173	0.645	0.157	0.217	0.209	0.529	0.105	0.278
	$p = 3$	0.163	0.61	0.143	0.199	0.206	0.522	0.105	0.275
	$p = 5$	0.155	0.588	0.137	0.186	0.203	0.517	0.104	0.27
	$p = 7$	0.151	0.575	0.133	0.172	0.202	0.516	0.103	0.266
	$p = 9$	0.148	0.567	0.13	0.161	0.203	0.521	0.103	0.264
A_2	-	0.221	0.708	0.235	0.3	0.219	0.539	0.114	0.312
	$p = 3$	0.205	0.66	0.217	0.291	0.215	0.535	0.113	0.309
	$p = 5$	0.172	0.607	0.151	0.261	0.212	0.531	0.111	0.301
	$p = 7$	0.157	0.579	0.137	0.194	0.209	0.529	0.109	0.293
	$p = 9$	0.151	0.57	0.132	0.172	0.209	0.531	0.108	0.288
A_3	-	0.263	0.734	0.29	0.328	0.23	0.548	0.122	0.342
	$p = 3$	0.243	0.721	0.268	0.323	0.226	0.545	0.121	0.338
	$p = 5$	0.201	0.649	0.177	0.314	0.22	0.542	0.119	0.33
	$p = 7$	0.17	0.601	0.147	0.277	0.217	0.54	0.116	0.319
	$p = 9$	0.158	0.582	0.139	0.211	0.215	0.54	0.114	0.309

- *Flow median filtering significantly improves optical flow results.* As it was discussed earlier, in the case when artifacts are present in image data, errors in the flow estimation which originate already on coarse image scales are propagated via a warping step to the next computation level. Then, the incorrect flow serves as an initialization for optical flow estimation. As a consequence, errors are accumulated during the incremental computation. The median filtering is an effective correction procedure to suppress flow errors already on earlier computation levels.
- *Large sizes of a median mask are useful to filter data outliers and improve overall accuracy of optical flow.* From the evaluation results on the *RubberWhale* dataset we can conclude that large sizes of a median mask are useful to suppress data artifacts. Surprisingly, it is also true for the original data with no additional artifact structures. In this case, the best results were given by a robust model with the median filtering with a spatial mask $p = 9$. With the increased amount of artifacts the importance of median filtering becomes even more prominent. As a conclusion - the stronger are the artifacts, the larger size of median mask is required.

- Optimum size of a median mask depends on the size of moving objects and the amount of flow discontinuities. To compare the performance of a median filtering with different sizes of spatial mask we also evaluated the *Hydrangea* dataset. This dataset is characterized by small sizes of moving objects (plant leaves) and large amount of flow discontinuities (See Section 5.2.1). As a result, for smaller amount of artifacts, the smaller size of a median mask is optimal (See Table 5.20). However, for the artifact level A_3 a median filter with a large mask size of $p = 9$ provides the best results.

The evaluation of the flow median filtering procedure on the *New Marble* dataset provides similar performance results (not shown here).

5.4.3.6 Experiment: Data Refinement

In this experiment we evaluate the procedure of data refinement (see Section 2.1.3.5) as an additional correction step. It intends to improve the robustness of optical flow around strong data outliers. For the evaluation we use a *baseline* model, which already includes a robust data term, but excludes other robust settings (e.g. the median filtering and the CLG approach).

Analyzing qualitative results of optical flow shown in Figure 5.14 we can notice, that flow errors due to the strong image artifacts are almost completely eliminated. The visual quality is more appealing than the corresponding results using the median filtering approach (see Section 5.4.3.5 and Figure 5.13). Quantitative results are given in Table 5.21.

Table 5.21: Performance evaluation of the data refinement procedure on images with data outliers. Artifact structures with thickness of $A = 0$ (no artifacts), 1, 2, 3 pixels are added to the original images to simulate data outliers. Method: *baseline*, model = {no refinement, with refinement}, optimized σ in the range [0.15..2.0], optimized α in the range [1.0..40].

noise	model	error			
		<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>
A_0	no ref	0.157	0.617	0.129	0.152
	refine	0.156	0.591	0.131	0.156
A_1	no ref	0.173	0.645	0.157	0.217
	refine	0.163	0.602	0.143	0.186
A_2	no ref	0.221	0.708	0.235	0.3
	refine	0.163	0.603	0.144	0.183
A_3	no ref	0.263	0.734	0.29	0.328
	refine	0.168	0.609	0.15	0.197

Results of our performance evaluation are:

- *Data refinement significantly improves the accuracy of optical flow computation in the presence of image artifacts.* Quantitative evaluation of optical flow model with data refinement reveals, that it is useful to weight down the contribution of a data term in the regions of strong image artifacts. As a result, accuracy of optical flow computation significantly improves.

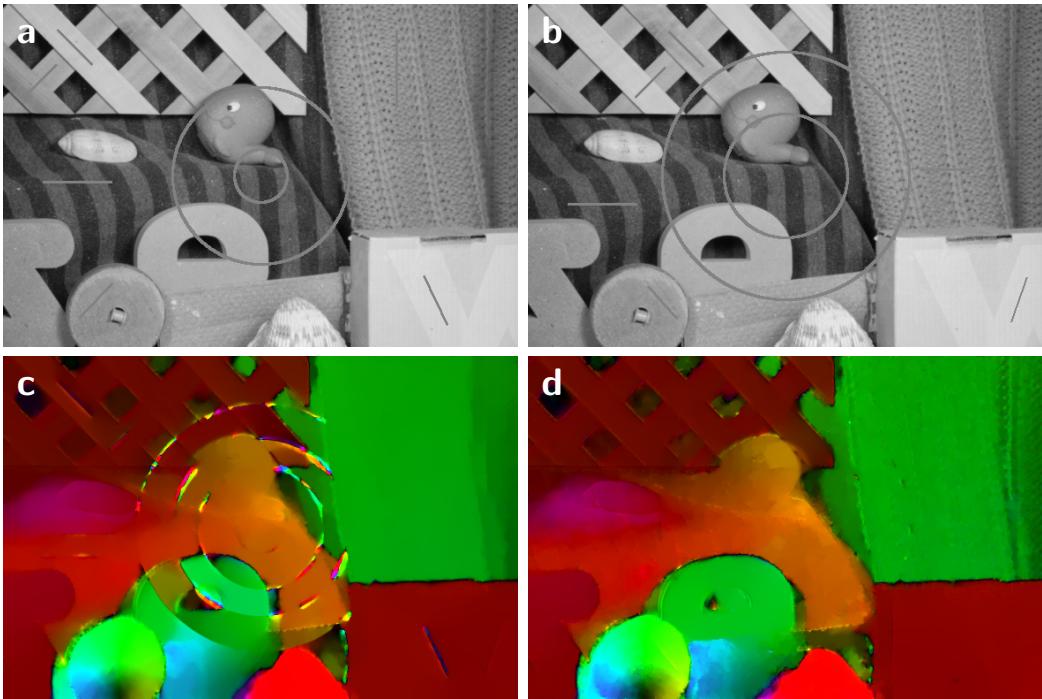


Figure 5.14: Performance evaluation of a data refinement procedure on the *RubberWhale* dataset with image artifacts $A_3 = 3$ pixels. (a) First frame of degraded image sequence. (b) Second frame of the sequence. (c) *Baseline* model *without* data refinement. (d) Optical flow model *with* data refinement. Flow errors due to image artifacts are almost completely eliminated.

- *Data refinement outperforms the median filtering approach for smaller sizes of a median mask ($p < 9$)*. From both qualitative (see Figure 5.14) and quantitative (see Table 5.21) we can observe, that the data refinement approach outperforms the robust model based on the flow median filtering (see Figure 5.13 and Table 5.20), except for the case when a larger mask of median filter is used ($p = 9$).
- *Data refinement is especially useful within or around regions with strong data outliers*. When compared with median filtering approach using a large spatial mask size ($p = 9$) data refinement approach provided significantly better results in the *outliers* regions, however was less accurate in other statistics regions (*all*, *disc*, *untext*). A straightforward improvement would be a combination of both approaches into a single optical flow method. We evaluate this possibility in our final experiment (see Section 5.5).

The evaluation of data refinement approach on *Hydrangea* and *New Marble* datasets gives similar performance results (not shown here).

5.4.4 Experiments: Confidence Measures

In this section we switch from the evaluation of different optical flow models to an important topic of automated confidence measures. This quantitative information could be obtained from the input image data and the computed flow field. As it was outlined

earlier in our work, this topic is not sufficiently covered in the literature on optical flow. However, for analysis of real life data, especially in the case when it is diverse in its quality and challenging to analyze, the topic of confidence measures is particularly important. With the appropriate use of such measures it is possible to evaluate the quality of the computed result and automatically optimize the model parameters. In the recent works of [SRB10, ZBW11] the authors demonstrated the benefits of such optimization.

In this section we evaluate a number of confidence measures presented in section 2.1.5. We do not present the evaluation of such measures as the gradient-based approach and the motion uniqueness criteria in our final results, since these measures did not show an adequate performance. In the current evaluation we test the following confidence measures:

- Data Constancy Error (see Section 2.1.5.3)
- Forward-Backward Check (see Section 2.1.5.5)
- Optimal Prediction Principle (see Section 2.1.5.6)

Additionally, we test the performance of confidence measures on different datasets (with varying amount of noise) and optical flow models. We use the *RubberWhale* dataset and the following list of computation models:

- Simple method C_1 : Horn and Schunck method + multi-level computation
- Advanced method C_2 : Model C_1 + flow-driven smoothness
- Baseline method C_3 : C_2 + robust data term (see Section 5.3.3)
- Robust method C_4 : C_3 + median filtering + combined local-global approach
- Noisy images $\sigma = 10$: we added *Gaussian* noise with zero mean and standard deviation values of $\sigma_n = 10$ to the original *RubberWhale* dataset. Method: robust method C_4
- Noisy images $\sigma = 20$: the same as the previous model, but with an increased level of noise ($\sigma = 20$). Method: robust method C_4

For the evaluation we vary the value of the smoothness parameter α and compare distribution values of each confidence metric with the distribution of the average endpoint error provided by the ground truth result (AEE). Quantitative evaluation of confidence measures on a selected number of experimental models is presented in Figure 5.15.

A summary of comparison between automatically selected smoothness parameter α using a confidence measure and the optimal parameter given by the ground truth is presented in Table 5.22.

Results of performance evaluation are:

- *Confidence measure based on the Optimal Prediction Principle is more accurate to predict quality of optical flow computation.* Based on the evaluation of all optical flow models (see Table 5.22) we conclude that OPP measure gives the best results to describe the quality of the computed flow field in the case when ground truth result is not known.

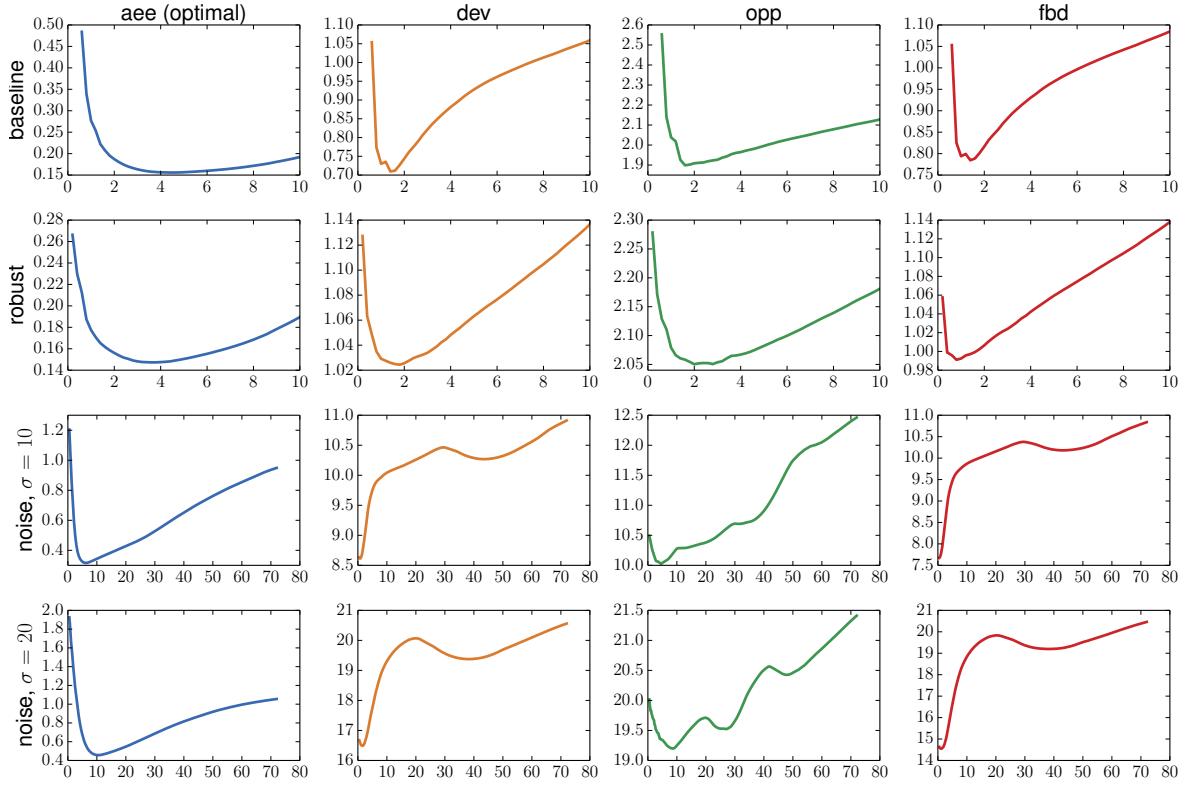


Figure 5.15: Comparison of confidence measures for different optical flow models on the *RubberWhale* dataset. The value of the corresponding confidence metric is plotted over a list of varying smoothness parameters α . **First Row:** Baseline method C_3 . **Second Row:** Robust method C_4 . **Third Row:** Robust method C_4 on noisy images ($\sigma = 10$). **Forth Row:** Robust method C_4 on noisy images ($\sigma = 20$). **First Column:** Average endpoint error given by the comparison of the computed flow field with the ground truth result (*aee*). **Second Column:** Data Constancy Error measure (*dev*). **Third Column:** Confidence measure based on the Optimal Prediction Principle (*opp*). **Forth Column:** Forward-Backward consistency check (*fbd*).

- *Performance of confidence measures depends on the optical flow model and input data.* Such measures as the Data Constancy Error (*dev*) and Forward-Backward Check (*fbd*) failed to provide optimal results. With an increase of noise this effect magnifies. Note, that we do not optimized other parameters, such as σ , to show the influence of an individual parameter. In general, all confidence measures performed less stable on the datasets with noise, which could potentially limit their application.
- *If a confidence measure based on the Optimal Prediction Principle cannot be applied due to violation of its assumptions the next optimal confidence measure is based on the Data Constancy Error.* This measure performed less accurate in comparison with OOP metric (see Table 5.22). However, this measure has a number of advantages: there are no assumptions about an image sequence and a motion model, except the ones incorporated in the data constancy assumption; it is general and could be applied for any optical flow model.

Table 5.22: Comparison of confidence measures for optical flow models $C_1 - C_4$ on the original *RubberWhale* dataset and datasets with noise. *Gaussian* noise with zero mean value and standard deviation values of $\sigma_n = 10, 20$ was added. The optimal value of a smoothness parameter α selected using the corresponding confidence measure approach is shown. Methods: Simple method C_1 (Horn and Schunck method + multi-level computation); Advanced method C_2 (Model C_1 + flow-driven smoothness); Baseline method C_3 (C_2 + robust data term); Robust method C_4 (C_3 + median filtering + combined local-global approach). No optimization of model parameters σ, ρ was performed.

Method	<i>optim</i>	<i>dev</i>	<i>opp</i>	<i>fbd</i>
Method C_1	55	15	24	15
Method C_2	20	15	22	15
Method C_3	4.4	1.4	1.6	1.4
Method C_4	3.6	1.8	2	0.8
Method C_4 + noise ($\sigma = 10$)	6.1	0.7	4.5	0.5
Method C_4 + noise ($\sigma = 20$)	10.5	1.7	8.7	1.1

5.5 Models Comparison

So far we have experimented on individual components of optical flow models. In this section we put all the successful model into a single technique. We examine step by step how additional optical flow features improve the accuracy of results. We do not provide the performance analysis on the original data, because it does not comply with the purpose of this work, which is focused on a challenging X-ray data. Instead, we use a dataset with a simulated Poisson noise ($P = 200$) and substantial amount of image artifacts. Starting from the simplest approach - the Horn and Schunck method we gradually include other optical flow features and end up with the most advanced and robust model.

For a shorter model notation we use the name of the previously introduced model and give names of additional features. The final model combines all the robust features. We evaluate the following optical flow models:

- Model M_1 : Classical Horn and Schunck model (see Section 1.4.2)
- Model $M_2 = M_1 +$ flow-driven smoothness term (see Section 2.1.4.3)
- Model $M_3 = M_2 +$ multi-level computation (see Section 2.2.5)
- Model $M_4 = M_3 +$ robust data term (see Section 2.1.3.1)
- Model $M_5 = M_4 +$ combined local-global approach (CLG) (see Section 2.1.3.3)
- Model $M_6 = M_5 +$ median filtering of an intermediate flow result
- Model $M_7 = M_6 +$ data refinement step (see Section 2.1.3.5) using the forward-backward check method as a confidence map (see Section 2.1.5.5)

Analyzing the results which are given in Figure 5.16 and the corresponding quantitative performance metrics from Table 5.23 we can make the following observations:

Table 5.23: Comparison of optical flow models $M_1 - M_7$ on the *RubberWhale* dataset with added noise and image artifacts. *Poisson noise*: Mean value of Poisson random distribution $P = 200$. Artifact structures with thickness of $A = 2$ pixels are added to original images to simulate data outliers. In addition to a common average endpoint error (AEE) we provide a robust statistics metric R1.0 (see Section 2.1.1.3), which helps to judge about qualitative performance. Method: according to model M_i , optimized ρ in the range [0.5..2.0], optimized σ in the range [0.35..2.5], optimized α in the range [3.0..2000].

model	AEE error				R1.0 statistics			
	<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>	<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>
$M_1 = \text{HS}$	0.599	0.963	0.72	0.704	17.9	36.7	24.8	25.1
$M_2 = M_1 + \text{flow-driven}$	0.784	1.011	1.014	1.125	26.6	35	41.2	41.2
$M_3 = M_2 + \text{warping}$	0.51	0.904	0.637	0.745	13.1	31.8	13.5	18
$M_4 = M_3 + \text{robust data term}$	0.43	0.834	0.522	0.462	9.8	29.6	9.4	9.9
$M_5 = M_4 + \text{CLG}$	0.378	0.829	0.406	0.416	9.4	29.4	9	10.2
$M_6 = M_5 + \text{median filtering}$	0.376	0.82	0.395	0.409	9.1	28.3	8.9	9.7
$M_7 = M_6 + \text{refinement}$	0.353	0.821	0.351	0.386	8.8	28.1	8.6	9.5

- As it was expected, the simplest optical flow model gives the worst accuracy. The flow magnitudes were not captured correctly, since no strategy to cope with large displacements were employed. The optimal model parameters are the large pre-smoothing parameter σ and the large smoothness regularization parameter α . By oversmoothning the initial data this helped to reduce the influence of data outliers (see Figure 5.16d).
- An interesting aspect of our evaluation is that the flow-driven smoothness approach performed significantly worse then a simple method using the homogeneous regularization. Adaptive smoothness regularization was not sufficient to cope with data outliers (see Figure 5.16e) and provide filling-in effect in homogeneous image regions (see Figure 5.16e).
- With the incorporation of a multi-level computation technique the flow in homogeneous regions is estimated correctly (see Figure 5.16f). Moreover, the estimation of the magnitude of the flow field improves.
- The use of the combined local-global approach significantly improves the robustness of the optical flow method under noise and data artifacts (see Figure 5.16g).
- Addition of the median filtering of intermediate flow results between levels of a coarse-to-fine computation strategy and subsequent selective refinement of the data term weight according to the automated confidence measure further improves robustness against artifacts (see Figure 5.16h).

To summarize, for all testing datasets with artifacts (Poisson noise $P = 200$ and artifact structures A_2) we compare the performance of a *baseline* method M_3 (see Section 5.3.3) with a robust optical flow method M_7 . The quantitative comparison is presented in Table 5.24 and qualitative visualization of computed flow fields is given in Figure 5.17. For all dataset the robust model, designed for challenging X-ray data provides significantly better results as measured using the average endpoint error metric for the entire image.

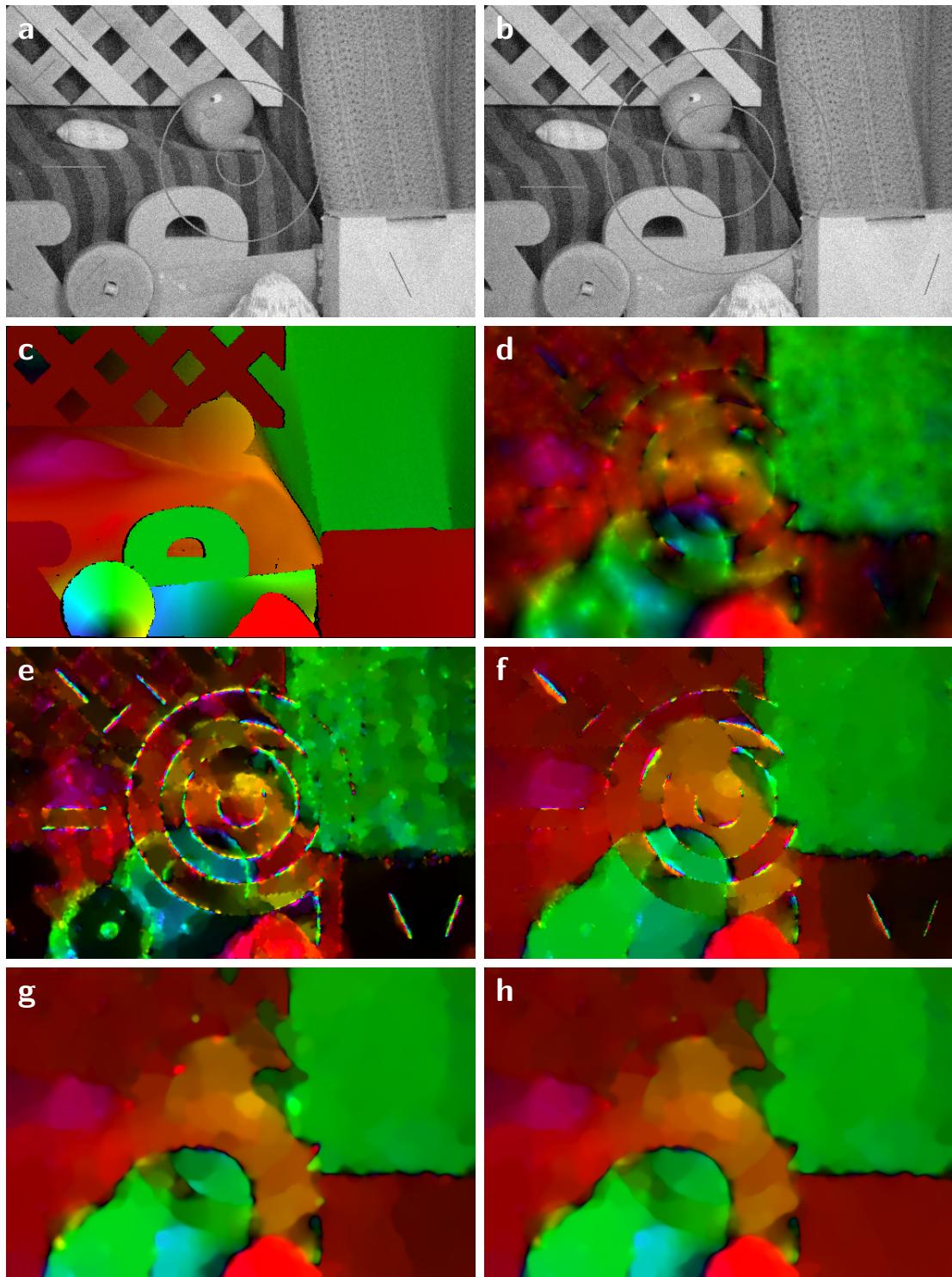


Figure 5.16: Comparison of optical flow models $M_1 - M_7$ on the *RubberWhale* dataset with added noise and image artifacts. (a) First frame of degraded image sequence. (b) Second frame of the sequence. (c) Ground truth result. (d) Model M_1 : Classical Horn and Schunck. (e) Model $M_2 = M_1 +$ flow-driven smoothness term. (f) Model $M_3 = M_2 +$ multi-level computation. (g) Model $M_5 = M_4 +$ robust data term + combined local-global approach. (h) Model $M_7 = M_5 +$ median filtering + data refinement step.

Table 5.24: Comparison of optical flow models M_3 and M_7 on the *RubberWhale*, *Hydrangea* and *New Marble* datasets with added noise and image artifacts. *Poisson noise*: Mean value of Poisson random distribution $P = 200$. Artifact structures with thickness of $A = 2$ pixels are added to the original images to simulate data outliers. Average endpoint error and Robust statistics are shown. Methods: according to models M_3 and M_7 , optimized ρ in the range [0.5..2.0], optimized σ in the range [0.35..2.5], optimized α in the range [3.0..40].

		AEE error				R1.0 statistics			
		<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>	<i>all</i>	<i>disc</i>	<i>untext</i>	<i>outliers</i>
<i>RubberWhale</i>	baseline	0.43	0.834	0.522	0.462	9.83	29.59	9.44	9.93
	robust	0.353	0.845	0.351	0.39	9.1	30.26	8.8	10.2
<i>Hydrangea</i>	baseline	0.728	0.63	0.737	0.918	18.32	16.45	17.05	27.49
	robust	0.343	0.632	0.279	0.413	5.84	15.98	4.19	8.02
<i>New Marble</i>	baseline	0.077	0.492	0.054	0.096	0.29	10.87	0	0.55
	robust	0.067	0.523	0.051	0.086	0.41	14.81	0	0.74

5.6 Evaluation Summary

In this section we provide a summary list of our quantitative experiments with synthetic data presented in the current chapter. We enlist all the conclusions as it is stated in the corresponding experimental section. This list of justified principles should serve us as a general reference for the choice of appropriate optical flow models and their parameters.

Experiments: Noise

- Gradient constancy assumption with the robust setting provides good results for both Gaussian and Poisson noise models.
- Higher order derivatives are sensitive to noise for the Gaussian, Poisson and Spike noise models.
- Grey value constancy assumption is more robust for the Spike noise model.
- Grey value constancy assumption is more robust for the Poisson noise model in the regions of image discontinuity.
- Integration of local information is useful to improve results on noisy data.
- Data normalization improves results on noisy data.
- Data normalization approach outperforms the Combined Local-Global approach (CLG) on flow boundaries.
- Data normalization can be further combined with the CLG approach to improve results.

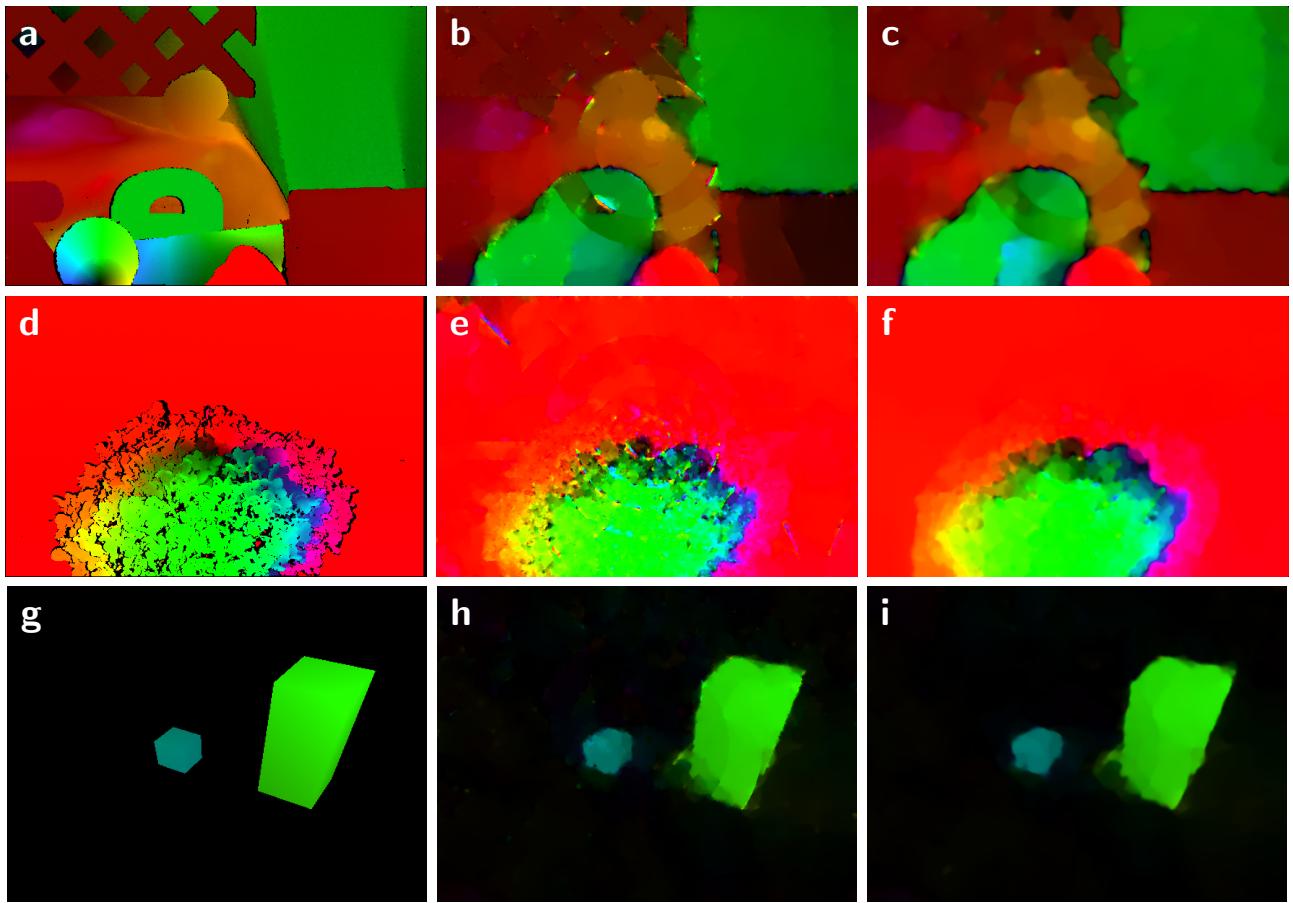


Figure 5.17: Comparison of advanced optical flow model M_4 and robust optical flow model M_7 on *RubberWhale* dataset with added noise and image artifacts. (a) Ground truth flow result for the *RubberWhale* dataset. (b) Flow field computed using M_4 optical flow model. (c) Flow field computed using the robust M_7 optical flow model. (d) Ground truth flow result for the *Hydrangea* dataset. (e) Flow field computed using M_4 optical flow model. (f) Flow field computed using the robust M_7 optical flow model. (g) Ground truth flow result for the *New Marble* dataset. (h) Flow field computed using M_4 optical flow model. (i) Flow field computed using the robust M_7 optical flow model.

Experiments: Preprocessing

- Noise filtering of initial data can be helpful.
- For each noise model there is an optimal filtering procedure.
- Gaussian filtering is a simple and effective filtering procedure.
- Gaussian filtering is useful to remove Poisson noise.
- Usefulness of Anisotropic diffusion filtering for complex noise scenarios.
- The appropriate choice of a filtering procedure and its parameters is crucial, however, a challenging task.

Experiments: Artifacts

- Brightness correction procedure significantly improves results of optical flow in the presence of brightness variations
- Multiplicative brightness change is the most complicated brightness change scenario.
- Contrast enhancement is not useful for noise-free images.
- Contrast enhancement could be helpful for low-contrast images with noise.
- Performance of contrast enhancement depends on the statistics of noise distribution.
- The contrast-to-noise ratio (CNR) is a crucial quality measure for analysis of X-ray data.
- Higher-order data terms provide better results for low-contrast images.
- Gradient value constancy assumption provides the best results for low-contrast data.
- Lower values of the smoothness parameter are required for low-contrast data.
- Gradient constancy assumption is the best data constraint for additive illumination changes.
- Log-derivatives based data term is robust with respect to multiplicative brightness changes.
- Log-derivatives based data term outperforms gradient data term only for large values of multiplicative factor.
- Data preprocessing of initial image data significantly improves robustness against artifacts related to changing brightness conditions.
- For low-contrast image data the robust data term performs worse than the original (quadratic) data term.
- Gradient constancy assumption outperforms the grey value constancy assumption in a robust setting for low-contrast data.
- Robust data term significantly improves optical flow results in the presence of image artifacts.
- Flow median filtering significantly improves optical flow results.
- Large sizes of a median mask are useful to filter data outliers and improve overall accuracy of optical flow.
- Optimum size of a median mask depends on the size of moving objects and the amount of flow discontinuities.
- Data refinement significantly improves the accuracy of optical flow computation in the presence of image artifacts.

- Data refinement outperforms the median filtering approach for smaller sizes of a median mask ($p < 9$).
- Data refinement is especially useful within or around regions with strong data outliers.

Experiments: Confidence Measures

- Confidence measure based on the Optimal Prediction Principle is more accurate to predict quality of optical flow computation.
- Performance of confidence measures depends on the optical flow model and input data.
- If a confidence measure based on the Optimal Prediction Principle cannot be applied due to violation of its assumptions (linear motion, more than two frames available) the next optimal confidence measure is based on Data Constancy Error.

Chapter 6

Applications in X-ray Imaging

In this chapter we show the application of the developed optical flow and data analysis methods on a number of scientific problems from different research fields. For each application we start by giving a short introduction to the subject of investigation. We describe important aspects of the X-ray imaging setup and sample preparation. Then, we give a short description of a problem and specify our tasks for data processing. We evaluate each dataset according to our data taxonomy presented in Section 5.1. Then we provide a detailed discussion on the usage of appropriate data preprocessing routines, derive a suitable optical flow model, optimize model parameters and perform data analysis, visualization and interpretation of the results.

6.1 Motion Analysis

6.1.1 Particles Fluid Dynamics in Semi-Solid Metals

6.1.1.1 Introduction

Semi-solid metals are two-phase mixtures, which exhibit unusual rheological properties [Fle91]. Without stress they behave like elastic solids, but when a shear force is applied a transition into a viscous mixture takes place.

The transition from a solid phase filled by liquid to a viscous state is accompanied by the breakup of the solid skeleton into individual particles and clusters which can flow independently. This breakup process is known as a *thixotropic effect* [SMF72]. In comparison to high pressure die casting, semi-solid casting (SSC) is performed at lower temperature conditions, which has a number of advantages [ZER⁺13].

The liquid phase is well-studied and can be described as a Newtonian fluid. However, the flow properties and rheology of a solid-liquid mixture are poorly understood. As an example, such characteristics as size and distribution of particles and their clusters, flow properties, morphology of particle groups and events of breaking apart and solidify are crucial to be considered for semi-solid casting.

A typical approach to study rheology of semi-solid metals is to use experimental rheometers by measuring viscosity of a system. With the use of this technique systematic studies were performed to evaluate the dependence on various parameters. However, these studies are only statistical and do not provide enough information to describe and predict the actual casting process. Therefore, in order to investigate more complex phenomena an

in situ imaging method is required. For this purpose X-ray imaging methods are highly appropriate due to the penetrating ability of the radiation.

6.1.1.2 Experimental Setup

The experiment was performed at a beamline station ID15a of the European Synchrotron Radiation Facility (ESRF). For our experiment a high absorption contrast is required to allow for high acquisition rates. For this purpose the high flux beamline ID15a was chosen to carry out this study. ID15a beamline was operated in a white beam mode: the radiation from the source was filtered by approximately 20 mm of silicon, resulting in a photon flux density of approximately 10^{15} photons/mm/s.

The X-ray detector was an indirect system: an optical lens projects the luminescence image of a scintillator screen via a mirror system onto the sensor of a CMOS camera. In order to acquire images with high frame rates, a Photron SA1 camera is used, which is based on a CMOS sensor with 1024×1024 pixels (size $20 \mu\text{m}$) and a true dynamic range of 800:1 grey levels (10 bit, with a 12 bit digitalization) (see Section 1.3.1 for details).

The experimental setup is shown in Figure 6.1. It consists of an *in situ* furnace with a linear motor driving injection of the sample material and an imaging system. The furnace is an aluminium case covered with vermiculite plates to shield the heat produced by four Osram Xenophot 64635 HLX heating lamps (150 Watts, 15 Volts). A sheet of stainless steel is attached to a linear stepping motor and serves as a piston, pushing downwards into the sandwich structure at 20 mm/s velocity, thereby injecting the semi-solid material into the channel.

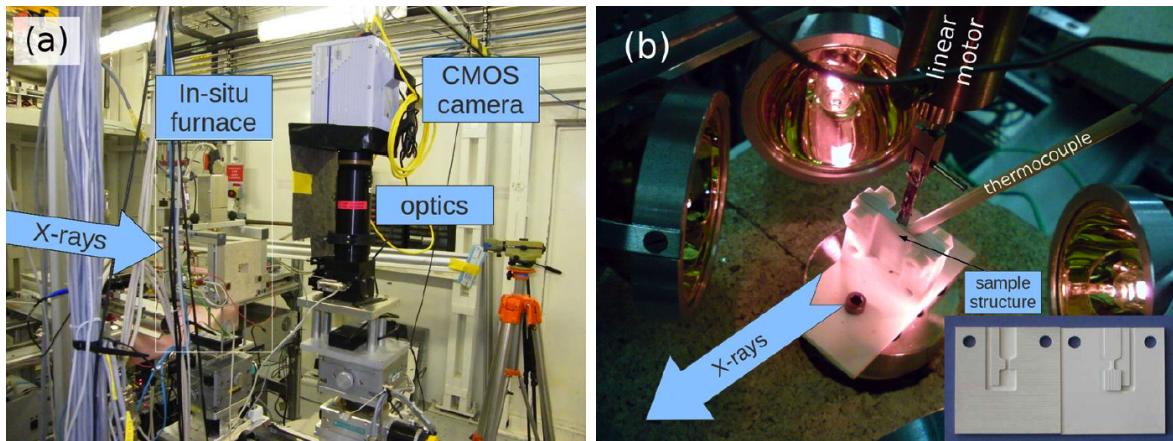


Figure 6.1: Setup for fast radiography experiment on semi-solid alloys. (a) Experimental hutch of the ID15a high energy beamline at the ESRF synchrotron facility with an *in-situ* furnace and a detector system. (b) Close-up view into the sample environment: open *in-situ* furnace, four Xenophot heating lamps, a linear stepping motor drives down the piston into the sandwich structure, thus injecting the semi-solid material during *in-situ* experiment. Image: [ZER⁺¹³]

The alloy Al-Ge32(wt.) was chosen for the experiment. The temperature for the SSC process is 420C for solidus temperature and liquidus temperature of 550C. After each experiment the samples were cooled down at the room temperature, extracted from the

sandwich structure and carried for metallographic investigation under the light microscope (see Figure 6.2). Further details on sample preparation and sample environment can be found in the original publication [ZER⁺13].

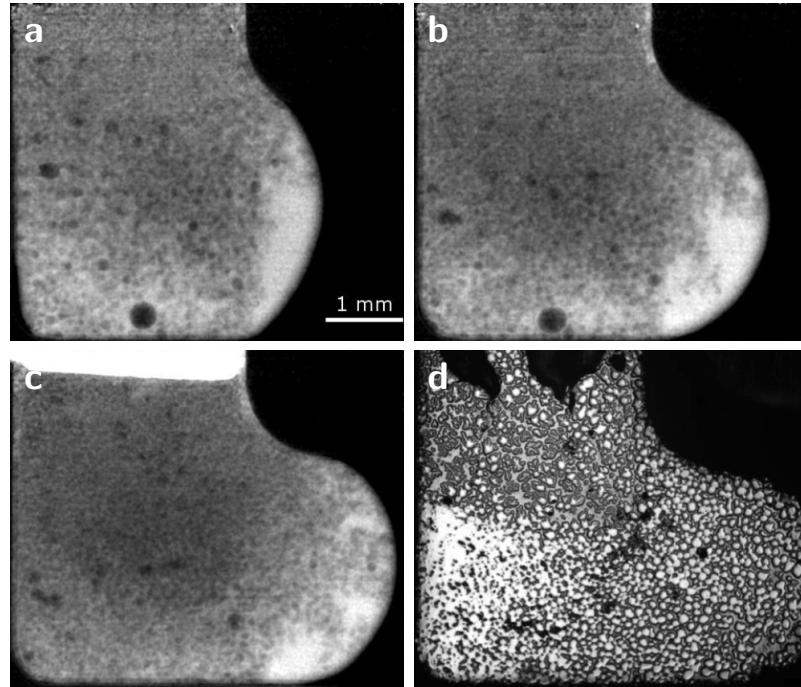


Figure 6.2: Radiographic sequence showing frames 250, 350 and 450 of the right-angle turn geometry. Images from (a) to (c) are 0.2 seconds apart. (d) Metallographic section obtained after the experiment to verify structural changes. Image: [ZER⁺13].

The employed Photron SA1 camera can acquire up to 5400 images/s in a full frame mode. Due to the optical magnification, the detector system operates with an effective pixel sampling of $5.5 \mu\text{m}$, which results in a spatial resolution of $R > 11 \mu\text{m}$ (see Section 1.3.1). For the current study the acquisition rate was 500 images/s for the bottle-neck geometry (Figure 6.3) and 1000 images/s for the right-angle (Figure 6.2).

6.1.1.3 Task for Data Processing

The main task for data processing is a quantitative investigation of complex flow dynamics in semi-solid alloy systems. For this purpose we perform:

- Quantitative motion estimation to measure the velocity of solid particles, clusters and air bubbles.
- Visualize trajectories to reveal flow patterns and interesting events during the casting process.
- Perform motion-based segmentation (using magnitude of the flow or flow direction as a measure) to classify particles and allow for extensive morphological analysis.

6.1.1.4 Data Evaluation

Prior to data processing and optical flow computation we perform an input data characterisation according to our data taxonomy 5.1:

- *Image noise*: The dataset contains high amount of noise due to non-optimal imaging conditions, in particular short exposure time. As a result we obtained image data which is contaminated with high amount of Poisson noise. Additionally, there is small amount of saturated ("dead") pixels within image data. A proper treatment of noise is mandatory. Moreover, an optical flow model which is robust under noise is required.
- *Contrast level*: The image contrast is low, the solid particles and air bubbles are hardly distinguishable from the background. A proper treatment of low-contrast data is needed.
- *Object size*: The solid particles, clusters and air bubbles are of average size. There are no small and large objects.
- *Object distribution*: Particles are distributed normally. In some cases the objects are close to each other and in some cases they are distributed sparsely. The object distribution should not pose a concern for data processing.
- *Object details*: Individual particles are of a homogeneous contrast, no details are present. As a result, the main source of information for the data term are the boundaries of each particles. In the case of particle clusters, which are moving conjointly the amount of details is higher and is composed of details of each individual particle. In general, the amount of details within particles is not sufficient, that is why the appropriate design of the smoothness term becomes important.
- *Image artifacts*: There are very strong variations of brightness both in the space and time. There artifacts may pose a major problem for the optical flow estimation. A proper data correction technique is mandatory.
- *Motion type*: The motion model may be classified as rigid. However, most of the circular particles cannot undergo the rotational motion (it is not captured for circular shapes). More complex clusters of particles may rotate, which frequently occurs during the solidification stage.
- *Motion range*: Motion of particles thorough out the process is slow, thanks to the correctly chooses parameters for data acquisition, which allowed adequately sample the process, yet avoid significant motion blur.
- *Motion discontinuities*: There is a certain amount of discontinuities between the moving particles and clusters. However, this should not affect the computation of flow fields in a substantial way. A more dangerous scenario is the presence of occlusions between moving particles and their clusters, which can happen on later stages of the thixocasting process.

Table 6.1: Evaluation of the semi-solid alloys dataset.

Image	Data	Motion
noise: high contrast: low artifacts: brightness	size: average dist: norm detail: low	type: rigid range: small disc: occlusions

A summary of evaluation of this dataset is given in Table 6.1.

Requirements on the Results

Accuracy: For this dataset we do not need sub-pixel accuracy. The error in the range of 1-3 pixels is sufficient to accomplish the task for data processing.

Density: Dense flow fields are not mandatory. The correct estimation of the flow vector for each particle is sufficient. The fact that this requirement is relaxed gives some degree of freedom for the data preprocessing step - we can use much stronger filters to get rid of noise and artifacts.

Motion components: The full flow is required, i.e. both the flow magnitude and the flow direction.

Consistency: We do not pose any consistency constraints.

Motion boundaries: We do not aim to capture strict flow on the boundaries of particles and particle cluster. Again, this gives us the possibility to increase the influence of the smoothness term and outcome the problem of low feature details.

Computation time: Computation time is not an issue, the processing is done offline.

Data size: A typical size of a single image frame is 1024×1024 pixels. An image sequence can consist of up to 1000 time frames. There is no restriction on processing due to the size of the input data - both image pairs are fitted into the memory and the processing is done sequentially.

6.1.1.5 Data Analysis: Preprocessing

In order to improve the raw data and meanwhile preserve spatial and temporal details we develop an preprocessing routine.

Flat- and dark-field correction. The routine starts with dark field filtering (3.1.2.2) to correct for the fixed-pattern noise, including "dead" and saturated pixels. Flat-field radiograms are not available for this type of experiment, which is a typical limitation for fast *in situ* experiments. Thus, temporal brightness variations have to be corrected by a dedicated filtering step.

Artifacts removal. Next we proceed with applying a 3D Hybrid median filter [http:](http://)

[//rsbweb.nih.gov/ij/plugins/hybrid3dmedian.html](http://rsbweb.nih.gov/ij/plugins/hybrid3dmedian.html), (Section 3.1.1.2) to eliminate the high contrast speckle noise and small image artifacts caused by dust and scratches on the scintillator screen. For this procedure a number of time-lapse frames are used ($N = 5$). The procedure effectively removes small artifacts.

Brightness variation correction. In order to equalize non-uniform brightness variations we perform an image leveling using the median filter with a large spatial mask (Section 3.1.2.4). This part of the filtering process removes varying large-scale brightness patterns, but retains local details.

Noise removal. Finally, we use an anisotropic diffusion filtering (Section 3.1.1.2) to eliminate the image noise and at the same time to enhance particle edges.

Background subtraction. To improve visual appearance of the filtered radiographs we subtract background which contains no structural details to reveal only the semi-solid material.

On Figure 6.3 we show the result of our preprocessing procedure.

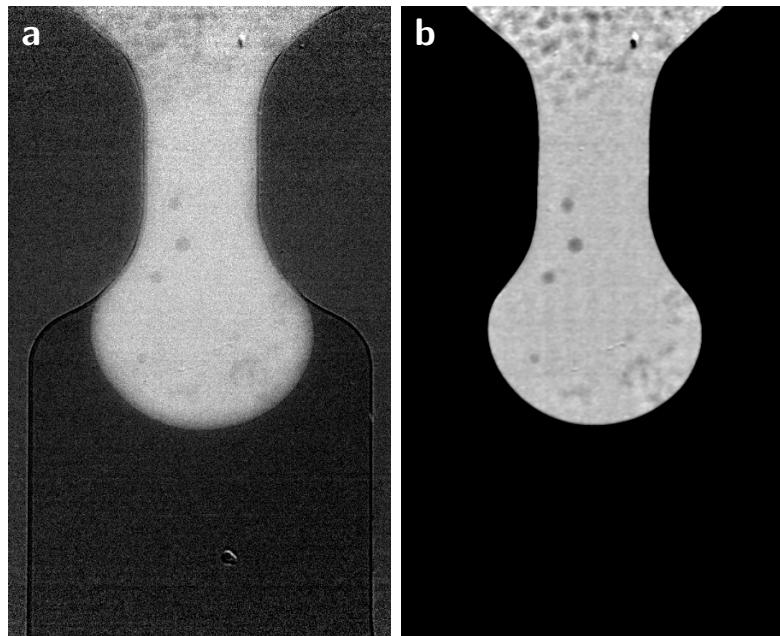


Figure 6.3: Preprocessing of the original radiographic sequence. (a) Original radiograph. (b) Filtered radiograph. The filtering procedure consists of noise removal, brightness normalization, anisotropic diffusion filtering and background subtraction.

6.1.1.6 Data Analysis: Flow Computation

In order to compute flow fields we employed an advanced optical flow model, which takes into account multiple image features (Section 2.1.2.5) and a priori information about motion model. For the construction of the data term we assume the constancy of image brightness (Section 2.1.2.2) and constancy of spatial intensity gradients (Section 2.1.2.3). This joined assumption is more suitable for the motion estimation, if there is a small

amount of spatial images features and objects are of low contrast. To provide robustness against noise and data artifacts that remain after a preprocessing step, the data term is extended by a combined local-global approach (Section 2.1.3.3). Since we do not aim to capture fine details of moving particles, air bubbles and particle clusters, and the correct estimation on a more coarse scale (particle shape) is sufficient, a large value of the integration parameter in the range of $\rho=[0.7...1.5]$ can be used.

For the motion constraint we choose a flow-driven smoothness approach (Section 2.1.4.3). The rather high image acquisition rate (500 images/s and 1000 images/s) results in a smooth gradual motion of the constituents, from which we can benefit by introducing an additional spatio-temporal smoothness constraint (Section 2.1.4.4). In this way, the final optical flow model takes into account motion between more than two successive frames ($N = 3$). Furthermore, to separate the motion of the liquid front from the flow of adjacent particles, the smoothness is adapted (see Section 2.1.4.5) to a spatial mask covering the air-liquid interface.

In order to cope with a large range of flow velocities a coarse-to-fine computation strategy is used (Section 2.2.5). In order to visualize the resulted flow fields we use a color pseudo coding, which indicates the direction by color and the velocity by its brightness.

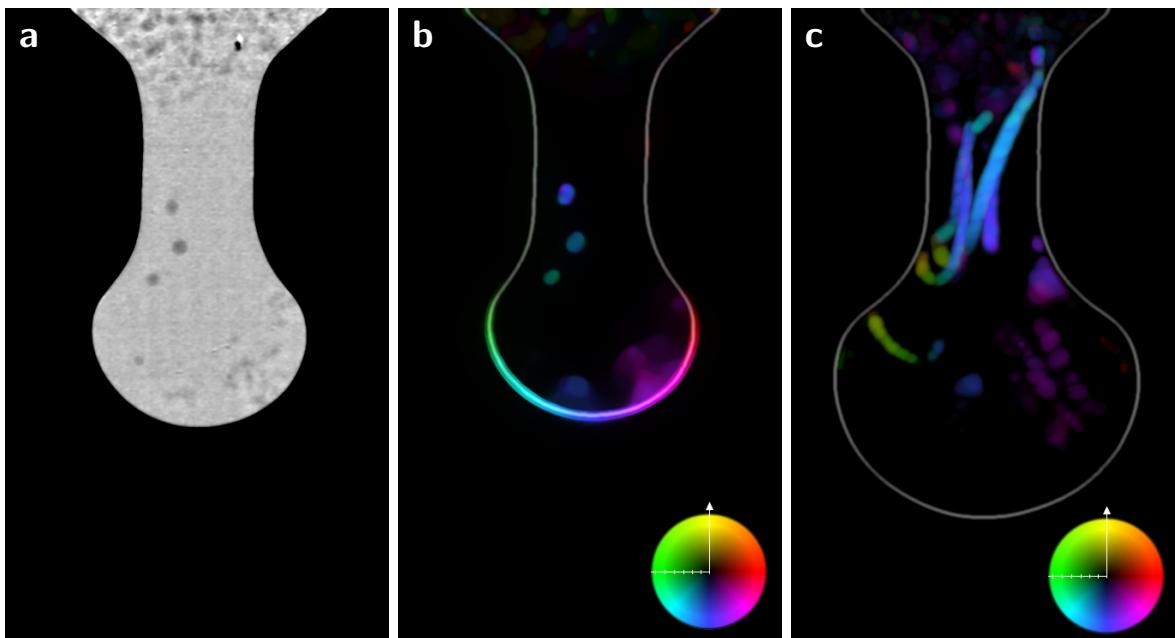


Figure 6.4: Flow computation during semi-solid casting using optical flow. (a) Filtered radiographic projection. (b) Calculated flow map. According to the color wheel at the bottom right a color shows direction and brightness encodes velocity. Arrows: (1) air bubbles, (2) cluster of particles moving at high speed on the crest of the liquid front, (3) a static particle around solidified region, (4) slowly moving bulk skeleton. (c) Projected average velocity, integrated from frame 50 to frame 250. It shows the curved trajectories of air bubbles, as well as several solid particle clusters which can be seen at the lower right.

6.1.1.7 Data Analysis: Flow Analysis

The results of optical flow computation on a bottle-neck sequence are shown in Figure 6.4. One can clearly distinguish between the liquid and the solid phase whereby the latter appears darker due to the lower X-ray density of aluminium compared to the Ge-rich liquid matrix. Air bubbles are difficult to distinguish from aluminium particles. Compared to the latter, they descend at higher velocities and appear darker in the radiographs (arrow 1 in Figure 6.4b). Two or three small clusters of solid particles detached from the bulk and move freely through the channel following the liquid-air interface (arrow 2 Figure 6.4b). One isolated particle appears to be completely static on this frame (arrow 3). The expanding liquid forms a semi-circular crest of constant velocity, as it can be expected for a Newtonian fluid. The movement of the solid bulk in the upper part of the radiograph is slow, which is captured by the computed flow field in Figure 6.4c (indicated by arrow 4). Evaluating the quantitative flow amplitude the liquid is found to advance at a maximum speed of 3.1 mm/s, with the air bubbles descending at relatively high speed, up to 3.9 mm/s. The velocity of the particle cluster at the lower liquid-air interface is 1.6 mm/s, and the average speed of the solid bulk in the upper part of Figure 6.4b remains less than 0.3 mm/s.

A projected time evolution of interior of semi-solid alloy is shown in Figure 6.4c. Here, the average velocity was projected for each pixel over several hundred time frames, i.e. integrating from frame 50 to frame 250, while removing the velocities of the liquid front with a automatically extracted contour mask. The resulting color-coded velocity projection shows dynamics of solid particles, clusters and air bubbles. The latter follow a curved trajectory until the downwards directed pressure is balanced by the buoyant force and their motion describes a hook trajectory.

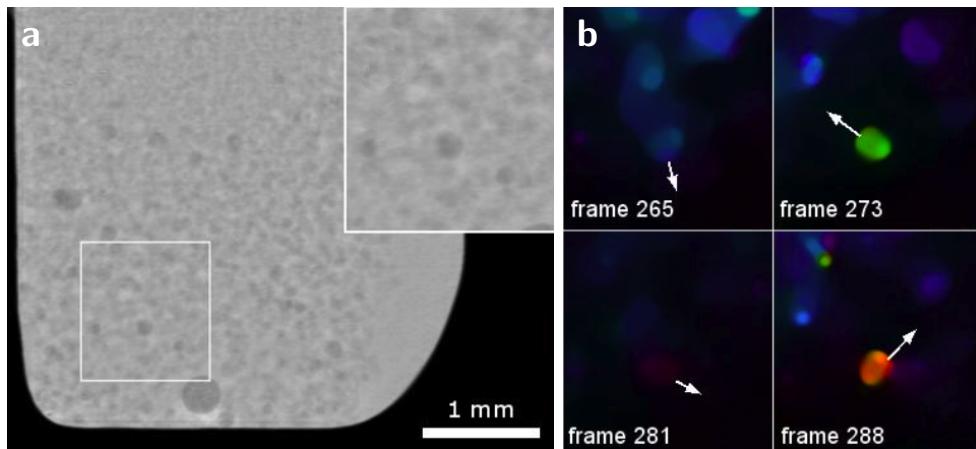


Figure 6.5: Example of turbulent motion in the right angle turn channel. (a) Input radiograph with magnified region of interest. (b) Color-coded turbulent motion of a single particle recorded over 23 frames. The arrows indicate the direction of particle motion as well as the velocity amplitude (arrow length).

When the experiment is carried out with a different channel geometry (right-angle turn) the flow behavior changes dramatically. The right-angle mold geometry was chosen to create a more turbulent flow, in the hope that the flow of both liquid and solid phases would be more conjoint compared to the bottleneck geometry. Judging from the metal-

lographic sections of the final structure (see Figure 6.2d) the solid phase appears indeed to stay closer to the liquid front. Furthermore, evaluation of visualized flow fields reveals numerous particles which exhibit turbulent, Brownian-like motion. An example of such event can be seen in Figure 6.5.

A spontaneous break-up of a larger particle cluster from the solid skeleton is displayed in Figure 6.6. A large region of the solid phase, approximately 1.5 mm in diameter remains frozen for approximately 35 ms (Figure 6.6), then suddenly the upper half of this cluster moves at high speed into separate directions, indicated by velocity magnitude flares, while the lower part of material remains still.

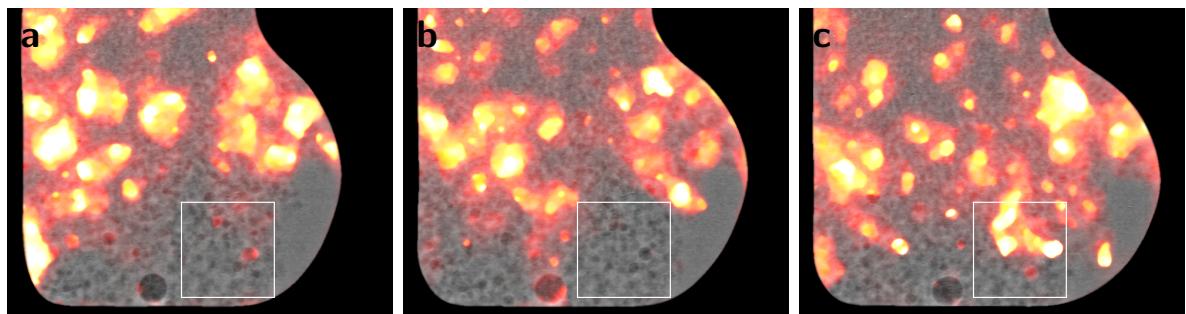


Figure 6.6: Example of sudden breakup of a larger cluster of particles, visualizing thixotropic effect. Images show radiographs of the right-angle turn geometry with the super-imposed velocity amplitude with brightness. (a) Frame 270, rectangle indicates a region where only little motion of the solid phase is observed. (b) In frame 296 the solid skeleton in the same region has become complete static (i.e. solidifies). (c) 34 frames later (frame 330, 35 ms apart) the static structure breaks up into many parts.

6.1.1.8 Results

In situ experiments by means of fast X-ray radiography with the optical flow analysis revealed a multitude of dynamical effects, particularly for right-angle turn geometry of the flow channel. Here, isolated turbulent particle motion and transition from a frozen solid skeleton to moving particle / clusters could be observed. This, we believe, is the *first visual proof* of the thixotropic break-up during semi-solid injection.

For the bottleneck channel in the narrow part only few particles traverse the bottleneck, the remains accumulate at the entrance. The solid movement comes to a complete halt, when the liquid has fully filled the recipient and compressed solid particles block further flow through the channel. Our results clearly show the inter-particle as well as the particle-wall friction to be the hindering force during semi-solid injections through thin cavities.

On the contrary, both solid and liquid phase moved relatively conjointly through the right-angle turn in the second experiment, which was found to locally exhibit turbulent flow resulting in Brownian-like particle motion and breakup of larger clusters into smaller particles. It appears that the less laminar flow of the liquid which takes place in such channel-turns and -corners allows the solid skeleton to advance at a similar speed with the liquid.

These investigation extended by others geometries or introduction of additional compounds, as well as temperature dependence studies should advance the understanding of semi-solid casting technology.

6.1.2 Analysis of Morphogenesis in Frog Embryos

6.1.2.1 Introduction

A primary goal in biology is to understand the behavior of cells during the development of an organism. One way to reach it is to image the actual morphological changes of embryonic structures *in vivo* with sub-cellular resolution. Many morphogenetic events take place in the process of embryogenesis. One particularly important stage is *gastrulation*, a stage when a series of dramatic changes occur and coordinated cell movements results in a transformation of a simple, homogeneous ball of cells into a complex multi-layered organism [KDS03].

In model organisms, such as *Xenopus laevis* and zebrafish, cell movements, tissue and organ formation have been studied by a number of well-established imaging methods: microscopy for explants and fixed embryos, *in vivo* using fluorescence microscopy [KSWS08, HS09] or microscopic magnetic resonance imaging [PBV⁺07]. However, these methods do not allow to visualize *in vivo* cell behavior in *optically opaque* living embryos with micrometre-scale spatial resolution.

To overcome these limitation we use *in vivo* time-lapse X-ray microtomography, based on a single-distance phase contrast and supplemented with a multi-scale motion analysis to examine the process of embryonic development.

Phase-contrast imaging exploits high coherence and high intensity properties of the radiation produced by modern synchrotrons. During early developmental stages vertebrate embryos are composed of light chemical elements (comparable with water). As a result, for hard X-rays phase changes in the transmitted wave dominate the effects of attenuation of the intensity. This allows to avoid high dose deposition attributed to a conventional X-ray absorption imaging, which is a crucial aspect for investigation of living organisms. Due to a high penetration depth, high spatial and temporal resolution phase-contrast microtomography is a genuine 3D imaging modality to visualize the whole scale of embryonic structures: from organs, tissues and single cells, down to subcellular structures such as nuclei and yolk platelets.

6.1.2.2 Experimental Setup

The experiment described in this work was performed at beamline station 2-BM-B of Advanced Photon Source synchrotron facility (X-ray beam energy $E = 30$ keV, photon flux density is $\sim 10^{12}$ photons/s/mm²). The experimental setup was optimized for a low dose deposition and longer observation time. Additionally, in order to avoid blurring artifacts due to cells motion and obtain sufficient image contrast a compromise between parameters of the acquisition process had to be made. After the phase retrieval algorithm was applied on projection radiography, a filter-back projection tomographic reconstruction method was used to obtain a 3D distribution of electronic density of the sample. The following parameters were used during the acquisition procedure: expose time per projection is 15 ms, number of projections $N = 2000$, which yields a 18 s acquisition time

for one tomogram with the continuous rotation of the sample. To obtain a time-lapse images of the development process the waiting time between subsequent tomograms was 10 minutes. These time intervals were long enough to provide an adequate observation time and sufficiently short to result in a gradual cell movement between the subsequent time frames, which permits a reliable motion estimation. The steps of sample preparation, such as in vitro fertilisation, embryo culture and staging were carried out as described in [KKK⁺09]. The X-ray phase-contrast microtomography setup is shown in Figure 6.7.

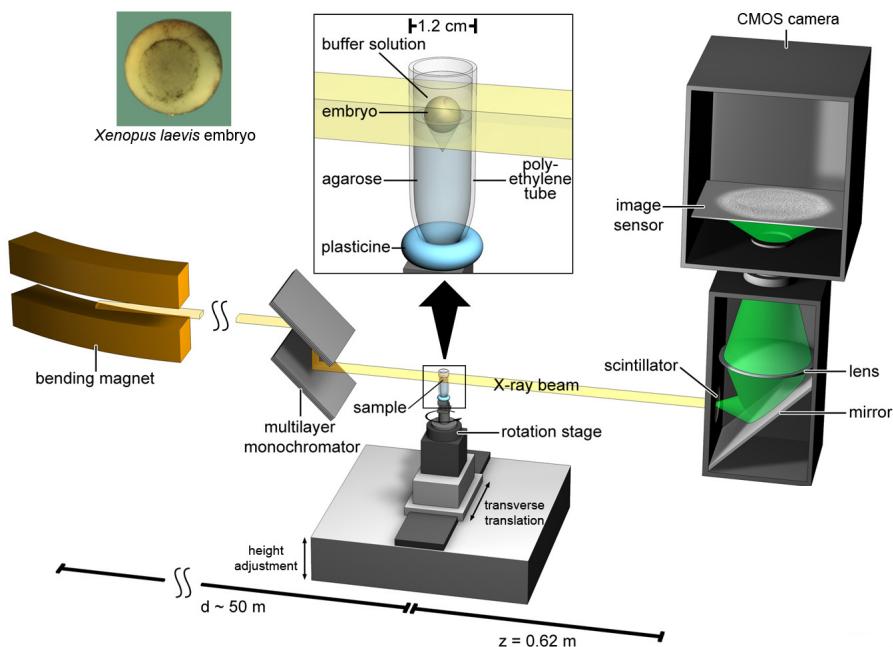


Figure 6.7: Experimental setup for phase-contrast X-ray microtomography. A parallel photon beam is produced by a synchrotron source. A monochromator is used to change the properties of the beam. X-rays propagate over a distance $d = 50\text{m}$ and interact with a living *Xenopus laevis* embryo immersed in a buffer solution and suspended by agarose. The sample is mounted on a rotation stage to perform a tomographic scan. The 2D detector consists of a scintillator, converting X-rays into visible light, a mirror system, an optical lens, and a CMOS camera with 2016×2016 pixel resolution and pixel size $11 \times 11 \mu\text{m}^2$, which yields an effective pixel size of $\Delta x = 2.2\mu\text{m}$.

Figure 6.8 shows 3D renderings of a mid-sagittally cut tomographic volumes of an embryo at the developmental stages 11.5, 12.0 and 12.5. Single cells, tissue layers and important embryonic structures are distinguishable, including the blastopore, archenteron, ventral and dorsal blastopore lips, blastocoel, blastocoel roof and floor, and the porous cell network between archenteron and blastocoel. A detailed, step by step protocol for the sample preparation, description of the experimental setup and data acquisition pipeline could be found in [MEW⁺14].

6.1.2.3 Task for Data Processing

Morphogenesis is a complex process, it involves many diverse and tightly coupled cell movements and tissue changes. The whole underlying dynamics can be captured by

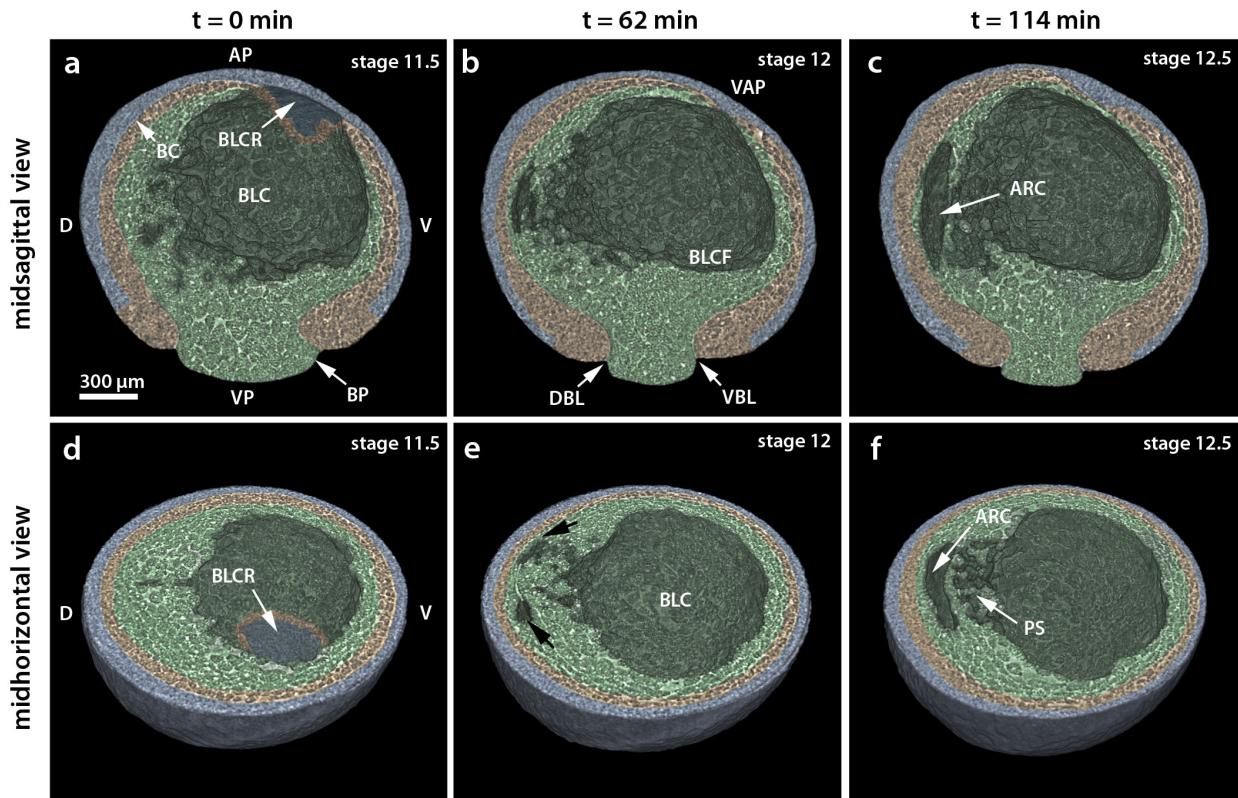


Figure 6.8: 3D time-lapse series of *Xenopus laevis* embryo during gastrulation stage. Mid-sagittal (**a-c**) and mid-horizontal (**d-f**) cuts of embryo 3D renderings at stages 11.5 (0 min), 12 (62 min) and 12.5 (114 min). Tissue layers are labeled as follows: ectoderm (blue), mesoderm (orange), and endoderm (green). The following structures are identified: Animal pole (AP), Archenteron (ARC), Brachets cleft (BC), blastocoel (BLC), blastocoel floor (BLCF), blastocoel roof (BLCR), blastopore (BP), dorsal and ventral sides (D, V), dorsal and ventral blastopore lip (DBL, VBL), porous cell system between archenteron and blastocoel (PS), ventral animal pole (VAP) and vegetal pole (VP).

optical flow methods. To analyze the entire multitude of morphogenetic movements and events we use a multi-scale approach. In this approach we distinguish the following steps:

- Perform dense, quantitative motion estimation of individual cells and tissues. Visualize global patterns of morphogenesis during mid-gastrulation stage.
- Distinguish collective motion and motion of individual cells. Discriminate between passive and active movements. Draw conclusions about cause-and-effect and dependency of morphogenetic processes.
- Track individual cells, analyze their dynamics to investigate interaction mechanisms between cells.

6.1.2.4 Data Evaluation

First, we evaluate our embryo datasets to identify possible problems and gather useful information for the design of an effective data processing pipeline. This data classification according to our data taxonomy 5.1 reads:

- *Image noise:* The original radiograms contain high amount of noise due the short exposure time, which was necessary to limit radiation damage to the sample. After the tomographic reconstruction the noise issue is partially solved. However, the data may still suffer from the presence of noise, that is why optical flow models which are robust with respect to noise should be used.
- *Contrast level:* The image contrast in the reconstructed tomograms after the phase-retrieval is sufficient. It is possible to distinguish between tissues, organs, individual cells and even subcellular structures, such as nuclei. The good contrast level is strong point of the dataset. (See Figure 6.12 a).
- *Object size:* The structures of each tissue or organ are individual cells of different types. Most of the cells are small, some of the cell types (ectoderm) are even hardly resolved. The size of objects can be a major problem for optical flow estimation. (See Figure 6.12 a).
- *Object distribution:* Cells are closely packed to form a tissue or an organ. The cells distribution can be classified as dense. Such allocation in a combination with the small size of cells pose a major challenge for data processing. (See Figure 6.12).
- *Object details:* The amount of feature in tissues, organs or individual cells is high (except very small cells of ectoderm). Again, this property is a strong point of the dataset. (See Figure 6.12 a).
- *Image artifacts:* There are very strong variations of brightness in the original radiographs (See Figure 6.9). The variations appear as horizontal stripes which are shifting in time. The artifacts are severely degrading the original data. After the reconstruction brightness inhomogeneities cause very pronounced ring artifacts. These artifacts pose a major problem for the optical flow estimation. The use of proper brightness and ring artifacts correction techniques is mandatory.
- *Motion type:* The motion of tissues, organs and individual cells is non-rigid. Cells may deform and change their shape. Such changes, however, do not occur rapidly and can be assumed to be locally smooth.
- *Motion range:* Motion of cells is slow and there is not much variation in the motion range. The frame rate of the detector system was adequate to capture morphogenetic movements.
- *Motion discontinuities:* There are a lot of motion discontinuities between the moving cells. Adjusted structures can move in completely different direction.

A summary of evaluation of embryo dataset is given in Table 6.2.

Requirements on the Results

Accuracy: For this dataset we require accurate results. The error in the range of 1-2 pixels would be sufficient, but exceeding values should be eliminated from the results or indicated.

Table 6.2: Evaluation of the embryo dataset.

Image	Data	Motion
noise: average	size: small	type: non-rigid
contrast: high	dist: dense	range: small
artifacts: brightness, rings	detail: high	disc: high

Density: Dense flow fields are mandatory to capture differences in cells motion.

Motion components: The full flow is required, i.e. both the flow magnitude and the flow direction.

Consistency: We do not pose any consistency constraints.

Motion boundaries: We aim to capture strict flow on the boundaries of moving cells. This information is crucial to analyze cell-to-cell behaviour.

Computation time: Computation time is not an issue, the processing is done offline. But taking into account the size of tomographic datasets (see next point), it would be a big advantage to perform the computation on GPU to speed up the process.

Data size: The original size of a single tomogram is $2016 \times 2016 \times 2016$, which corresponds to 32 Gb of floating-point data. This tremendous amount of data is impossible to process on CPU in a reasonable time. That is why data should be reduced or processed using GPU technology. Even in this case memory management is a major challenge.

6.1.2.5 Data Analysis: Preprocessing

For the experiment described in this section the imaging conditions were optimized for a low-dose scenario and a strict compromise have been made between data quality and embryo viability. As a consequence, the dataset possess all kind of image degradation artifacts and is very challenging to analyze. Here we enlist data preprocessing steps to remove artifacts and improve signal-to-noise ratio as a preparation step for further data analysis.

Removal of hot/dead pixels and brightness variations. The procedure starts with a removal of hot and dead pixels (Section 3.1.2.2). A single dark field image D is computed using a median of 60 dark-field images $D_1 \dots D_{60}$, recorded independently (to acquire image statistics). Then, flat-field images and original radiographs are dark-field corrected by taking a difference between the image D every flat field image, as well as with the original radiograph. Due to cooling-induced thermal instabilities of the multilayer monochromator at the 2-BM-B beamline a temporally varying horizontal stripe brightness pattern significantly degrades the image quality. A crucial step is the correction of the initial radiographs and the flat-field images prior to any image reconstruction procedure. For this purpose before the flat-field correction gray values in each pixel along a given horizontal line are normalized to the mean value in this line (Section 3.1.2.4). After the image level-

ing of flat-fields images a single flat-field image F is calculated as a median. Normalized radiographs are then flat-field corrected using image F . Individual steps of the filtering procedure are shown on Figure 6.9.

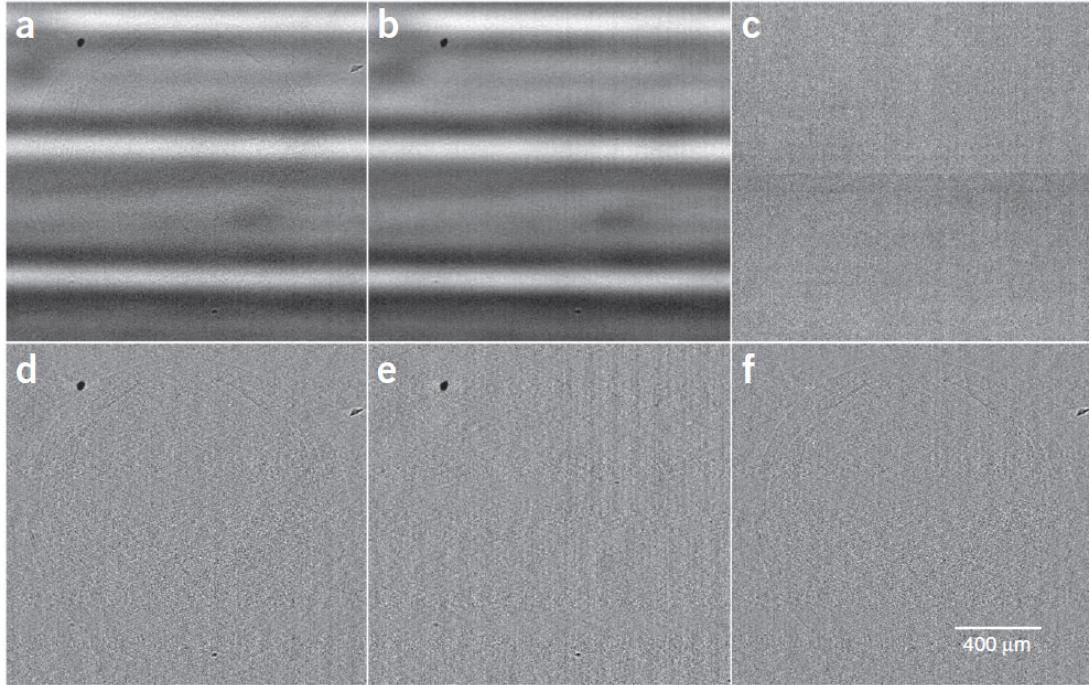


Figure 6.9: Removal of brightness variation artifacts, flat-field and dark-field correction. (a) Raw projection radiograph. (b) Flat-field image. (c) Dark-field image. (d) Projection radiograph with brightness artifacts removed. (e) Flat field image with brightness artifacts removed. (f) Flat- and dark-field corrected projection image with brightness artifacts removed.

Ring-artifact removal. Because of short exposure times and, as a consequence, low count rates, the CMOS camera sensor produces characteristic vertical stripe artifacts. Since such patterns are sensor related and independent of the angle of rotation, they give rise to ring artifacts in the reconstructed volume. Ring artifacts can be effectively filtered in polar coordinates, in which circular brightness patterns become straight lines and can be easily identified. To implement this a filtering procedure is performed in the Fourier domain. As a first step, the Fourier transformation along horizontal and angular coordinates is done. Then, grey values of the plane corresponding to a zero angle are substituted by median values of adjacent planes. As a final step, an inverse Fourier transformation is computed to obtain a stack of filtered sinograms. Figure 6.10 shows results of the ring artifacts removal procedure.

Phase retrieval. To extract a 3D distribution of refractive index a simple linear phase retrieval is performed [PN98].

Tomographic Reconstruction. The final step of data processing is 3D tomographic reconstruction using the filtered back-projection method (See section 1.3.2.2). Since the

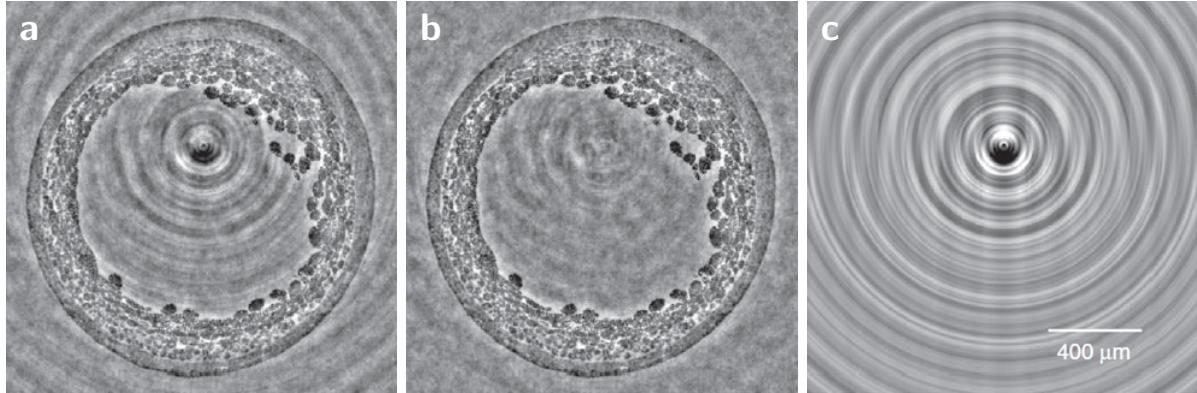


Figure 6.10: Removal of ring artifacts. (a) A horizontal slice through the reconstructed volume (no ring artifacts filtering was used). (b) Reconstructed slice with the ring artifacts reduction procedure applied. (c) Difference between processed and unprocessed images shows the removed ring artifacts.

number of acquired projections is sufficient and an efficient data preprocessing pipeline is established, we achieve a good quality tomographic reconstruction.

6.1.2.6 Data Analysis: Flow Computation

Even after an extensive pre-processing procedure, time-lapse sequences are still challenging to analyze in terms of automated techniques. For the reliable computation of flow fields we use a robust 3D variational optical flow method. For the modeling of the data term we choose the constancy of pixel brightness (Section 2.1.2.2). Since embryonic structures contain a lot of image details which are highly textured, and the grey value constancy assumption is more robust in the presence of noise, there is no benefit to use the gradient constancy assumption (as we concluded in our experimental evaluation). For the regularization of the motion field we use the flow-driven smoothness assumption (Section 2.1.4.3).

Before motion computation the input images are smoothed using a Gaussian filter with a large spatial mask (optimal choice is in the range $\sigma = [1.2 \dots 2.0]$). To further improve the performance with respect to noisy data, we incorporate a local-global approach (Section 2.1.3.3), that takes into account image data from a local neighborhood. The optimal values for the integration scale parameter are in the range $\rho = [0.5 \dots 1.2]$. We use a multi-level computation procedure (Section 2.2.5) to capture the entire range of cell velocities. The number of warping levels and the warping scale parameter are the same as for the *baseline* method (Section 5.3.3). A median filtering of the flow fields at the intermediate computation levels is a crucial step to exclude data outliers due to the existing image artifacts. In general, we can also use a forward-backward checking as an automated confidence measure (Section 2.1.5.5) to discard flow vectors which are not reliable. As a result of optical flow computation, we obtain a dense 3D velocity field, which we use for further analysis. A visualization of the computed optical flow is given in Figure 6.11, using sparse 3D vector glyphs. The assessment of the accuracy of the optical flow computation we present in the section, dedicated to cell tracking (See Section 6.1.2.8).

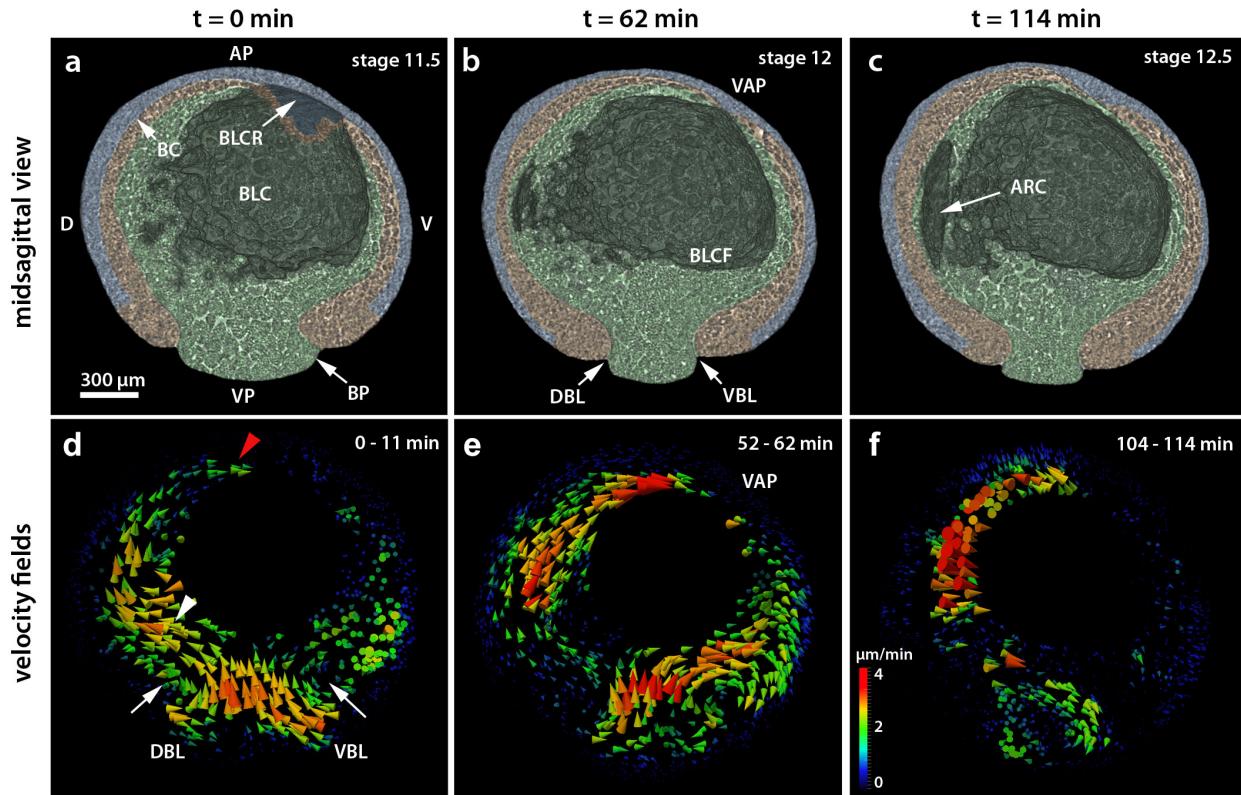


Figure 6.11: 3D time-lapse series of *Xenopus laevis* embryo during mid-gastrulation. (a-c) Mid-sagittally halved embryo renderings at stages 11.5 (0 min), 12 (62 min) and 12.5 (114 min). (d-f) Velocity fields on a 180-mm thick 3D slab centred about the cutting planes of (a-c). Colour bar indicates velocity magnitude representation.

6.1.2.7 Data Analysis: Flow Analysis

As we already mentioned when stating the tasks for data processing , morphogenesis involves hierarchical, multi-scale cell movements. The whole range of underlying dynamics can be captured by the optical flow method and expressed as a dense motion field \vec{v} . We pursuit a multi-level strategy for motion analysis and investigate: i) global, collective aspects of the displacement field \vec{v} , ii) its local differential behavior and iii) a long range individual cells behavior using cell trajectories.

The computed motion fields for time-lapse series of *Xenopus laevis* embryo during a mid-gastrulation reveal global patterns of cell movements. These patterns comprise several morphogenetic processes, including rotation of the vegetal endoderm (white arrowhead in Figure 6.11g), involution of the mesendoderm at the dorsal and ventral blastopore lips (white arrows in Figure 6.11d), and migration of the mesendoderm on the blastocoel roof towards the animal pole (red arrowhead in Figure 6.11d). By stage 12, prominent horizontal movement within the endoderm layer is observed on the ventral side of the embryo (Figure 6.11e). Additionally, a number of interesting events can be observed: rapid migration of leading-edge head mesendodermal cells on the blastocoel roof, a bulk movement of the mesendoderm towards the animal pole, and local cell dispersion due to the onset of archenteron formation. After the dorsal and ventral edges of the mesendodermal mantle make contact and cover the blastocoel, an overall halt of cell movements

is observed on the ventral side (Figure 6.11f, stage 12.5). It is important to note, that before these studies, most of such morphogenetic processes in *Xenopus laevis* were never captured *in-vivo* and quantitatively analyzed with high resolution and accuracy.

For the next level in our hierarchical motion analysis approach, we examine a differentiated cell and tissue movements using a phase of the motion field (See Section 3.2.1.4). This measure defined as a magnitude of the spatial gradient of the motion field $\mathbf{v} = (v_x, v_y)$ and is given by $P = |\nabla \mathbf{v}|^2$, where $|f| = \sqrt{f_x^2 + f_y^2}$ denotes the spatial magnitude and $\nabla = (\partial_x, \partial_y)$ the spatial gradient. Small values of measure P indicate smooth, collective motion, while large values capture local changes in velocity magnitude and/or variations in the direction of \mathbf{v} . In addition, we use a magnitude of the motion field $|\mathbf{v}|$ (See Section 3.2.1.1) to analysis an activity level of cells in a particular time point.

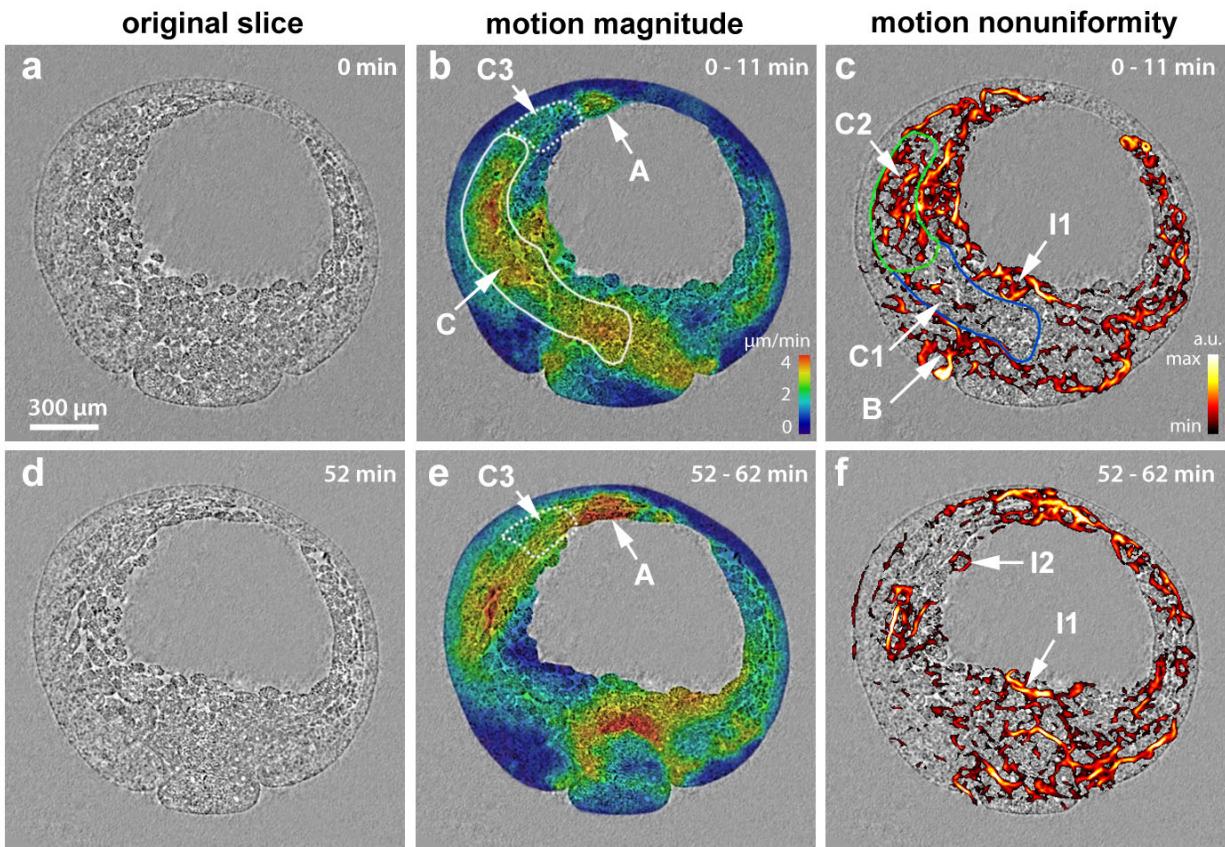


Figure 6.12: Collective/differentiated motion and cells activity analysis. (a, d) 2D slices through tomographic volumes at times 0 and 52 min. (b, e) Magnitude $|\mathbf{v}|$ of the velocity field on a sagittal slice for two subsequent time frames. (c, f): Field P for the same slice and times frames.

Figure 6.12 shows a mapping of magnitude and phase of the motion field measures. Using these visual maps it is possible to analyze, interpret and draw conclusions about causal relations between different morphogenetic movements and events. Contour C encloses a large, collectively moving cell mass (Figure 6.12a), extending from the vegetal region ($C1$, Figure 6.12c) along the posterioranterior axial mesendoderm and endoderm on the dorsal side ($C2$, Figure 6.12c).

The movement of cells within the $C1$ region is mostly collective (small values of P),

driven by blastopore closure. But in region the *C*2, a similar velocity magnitude is a result of involution of the dorsal mesendoderm (note the transition from *C*1 to *C*2 in Figure 6.12c).

Interestingly, despite the fact that the motion magnitude in region *C*2 is similar, cells do not move coherently. The relative cell motion within *C*2 can be attributed to mediolateral intercalation associated with convergent extension. This can explain the acceleration (Figure 6.12 g, h) of the anterior cells within *C* (Figure 6.12a) towards the animal pole. Another interesting observation is a velocity gap (i.e. discontinuity) in region *C*3 (Figure 6.12a). It is seen from comparing Figures 6.12a and 6.12b in *C*3 region. This gap indicates active migration of the leading-edge cells in the small region *A* (Figure 6.12a,b) which was reported in the studies with explants.

Cells with individual movements are also discernible: cells (*I*1) crawl on the blastocoel floor (Figure 6.12c,d), and a single cell (*I*2) migrates on the endodermal cell mass (Figure 6.12f).

6.1.2.8 Data Analysis: Tracking

In order to analyze individual cells behavior for a longer time period we perform cell tracking. We use an automated tracking algorithm based on points tracking and optical flow (See Section 3.2.2). Cell trajectories are obtained by time integration of a 3D flow field, starting with manually selected initial positions. As objects we identify the centers of cell nucleus, lining both sides of archenteron walls (Figure 6.13). We checked the accuracy of the optical flow algorithm by comparing automatically estimated cell trajectories with the corresponding trajectories tracked manually by the user expert. For an average traveled distance $58 \mu\text{m}$ over four successive volumes used for the trajectory computation, the average error was $4 \mu\text{m}$, which roughly matches the resolution limit of $4.4 \mu\text{m}$. This indicates a reliability of the optical flow computation.

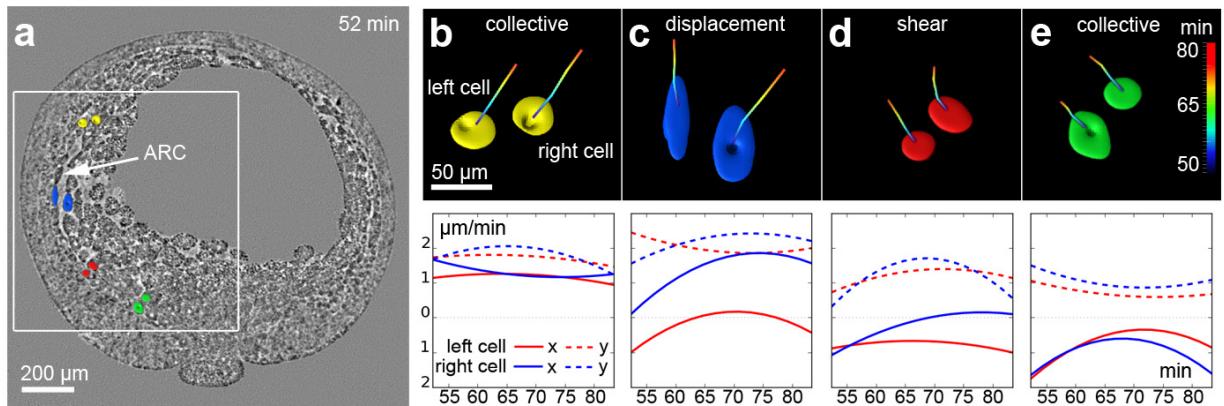


Figure 6.13: Trajectories and velocities of single cells lining the archenteron. (a) parasagittal slice at 52 min with highlighted cell pairs lining the region along the archenteron. (b-e), trajectories of these cell pairs (top row) and associated velocity components in x-y (parasagittal) plane (bottom row; left cell: red, right cell: blue; solid: x-component, dashed: y-component).

After trajectories of moving cells are estimated by our automated tracking procedure, parameters of their dynamics and interaction can be studied. The two cell pairs most

proximal to the animal pole (Figure 6.13a,b) and the blastopore (Figure 6.13a,e) display near-parallel trajectories in the direction of mesendodermal mantle, which is an indicator of collective motion. In contrast, for cell pairs occupying more medial positions along the posterior-anterior extent of elongation a distinct directional variation is prominent (Figure 6.13c,d). For the anterior cells (Figure 6.13c) there is a large difference between radial and tangential displacement. This large horizontal displacement is a consequence of the onset of archenteron inflation and vertical component is a result of active push from the side of vegetal cell mass. The dynamical characteristics are quantified and visualized in Figure 6.13.

6.1.2.9 Results

The results presented in this section demonstrate that phase-contrast X-ray microtomography, combined with optical flow and multi-level motion analysis, is a powerful tool for 4D *in vivo* investigation of embryonic development. Analysis of differentiated motion allows to distinguish between active or passive tissue and cell movements, giving insights into how collective morphogenetic events are powered by individual cell behaviour. Such studies can advance our understanding of the molecular mechanisms and biomechanical processes which drive embryogenesis.

6.2 Temporal Changes Detection

6.2.1 Stability Study During Foaming Process

6.2.1.1 Introduction

The production, characterization and application of foams and cellular structures has attracted increasing attention in the recent decade. Foams and cellular structures exhibit remarkable properties, such as high strength and mechanical stiffness, low density and the ability to absorb large amounts of energy. The porous materials are widely used for a broad range of industrial, medical and scientific applications.

During a foaming process a complex interplay of different physical phenomena takes place. Such processes include: liquid drainage, coalescence, coarsening, topological reorganization, pore inflation and many others. A sudden instability in a film leading to its disappearance is called *coalescence*. Such process triggers two (or more) neighbor bubbles to merge onto a single one by the rupture of separating film walls. Coalescence events are the major phenomena which affects the spatio-temporal stability of the foam.

An understanding of the underlying processes of foaming dynamics is a challenging task that is important for future technological progress. Theoretical models of drainage and coalescence are available in the literature [BR97, Ire09]. However, the adequacy of these models with real physical phenomena is still far from being completely established. A number of experimental studies for different variety of foams have been performed to investigate coalescence events, for example, using conductivity profiles, acoustic measurements, neutron and visible-light scattering [MEH⁺12]. Bubble collapses in transparent aqueous foams can also be captured using a visible-light system [RCAVAH03]). However, these techniques have their limitations and do not allow to detect and quantify coalescence with high spatial and temporal resolution.

X-ray imaging is particularly well-suited for the study of opaque materials. High-intensity synchrotron x-ray radioscopy was exploited to obtain real-time images of foaming metals [BSHB01].

The use of computed tomography already has proven to be a powerful tool to study the 3D structure of foams, with the primary focus on morphological and physical properties [HBP⁺05, RHB⁺09]. However, tomography imposes constraints on the temporal resolution due to the requirement of 3D data acquisition. Therefore, *in situ* 3D imaging could be realized only for sufficiently stable foams, in which structural changes occur on a time scale that is longer than the duration of a single tomography scan. Compared with the coarsening process due to gas diffusion between bubbles, coalescence is a much faster event and therefore requires higher data acquisition rates for an adequate process representation.

Sample environment (e.g. heat chambers, high-pressure devices) could also restrict the rotation of a sample during tomographic acquisition.

In this work we employ *in situ* synchrotron radiography to observe the evolution of foams with high temporal and spatial resolution [GMRH⁺08, RGMS⁺10]. To perform an automated and reliable quantification of coalescence events in radiographic images we develop a detection procedure based on optical flow methods.

6.2.1.2 Experimental Setup

The experiments were conducted at the ID19 beamline [WTB⁺10] of the European Synchrotron Radiation Facility (ESRF) in Grenoble, France. The details of a sample preparation and experimental conditions are given in Table 6.3.

Table 6.3: Experimental conditions and sample environment used for the studies of different foams at ID19 beamline [WTB⁺10]

Foam	Sample	X-ray energy	Detector pixel size	Frame rate	Conditions
Aqueous foam 6.2.1.9	Aqueous solution with 0.1% dishwasher liquid	17.7 keV	1.75 μm	3 Hz	Foaming in a test tube at room temperature
Metal foam 6.2.1.10	AlSi7 powder mixed with TiH ₂ (0.5 wt%)	33 keV	40 μm	2 Hz	Foaming in a steel mold, and furnace at T \simeq 998 K
Polymer foam 6.2.1.11	Polydimethylsiloxane (PDMS)	20 keV	2.8 μm	10 Hz	Foaming in a test tube at room temperature

To identify coalescence events, a data acquisition frame rate is an important aspect. In the current work, in order to implement a coalescence event detection procedure, we assume that we are able to follow the structural changes which are not related to a sudden film ruptures (e.g. foam expansion, coarsening, smooth displacements, etc). Thus, the coalescence events can be attributed to the discontinuous (untrackable) part of the foaming process.

6.2.1.3 Task for Data Processing

Perform estimation of coalescence event in metal foams during formation process using a sequence of radiographic images. Evaluate spatial and temporal distribution of coalescence events.

In order to perform estimation of coalescence events a previously described occlusion detection approach was applied. The following list of aims should be accomplished:

- Quantify a number of coalescence events within a single pair of subsequent images
- Evaluate quantitative distribution of coalescence events during foam evolution
- Perform analysis of spatial distribution of coalescence events
- Study different visualization and analysis approaches to get insights about overall foam stability: in both temporal and spatial domain.

6.2.1.4 Data Evaluation

Prior to any data processing steps we evaluate the metal foam dataset according to our data taxonomy 5.1:

- *Image noise*: The original radiograms contain low amount of noise.
- *Contrast level*: The image contrast is sufficient. It is possible to distinguish between different bubbles and identify borders. (See Figure 6.14).
- *Object size*: Foam bubbles are of different sizes, ranging from very large to very small. Additionally, an important structure to identify are bubbles boundaries, which are typically several pixels thick. This means that the objects sizes are mixed and we should pay additional attention to preserve such structures during image preprocessing and optical flow computation stages.
- *Object distribution*: Foam constituents are distributed densely, which complicates the task of bubbles identification and optical flow computation.
- *Object details*: Despite the fact, that the foam itself is highly textured, the individual bubbles are homogeneous and contain no details. The most characteristic feature of each bubble is its outline.
- *Image artifacts*: A number of saturated pixels, as well as artifacts caused by dust and scratches on the scintillator screen are present. That is why an artifact removal procedure is required.
- *Motion type*: The motion of foam and its constituents is non-rigid, non-linear, including expansion and contraction. During bubble collapses events a dramatic rearrange of foam bubbles may occur. Tracking of such events is the core idea behind our approach on coalescence events estimation (See next sections).
- *Motion range*: The arrangements (when linear) and foam expansion is a slow, smooth event. However, during coalescence events very large displacements are possible.

- *Motion discontinuities:* There are a lot of motion discontinuities between bubbles. During coalescence events the temporal continuity of object motion is highly violated.

A summary of evaluation of foams dataset is given in Table 6.4.

Table 6.4: Evaluation of foam datasets.

Image	Data	Motion
noise: average	size: mixed	type: non-rigid, non-linear
contrast: average	dist: dense	range: small
artifacts: low	detail: low	disc: high

Requirements on the Results

Accuracy: For this dataset we do not require accurate for optical flow estimation itself. As it is stated in the task for data processing (See Section 6.2.1.3), we aim to estimate the coalescence events instead.

Density: Dense flow fields are mandatory to capture differences temporal flow and recognise the shape of collapsed bubbles.

Motion components: In general, no flow components are required for the final result. However, the estimation technique employed in this work makes use of a full vector representation.

Consistency: The assumption, that the flow vectors are consistent in the forward and backward directions is the main idea behind our estimation approach.

Motion boundaries: We do not aim to capture strict flow boundaries of moving bubbles.

Computation time: Computation time is not an issue, the processing is done offline.

Data size: The original size of a single radiographic image is 2016×2016 . The processing is done sequentially and can be performed either on CPU or GPU.

6.2.1.5 Data Analysis: Preprocessing

In order to correct uneven brightness patterns and get rid of artifacts, caused by the dust and scratches on the scintillator screen, we perform flat- and dark-filed correction as described in Section 3.1.2.2. Further data pre-processing is not required, since the noise level and image contrast allow to proceed with the data analysis.

6.2.1.6 Data Analysis: Events Detection

The violation of data constancy assumption attributed to coalescence events results in errors in optical flow computation. The image features which cause this are treated

by conventional optical flow methods as data outliers which should be eliminated or suppressed, e.g. using robust data term 2.1.3.1. In our approach we employ the detection or evaluation of discontinuous features in optical flow field to capture coalescence events.

A naive approach for the event detection is to assume a slow motion of objects, such as pixels displacements u, v are small. Thus, a high value of difference between the successive image frames, which is greater than a predefined threshold:

$$|f(x, y, t + 1) - f(x, y, t)| > T,$$

indicates for the position x, y an occurrence of a coalescence event. The naive approach fails to discriminate between foam motion accompanied by geometrical distortions (e.g. expansion, coarsening) and bubble collapses, thus it is restricted to relatively static foams.

The naive detection model can be extended by applying a set of morphological operators on the result of difference image to filter structures presumably originated from foam's motion. In the work of [GMFB04] a similar approach was used to improve the detection of coalescence events by means of image difference. As a result, the amount of incorrectly detected events is reduced. However, this approach discriminates changing structures only based on their shape and size and still do not take into account motion, thus is prone to fail for fast movements within foam constituents.

The concept of a static or slowly changing foam can be extended by a more physically realistic assumption of linear motion. For a local image region it is assumed that objects can shift globally. The computed displacements $\Delta x, \Delta y$ are then used to perform a *motion-compensated difference*:

$$|f(x + \Delta x, y + \Delta y, t + 1) - f(x, y, t)| > T,$$

which indicates motion-aware temporal changes at the image location x, y . A popular choices for global motion estimation are correlation-based techniques. In general, simple implementations of correlation approach show only average performance. In order to cope with rotational motion, large pixel displacements, more sophisticated modifications of the correlation method are required.

An alternative method to deal with large displacements is based on Fourier analysis. According to the Fourier shift theorem the moving image regions have identical amplitude spectra. Exploiting this, a high difference value:

$$|FT\{f(x, y, t + 1)\}| - |FT\{f(x, y, t)\}| > T,$$

indicates the presence of a coalescence event. This approach was used by [MHB09]. The major drawback of the method is similar to that of correlation-based methods, i.e. non-linear distortions or motion are not taken into account, despite the fact that such events are highly probable owing to expansion and deformation of pores. This leads to mistaken attribution of these deformations/movements to coalescence events.

In this work for the motion computation we use variational optical flow methods, extensively discussed in Chapter 2. Two alternative approaches might be used to detect coalescence events: motion-compensated difference and forward-backward check 2.1.5.5. We propose to use a forwardbackward check for the construction of an improved coalescence event detector. For trackable (only motion, no film ruptures) image features the vector fields should be consistent in forward and backward direction. High values of the

forward-backward discrepancy vector are give a robust indicator revealing bubble collapses. The advantages of the proposed optical flow based approach are: a high accuracy of motion estimation, robustness against noise and small artifacts, and the possibility to select an appropriate model for the given imaging conditions and motion types.

6.2.1.7 Data Analysis: Motion Estimation

For the data term we choose brightness constancy assumption 2.1.2.2, since this constraint is highly violated during coalescence event. An important optical flow model setting is the integration of combined local-global approach 2.1.3.3 into a data term. This allow to reduce artifacts arising from superposition of internal foam structures on the radiographic image plane. A large integration scale parameter should be used in an optimal range of $\rho = [0.7...2.0]$. A crucial design aspect of our coalescence detector is that we intentionally choose a non-robust setting (quadratic) for data term over a robust one. In this way, optical flow model becomes more sensitive for violation of data constraints, which significantly improves the performance of the forwardbackward check method. In order to solve for large displacements a multi-level computation scheme is employed 2.2.5 according to baseline algorithm 5.3.3.

6.2.1.8 Data Analysis: Performance

To evaluate the performance of coalescence detection methods we consider these techniques as binary classifiers, which decide for each pixel location whether or not coalescence event happened. For each classifier we count hits and false alarm events (Wickens, 2001). With this approach we define the *true positive rate* (TPR, sensitivity) as:

$$TPR = TP/P,$$

where TP is the number of correctly attributed events normalized by the total area of coalescence event P , and the *false positive rate* (FPR, fall-out) as:

$$FPR = FP/N,$$

where FP is the number of mistakenly attributed events normalized by the total area of no coalescence N . To make quantitative performance analysis we employ a simple simulation model which takes into account foaming with linear, fast motion and expansion, accompanied with coalescence event. To implement it, we take a 3D tomographic image of a real metal foam and generate a radiographic projection using a ray-tracing algorithm (Figure 6.14). Then, the dataset was translated with a constant velocity and elastic deformation was performed to model foam expansion. A coalescence event is modelled by setting the material density to zero within a spherical region. A series of radiographic images, emulating temporal evolution of foams, is generated in this way.

The quantitative evaluation of several coalescence detectors is given in Figure 6.14. The curves represent the receiver operation characteristic (ROC) of a given method, which maps the false positive (fall-out) and true positive rates (sensitivity) for different thresholds parameters. The dashed line denotes the ROC curve for the worst possible detector, which classifies the events equivalently to a random guess with the uniform distribution.

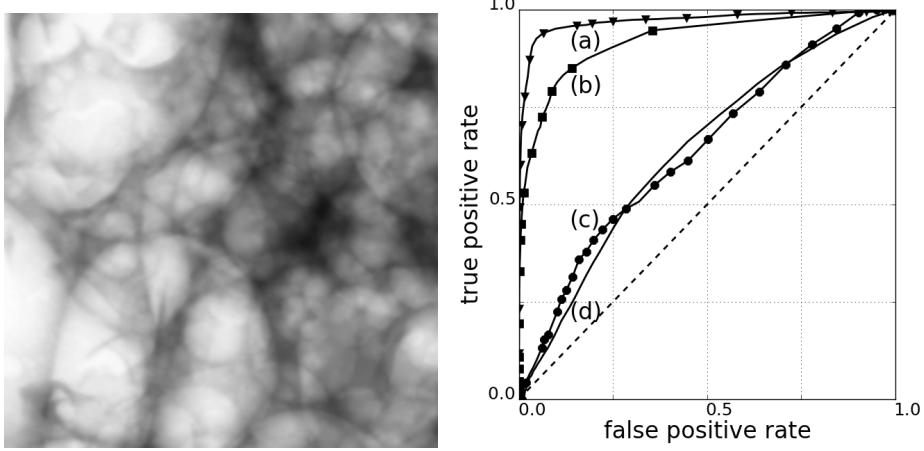


Figure 6.14: Example of ROC curves analysis of coalescence detection methods. **Left:** Simulated radiograph. **Right:** ROC curves on simulated data for the simple difference approach **(d)**, the difference approach extended with morphological operations **(c)**, the Fourier shift detection approach **(b)** and variational optical flow with the forward-backward check **(a)**.

For the comparison we check the performance of four coalescence detectors: (i) a simple difference approach, (ii) difference extended by morphological filtering, (iii) Fourier shift detection method [MHB09], and (iv) our proposed variational optical flow approach combined with a forward backward check.

Inspecting the ROC characteristics it is evident that both difference-based approaches are unable to provide high detection quality. As a single measure of method accuracy we may use an area under the ROC curve (AUC). For difference-based methods it does not attain 0.7, while based on the theory of signal detection the value 0.75 assumed to be a acceptable classifier [Swe88]. Motion-aware Fourier shift detection is much more reliable even for the sequence with non-linear distortions ($AUC = 0.93$). The proposed method, based on high accuracy optical flow based, significantly outperforms other methods. Its estimated AUC value is in the range of 0.97-0.98 for all input sequences, which is close to ideal detector.

In the next section we demonstrate a number of studies, in which optical flow combined with the forwardbackward check is used as an analysis tool to perform coalescence studies and to assess the stability of evolving foams. In particular, we consider aqueous, metal and polymer foam samples monitored by X-ray radiography.

6.2.1.9 Results: Aqueous Foams

The first application example of the devised optical flow detection technique is an automatic detection and registration of individual pore collapses. For our X-ray experiment on aqueous foams we generated a long radiographic sequence consisting of 500 frames and depicting the evolution of foam. Due to high stability of the sample only a few collapses occur throughout the whole sequence. On Figure 6.15 we present two typical cases of bubble rearrangements.

The top row of Figure 6.15(a)(e) corresponds to the case where only displacements

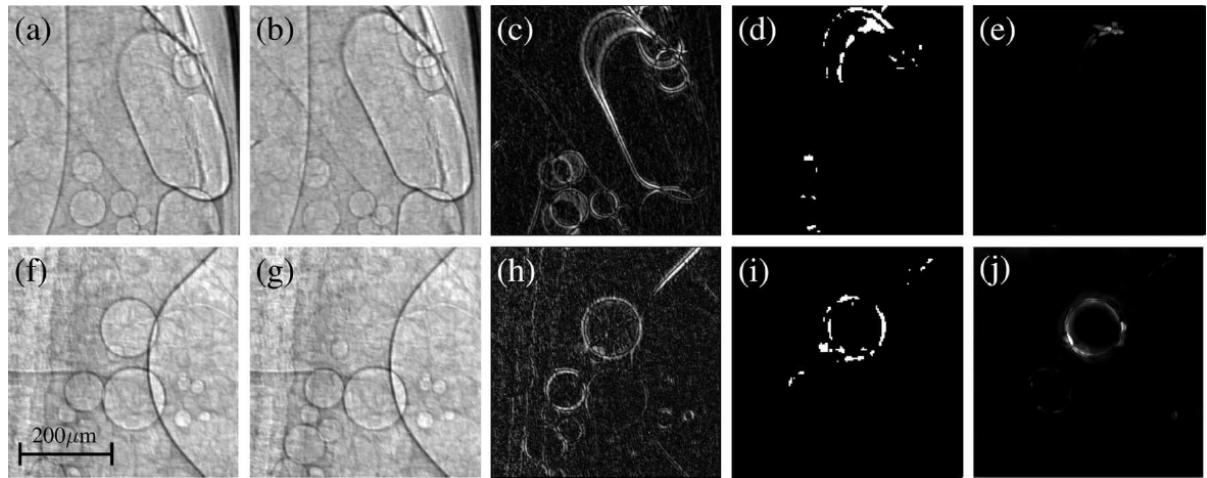


Figure 6.15: Detection of single coalescence events in aqueous foam. **Top Row:** Displacement of foam constituents, no coalescence events. **Bottom Row:** Collapse of a bubble combined with the motion of foam interior. (a, b) and (f, g): original successive radiographs; (c, h): difference images between the two successive frames; (d, i): coalescence maps produced by the Fourier shift detection; (e, j): coalescence maps produced by the optical flow based. Both difference images (c, h) contain strong motion-related artifacts, which are also partly present in (d, i), while the map (j) correctly identifies the underlying bubble collapse and gives no false detections

of foam constituents are present and no coalescence events happen. The images in the bottom row in Figure 6.15(f)-(j) show the collapse of a bubble combined with the motion of surrounding constituents between the successive frames Figure 6.15(f, g). The difference method in Figure 6.15(c, h) do not make the detection correctly since it produces a lot of false alarms because of structural motion and distortions. The detection method using Fourier analysis (Myagotin et al. 2009) performs better (Figure 6.15(d, i)), but still includes a number of artifacts. The developed approach is able to distinguish pore collapses from simple rearrangements in a much reliable manner (Figure 6.15(e, j)) This is also constituent with the results of performance evaluation using ROC curves analysis on synthetic dataset (see Section 6.2.1.8).

6.2.1.10 Results: Metal Foams

In this section we show *in situ* coalescence analysis of metal foaming and capture spatial distribution of bubble collapses, which allows to assess evolution of foams for different mold geometries. In the following experiment two baking cups made from steel are used. We select two mold geometries for which completely different foaming behavior is presumed - wedge-shaped and L-shaped molds, each 10 mm in depth. On Figure 6.16 a time-lapse sequences of foaming process and estimated velocity vector fields are shown. At the beginning samples expand mostly on the left-hand side producing bumps of semi-liquid material. This is a result of a non-uniform heating inside the molds. With the course of time the right-hand foam front overtakes that on the left-hand side. In the final expansion stage we observe a slow continuous flow of the foamed material oriented parallel to the inclined border in the wedge-shaped mold (Figure 6.16(d)) and along the cavity to the left

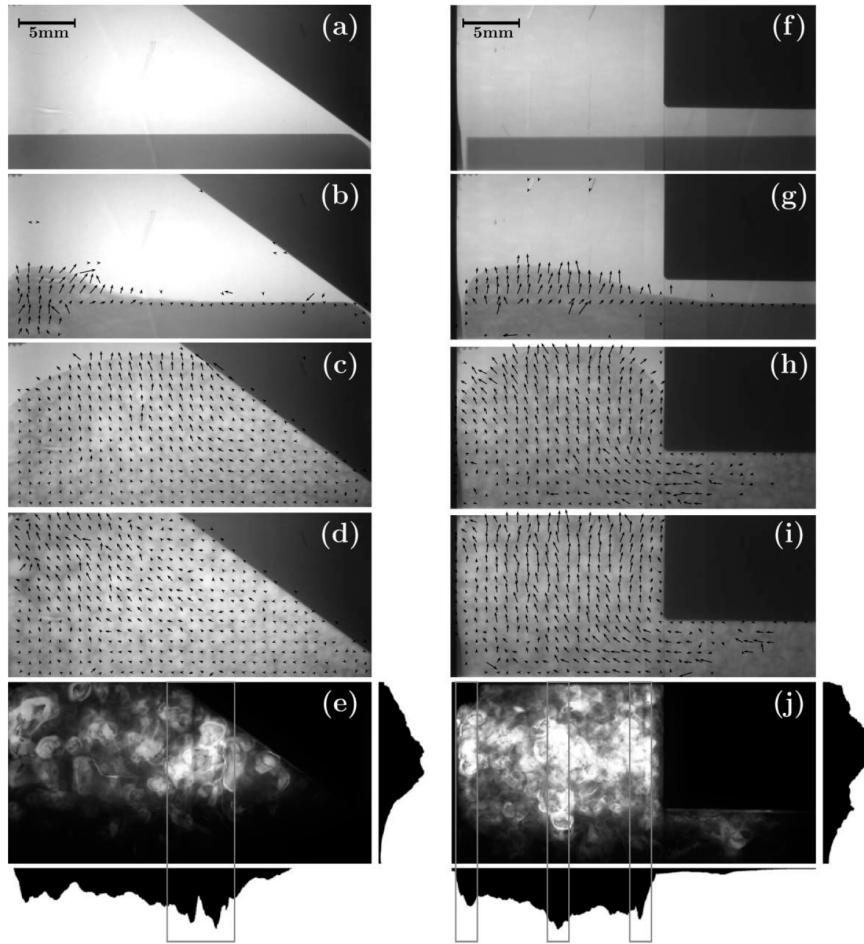


Figure 6.16: Estimated velocity fields for two metal foaming geometries at different expansion stages. **Left column:** foaming in a wedge-shaped mold. **Right column:** foaming in an L-shaped mold. Plots (e) and (j) show integral coalescence distributions. The vertical coalescence event profiles reveal a lower events fraction in the bottom foam layers, where film thinning is slowed down by a liquid material flow from upper layers.

and upwards in the L-shaped one (Figure 6.16(i)). As expected, foams in wedge-shaped and L-shaped molds behaved differently.

Now we investigate spatial distribution of coalescence events. For this purpose we constructed integral coalescence maps for both samples. The corresponding plots are demonstrated in Figure 6.16(e) and Figure 6.16(j). Additionally, the horizontal and vertical integral coalescence distribution histograms are plotted. A vertical event distribution reflects a smaller number of collapses on the bottom and a higher fraction of events in the middle part of samples, as to be expected. The downward liquid flows (induced mostly by gravitational forces) supply the bottom layers of the foam with liquid melt. This delays the thinning of the bottom foam films, and produces the vertical gradient of coalescence events. The estimated data are well in accordance with data estimated earlier (Babcsan et al., 2007).

The horizontal plots reveal an anomalous and unpredictable effect. One recognizes a high fraction of coalescence events on the right-hand side of the wedge-shaped sample

(Figure 6.16e)); in the plot (j) on Figure 6.16 we observe an increased fraction of coalescence events in the middle and on the left- and right-hand sides of the mold. The corresponding image regions are highlighted by the grey rectangles on Figure 6.16. The possible interpretation of this effect could be a low foam stability due to significant friction forces between the upward moving foam front and the steel mold. Friction forces promote film stretching and additional topological rearrangements of the foam films which destabilize the foam structure, especially at the molds faces. The important result of this observation is that already during the manufacturing stage of lightweight components one can predict the locations of possible detachments of the metal foam from the requested profile.

6.2.1.11 Results: Polymer Foams

A common receipt to improve the stability of foams is to include solid particles in the liquid phase. The particles are thought to increase the viscosity of the liquid and thus stabilize cell walls (Verdejo et al., 2009). In this section we analyze this technique by comparing unfilled and filled silicone polymer foams. To provide quantitative information about foam stability we analyze temporal distribution of coalescence events.

For the experiment foaming takes place at room temperature as an exothermic reaction. Foaming takes place at room temperature as an exothermic reaction. Aligned multiwalled carbon nanotubes (MWNT) were synthesized in-house by a chemical vapour deposition (CVD) technique. The carbon nanotubes were dispersed by high shear mixing in the polymethylhydrogensilane reactant. The two compounds were then mixed at a 1:1 ratio for 1 min. To study foam stability we consider two samples containing 0 and 0.2 wt% of nanotubes (MWNT), called hereafter CONT and CVD, respectively.

In order to quantify coalescence processes, a running average for the coalescence rate $\langle a \rangle(t)$ and integral coalescence fraction $A(t)$ are plotted for both samples in Figure 6.17.

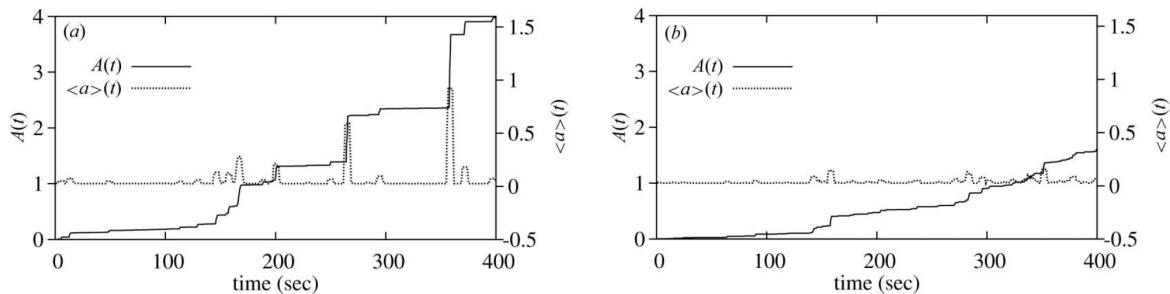


Figure 6.17: Coalescence rates for two kinds of polymer foams. **Left:** CONT sample (0 wt% of MWNT nanotubes). **Right:** CVD sample (0.2 wt% of MWNT nanotubes) (all characteristics are given in arbitrary units). A lower slope of the integral coalescence fraction $A(t)$ in the case of the CVD foam shows evidence of a higher stability of the foam filled with nanoparticles.

The former measure corresponds to the number of pixels covering detected coalescence area. The latter is the total coalescence rate $\langle a \rangle(t)$ integrated over a time period $[0, t]$. The formal definitions and interpretations for these characteristics are given in Myagotin et al. (2009). For both samples, already short after the onset of foaming the first coalescence events are detected which are represented by peaks on the coalescence rate $\langle a \rangle(t)$ plot.

With the course of time the two samples evolve differently. Exploring the plots, it becomes evident that in the case of CONT foam sample relatively rare events happen with rather high amplitude (large detected area) which indicates large coalescing bubbles.

The integral coalescence of the CVD sample, in contrast, is a more linearly increasing function, which shows an almost constant number of low-intensity film ruptures per unit time. Comparing the two samples one may notice that the CONT foam exhibits a significantly higher integral coalescence than CVD sample. These observations support the hypothesis that, owing to an increased viscosity of the liquid or decoration of the cell walls by the nanoparticles, capillary drainage in the filled sample is significantly reduced. This in turn leads to a reduced film thinning rate and to more stable liquid films separating the bubbles, and consequently to more stable foams.

By the systematic use of X-ray radiography combined with optical flow it is possible to analyze coalescence processes for a diversity of foam types. By introducing a concept of the forwardbackward check we are able to increase reliability of detection reliability and improve quantification of coalescence events. The presented approach is suitable for the detection and localization of individual film or bubble collapses, as well as for the estimation of spatial and temporal distributions of the coalescence events. These studies can provide insights about the underlying processes of foaming dynamics which is important for understanding of physics of foams and their production.

6.3 Tracking

6.3.1 Kinematics Analysis of Joint Parts in Living Insect

6.3.1.1 Introduction

Arthropods are the most numerous species and constitute more than 80% of all animals. Biologists have a major interest in studying their morphological diversity and related physiology. However, remarkably little is known about their functional morphology. The main reason is, until recently there has been no imaging method, which allows *in vivo* investigation of internal structures in 3D and real-time for non-transparent organisms such as most arthropods.

For optically opaque living organisms X-ray imaging methods are ideal techniques due to the high penetration ability of the radiation. For *static* specimens synchrotron-based X-ray computed microtomography (SR- μ CT) has proven to be a powerful tool to study arthropod morphology [WSL08]. To analyze *in vivo* dynamics of internal structures sequences of 2D X-ray projection radiographs were recorded [Ref 7,8]. However, in this case the depth information is lost, structures are superimposed, which complicates their identification and analysis.

Recently, a system for *in vivo* X-ray cine-tomography was designed, which aims to investigate previously inaccessible 3D morphological dynamics with feature sizes in the micron range and with temporal resolution down to a fraction of a millisecond [dSREvdKB14]. An important component of the X-ray 4D cine-tomography is not only an image acquisition system, realized by means of ultra-fast SR- μ CT, but also an automated data processing and motion analysis procedures. This combination allows to provide complete 4D quantitative information on the functional morphology of the living specimen.

6.3.1.2 Experimental Setup

Data acquisition was performed at the TOPO-TOMO beamline at the ANKA synchrotron facility operated by Karlsruhe Institute of Technology. A white X-ray beam was filtered by 0.2 mm aluminum plate giving a photon energy window of 9.6 - 24 keV with the maximum flux density at 14.5 keV.

The indirect X-ray detector system was optimized to achieve an effective pixel size of was 1.2 μ m, with a field of view of 2.464 mm. A camera system is a pco.dimax CMOS camera with 2016×2016 pixel resolution.

Static overview radiographs (Figure 6.18c) and reconstructed tomographic images (Figure 6.18d) were acquired using a detector system equipped with a Photron SA1.1 camera. The resulting pixel size was 6.6 μ m, with a field of view of 6.6 mm. These images were used to examine the insect's internal structures and identify location of the region of interest (screw-and-nut joint system, see arrows on Figure 6.18d).

To acquire *in vivo* tomograms of moving weevil and roughly estimate parameters of its motion, the specimen was imaged with a temporal resolution of 20 tomograms per second by continuous rotation with 10 revolutions per second. The camera recording frame rate was 5.000 frames per second and 250 projections were used per tomogram. The distance between sample to detector was 50 cm.

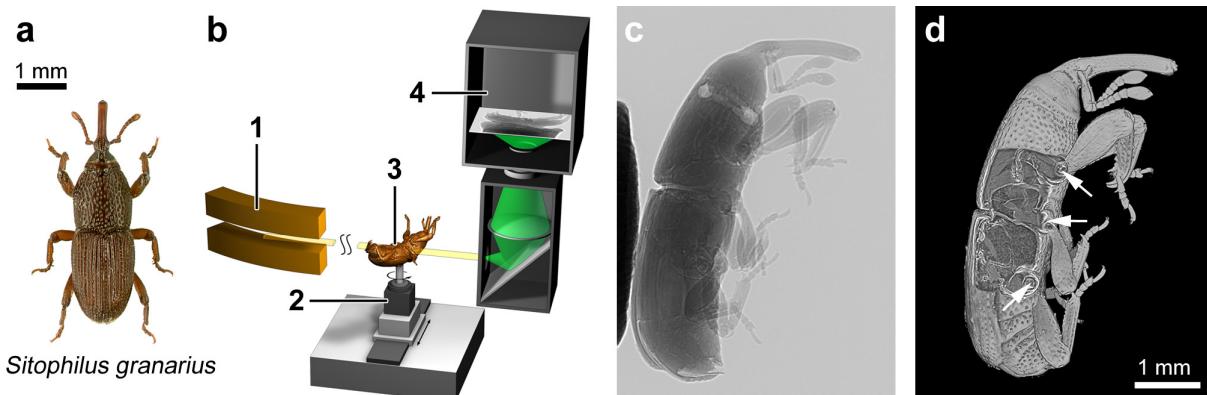


Figure 6.18: In vivo X-ray 4D cine-tomography experiment. (a) Photograph of *Sitophilus granarius*, dorsal view. (b) Experimental set-up for ultra-fast X-ray microtomography showing bending magnet (1), rotation stage (2), fixed specimen (3) and detector system (4). (c) Radiographic projection. (d) 3D rendering of the reconstructed volume with thorax cut open and revealing hip joints (arrows).

As the last level of our data acquisition pipeline we recorded high resolution radiographs of moving hip joint region (Figure 6.19a-c) while the specimen was continuously rotating with 3.25 revolutions per second. The images were recorded with a rate of 1500 frames per second. To acquire one tomogram we used 200 radiographs, instead of 250 for high-speed tomographic setup, the resulting temporal resolution was 7.5 tomograms per second. The sample to detector distance was 20 cm. A combination of these parameters was a good compromise between fast acquisition time and image resolution, which allowed us to study both structure and dynamics of the screw joint.

Additionally, after *in vivo* data acquisition, the same specimen was scanned *post mortem* (Figure 6.19d) to obtain high-quality static volumes of the joint region. For this case we increased the exposure time and 1000 projections were collected with a frame rate of 100 frames per second.

6.3.1.3 Data Evaluation

Before we proceed with a development of a data processing workflow, which is required to accomplish the task of quantitative estimation of kinematics in the joint system of a living bug, we evaluate the screw joint dataset according to our data taxonomy: 5.1:

- *Image noise*: The original radiograms contain high amount of noise due the short exposure time, which was necessary to limit radiation damage to the sample. After the tomographic reconstruction the amount of noise is reduced. However, the image data still suffers from the presence of noise. Therefore a dedicated preprocessing routine to perform noise reduction is required. Moreover, an optical flow model which is robust with respect to noise should be used.
- *Contrast level*: The image contrast is low, different structures of the joint system are hardly resolved between each other (See Figure 6.19). A proper treatment of low-contrast data is needed. The employed optical flow model should be adjusted to low-contrast data. The insufficient contrast is the major challenge of this dataset.

- *Object size:* The structures the joint system are of different sizes. In general, according to our data classification, they can be described as average.
- *Object distribution:* Different parts of the joint system are distributed normally.
- *Object details:* The amount of feature in various parts of inner structures is not sufficient. All parts are depicted as homogeneous structures (See Figure 6.19). Low amount of image details is the second main challenge of this dataset.
- *Image artifacts:* The main source of image artifacts is motion blur. Since during this *in vivo* experiment, the bug was moving, the acquisition parameters were adjusted in such a way that to achieve a compromise between dose, temporal resolution, image contrast and signal-to-noise ratio. As a result, temporal resolution was not sufficient to allow to follow fast movements in real time. After the image reconstruction motion-related artifacts emerge. In order to compute reliable optical flow the artifacts should be treated. Otherwise, the artifacts could obstruct the successful application of optical flow methods.
- *Motion type:* The motion of the screw joint and other structures is strictly rigid. The trochanter (an actual screw) undergoes fast rotation, as well as the translation within the coxa part (a box that contains the screw). Using the rigidity assumption we can design a special tracking procedure, which allows to overcome problem with image artifacts or insufficient amount of image details (See next sections).
- *Motion range:* Rotation of the trochanter structure is very fast, depending on a distance to the axis of rotation. However, the translational part of the movement is relatively slow. These aspects should be taken into account when modeling a motion estimation procedure.
- *Motion discontinuities:* The amount of motion discontinuities is normal and should not cause any problems for the optical flow estimation.

A summary of evaluation of screw joint dataset is given in Table 6.5.

Table 6.5: Evaluation of the screw joint dataset.

Image	Data	Motion
noise: high contrast: low artifacts: motion	size: average dist: norm detail: low	type: rigid range: mixed disc: norm

Requirements on the Results

Accuracy: For this dataset we require accurate results. Since the main task is to reconstruct the kinematics of the screw joint system over period of time, we require accuracy not only between two successive frames, but also for the whole motion cycle. The error in the range of 1-2 pixels would be sufficient for a pair of images, but exceeding values should be eliminated from the results or indicated. For the whole sequence the error of more than 5-7 pixels is already a substantial mistake, if the structure have displaced for

around 20 pixels. In other world the total error depends on the total displacement and should not exceed 25%.

Density: Dense flow fields are not required, since the main task is to reconstruct motion components (rotation and translation) during the movement. The fact that this requirement is relaxed gives the possibility to evaluate the computed flow field, filter unreliable flow vectors using internal confidence measures and perform analyses of flow components using a sparse flow field.

Motion components: The full flow is required, i.e. both the flow magnitude and the flow direction.

Consistency: The assumption, that the flow vectors are consistent in the forward and backward directions is the main idea behind our approach to filter unreliable flow vectors (see the description of data processing workflow).

Motion boundaries: We do not aim to capture strict flow on the boundaries of moving structures. Only general information about each structure is required - location, translation and rotation from the previous frame.

Computation time: Computation time is not an issue, the processing is done offline. Taking into account the size of tomographic datasets, it is advantageous to perform the computation on GPU.

Data size: The original size of a single tomogram is $2016 \times 2016 \times 2016$, which corresponds to 32 Gb of floating-point data. Each pair of the image sequence can be stored in the memory both on CPU and GPU.

6.3.1.4 Data Analysis: Preprocessing

Flat-field correction. Before tomographic reconstruction, the original radiographs were normalized by flat field correction 3.1.2.2.

Noise filtering. Due to the short exposure time during fast acquisition process the images are contaminated with high amount of noise. Prior to tomographic reconstruction we perform a noise filtering. Another challenge with respect to image quality is a low separation contrast between different structures. This should be taken into account for the design of filtering procedure. As an example, image denoising using the Gaussian low-pass filtering is a common approach (See section 3.1.1.2). However, the filter restores the pixel value by averaging its spatial neighborhood, thus blurring important image details. For this reason an edge-preserving filtering approach is desirable.

In the review of image denoising algorithms a non-local means algorithms (NLM) was presented [BCM05]. It was shown that NLM algorithm gives minimum mean-square error between the noisy and the uncorrupted image for signal-independent additive white noise. Additionally, in the comparison with other methods such as Gaussian filter, anisotropic filter and total variations (TV), NLM algorithm showed better performance.

However, as we discussed in section 3.1.1.2 for X-ray data the most influential noise

model is described by a Poisson distribution, and Poisson noise is not independent of the signal intensity, as required for NLM methods. For this reason we perform a noise variance equalizing transformation before noise filtering . After the noise removal by NLM algorithm, feature details are preserved and high-frequency noise is significantly decreased.

Tomographic reconstruction. 3D image reconstruction is performed using the filtered back-projection method (See section 1.3.2.2) using a Ramp filter [Buz08]. This approach differs from a conventional filtering during back projection procedure, when Shepp-Logan or the Hamming filters are used [Buz08]. Both filters allow to reduce noise by dumping high spatial frequencies. However, in our preprocessing procedure we use a separate noise filtering step, which allows us to employ a non-windowed filter function. As a consequence, image feature are better preserved, as required for motion estimation. The reconstructed 3D images can still be degraded by motion artifacts, since no motion-aware reconstruction was used. To tackle with the remaining artifacts a robust optical flow computation method is employed.

6.3.1.5 Data Analysis: Motion Computation

To capture dynamics of the screw-and-nut system we develop a robust 3D variational optical flow method. It is designed to be robust against image artifacts and yet be sensitive to low-contrast image details represented different structures. For the data term we use constancy of image brightness 2.1.2.2. Instead of using a common total variation (TV) approach to model the data term (robust approach 2.1.3.1), we employ quadratic penalization to enhance the contribution of image gradients (representing boundaries between structures), which were deliberately and carefully preserved throughout the entire image processing and reconstruction pipeline (see noise filtering and reconstruction steps). Since motion artifacts could still appear within image data, to discard them we selectively model data constancy using an automated confidence measure based on cross-checking 2.1.5.5. A first computation of optical flow is performed to localize data outliers and construct a confidence map for each pixel position. This confidence map is then applied during the second run of optical flow algorithm to reduce the contributions from problematic image areas and thus to refine the motion field 2.1.3.5. We further improve robustness of data term against noise using a combined localglobal approach 2.1.3.3, which takes in account an information from a local region. Large value of integration scale is used, with the optimal value in the range $\rho = [0.7 \dots 1.5]$. For the regularization of the motion field we use a combined image and flow-driven smoothness method 2.1.4.6. This approach imposes more strict smoothness constraints inside homogeneous image regions, but preserves motion boundaries near separate structures. Again, large values of smoothness parameters are required to give reliable results.

To capture the entire range of movements, including fast rotation of the trochanter, we use a multi-level computation procedure. The parameters are the same as for *baseline* model 5.3.3. Between computation levels we use median filtering 2.2.6 with large spatial mask ($m = 5$) to suppress data outliers from coarser levels.

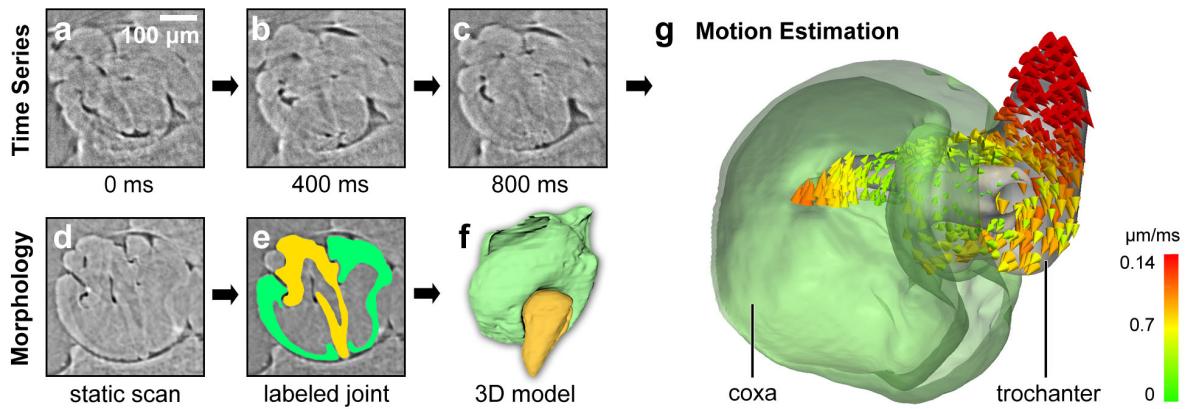


Figure 6.19: Morphological dynamics and kinematics analysis of the screw joint during defensive movement. (a-c) Time-lapse sequence of tomographic slices for times 0, 400 and 800 ms, respectively. (d) Tomographic slice of *post mortem* scan with increased exposure time and, as a result, image contrast. (e) Manual labeling of coxa (green) and trochanter (yellow). (f) 3D model of the screw joint based on manual labeling. (g) 3D motion field computed from 0 ms and 130 ms and represented as 3D glyphs colored according to magnitude

6.3.1.6 Data Analysis: Tracking

To perform a fully automated tracking of individual parts of the screw joint we use the results of optical flow computation on subsequent volume images. For the first frame of in vivo image sequence we automatically distribute 70 and 130 landmarks for coxa and trochanter, respectively. Part of the landmarks (50 %) were placed using a grid with equal spacing to uniformly cover both joint parts. The locations for rest of landmarks were selected based on image edges 3.2.2. For each landmark $L_t(\mathbf{x})$ at a time frame t , the corresponding landmark on time frame $t + 1$ is provided by the displacement vector $\mathbf{u}(\mathbf{x})$.

In order to improve accuracy of the tracking procedure we filter unreliable trajectories using a forward-backward cross-check 2.1.5.5. This method ensures consistency of tracking results computed in forward and reversed directions by evaluating the difference between the location of initial landmark and its position followed in opposite direction a possible errors could be identified. We eliminate data points with the discrepancy of forward-backward check of more than 3 pixels tracked throughout the entire sequence. As an additional constraint we perform rigidity check by analyzing relative spatial arrangement of landmarks 3.2.2.3. We filter out 30% of landmarks showing the most deviations from rigidity constraint. The remaining high-confidence landmarks are used to estimate global transformations of moving joint parts.

6.3.1.7 Data Analysis: Kinematics

For the estimation of global transformation between two sets of landmarks and extraction of motion components we use a dedicated Python module for matrix transformations <http://www.lfd.uci.edu/~gohlke/>. A transformation matrix M_t , which performs rigid transform of landmark positions between successive time frames, was calculated using Kabsch algorithm [Kab76] (See Section 3.2.3.3). The resulting matrix was then decom-

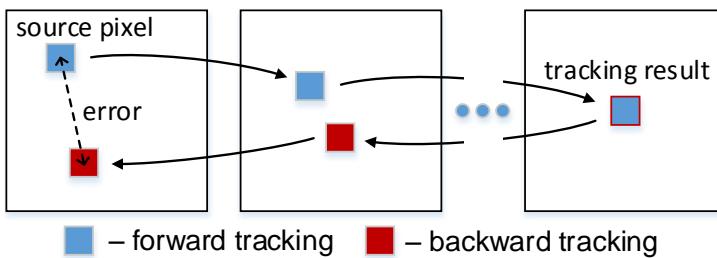


Figure 6.20: Cross-checking principle.

posed to extract translational and rotational components (Figure 6.21i).

For the trochanter part, displacement towards the coxa and rotation angles are evaluated with respect to its longer axis on the first time frame. To check the accuracy of automated tracking via motion analysis we compared it with a manual tracking performed by the expert. We achieved an average error for coxa displacement of 0.7 pixels with the average / maximum actual displacement of 3.7 / 5.2 pixels; an average error for trochanter displacement of 3.7 pixels with the average / maximum actual displacement of 10.7 / 18.0 pixels; an average error of trochanter rotation of 2.5 degrees with the average / maximum actual rotation of 55.4 / 94.8 degrees. Such performance in terms of accuracy was sufficient for a fully automated tracking procedure.

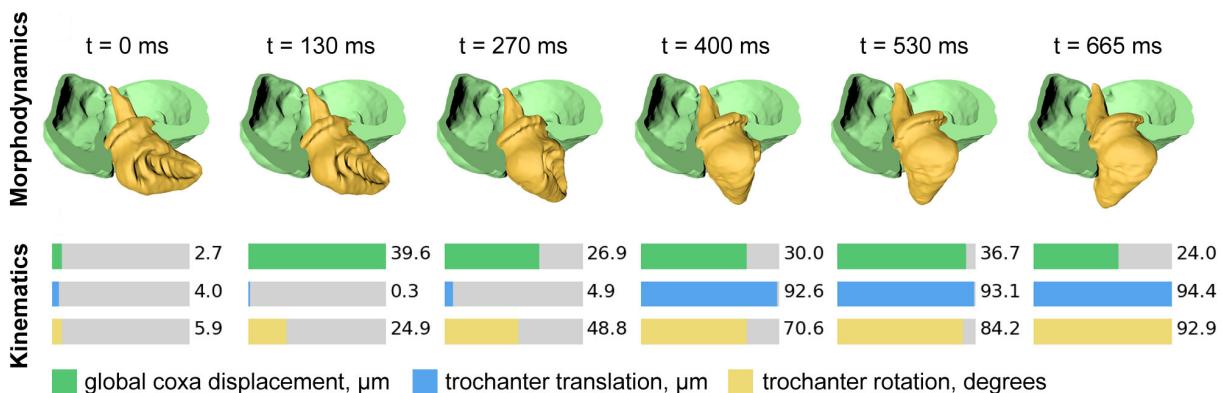


Figure 6.21: Morphological dynamics and kinematics analysis of the moving screw joint. **Top row:** *In vivo* dynamics of joint parts based on the 3D segmented static model 6.19f and automated tracking with motion analysis 6.19g. **Bottom row:** Analysis of kinematics. Global displacement of the whole screw-and-nut system with respect to the main body (green plot); fast translation of the trochanter inside the coxa (blue plot); linear rotation (screwing process) of the trochanter (yellow plot).

To visualize morphological parts in dynamics the 3D volumes of the static and the first frame of in-vivo scans were imported into the data analysis software Amira 5.4. Coxa and trochanter parts were then labeled manually. Next, coxa and trochanter from the static scan were aligned manually to match the first frame of the in vivo scan. The high quality labeled data from the static scan was then transformed according to its actual motion calculated during kinematics analysis from the previous step 6.3.1.7. As a result we obtain an *in vivo* quantitative morphological dynamics of moving joint parts.

6.3.1.8 Results

In this work we capture 4D spatio-temporal information about complex kinematics in living insect using optical flow and automated tracking, which allows us to visualize the functionality of internal structures (Figure 6.21). To determine quantitative characteristics of screwing process we separate the angular velocities from translational motion of trochanter inside the coxa. The results show that the trochanter rotated within 0.8 s around 92.9 degrees clockwise (from the view point of an outside observer) 6.21. The rotation was accompanied with an inward translation along the axis of rotation of 94.4 μm (Figure 6.21). The relation between rotary and translatory movements appears to be nonlinear, which can be a consequence of comparatively wide opening space observed between the trochanteral and coxal thread. Such type of motion is unlikely for the narrow hip joints described for *Trigonopterus oblongus* [vdKVBR11], as the defensive strategy of the genus relies on more narrow type of screw joints. Therefore, the current studies also provide an evidence for the functional variability of the constriction of leg joints in weevils, which can be closely related to their ecology.

A X-ray 4D cine-tomography has been proven to be a promising tool to study morphological dynamics in millimeter-sized animals such as insects and can be employed to investigate other biological specimens and processes.

Chapter 7

Summary and Outlook

7.1 Conclusions

In this work we investigated the application of optical flow methods in the field of X-ray Imaging. We conclude that optical flow methods are well suited for *automated* analysis of time-resolved X-ray data for a wide range of scientific problems. As a core technique we use the *variational* optical flow methods. A variety of existing models - both for the design of data terms and the flow field regularization, the robustness under noise and changing brightness conditions, the subpixel precision, the well-posedness of a solution and possibility to compute all types of motion, makes variational optical flow a flexible and an effective tool.

Depending on an application, an imaging technique, a data acquisition protocol and a type of the investigated process, the input X-ray data may be immensely diverse. That is why, in contrast to classical optical flow applications, where a visible light imaging is used to obtain a high-resolution, low-noise, color images, no single optical flow method can be universally and effectively used. Therefore, a dedicated motion estimation procedure should be developed and implemented for each application class. This procedure should take into account all properties related to the input data, the process of investigation and requirements on the results.

To get insights about the performance of each optical flow model for a particular imaging condition, we performed quantitative studies on the synthetic data. Based on an extensive evaluation of experimental data we learned a number of principles and aspects which are integral for the analysis of X-ray images:

- The gradient based approach, compared with the image brightness constancy, has an advantage of robustness under spatio-temporal brightness variations. However, it is reasonable to assume that derivatives-based gradient constancy assumption will be more sensitive to noise, than the data assumption on brightness values. Our performance analysis revealed that gradient constancy assumption in a robust setting provides better results than the grey value constancy even for large amounts of noise. This makes gradient based data term a superior model for most cases.
- Our experiments show that the robust data term provides worse performance for low-contrast data, than the quadratic terms. This is in contradiction to popular state-of-the-art optical flow models [BSL⁺11], which employ robust data term as a

default setting for most application scenarios. We conclude, that in the case of a low-contrast data a dedicated optical flow model should be employed. Furthermore, the preprocessing step to enhance the image contrast shows to be a useful strategy. However, a special care should be devoted to the noise properties of the input data, especially to such parameter as a contrast-to-noise ratio (CNR).

- The usage of quadratic data term can be beneficial to capture data outliers, if one is interested to recognize appearing/disappearing information. This can be useful for various applications such as analysis of bubble collapses in foams, crack propagation in materials or structures, and changes in other morphological properties.
- Usefulness of data preprocessing. For difficult image quality scenarios - high noise, low-contrast, presence of artifacts - it makes sense to augment the optical flow computation with a dedicated image preprocessing. Our experiments with the synthetic datasets clearly indicate this. Interestingly, in some cases, a simpler optical flow model with a highly optimized preprocessing step may provide better results than an advanced method with the explicit modeling of challenging data. This can be explained by a more efficient decoupling of interrelated problems - denoising, brightness correction and motion estimation. As a result, such decoupling may give more degrees of freedom for the optimization process.

To justify the choice of a processing workflow we provide a systematic data taxonomy on which we base the selection of the appropriate optical flow model.

Additional advantage of optical flow methods which we evaluated in this work is a possibility to use confidence measures to ensure the accuracy of results and to simplify optimization of model parameters. Such feature is crucial for practical applications, especially taking into account the variability of X-ray data.

On top of the optical flow techniques, we implemented an extensive framework for further analysis of optical flow results. Based on the result of optical flow we are able to perform motion analysis, object tracking, image registration and motion-based object recognition.

The implementation of the developed techniques incorporates advanced numerical schemes and computations on Graphical Processing Units (GPU). Thereby, the processing of a vast amount of X-ray data is feasible. This is especially important for analysis of 3D tomographic datasets.

After we establish methods for motion estimation and data analysis, and implement efficient computation routines we apply the devised techniques to a number of exemplary scientific problems from various research fields. These examples include flow analysis and particle segmentation in semi-solid alloys, analysis of morphogenesis in living frog embryos, coalescence events estimation and stability studies during the foaming process, and tracking of morphological dynamics in living insects. For all presented applications an automated, robust and accurate analysis of time-resolved data was central to get important insights about the given research topic. Therefore, we conclude that optical flow is a promising tool for automated data analysis, which can be effectively used for a wide range of other scientific problems in the fields of Medical Imaging, Material and Life Sciences, and Quality Engineering.

7.2 Further work

Despite the fact that nowadays the optical flow is the established field, there is still a plenty of space for further development. Here we describe a number of promising directions and aspects, which could improve the accuracy of optical flow methods and advance their applicability for challenging data, such as X-ray images.

Advanced optical flow techniques. Novel optical flow methods are constantly being developed in the field of Computer Vision, as well in the applied fields (e.g. Medical Image Processing). Some of the promising methods include - modeling temporal coherence [VBVZ11], motion and segmentation methods , optical flow based on superpixels [AMK13].

Modeling of motion blur. A proper modeling of motion blur within the optical flow framework could significantly improve the results and allow to perform quantitative motion analysis for faster processes and events, thus reducing the limiting factor of the maximum possible frame rate. Some works already propose a number of possibilities using a special optimization strategy [SB09].

Joint motion estimation and segmentation. If images can be segmented into coherently moving regions, many of the presented methods can be used to accurately estimate the flow within such regions. From the other hand, if the flow were accurately known, segmenting it into coherent regions would be feasible. Therefore, a joint motion estimation and image segmentation should be a major improvement for both tasks.

Motion in multilayers. The use of parametric models that estimate motion in layers [JB93, WA93, JBJ96] is a promising concept to properly apply optical flow on radiographic images, where 3D structures overlap due to the protective geometry.

Incorporation of geometrical constraints on motion. In the case when a sparse optical flow result is sufficient (e.g. for estimation of rigid motion), the use of geometrical constraints on a set of sparse landmarks could improve the robustness of the results. A similar approach was used in this work in the section 6.3.1. It was implemented as separate post-processing step. However, such procedure can be explicitly modelled within a variational optical flow framework.

Fusion of different contrast modalities. Multiple image assumptions might be useful in order to incorporate information obtained using different contrast mechanisms, for example an absorption contrast and contrast from phase- and dark- field imaging. Taking into account more physically meaningful information should enhance the credibility of the data term.

Learning optical flow methods. Instead of tuning parameters of the optical flow manually or using a brute-force search, one may use learning methods to train the optical flow on a set of training datasets. A number of such methods already been proposed [SLB08].

Combination of optic flow with ART reconstruction methods. Incorporate the problem

of dense optical flow estimation with algebraic reconstruction techniques [GBH70] into a single computational framework.

Data correction using motion-based in-painting. Recover missing information in radiographic images (e.g. corrupted by dust or scratches in the detector optics) using a motion based in-painting [WPUB11]

Database for medical and microscopy imaging and optical flow. Middlebury database for the evaluation of optical flow techniques has led to rapid improvements and understanding of optical flow methods [BSL⁺11]. A similar approach could be of great use for the field of Medical Imaging. Such work on for the X-ray data is ongoing.

More applications of optical flow methods. As we stated in the concluding section of our work, optical flow methods are general and well-suited for a number of scientific tasks in various research fields. Therefore, more application from the fields outside of the native Computer Vision are foreseen and expected to give useful results. The fields which can benefit from a thorough and systematic use of optical flow methods are Medical Imaging, Material Science and Quality Engineering.

Appendix A

List of Publications

1. K. Heim, A. Ershov, F. García-Moreno, A. Rack, J. Banhart. Investigation of particle movements and drainage in liquid metallic films by synchrotron radioscopy. *International Conference on Porous Metals and Metallic foams Technology, "Met-Foam 2015"* submitted (2015)
2. T. dos Santos Rolo, A. Ershov, T. van de Kamp, and T. Baumbach. In vivo X-ray cine-tomography for tracking morphological dynamics. *Proceedings of the National Academy of Sciences* 111(11):3921-3926 (2014)
3. J. Moosmann, A. Ershov, V. Weinhardt, T. Baumbach, M. Prasad, C. LaBonne, X. Xiao, J. Kashef, R. Hofmann. Time-lapse X-ray phase-contrast microtomography for in vivo imaging and analysis of morphogenesis. *Nature Protocols* 9 (2) , 294–304 (2014).
4. J. Moosmann, A. Ershov, V. Altapova, T. Baumbach, M. Prasad, C. LaBonne, X. Xiao, J. Kashef, R. Hofmann. X-ray phase-contrast in vivo microtomography probes new aspects of Xenopus gastrulation. *Nature* 497 (7449) , 374–377(2013).
5. T. van de Kamp, A. Ershov, T. dos Santos Rolo, A. Riedel, T. Baumbach. Insect imaging at the ANKA Synchrotron Radiation Facility. *Entomologie heute* 25, 147–160(2013).
6. S. Zabler, A. Ershov, A. Rack, F. García-Moreno, T. Baumbach, J. Banhart. Particle and liquid motion in semi-solid aluminium alloys: a quantitative in situ micro-radioscopy study. *Acta Materialia*, vol. 61, issue 4, 12441253 (2013)
7. A. Myagotin, A. Ershov, L. Helfen, R. Verdejo, A. Belyaev and T. Baumbach. Coalescence analysis for evolving foams via optical flow computation on projection image sequences. *Journal of Synchrotron Radiation*, 19, 483-491 (2012)
8. V. R. Altapova, A. Ershov, T. d. S. Rolo, E. Reznikova, J. Mohr, Yu. L. Pivovarov, V. F. Pichugin, G. T. Baumbach. Imaging Methods and Their Application at ANKA Light Source. *Journal of Surface Investigation: X-ray, Synchrotron and Neutron Techniques*. Volume 6, Number 3 394-397 (2012)

9. A. Rack, F. García-Moreno, C. Schmitt, O. Betz, A. Cecilia, A. Ershov, T. Rack, J. Banhart, S. Zabler. On the possibilities of hard X-ray imaging with high spatio-temporal resolution using polychromatic synchrotron radiation. *Journal of X-Ray Science and Technology*, vol. 18, issue 4, 429-441 (2010)
10. S. Zabler, A. Rack, F. García-Moreno, A. Ershov, T. Baumbach, J. Banhart. Th. Kannengiesser, S.S. Babu, Y.-I. Komizo, A.J. Ramirez (Eds.). Imaging fast processes in liquid metal foams and semi-solid alloys using synchrotron radioscopy with spatio-temporal micro-resolution in: *In-situ Studies with Photons, Neutrons and Electrons Scattering*, Springer, ISBN 978-3-642-14793-7, p. 149-158 (2010)
11. T. Baumbach, T. dos Santos Rolo, A. Ershov, L. Helfen, D. Lübbert, P. Modregger, D. Pelliccia, P. Vagovic, F. Xu. Moderne 2-D und 3-D abbildende Rntgenverfahren mit Synchrotronstrahlung. *MP Materials Testing* 51 10 (2009)
12. O. Betz, A. Rack, C. Schmitt, A. Ershov, A. Dieterich, L. Körner, D. Haas, T. Baumbach. High-Speed X-ray Cineradiography for Analyzing Complex Kinematics in Living Insects. *Synchrotron Radiation News* 21 (5), 34-38 (2008)

Bibliography

- [AER⁺12] V.R. Altapova, A. Ershov, T.d.S. Rolo, E. Reznikova, J. Mohr, Yu.L. Pivovarov, V.F. Pichugin, and G.T. Baumbach. Imaging methods and their application at the anka synchrotron light source. *Journal of Surface Investigation. X-ray, Synchrotron and Neutron Techniques*, 6(3):394–397, 2012.
- [AMK13] Fernando Amat, Eugene W. Myers, and Philipp J. Keller. Fast and robust optical flow for time-lapse microscopy using super-voxels. *Bioinformatics*, 29(3):373–380, 2013.
- [ASW99] Luis Alvarez, Javier Snchez, and Joachim Weickert. A scale-space approach to nonlocal optical flow calculations. In Mads Nielsen, Peter Johansen, OleFogh Olsen, and Joachim Weickert, editors, *Scale-Space Theories in Computer Vision*, volume 1682 of *Lecture Notes in Computer Science*, pages 235–246. Springer Berlin Heidelberg, 1999.
- [AVBSZ07] Y. Adato, Y. Vasilyev, O. Ben-Shahar, and T. Zickler. Toward a theory of shape from specular flow. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.
- [BA91] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 296–302, 1991.
- [BA96] Michael J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75 – 104, 1996.
- [BBH03] M.Z. Brown, D. Burschka, and G.D. Hager. Advances in computational stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(8):993–1008, 2003.
- [BBPW04] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optic flow estimation based on a theory for warping. In T. Pajdla and J. Matas, editors, *Computer Vision – ECCV 2004*, volume 3024 of *Lecture Notes in Computer Science*, pages 25–36. Springer, Berlin, 2004.
- [BCM05] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

- [BD96] Folkmar A Bornemann and Peter Deuflhard. The cascadic multigrid method for elliptic problems. *Numerische Mathematik*, 75(2):135–152, 1996.
- [BFB94] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *Int. J. Comput. Vision*, 12(1):43–77, February 1994.
- [BJ96] Michael J. Black and Allan D. Jepson. Estimating optical flow in segmented images using variable-order parametric models with local deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):972–986, 1996.
- [BJKY14] Linchao Bao, Hailin Jin, Byungmoon Kim, and Qingxiong Yang. A comparison of tv-l1 optical flow solvers on gpu. In *GPU Technology Conference (GTC)*. NVIDIA, 2014.
- [BK89] Ruzena Bajcsy and Stane Kovačič. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 46(1):1 – 21, 1989.
- [BKD01] Kevin Bowyer, Christine Kranenburg, and Sean Dougherty. Edge detector evaluation using empirical roc curves. *Computer Vision and Image Understanding*, 84(1):77 – 103, 2001.
- [BL97] J L Barron and A Liptay. Measuring 3-d plant growth using optical flow. *Bioimaging*, 5(2):82–86, 1997.
- [BPT88] M. Bertero, T.A. Poggio, and V. Torre. Ill-posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–889, 1988.
- [BR97] A. Bhakta and E. Ruckenstein. Drainage and coalescence in standing foams. *Journal of Colloid and Interface Science*, 191:184–201, 1997.
- [BRE⁺09] T. Baumbach, T. D. S. Rolo, A. Ershov, L. Helfen, D. Lübbert, P. Modregger, D. Pelliccia, P. Vagovic, and F. Xu. Moderne 2-d und 3-d abbildende rntgenverfahren mit synchrotronstrahlung. *Testing, Material-pruefung/Materials*, 51:642–651, 2009.
- [Bro92] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM Comput. Surv.*, 24(4):325–376, December 1992.
- [BRS⁺08] O. Betz, A. Rack, C. Schmitt, A. Ershov, A. Dieterich, L. Körner, D. Haas, and T. Baumbach. High-speed x-ray cineradiography for analyzing complex kinematics in living insects. *Synchrotron Radiation News*, 21(5):34–38, 2008.
- [Bru06] A. Bruhn. *Variational Optic Flow Computation – Accurate Modelling and Efficient Numerics*. PhD thesis, Department of Mathematics and Comptuer Science, Saarland University, Germany, July 2006.
- [BSHB01] J. Banhart, H. Stanzick, L. Helfen, and T. Baumbach. Metal foam evolution studied by synchrotron radioscopy. *Applied Physical Letters*, 78:1152–1154, 2001.

- [BSL⁺11] Simon Baker, Daniel Scharstein, J.P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011.
- [Buz08] Thorsten M. Buzug. *Computed tomography from photon statistics to modern cone-beam CT*. Springer, Berlin, 2008.
- [BW05] A. Bruhn and J. Weickert. Towards ultimate motion estimation: combining highest accuracy with real-time performance. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 749–755 Vol. 1, 2005.
- [BW06] A. Bruhn and J. Weickert. A confidence measure for variational optic flow methods. In Reinhard Klette, Ryszard Kozera, Lyle Noakes, and Joachim Weickert, editors, *Geometric Properties for Incomplete data*, volume 31, pages 283–298. Springer Netherlands, 2006.
- [BWF⁺05] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr. Variational optical flow computation in real time. *Image Processing, IEEE Transactions on*, 14(5):608–615, 2005.
- [BWKS06a] Andrés Bruhn, Joachim Weickert, Timo Kohlberger, and Christoph Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *International Journal of Computer Vision*, 70(3):257–277, 2006.
- [BWKS06b] Andrés Bruhn, Joachim Weickert, Timo Kohlberger, and Christoph Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *International Journal of Computer Vision*, 70(3):257–277, 2006.
- [BWS02] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Combining the advantages of local and global optic flow methods. In Luc Gool, editor, *Pattern Recognition*, volume 2449 of *Lecture Notes in Computer Science*, pages 454–462. Springer Berlin Heidelberg, 2002.
- [BWS05] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.
- [Can86] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679–698, November 1986.
- [CBFAB94] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 2, pages 168–172 vol.2, November 1994.

- [CCK91] C. Chang, S. Chatterjee, and P.R. Kube. On an analysis of static occlusion in stereo vision. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 722–723, 1991.
- [CKS95] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 694–699, June 1995.
- [CLB⁺99] Peter Cloetens, Wolfgang Ludwig, Jos Baruchel, Jean-Pierre Guigay, Petra Pernot-Rejmankov, Murielle Salom-Pateyron, Michel Schlenker, Jean-Yves Buffire, Eric Maire, and Gilles Peix. Hard x-ray phase imaging using simple propagation of a coherent synchrotron radiation beam. *Journal of Physics D: Applied Physics*, 32(10A):A145, 1999.
- [Coo67] S. A. Coons. Surfaces for computer-aided design of space forms. Technical report, Cambridge, MA, USA, 1967.
- [Dou09] Geoff Dougherty. *Digital image processing for medical applications*. Cambridge University Press, Cambridge, UK; New York, 2009.
- [dSREvdKB14] Tomy dos Santos Rolo, Alexey Ershov, Thomas van de Kamp, and Tilo Baumbach. In vivo x-ray cine-tomography for tracking morphological dynamics. *Proceedings of the National Academy of Sciences*, 111(11):3921–3926, 2014.
- [Els62] Lev Ernestovich Elsgolts. *Calculus of variations*. Pergamon Press, 1962.
- [Ers08] A. Ershov. *Coalescence Rate Estimation using Optical Flow and Occlusion Detection*. Master thesis, Mathematical Image Analysis Group, Computer Science Department, Saarland University, Germany, March 2008.
- [Fea03] Bernd Fischer and et al. Combination of automatic non-rigid and landmark based registration: the best of both worlds, 2003.
- [FJ90] David J. Fleet and Allan D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990.
- [Fle91] Merton C. Flemings. Behavior of metal alloys in the semisolid state. *Metallurgical Transactions A*, 22(5):957–981, 1991.
- [Fua93] Pascal Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49, 1993.
- [GBH70] Richard Gordon, Robert Bender, and Gabor T Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of theoretical Biology*, 29(3):471–481, 1970.

- [GET01] S.M. Gruner, E.F. Eikenberry, and M.W. Tate. Comparison of x-ray detectors. In M.G. Rossmann and E. Arnold, editors, *International Tables for Crystallography Volume F: Crystallography of biological macromolecules*, volume F of *International Tables for Crystallography*, pages 143–147. Springer Netherlands, 2001.
- [GLU12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [GMFB04] F. García-Moreno, M. Fromme, and J. Banhart. Real-time x-ray radioscopy on metallic foams using a compact micro-focus source. *Advanced Engineering Materials*, 6(6):416–420, 2004.
- [GMRH⁺08] F. García-Moreno, A. Rack, L. Helfen, T. Baumbach, S. Zabler, N. Babcsán, J. Banhart, T. Martin, C. Ponchut, and M. Di Michiel. Fast processes in liquid metal foams investigated by high-speed synchrotron x-ray microradioscopy. *Applied Physics Letters*, 92(13):–, 2008.
- [GU98] G.J. Grevera and J.K. Udupa. An objective comparison of 3-d image interpolation methods. *Medical Imaging, IEEE Transactions on*, 17(4):642–652, August 1998.
- [GW08] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2008.
- [GZG⁺12] Pascal Gwosdek, Henning Zimmer, Sven Grewenig, Andrés Bruhn, and Joachim Weickert. A highly efficient gpu implementation for variational optic flow based on the euler-lagrange framework. In KiriakosN. Kutulakos, editor, *Trends and Topics in Computer Vision*, volume 6554 of *Lecture Notes in Computer Science*, pages 372–383. Springer Berlin Heidelberg, 2012.
- [HBP⁺05] L. Helfen, T. Baumbach, P. Pernot, P. Cloetens, H. Stanzick, K. Schladitz, and J. Banhart. Investigation of pore initiation in metal foams by synchrotron-radiation tomography. *Applied Physics Letters*, 86(23):–, 2005.
- [HS81] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [HS09] J Huisken and D Y Stainier. Selective plane illumination microscopy techniques in developmental biology. *Development*, 136(12):1963–1975, June 2009.
- [HSSW01] W. Hinterberger, O. Scherzer, C. Schnörr, and J. Weickert. Analysis of optical flow models in the framework of calculus of variations. Technical Report 8/2001, Computer Science Series, University of Mannheim, Germany, April 2001.

- [HTWM04] Weiming Hu, Tieniu Tan, Liang Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(3):334–352, 2004.
- [Ire09] Peter M. Ireland. Coalescence in a steady-state rising foam. *Chemical Engineering Science*, 64(23):4866 – 4874, 2009.
- [JB93] A. Jepson and M.J. Black. Mixture models for optical flow computation. In *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR '93., 1993 IEEE Computer Society Conference on*, pages 760–761, 1993.
- [JBJ96] S.X. Ju, M.J. Black, and A.D. Jepson. Skin and bones: multi-layer, locally affine, optical flow and regularization with transparency. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR '96, 1996 IEEE Computer Society Conference on*, pages 307–314, 1996.
- [Kab76] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.
- [Kar15] I. Karlinskiy. *Fast GPU-based 3D optical flow with interactive visualization*. Master thesis, Institute for Visualization and Data Analysis (IVD), Department of Informatics, Karlsruhe Institute of Technology, Germany, September 2015.
- [KDS03] R Keller, L A Davidson, and D R Shook. How we are shaped: the biomechanics of gastrulation. *Differentiation*, 71(3):171–205, April 2003.
- [KKK⁺09] J Kashef, A Köhler, S Kuriyama, D Alfandari, R Mayor, and D Wedlich. Cadherin-11 regulates protrusive activity in xenopus cranial neural crest cells upstream of trio and the small gtpases. *Genes Dev*, 23(12):1393–1398, June 2009.
- [KS01] Avinash C. Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, Philadelphia, 2001.
- [KSWS08] P J Keller, A D Schmidt, J Wittbrodt, and E H Stelzer. Reconstruction of zebrafish early embryonic development by scanned light sheet microscopy. *Science*, 322(5904):1065–1069, November 2008.
- [KZK03] V Kastrinaki, M Zervakis, and K Kalaitzakis. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4):359 – 381, 2003.
- [LK81] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*, IJCAI'81, pages 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

- [Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [LRR08] Victor Lempitsky, Stefan Roth, and Carsten Rother. Fusionflow: Discrete-continuous optimization for optical flow estimation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.
- [LV98] Shang-Hong Lai and BabaC. Vemuri. Reliable and efficient computation of optical flow. *International Journal of Computer Vision*, 29(2):87–105, 1998.
- [MB87] D. W. Murray and B. F. Buxton. Scene segmentation from visual motion using global optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(2):220–228, February 1987.
- [MBW07] Yana Mileva, Andrés Bruhn, and Joachim Weickert. Illumination-robust variational optical flow with photometric invariants. In FredA. Hamprecht, Christoph Schnrr, and Bernd Jhne, editors, *Pattern Recognition*, volume 4713 of *Lecture Notes in Computer Science*, pages 152–162. Springer Berlin Heidelberg, 2007.
- [MEA⁺13] Julian Moosmann, Alexey Ershov, Venera Altapova, Tilo Baumbach, Maneeshi S. Prasad, Carole LaBonne, Xianghui Xiao, Jubin Kashef, and Ralf Hofmann. X-ray phase-contrast *in vivo* microtomography probes new aspects of *xenopus* gastrulation. *Nature*, 497(7449):374–377, May 2013.
- [MEH⁺12] A Myagotin, A Ershov, L Helfen, R Verdejo, A Belyaev, and T Baumbach. Coalescence analysis for evolving foams via optical flow computation on projection image sequences. *J Synchrotron Radiat*, 19(Pt 4):483–491, July 2012.
- [MEW⁺14] Julian Moosmann, Alexey Ershov, Venera Weinhardt, Tilo Baumbach, Maneeshi S Prasad, Carole LaBonne, Xianghui Xiao, Jubin Kashef, and Ralf Hofmann. Time-lapse x-ray phase-contrast microtomography for *in vivo* imaging and analysis of morphogenesis. *Nat. Protocols*, 9(2):294–304, February 2014.
- [MHB09] A Myagotin, L Helfen, and T Baumbach. Coalescence measurements for evolving foams monitored by real-time projection imaging. *Measurement Science and Technology*, 20(5):055703, 2009.
- [MHSK13] J.R. McClelland, D.J. Hawkes, T. Schaeffter, and A.P. King. Respiratory motion models: A review. *Medical Image Analysis*, 17(1):19 – 42, 2013.
- [MP98] E. Memin and P. Perez. A multigrid approach for hierarchical motion estimation. In *Computer Vision, 1998. Sixth International Conference on*, pages 933–938, 1998.

- [Nag83] Hans Helmut Nagel. Constraints for the estimation of displacement vector fields from image sequences. In *In International Joint Conference on Artificial Intelligence*, pages 945–951, 1983.
- [Nag90] Hans-Hellmut Nagel. Extending the oriented smoothness constraint into the temporal domain and the estimation of derivatives of optical flow. In O. Faugeras, editor, *Computer Vision ? ECCV 90*, volume 427 of *Lecture Notes in Computer Science*, pages 139–148. Springer Berlin Heidelberg, 1990.
- [NE86] Hans-Hellmut Nagel and Wilfried Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(5):565–593, 1986.
- [NHN87] Ari Nieminen, P. Heinonen, and Y. Neuvo. A new class of detail-preserving filters for image processing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-9(1):74–90, January 1987.
- [NVia] NVidia. Cuda toolkit documentation.
- [NVib] NVidia. Nvidia developer zone.
- [Nyq28] Harry Nyquist. Certain topics in telegraph transmission theory. *American Institute of Electrical Engineers, Transactions of the*, 47(2):617–644, 1928.
- [ON94] M. Otte and H.-H. Nagel. Optical flow estimation: Advances and comparisons. In Jan-Olof Eklundh, editor, *Computer Vision ? ECCV '94*, volume 800 of *Lecture Notes in Computer Science*, pages 49–60. Springer Berlin Heidelberg, 1994.
- [PAA⁺87] Stephen M. Pizer, E. Philip Amburn, John D. Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart Ter Haar Romeny, and John B. Zimmerman. Adaptive histogram equalization and its variations. *Comput. Vision Graph. Image Process.*, 39(3):355–368, September 1987.
- [PBB⁺06] Nils Papenberg, Andrés Bruhn, Thomas Brox, Stephan Didas, and Joachim Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int. J. Comput. Vision*, 67(2):141–158, April 2006.
- [PBV⁺07] C Papan, B Boulat, S S Velan, S E Fraser, and R E Jacobs. Two-dimensional and three-dimensional time-lapse microscopic magnetic resonance imaging of xenopus gastrulation movements using intrinsic tissue-specific contrast. *Dev Dyn*, 236(2):494–501, February 2007.
- [PKT83] J. Anthony Parker, Robert V. Kenyon, and D. Troxel. Comparison of interpolating methods for image resampling. *Medical Imaging, IEEE Transactions on*, 2(1):31–39, March 1983.

- [PM90] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(7):629–639, July 1990.
- [PN98] D. Paganin and K. A. Nugent. Noninterferometric phase imaging with partially coherent light. *Phys. Rev. Lett.*, 80(12):2586–2589, March 1998.
- [R96] Wilhelm Röntgen. On a New Kind of Rays. *Nature*, 53:274–276, January 1896.
- [Rad86] Johann Radon. On the determination of functions from their integral values along certain manifolds. *Medical Imaging, IEEE Transactions on*, 5(4):170–176, December 1986.
- [RCAVAH03] Florence Rouyer, Sylvie Cohen-Addad, Michèle Vignes-Adler, and Reinhard Höhler. Dynamics of yielding observed in a three-dimensional aqueous dry foam. *Phys. Rev. E*, 67(2):021405, February 2003.
- [RGMS⁺10] A Rack, F García-Moreno, C Schmitt, O Betz, A Cecilia, A Ershov, T Rack, J Banhart, and S Zabler. On the possibilities of hard x-ray imaging with high spatio-temporal resolution using polychromatic synchrotron radiation. *J Xray Sci Technol*, 18(4):429–441, 2010.
- [RHB⁺09] A. Rack, H.-M. Helwig, A. Btow, A. Rueda, B. Matijaevi-Lux, L. Helfen, J. Goebels, and J. Banhart. Early pore formation in aluminium foams studied by synchrotron-based microtomography and 3-d image analysis. *Acta Materialia*, 57(16):4809 – 4821, 2009.
- [ROF92] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [Rus11] John C. Russ. *The image processing handbook*. CRC Press, Boca Raton, 2011.
- [Saa03] Yousef Saad. *Iterative methods for sparse linear systems*. Siam, 2003.
- [SACF⁺12] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, Jean-Yves Tinevez, Daniel James White, Volker Hartenstein, Kevin Eliceiri, Pavel Tomancak, and Albert Cardona. Fiji: an open-source platform for biological-image analysis. *Nat Meth*, 9(7):676–682, July 2012.
- [SAH91] E.P. Simoncelli, E.H. Adelson, and D.J. Heeger. Probability distributions of optical flow. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 310–315, 1991.
- [SB09] Steven M. Seitz and Simon Baker. Filter flow. In *In Proceedings of the IEEE International Conference on Computer Vision*, 2009.

- [Sch93] C. Schnörr. On functionals with greyvalue-controlled smoothness terms for determining optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1074–1079, 1993.
- [Sch94] C. Schnörr. Segmentation of visual motion by minimizing convex non-quadratic functionals. In *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision and Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 661–663 vol.1, 1994.
- [SH89] D. Shulman and J.-Y. Herve. Regularization of discontinuous flow fields. In *Visual Motion, 1989., Proceedings. Workshop on*, pages 81–86, 1989.
- [SL96] G. Sudhir and John C.M. Lee. Video annotation by motion interpretation using optical flow streams. *Journal of Visual Communication and Image Representation*, 7(4):354 – 368, 1996.
- [SLB08] Deqing Sun, J. P. Lewis, and Michael J. Black. Learning optical flow. In *In Proc. ECCV*, pages 83–97, 2008.
- [SMF72] D.B. Spencer, R. Mehrabian, and M.C. Flemings. Rheological behavior of sn-15 pct pb in the crystallization range. *Metallurgical Transactions*, 3(7):1925–1932, 1972.
- [SRB10] Deqing Sun, S. Roth, and M.J. Black. Secrets of optical flow estimation and their principles. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2432–2439, 2010.
- [SRB14] Deqing Sun, Stefan Roth, and MichaelJ. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014.
- [SRE12] Caroline A Schneider, Wayne S Rasband, and Kevin W Eliceiri. NIH image to imagej: 25 years of image analysis. *Nat Meth*, 9(7):671–675, July 2012.
- [SS02] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.
- [ST82] Klaus Stüben and Ulrich Trottenberg. *Multigrid methods: Fundamental algorithms, model problem analysis and applications*. Springer, 1982.
- [Swe88] J A Swets. Measuring the accuracy of diagnostic systems. *Science*, 240(4857):1285–1293, June 1988.
- [TBKP12] Michael Tao, Jiamin Bai, Pushmeet Kohli, and Sylvain Paris. Simpleflow: A non-iterative, sublinear optical flow algorithm. *Computer Graphics Forum*, 31(2pt1):345–353, 2012.
- [TD05] D Tschumperlé and R Deriche. Vector-valued image regularization with pdes: a common framework for different applications. *IEEE Trans Pattern Anal Mach Intell*, 27(4):506–517, April 2005.

- [TM98] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, January 1998.
- [TP06] E. Trucco and K. Plakas. Video tracking: A concise survey. *Oceanic Engineering, IEEE Journal of*, 31(2):520–529, April 2006.
- [UGVT88] S. Uras, F. Girosi, A. Verri, and V. Torre. A computational approach to motion perception. *Biological Cybernetics*, 60(2):79–87, 1988.
- [VBVZ11] S. Volz, A. Bruhn, L. Valgaerts, and H. Zimmer. Modeling temporal coherence for optical flow. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1116–1123, November 2011.
- [vdKEdSR⁺13a] Thomas van de Kamp, Alexey Ershov, Tomy dos Santos Rolo, Alexander Riedel, and Tilo Baumbach. Insect imaging at the anka synchrotron radiation facility. *Entomologie heute*, 25:147–160, 2013.
- [vdKEdSR⁺13b] Thomas van de Kamp, Alexey Ershov, Tomy dos Santos Rolo, Alexander Riedel, and Tilo Baumbach. Insect imaging at the anka synchrotron radiation facility. *Entomologie heute*, 25:147–160, 2013.
- [vdKVBR11] T van de Kamp, P Vagovi, T Baumbach, and A Riedel. A biological screw in a beetle’s leg. *Science*, 333(6038):52–52, July 2011.
- [vdWG04] J. van de Weijer and T. Gevers. Robust optical flow from photometric invariants. In *Image Processing, 2004. ICIP ’04. 2004 International Conference on*, volume 3, pages 1835–1838 Vol. 3, 2004.
- [VLW07] Peter Vennemann, Ralph Lindken, and Jerry Westerweel. In vivo whole-field blood velocity measurement techniques. *Experiments in Fluids*, 42(4):495–511, 2007.
- [WA93] J.Y.A. Wang and E.H. Adelson. Layered representation for motion analysis. In *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR ’93., 1993 IEEE Computer Society Conference on*, pages 361–366, 1993.
- [Wei98] Joachim Weickert. *Anisotropic diffusion in image processing*, volume 1. Teubner Stuttgart, 1998.
- [Wol96] W. Wolf. Key frame selection by motion analysis. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 2, pages 1228–1231 vol. 2, May 1996.
- [WPUB11] Manuel Werlberger, Thomas Pock, Markus Unger, and Horst Bischof. Optical flow guided tv-l1 video interpolation and restoration. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 273–286. Springer, 2011.

- [WPZ⁺09] Andreas Wedel, Thomas Pock, Christopher Zach, Horst Bischof, and Daniel Cremers. An improved algorithm for tv-l 1 optical flow. In Daniel Cremers, Bodo Rosenhahn, Alan L. Yuille, and Frank R. Schmidt, editors, *Statistical and Geometrical Approaches to Visual Motion Analysis*, volume 5604 of *Lecture Notes in Computer Science*, pages 23–45. Springer Berlin Heidelberg, 2009.
- [WS01a] J. Weickert and C. Schnörr. A theoretical framework for convex regularizers in PDE-based computation of image motion. *International Journal of Computer Vision*, 45(3):245–264, December 2001.
- [WS01b] Joachim Weickert and Christoph Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245–255, 2001.
- [WSL08] M W Westneat, J J Socha, and W K Lee. Advances in biological structure, function, and physiology using synchrotron x-ray imaging*. *Annu Rev Physiol*, 70:119–142, 2008.
- [WTB⁺10] Timm Weitkamp, Paul Tafforeau, Elodie Boller, Peter Cloetens, Jean-Paul Valade, Pascal Bernard, Franoise Peyrin, Wolfgang Ludwig, Lukas Helfen, and Jos Baruchel. Status and evolution of the esrf beamline id19. *AIP Conference Proceedings*, 1221(1):33–38, 2010.
- [XCJ08] Li Xu, Jianing Chen, and Jiaya Jia. A segmentation based variational model for accurate optical flow estimation. In David Forsyth, Philip Torr, and Andrew Zisserman, editors, *Computer Vision, ECCV 2008*, volume 5302 of *Lecture Notes in Computer Science*, pages 671–684. Springer Berlin Heidelberg, 2008.
- [XJM10] Li Xu, Jiaya Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1293–1300, 2010.
- [XPI⁺10] Chun Xu, James J Pilla, Gamaliel Isaac, Joseph H Gorman, Aaron S Blom, Robert C Gorman, Zhou Ling, and Lawrence Dougherty. Deformation analysis of 3d tagged cardiac images using an optical flow method, 2010.
- [YJS06] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4), December 2006.
- [You71] D. M. Young. *Iterative Solution of Large Linear Systems*. Academic Press, New York, 1971.
- [ZBW11] Henning Zimmer, Andrés Bruhn, and Joachim Weickert. Optic flow in harmony. *International Journal of Computer Vision*, 93(3):368–388, 2011.

- [ZER⁺13] S. Zabler, A. Ershov, A. Rack, F. García-Moreno, T. Baumbach, and J. Banhart. Particle and liquid motion in semi-solid aluminium alloys: A quantitative *in situ* microradioscopy study. *Acta Materialia*, 61(4):1244 – 1253, 2013.
- [ZF03] Barbara Zitov and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.
- [ZJK05] C.L. Zitnick, Nebojsa Jojic, and Sing Bing Kang. Consistent segmentation for optical flow estimation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1308–1315 Vol. 2, 2005.
- [ZRGM⁺10] S. Zabler, A. Rack, F. García-Moreno, A. Ershov, T. Baumbach, and J. Banhart. Imaging fast processes in liquid metal foams and semi-solid alloys using synchrotron radioscopy with spatio-temporal micro-resolution. In Thomas Kannengiesser, Sudarsanam Suresh Babu, Yu-ichi Komizo, and Antonio J. Ramirez, editors, *In-situ Studies with Photons, Neutrons and Electrons Scattering*, pages 149–158. Springer Berlin Heidelberg, 2010.